Supplementary materials for the article:

Detecting hydrological connectivity using causal inference from time-series: synthetic and real karstic study cases

Damien Delforge [1,2,*]**, Olivier de Viron** [3]**, Marnik Vanclooster** [1]**, Michel Van Camp** [2]**, and Arnaud Watlet** [4]

[1] Earth and Life Institute, Université catholique de Louvain, Louvain-la-Neuve, Belgium;

[2] Royal Observatory of Belgium, Brussels, Belgium;

[3] Littoral, Environnement et Sociétés, Université de La Rochelle and CNRS (UMR7266), La Rochelle, France

[4] British Geological Survey, Nottingham, UK

* Correspondence: damien.delforge@uclouvain.be

SM1: Causal Inference Methods (CIMs)

SM1.1 Cross-Correlation Function (CCF)

For a driving variable X_t and a response variable Y_t of N samples, causality is framed through the computation of the cross-correlation function (CCF) and the principle of priority of the cause. For a window of absolute delays $[0, d_{max}]$ with $d_{max} > 0$, the Pearson's correlation coefficient ρ is computed between the response and the delayed driver on their overlapping time-domain:

$$CCF(d) = \rho(X_{t-d}, Y_t), \tag{1}$$

with $d \in [0, d_{max}]$. The Pearson's ρ between two time-series X_t and Y_t is the ratio between their covariance and the product of their standard deviations:

$$\rho(X_t, Y_t) = \frac{\text{cov}(X_t, Y_t)}{\sigma_X \sigma_T}, \tag{2}$$

The ρ coefficient is a standardized measure of linear dependencies that can be interpreted as the slope of a linear regression between the two standardized variables (i.e., zero mean and unit variance). Accordingly, ρ is ranging between -1 and 1, meaning respectively perfectly anti-correlated or correlated. A ρ of zero indicates the absence of linear dependencies. The significance of the hypothesis that ρ is different from zero is usually assessed analytically through a Student's-t test reporting a p-value. The p-value estimates the probability that the correlation between the two time-series is the output of an uncorrelated process. The p-value is sensitive to the number of overlapping samples such that more samples are required to have a significant p-value if $|\rho|$ is low. For a significance level α, significant relationships are considered when the p-value is lower than α. In the main manuscript, significant correlations output by the CCF method are considered as a way to reveal potentially causal links between two time-series at a causal delay d, in virtue of the principle of priority. The case of $d = 0$ refers to a contemporaneous dependency and does not allow to infer a direction for the causal relationship. Correlation and the significance test were performed using Python and the Scipy library (Virtanen et al., 2020).

SM1.2 Convergent Cross-Mapping (CCM)

CCM is a CIM rooted in the theory of nonlinear dynamical systems. It aims at detecting weak nonlinear associations between two time-series (Sugihara et al., 2012). CCM goes beyond linear correlation by checking if two variables behave consistently when the system revisits the same states. To approximate the states of a system, CCM relies on Takens's embedding theorem (Takens, 1981). To address whether X_t causes Y_t, the response variable Y_t is first embedded using Takens's state space reconstruction:

$$M_Y = \{Y_t, Y_{t-\tau}, \dots, Y_{t-(m-1)\tau}\}, \tag{3}$$

M_Y is the reconstructed state space, i.e., a trajectory matrix defined by the embedding delay τ and the embedding dimension m.

The state space reconstruction of the potential response variable Y_t is the starting point of the CCM flowchart (Figure SM.1). CCM uses a nearest-neighbor algorithm to make forecasts of the states $x_{i+d} \in X_t$, with X_t the potential cause of Y_t and d the time to prediction and potential causal delay (up to d_{max}). *REF* and *LIB* are two user-defined set of time-series indices. *REF* spots the time-indices of reference, such that the forecast being performed would be x_{i+d} for all $i \in REF$, with d the forecast horizon and potential causal delay. The set *LIB* lists the time-indices where nearest-neighbors can potentially be identify. By default, no restrictions apply on *LIB* and *REF* and these sets cover the whole time domain, with the exception of the indices truncated by the reconstruction or the prediction delay d. In practice, L trajectory samples are randomly selected from the *LIB* set. Working on subsamples allows repeating forecasts N_{SAM} times to bootstrap statistics and makes more reliable forecasts. To perform a single forecast, $m + 1$ nearest-neighbors of the state $\dot{y}_i = \{y_i, y_{i-\tau}, \dots, y_{i-(m-1)}\}$ in M_Y are identified using Euclidean distance. The $m + 1$ is the default number of nearest-neighbors. We denote this set of nearest-neighbor states $\{\dot{y}_{e1}, \dot{y}_{e2}, \dots, \dot{y}_{e(m+1)}\}$ with a corresponding set of Euclidean distances to $\dot{y}$ noted $\{e_1, e_2, \dots, e_{m+1}\}$. Based on the time-indices of the nearest-neighbor set, corresponding points in X_t are identified and shifted by d time-steps as $\{x_{e1+d}, x_{e2+d}, \dots, x_{e(m+1)+d}\}$. These $m + 1$ states are averaged using exponential weights w_j defined based on the vector of distances:

$$w_j = \exp - \frac{e_j}{\min\{e_1, e_2, \dots, e_{m+1}\}}, \text{ with } j \in \{1, 2, \dots, m+1\} \qquad (4)$$

