Reply to Reviewer 3

Thank you very much for taking the time to review our manuscript and for providing constructive feedback. Your comments have helped improve the quality of our work. We have revised the manuscript and repeated all analyses, taking into account the points you raised. Additionally, we increased the number of model runs from 100 to 500 using a high-performance computing (HPC) system. The updated code used in our analysis has also been uploaded to GitHub for transparency and reproducibility. We hope that the changes we have made address your concerns. Below, we summarize the main points we have revised:

1. **First of all, I want to point out that the output of the PC algorithm is a CPDAG, i.e., the Markov equivalence class. A CPDAG is a graphical representation of a set of DAGs where the distribution satisfies the Markov property relative to every single DAG in that set. This means that given a distribution which satisfies the Markov property for one DAG, there could also be several other DAGs for which the Markov assumption is satisfied. In other words, given a distribution, there are several DAGs where the distribution fulfills the Markov assumptions with respect to those DAGs. When orienting undirected edges to obtain a DAG from this set of DAGs, you should be careful not to introduce new unshielded colliders / v-structures. Additionally, you can also introduce background knowledge before running the PC algorithm. See for example "Interpreting and using CPDAGs with background knowledge" (2017) or "Constraint-based causal discovery with tiered background knowledge and latent variables in single or overlapping datasets" (2025).**

   Thank you for this comment. We totally agree that we should be aware not to introduce any new unshielded colliders when orienting undirected edges to obtain the DAG. To derive DAGs from CPDAGs, we previously relied on expert knowledge before and after running the PC algorithm. Therefore, to ensure the validity of our DAGs, and in response to your third comment regarding variable selection, we repeated the entire analysis using a consistent set of 22 variables across all runoff signature (target) variables. In this revised analysis, we incorporated general edge assumptions as background knowledge, considering the references you provided, prior to running the PC algorithm. We placed these implausible links in a blacklist and then ran the PC algorithm. This helped have a valid CAPDAG with a minimum number of Markov equivalence classes. Using this blacklist, we obtained a CPDAG with only one undirected edge between mean elevation and mean slope. Since neither of these variables has any parent nodes, the resulting CPDAG corresponds to two Markov equivalence classes. Introducing a directed edge, either from mean elevation to mean slope or vice versa, does not create any unshielded colliders. Therefore, we oriented the edge from mean elevation to mean slope. These points are addressed in the newly added Section 3.2 of the Methods, and we have rewritten and updated the entire explanation of the causal discovery section accordingly.

2. **Depending on the specific algorithm and plot function you are using, you will obtain bi-directed edges. If you use R and the pcalg package, there are cases where the direction could not be determined (but not because of the Markov equivalence class). This leads to an invalid CPDAG, meaning that your output is not representing a Markov equivalence class. There are several reasons for this, but it is important to mention that undirected and bi-directed edges are not the same. There are at least three violations of assumptions which may lead to an invalid CPDAG: cycles, hidden common causes, and selection bias. It could be that you set a direction for an edge without realizing that you are violating something else (for example, introducing new unshielded colliders / v-structures or creating cycles).**

   Thank you for raising this point and for your clear explanation. We used the *bnlearn* package in R along with the *graphviz.plot* function to generate the CPDAGs, which correctly displays undirected

edges when appropriate. To enhance visual clarity and ensure consistency across all figures, we manually recreated the graphs using Inkscape, a vector graphics editor. No bidirected edges were introduced during the causal discovery process. Additionally, we took care not to orient any edges in a way that would violate the assumptions of a valid CPDAG, such as by creating cycles, introducing unshielded colliders (v-structures), or misrepresenting the underlying Markov equivalence class.

3. **You wrote: "It is worth mentioning that we attempted to include all continuous variables in the causal discovery process without applying variable selection. This approach was tested to address the causal sufficiency assumption in the PC algorithm, which requires that all common causes of the target variables are accounted for. Despite this, we observed challenges such as the generation of disconnected DAGs with independent nodes or groups of nodes lacking causal relationships with runoff signatures." This should be a big red warning signal for your analysis and needs further investigation. The output of the PC algorithm heavily depends on the alpha value, this is the significance level for the tests. Have you optimized this value somehow? How does the results depends on alpha? Additionally, it seems strange to me to perform feature selection methods before applying the PC algorithm. The PC algorithm also estimates Pearson correlations, and the first step of the PC algorithm (learning the skeleton) is based purely on conditional independence tests. These tests are, for example, partial correlation for two variables conditional on a third one, or in the first step just a correlation between two variables. Why should you use an extra step of correlation analysis if the PC algorithm will do the same and additionally will save the information on potential unshielded colliders / v-structures?**

   Thank you for this comment. We acknowledge that the detected edges by the PC algorithm are sensitive to the choice of the alpha level, which controls the threshold for conditional independence tests. In our initial experiments using all available continuous variables (without feature selection), we applied the alpha value of 0.05. However, this resulted in sparse CPDAGs, sometimes with disconnected nodes, for example, geological permeability. Additionally, domain-reasonable relationships, such as the influence of geological attributes on the baseflow index, failed to appear under this setting. Therefore, applying the PC algorithm to a selected set of variables with an alpha value of 0.05 resulted in physically meaningful graphs. However, due to concerns raised by the reviewers regarding the variable selection process, we removed this step and instead applied the PC algorithm to a consistent set of 22 variables for each runoff signature in the revised manuscript. We acknowledge that the alpha value significantly influences the algorithm's results; however, the relatively small sample size (around 670 data points) may also contribute to graph sparsity or potential underfitting (Zuk et al., 2012).

   Therefore, to address the loss of information on potential v-structures and account for the relatively small sample size, we applied the PC algorithm to all 22 variables without performing variable selection, using a significance threshold of 0.2 to allow for a more inclusive initial edge selection. In this setting, the edge assumptions, which are explained in the first comments, avoid appearances of spurious links. To assess the stability of the discovered edges, we performed 1,000 bootstrap resamples of the data and applied the PC algorithm to each resample, using a significance threshold of 0.05 for conditional independence tests. We then measured the strength of each edge based on its frequency across the bootstrap iterations. The resulting edge strength estimates, which represent the proportion of bootstrap samples in which each edge appears, were then mapped onto the initial CPDAG obtained from the PC algorithm. This approach enabled us to evaluate the stability of the inferred causal relationships. This method is inspired by the work of Petersen et al. (2021), which you mentioned in your sixth comment. We addressed these points in Section 3.2 and reflected them in the Results and Discussion sections accordingly.

4. **Furthermore, the authors are mainly interested in discovering the parent set of the outcome Y. Why are you not considering methods which are designed for this task? I mean Invariant Causal Prediction (ICP). I would spend a bit more time on the limits of causal discovery and non-linear methods. I would recommend reading: 1) Model-Based Causal Feature Selection for General Response Types (2024); 2) Invariant Causal Prediction for Nonlinear Models (2018); 3) Causal inference by using invariant prediction: identification and confidence intervals (2016).**

   Thank you for your suggestion. While the main focus of this study is indeed on identifying the

causal parents of the runoff signatures, one of the broader objectives is to construct interpretable DAGs that reflect the underlying hydrological processes. These graphs aim to provide a more comprehensive understanding of the system beyond the immediate causal predictors of runoff signatures. We also examined model performance across clusters defined by climate, soil, geology, topography, and vegetation categories. The availability of DAGs allowed us to explore the causal relationships among variables within each category.

We appreciate the recommendation to consider Invariant Causal Prediction (ICP). We have explored the use of ICP as part of our preliminary analysis; however, we found that applying it meaningfully in this context, especially given the non-linear dependencies in hydrological data, requires further investigation and careful adaptation, which is beyond the scope of the current study. Nonetheless, we agree that ICP is a promising approach, and we included a discussion of its potential, highlighting it as an avenue for future research. We discussed this method as a potential approach for future work in the context of our study in the Discussion section.