Each weight w_j is then divided by the sum of all weight so that they sum up to 1. The weighted average of the time-series values $\{x_{e1+d}, x_{e2+d}, \dots, x_{e(m+1)+d}\}$ provides the estimate of x_{i+tp}. As the sampling of the L random states (without replacement) in M_Y is repeated N_{SAM} times, the output forecast matrix is of size $L \times N_{SAM}$. CCM forecasting skills are typically addressed with the mean Pearson correlation (Eq. 2) $\bar{\rho}$ between the N_{SAM} vectors of L forecasts and the corresponding observed values.
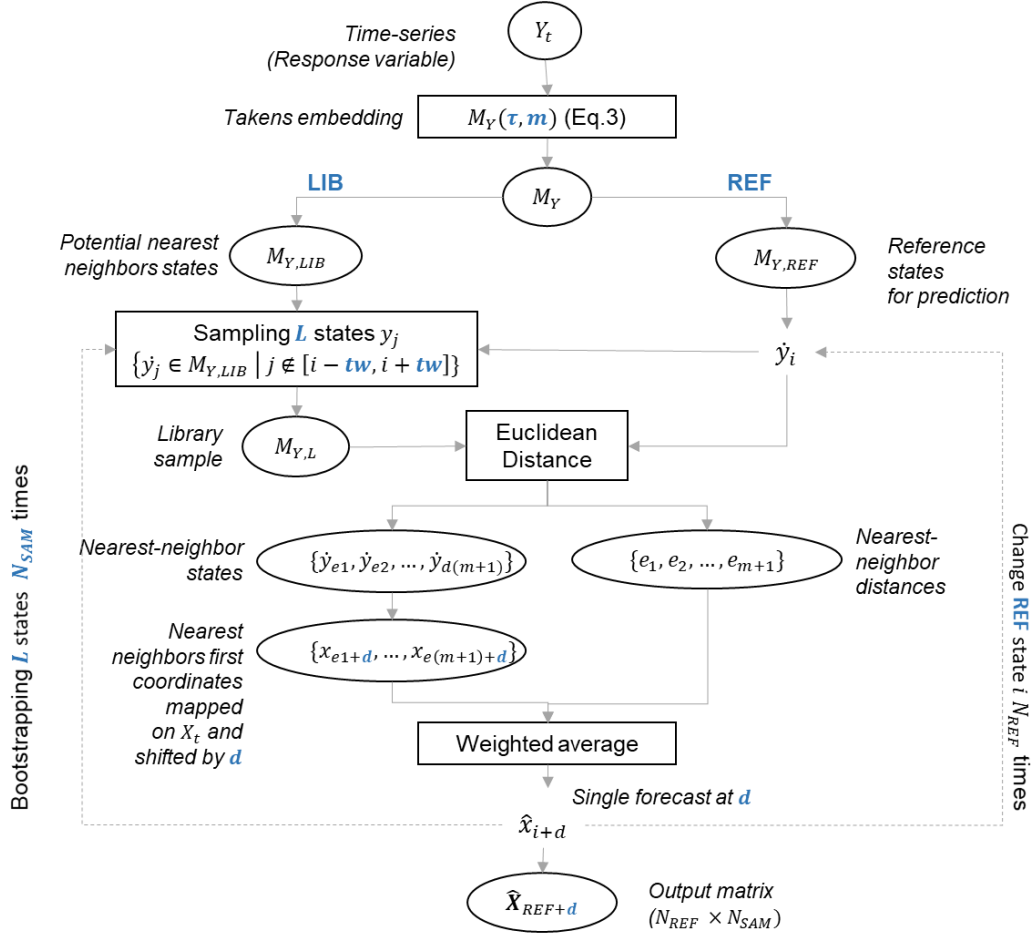
Figure SM.1. CCM Algorithm Flowchart. User-defined parameters are displayed in blue.

In fine, the fact that x_{i+d} can be significantly predicted from other points in X_t identified from time-series segments in Y_t, that are nearest-neighbors to $\dot{y}_i$, is used as an indicator that X_t causes Y_t. The effective cross-state then cross-time mapping suggests that X_t dynamic is embedded in Y_t, which suggests direct causality, or at least a common belonging to the same dynamical system. In the original paper (Sugihara et al., 2012), causality is inferred using a prediction horizon $d = 0$ and from the principle of convergence, meaning that $\bar{\rho}$ should progressively increase with larger sample sizes L. Our implementation does not consider convergence as a sufficient criterion for causality. We assume that if $\bar{\rho}$ is significantly high, convergence is expected. Similarly to the CCF method, we instead vary the prediction horizon d to identify significant causal delays and discriminate the driver from the response based on the principle of priority of the cause. This CCM approach is suggested for variables exhibiting a correlated and synchronous behavior (Ye et al., 2015), such as hydrological variables. The significance of the average forecast skills $\bar{\rho}$ is conducted through a Student's-t test, considering the sample length L. This means that $\bar{\rho}$ would be significantly different from the expected correlation between two white noise signal of length L, i.e., zero.

With respect the main manuscript, we applied no restriction on REF and LIB. A fixed sample length of L of 100 was used 100 times (N_{SAM}). The optimal embedding parameters (Eq. 3, m and τ) for Y_t can be selected by optimizing the self-forecasting skills (Sugihara and May, 1990). These self-prediction skills are obtained by applying the CCM algorithm on a single variable ($Y_t = X_t$ in Figure SM.1). The embedding delay τ was set at the time-series resolution of 1 day, while $m = 2$ was found to provide the best forecasting skills in general. Besides, we also considered a time exclusion window of 10 days, known as the Theiler window (Theiler, 1986), such that nearest-neighbor states in M_Y are not neighbors in time but remote by at least 10 days and picked whenever Y_t revisits recurrent trajectories. The Theiler window ensures that predictive power arises from state dependencies, not from auto-correlation patterns.

Finally, we chose to post-process the retrieved significant $\bar{\rho}$ to remove an undesirable effect of the embedding parameter m. For a single causal delay d, the embedding has the effect of providing significant predictive skills over the embedding window, such that the delays between d and $d + (m-1)$ are usually significant [see 24]. This effect is illustrated in Figure SM.2. To discard the undesirable sustain of predictive skills, we truncated all significant segments of consecutive delays by 1 day, since m is 2, as a post-filtering of the significant CCM relationship. Consequently, isolated significant delays are also removed. Besides, negative $\bar{\rho}$ are not considered since they are meaningless.
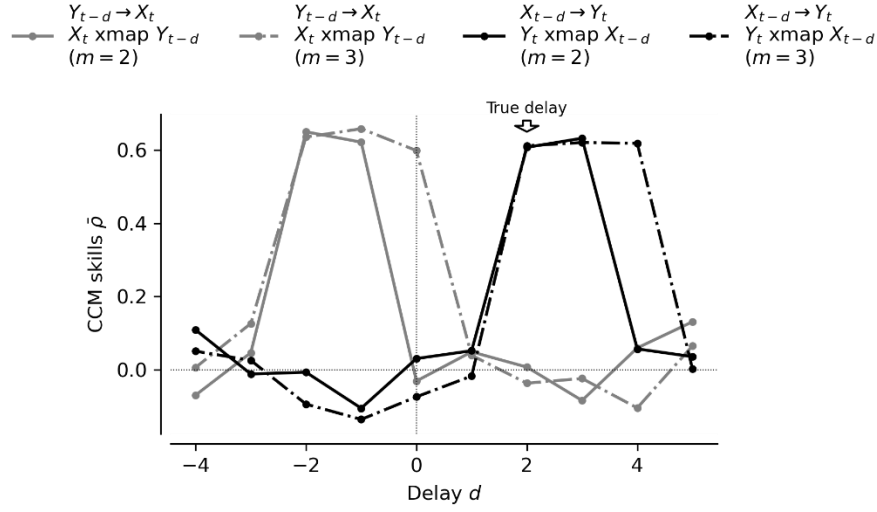
Figure SM.2. Example of CCM on a linear stochastic model $Y_t = 0.2Y_{t-1} + 0.2X_{t-2} + 0.2\varepsilon$ with ε a standard white noise. The model assumes a univariate causal relationship $X_{t-2} \to Y_t$ that should be revealed by CCM forecast (xmap) from Y_t to X_{t-2} (True delay $d = 2$). The figure shows the results of CCM forecasts from Y_t to X_t (black) and vice versa (gray) for delays ranging between -4 and 5. CCM skill $\bar{\rho}$ is the mean Pearson correlation between the $N_{SAM} = 100$ predicted vectors of length $L = 100$ and the corresponding true values. The effect of the embedding dimension is clearly visible. Significant predictive skills are sustained at least over a window of size m (when $\tau = 1$). Hence, island of significant dependencies should be tailed from the right to reveal the true delays associated with positive d. The truncation also prevents future dependencies ($d < 0$) from extending past dependencies ($d > 0$) and being misinterpreted as causal relationships.

Regarding the CCM code, CCM is applied with a Python version developed in (Delforge et al., 2020). A computationally faster and official Python version of CCM, as well as R and C++ implementation, are currently available from the repository: https://github.com/SugiharaLab/pyEDM.

PCMCI is a 2-step procedure: PC, named after its authors Peter and Clark (Spirtes and Glymour, 1991), and MCI, standing for Momentary Conditional Independence (Runge et al., 2019). In general, considering a multivariate time-series process $X_t = \{X_t^1, \dots, X_t^p\}$ of p time-series, PCMCI allows recovering a causal graph based on the principle of conditional independence. A delayed time-series X_{t-d}^i with $i \in [0, p]$ causes itself or another series X_t^j with $j \in [0, p]$, if conditional independence to the past of the process X_t^- excluding X_{t-d}^i is rejected:

$$X_{t-d}^i \rightarrow X_t^j \Leftrightarrow X_{t-d}^i \neg \perp\!\!\!\perp X_t^j \mid X_t^- \backslash \{X_{t-d}^i\}, \tag{5}$$

where the symbols respectively mean: $\rightarrow$: "causes"; $\Leftrightarrow$: "implies"; $\neg \perp\!\!\!\perp$: "not independent"; |: "conditioned to"; and \: "excluding".

The Full Conditional Independence algorithm (FullCI) is entirely based on Eq. 5. However, FullCI suffers from the curse of dimensionality if the conditioning involves too many variables in the conditioning set X_t^- (Runge et al., 2019). Hence, the purpose of the prior PC step is to estimate first the potential parents $\widehat{\wp}(X_t^j)$ for each variable X_t^j. Tigramite relies on the PC$_1$ iterative procedure by default. Initially, all potential parents are considered as $\widehat{\wp}(X_t^j)$. In the first step, all the parents that are unconditionally independent to X_t^j are removed, and the parent presenting the strongest dependency is identified. In the second one, the parents that are independent to X_t^j conditionally to the strongest parent of step 1 are removed, and a second parent with the highest conditional dependence is identified as an additional condition for step 3. The operation is repeated considering a 1-by-1 increasing number of conditions up to a point there are no more conditions to test in $\widehat{\wp}(X_t^j)$. Then, the MCI second step starts and tests for conditional independence on the dimensionally reducedset of parents resulting from PC1, such that:

$$X_{t-d}^i \rightarrow X_t^j \Leftrightarrow X_{t-d}^i \neg \perp\!\!\!\perp X_t^j \mid \widehat{\wp}(X_t^j) \backslash \{X_{t-d}^i\}, \widehat{\wp}(X_{t-d}^i) \tag{6}$$

Resulting from the MCI step, links where conditional independence cannot be rejected are considered as true causal parents, i.e., causally inferred sufficient causes. These links are reported in the resulting DAG.

PCMCI flexibly allows us to consider different conditional independence tests: a linear method by assessing Partial Correlations (ParCorr) and a nonlinear one relying on Conditional Mutual Information (CMI). Partial correlations are Pearson's correlations (Eq. 2) between X_{t-d}^i and the residuals of the multivariate linear regression model of X_t^j against its conditions, for instance, $\widehat{\wp}(X_t^j) \backslash \{X_{t-d}^i\}, \widehat{\wp}(X_{t-d}^i)$ in the case of MCI. The linear model is fit using Ordinary Least Square regression, and the correlation significance is estimated with p-value resulting from a Student's t-test. This framework is very similar to the popular Granger causality, which is based on vector auto-regressive models (Granger, 1969). However, PCMCI-ParCorr differs from the usual Granger causality in three aspects: (1) Granger causality does not rely on the PCMCI procedure and, thus, suffers from the curse of dimensionality; (2) GC does not report contemporaneous dependencies ($d = 0$); and (3) Granger causality relies on a F-test, testing if including a potential driver X_{t-d}^i in the multivariate model of X_t^j significantly reduces the variance of residuals.

In contrast, CMI can be seen as a multivariate extension of the transfer entropy method (Schreiber, 2000). In the information theory, CMI, or $I_{X,Y|Z}$, is the mutual information between two variables X_t and Y_t conditioned to Z_t:

$$I_{X,Y|Z} = \iiint p(x, y, z) \log \frac{p(x, y|z)}{p(x|z)p(y|z)} dx dy dz \tag{7}$$

If $I_{X,Y|Z} = 0$, X_t and Y_t are conditionally independent to Z_t, and, therefore, not directly causally related, given that the probability densities are correctly estimated. For this purpose, PCMCI with the CMI conditional independence test offers three different methods: Gaussian Process and Distance Correlation (GPDC); a k nearest neighbor estimator (CMIknn); and an estimator based on kernel measures of CMI (RCOT). The author recommends using the most general conditional independence test, CMIknn, where multiplicative noise is expected, as in hydrology [e.g., 12], and where the sample size is lower than 1000, which is the case in this study. CMIknn is, therefore, considered. The latter relies on a nearest-neighbor CMI estimator (Frenzel and Pompe, 2007; Vejmelka and Paluš, 2008) combined with a local permutation scheme as a nonparametric test for conditional independence (Runge, 2018).