5. **Furthermore, you are using GAMs without specifying which GAMs you are using. Which link function do you use? Which family of distribution is assumed for the outcome y? Generalized linear models are then useful if your outcome is discrete (Poisson, ...), binary (logistic, c-log-log, ...), or continuous but only positive valued (exponential, gamma, ...). I see why you are using the additive components (splines), but I miss some more information about the model specification. Without this information, reproducibility is not possible.**

   Thank you for your comment. We agree that more details on the model specification are important for clarity and reproducibility. In our study, we used Generalized Additive Models (GAMs) implemented via the *mgcv* package in R. Specifically, we used cubic regression splines (*bs = "cr"*) for the smooth terms. The outcome variable is continuous, and we used the default identity link function with a Gaussian error distribution (*family = gaussian()*). The GAMs were fitted using Restricted Maximum Likelihood (*REML*) to estimate the smoothing parameters. We explained the model's specifications in Section 3.3.2.

6. **A further point to note is that while the PC algorithm basically only uses (partial) correlation, testing linear dependency, the subsequent use of additive models assumes a potentially non-linear association between X and the mean of Y. This is not necessarily a flaw in your approach, but it is crucial to be aware that while the PC algorithm relies on (partial) correlations, which inherently assess linear relationships (keeping in mind the fundamental connection between linear regression and correlation), the application of GAMs (the additive part of it) implies that a non-linear relationship between the predictors and the mean of y is assumed. What you could try is something like using regression modeling with cubic spline as a heuristic test of conditional independence. See for an data example: "Data-Driven Model Building for Life-Course Epidemiology (2021)".**

   Thank you for this helpful point. We fully agree that the PC algorithm, in its standard form, relies on (partial) correlations to test for conditional independence, which inherently assumes linear relationships among variables. However, given the known nonlinearity of hydrological processes, we tried to address this limitation in two ways.

   First, for the conditional independence tests during causal discovery, we used a non-parametric approach based on mutual information with the James-Stein shrinkage estimator (Hausser and Strimmer, 2009), as implemented in the *bnlearn* package. This method does not assume linearity and is better suited than partial correlation for capturing the complex dependencies commonly found in environmental and hydrological systems.

   Second, following your suggestion, we used cubic spline regression to further explore potential non-linear associations and assess the robustness of the inferred edges. This served as a heuristic test to evaluate the significance of connections between adjacent nodes in the learned DAG, aligning with your suggestion. Using these spline-based models, we assessed the significance of relationships between adjacent variables in the DAGs by reporting the p-values from likelihood ratio tests for each edge.

   We clarified these methodological choices in Section 3.2 of the manuscript, emphasizing that our approach complements the PC algorithm by addressing its linearity assumption both during and after the structure learning phase.

References

Hausser, J. and Strimmer, K.: Entropy inference and the James-Stein estimator, with application to nonlinear gene association networks., Journal of Machine Learning Research, 10, 2009.

Petersen, A. H., Osler, M., and Ekstrom, C. T.: Data-Driven Model Building for Life-Course Epidemiology, AMERICAN JOURNAL OF EPIDEMIOLOGY, 190, 1898–1907, https://doi.org/10.1093/aje/kwab087, 2021.

Zuk, O., Margel, S., and Domany, E.: On the number of samples needed to learn the correct structure of a Bayesian network, arXiv preprint arXiv:1206.6862, 2012.

List of Changes in the Manuscripts

The list of changes in the manuscript is as follows:

Table 1: Changes in the manuscript.

Section	**Changes**
Section 1 (Introduction)	Add a short introduction on the PC algorithm, and remove the section about the variable selection.
Section 2 (Data)	add more explanation about how the data is used for clustering, causal discovery, and prediction. Modify Table 1 to clarify the data used in the study.
Section 3 (Methods)	Add explanation about causal discovery, including causal discovery with the PC, background knowledge, and PC implementation. Add more information about GAM specification.
Section 4 (Results)	Update the whole Section 4.2, 4.3, and 4.4 according to the new results.
Section 5 (Discussion)	Update this section according to the new results.
Section 6 (Conclusion)	Update this section according to the new results.
Appendix A	Update the tables in this section according to the new results.
Supplementary Materials	Update sections 2 and 3, according to the new results.