Besides the maximum lag d_{max}, PCMCI requires other arguments. The PC stage retrieves parents according to a regularization parameter α_{PC} ranging between 0 and 1. The higher α_{PC}, the higher the number of parents, with $\alpha_{PC} = 1$ corresponding to the FullCI algorithm (Eq. 5). If α_{PC} is too low, true parents might be missing. If α_{PC} is too high, the MCI step may retrieve spurious results due to the curse of dimensionality. When the ParCorr conditional independence test is selected, PCMCI allows optimizing α_{PC} while minimizing the Akaike Information Criterion (Akaike, 1974). In the main manuscript, this feature is used to generate the ParCorr DAG. For CMI, we tested the method considering two values: the default $\alpha_{PC} = 0.2$ and a more restrictive $\alpha_{PC} = 0.05$.

The CMIknn method further relies on two hyperparameters (Runge, 2018). The first, k_{CMI}, defines the size of the neighborhood for the knn estimator of CMI. The k_{CMI} mostly acts as a smoothing parameter regarding the CMI, and should not be too small. The second parameter, k_{perm}, defines the size of the neighborhood for the local permutation scheme for the shuffling test. The k_{perm} parameters has less importance than k_{CMI} regarding the sensitivity of the outcomes.

The PCMCI algorithm (Runge et al., 2019) is implemented in the Tigramite Python package for causal time-series analysis (version 4.1 for the main manuscript). The GitHub repository contains further information and tutorials to run the causal discovery framework: https://github.com/jakobrunge/tigramite/.

SM2: Supplementary Results

SM2.1. Cross-Correlation Function

Figure SM.3 shows the CCF time-dependencies in the form of a correlogram. CCF was applied on the first-order differenced real dataset (see the main manuscript).
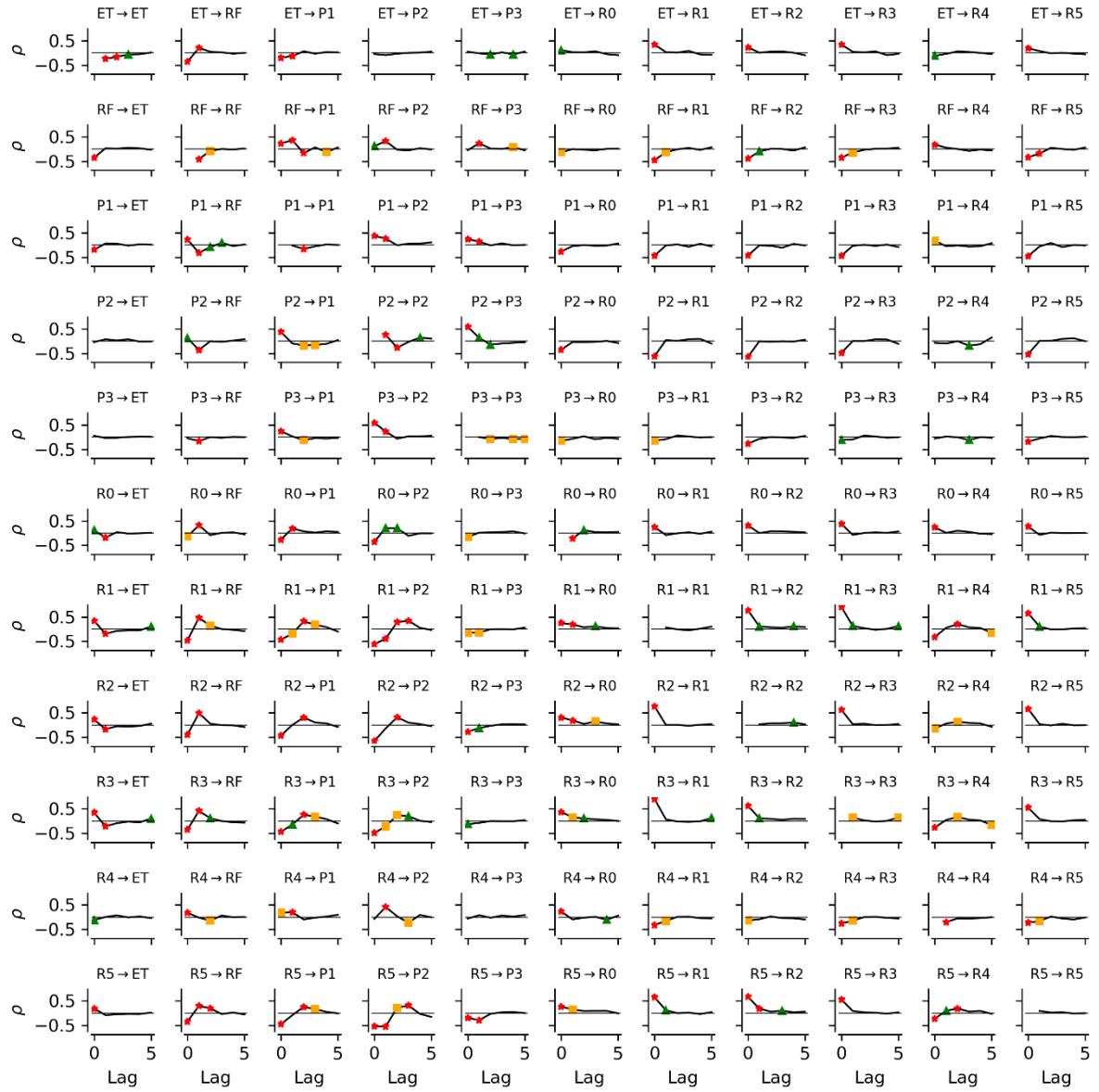
Figure SM.3. CCF time-dependencies applied to the first order differenced data. Lagged dependencies significantly different from zero are reported with red stars (p-val < 0.001), orange squares (0,001<p-val<0.01), and green triangles (0.01<p-val<0.05).

Figure SM.4 shows the CCM time-dependencies on the first order differenced dataset. Note that the significant dependencies are expected to sustain over an additional delay since the embedding dimension m (Eq. 3) is two days. This effect was removed in the main manuscript using post-filtering (see SM1.2).
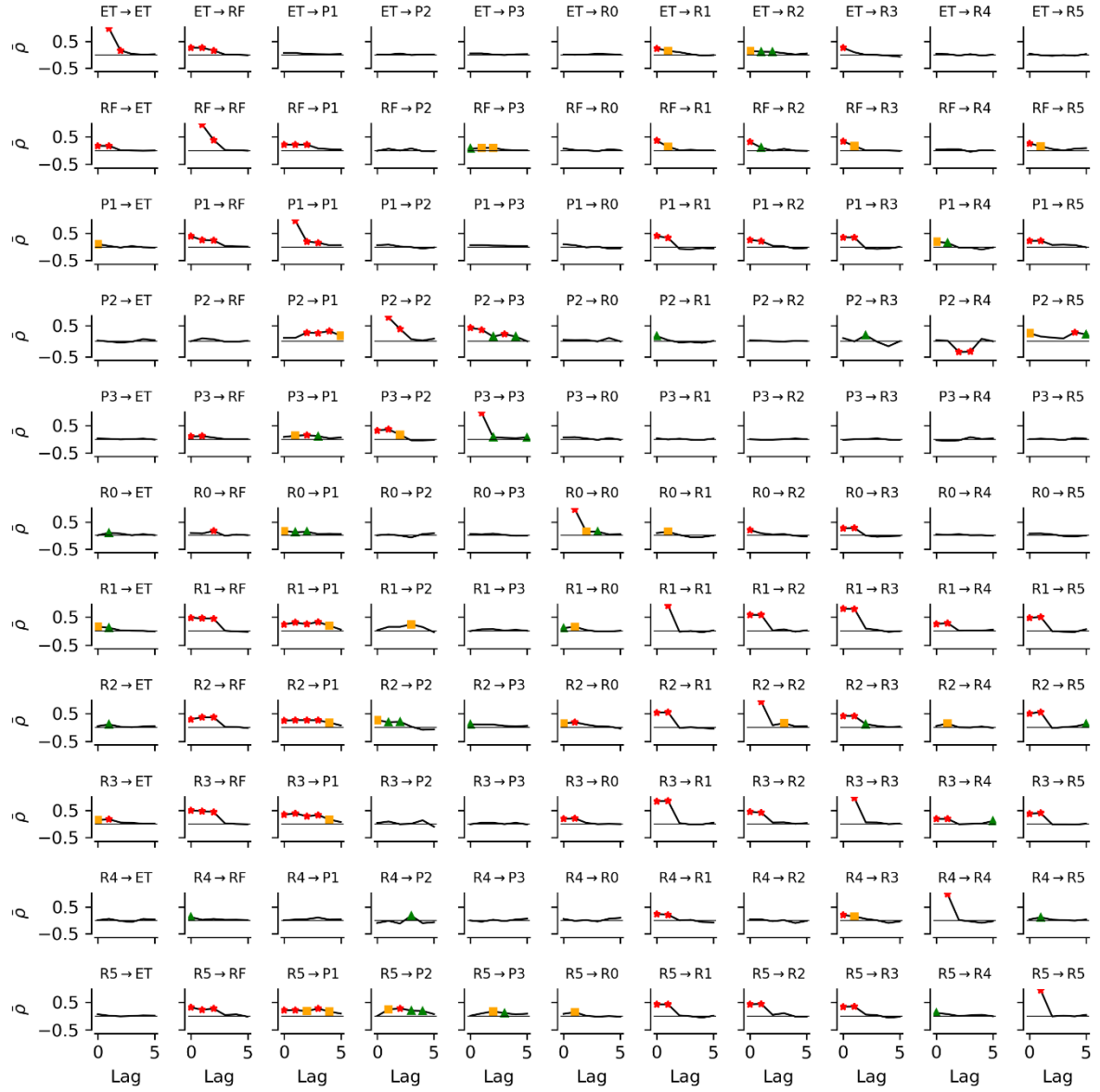
Figure SM.4. CCM time-dependencies applied to the first order differenced data. Lagged dependencies significantly different from zero are reported with red stars (p-val < 0.001), orange squares (0,001<p-val<0.01), and green triangles (0.01<p-val<0.05). The results are not post-processed (see SM1.2).

PCMCI with the CMIknn applied on the real dataset was unstable, potentially due to the small size of the overlapping time-domain of the conditioned variables. To illustrate this instability, Figure SM.5 and SM.6 reports the causal graph outputs for two runs obtained with the same parameter configuration (α_{PC}, k_{CMI}, k_{perm}). In particular, some strange causal loops sometimes appear between resistivity variables (Figure SM.5c R0-R3, or Figure SM.6a R0-R1). This observation led us to believe that the highly smoothed averaged resistivity time-series obtained from the inverted time-lapse Electrical Resistivity Tomography model contributes to the instability.
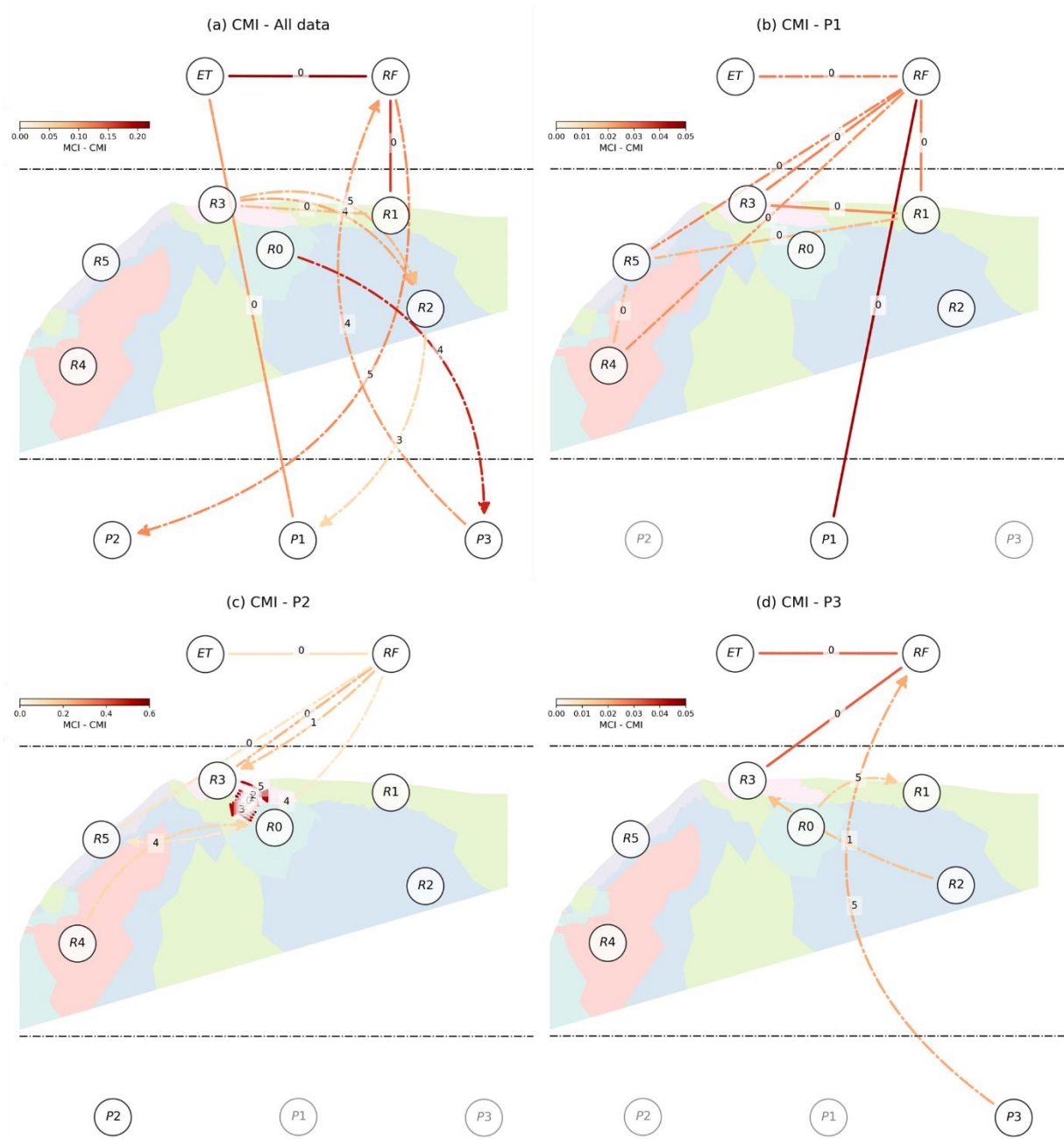
Figure SM.5. Graph of CMI cross-dependencies (RUN 1): (a) considering all data,(b) excluding P2, P3, (c) excluding P1, P3, (d) excluding P1, P2. Contemporaneous dependencies are represented by a undirected straight arrow. Delayed dependencies are shown using directed curved arrows. All corresponding delays d are displayed in the middle of its corresponding arrow. The color of arrows maps to the strength of dependencies. Solid and dash-dotted arrows represent respectively significant dependencies with p-value < 0.001 and < 0.01.

Figure SM.6. Graph of CMI cross-dependencies (RUN 2): (a) considering all data,(b) excluding P2, P3, (c) excluding P1, P3, (d) excluding P1, P2. Contemporaneous dependencies are represented by a undirected straight arrow. Delayed dependencies are shown using directed curved arrows. All corresponding delays d are displayed in the middle of its corresponding arrow. The color of arrows maps to the strength of dependencies. Solid and dash-dotted arrows represent respectively significant dependencies with p-value < 0.001 and < 0.01. For each graph, the size of the overlapping time-domain between the variables changes as follows: 48 days (a), 184 days (b) 62 days (c), and 218 days (d).

To better evaluate how dissimilar two causal graphs are and in which case, we performed a full sensitivity analysis. This sensitivity analysis varies the datasets as well as the parameters of the algorithm, and runs 5 tests for each configuration to investigate convergence problems. The sensitivity factors are:

- 'FD', 'P1', 'P2', 'P3': the four dataset considered previously with 'FD' being all data, 'P1' is FD excluding P2, P3, 'P2' is FD excluding P1, P3, and 'P3' is FD excluding P1, P2.

- The α_{PC} parameter defined as either 0.2 or 0.05;

- The k_{CMI} parameter belonging to [5, 10, 15, 20, 30, 40]. We cannot go beyond the absolute value of 48 which is the size of the overlapping domain for FD conditioned variables.

- The k_{perm} parameter as in [5, 10, 15, 20].

This parameter space covers 960 simulations executed with d_{max} equal to 5 days. To give an account of the high computational cost of CMIknn, the sensitivity analysis took about two weeks on a single PC laptop machine, 12 threads, intel core i7 9th generation. Figure SM.7 and SM.8 shows the mean Jaccard similarity between each combination of the 5 causal graphs retrieved for each parameter set. The Jaccard similarity (implemented in (Pedregosa et al., 2011)) reports the size of the intersection divided by the size of the union of two vector of labels, i.e., labeled as causal or non-causal in this case. For Figure SM.7 and SM.8, the significance p-value threshold was set respectively to 0.01 and 0.05. However, the Jaccard similarity capture the similarity that is due to chance. Figure SM.9 and SM.10. reports the Adjusted Rand Index (ARI), also implemented in (Pedregosa et al., 2011), which is a similarity metric that account for chance (0 being the expected score of random similarity). In overall, a slightly lower similarity obtained with $\alpha_{PC} = 0.2$, which allows more causal parents. The effect of the dataset is visible with ARI, especially with $\alpha_{PC} = 0.2$) showing that the FD and P2 appears more random due to the small size of their overlapping time domain (Figure SM.9 and SM.10). P1 is the less random, which is also where we expect an hydrological connection to be part of the graph. Regarding the optimal choice of k_{CMI} or k_{perm}, no clear-cut pattern appears. In the main manuscript, we therefore chose to build the consensual (majority) causal graph using all the causal graphs obtained through this sensitivity analysis.
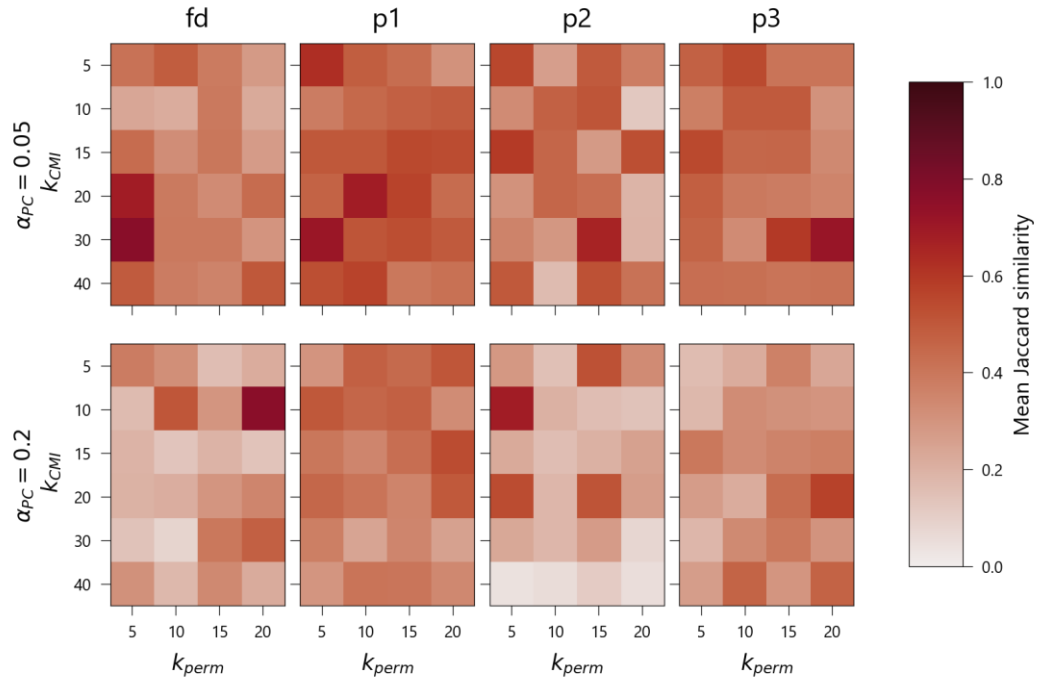
Figure SM.7. Mean Jaccard Similarity between the combinations of 5 graphs obtained with the same parameter sets and significance threshold p-value of 0.01.
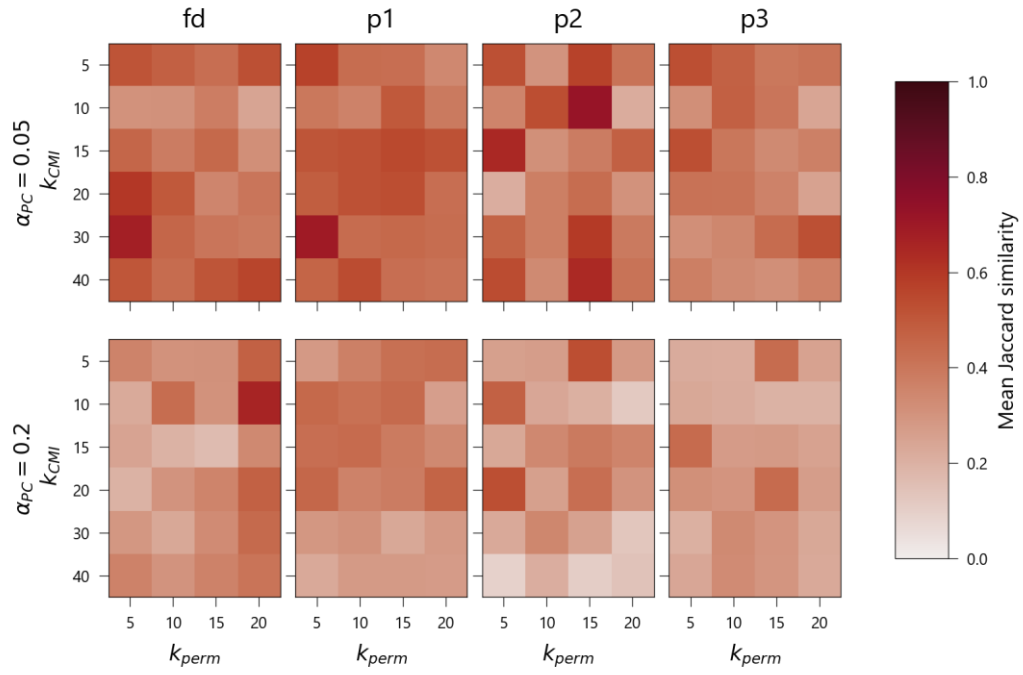
Figure SM.8. Mean Jaccard Similarity between the combinations of 5 graphs obtained with the same parameter sets and significance threshold p-value of 0.05.
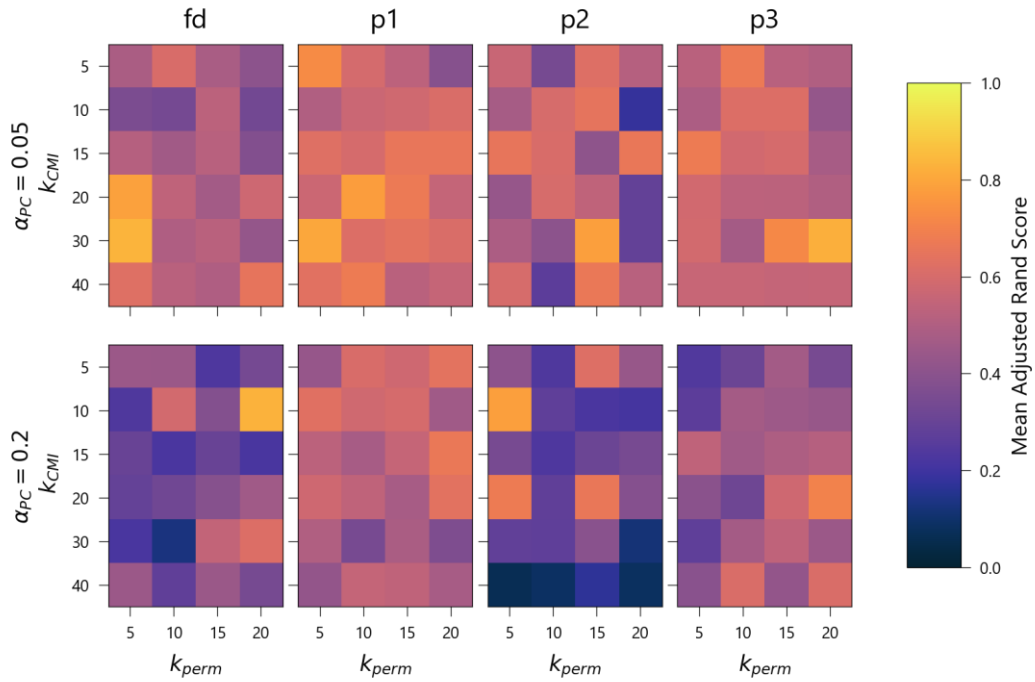
Figure SM.9. Mean Adjusted Rand Index between the combinations of 5 graphs obtained with the same parameter sets and significance threshold p-value of 0.01.
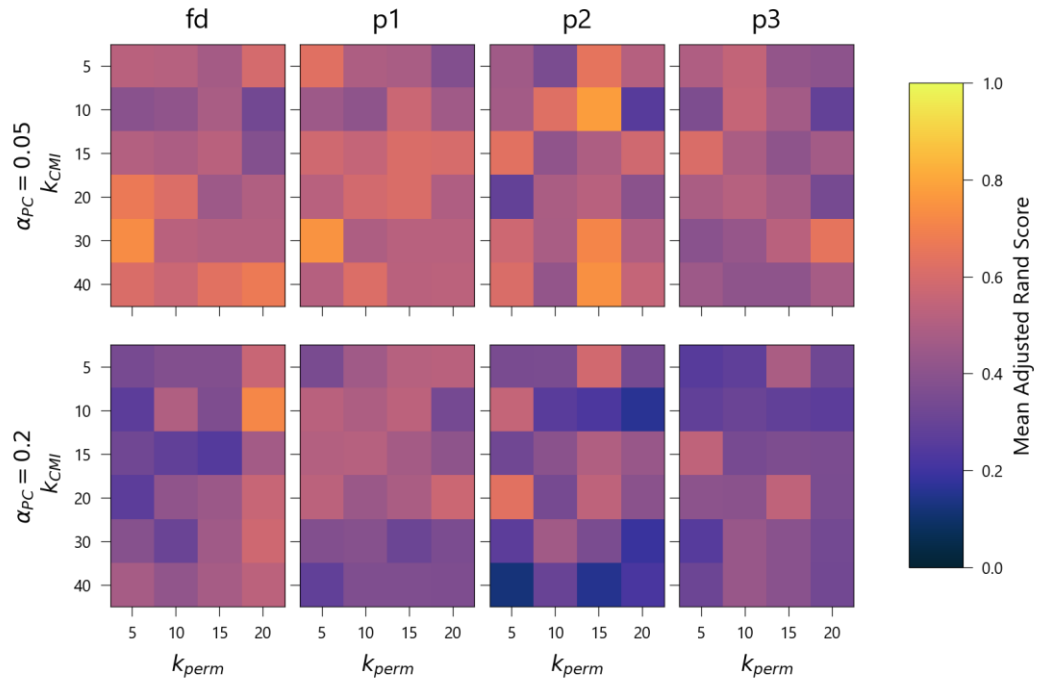
Figure SM.10. Mean Adjusted Rand Index between the combinations of 5 graphs obtained with the same parameter sets and significance threshold p-value of 0.05.

References

Akaike, H.: A new look at the statistical model identification, 19, 716–723, https://doi.org/10.1109/TAC.1974.1100705, 1974.

Delforge, D., Muñoz-Carpena, R., Van Camp, M., and Vanclooster, M.: A Parsimonious Empirical Approach to Streamflow Recession Analysis and Forecasting, 56, e2019WR025771, https://doi.org/10.1029/2019WR025771, 2020.

Frenzel, S. and Pompe, B.: Partial mutual information for coupling analysis of multivariate time series, Phys. Rev. Lett., 99, 204101, https://doi.org/10.1103/PhysRevLett.99.204101, 2007.

Granger, C. W. J.: Investigating Causal Relations by Econometric Models and Cross-spectral Methods, Econometrica, 37, 424–438, https://doi.org/10.2307/1912791, 1969.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E.: Scikit-learn: Machine Learning in Python, 12, 2825–2830, 2011.

Rodriguez-Iturbe, I., Entekhabi, D., and Bras, R. L.: Nonlinear Dynamics of Soil Moisture at Climate Scales: 1. Stochastic Analysis, 27, 1899–1906, https://doi.org/10.1029/91WR01035, 1991.

Runge, J.: Conditional independence testing based on a nearest-neighbor estimator of conditional mutual information, in: International Conference on Artificial Intelligence and Statistics, International Conference on Artificial Intelligence and Statistics, 938–947, 2018.

Runge, J., Nowack, P., Kretschmer, M., Flaxman, S., and Sejdinovic, D.: Detecting and quantifying causal associations in large nonlinear time series datasets, 5, eaau4996, https://doi.org/10.1126/sciadv.aau4996, 2019.

Schreiber, T.: Measuring Information Transfer, Phys. Rev. Lett., 85, 461–464, https://doi.org/10.1103/PhysRevLett.85.461, 2000.

Spirtes, P. and Glymour, C.: An Algorithm for Fast Recovery of Sparse Causal Graphs, Social Science Computer Review, 9, 62–72, https://doi.org/10.1177/089443939100900106, 1991.

Sugihara, G. and May, R. M.: Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series, 344, 734–741, https://doi.org/10.1038/344734a0, 1990.

Sugihara, G., May, R., Ye, H., Hsieh, C. -h., Deyle, E., Fogarty, M., and Munch, S.: Detecting Causality in Complex Ecosystems, 338, 496–500, https://doi.org/10.1126/science.1227079, 2012.

Takens, F.: Detecting strange attractors in turbulence, in: Dynamical Systems and Turbulence, Warwick 1980, edited by: Rand, D. and Young, L.-S., Springer Berlin Heidelberg, 366–381, https://doi.org/10.1007/BFb0091924, 1981.

Theiler, J.: Spurious dimension from correlation algorithms applied to limited time-series data, Phys Rev A Gen Phys, 34, 2427–2432, 1986.

Vejmelka, M. and Paluš, M.: Inferring the directionality of coupling with conditional mutual information, Phys. Rev. E, 77, 026214, https://doi.org/10.1103/PhysRevE.77.026214, 2008.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, 17, 261–272, https://doi.org/10.1038/s41592-019-0686-2, 2020.

Ye, H., Deyle, E. R., Gilarranz, L. J., and Sugihara, G.: Distinguishing time-delayed causal interactions using convergent cross mapping, 5, 14750, https://doi.org/10.1038/srep14750, 2015.