Karlsruhe Institute of Technology

Karlsruhe Institute of Technology

Causal Networks for Climate Model Evaluation: The Role of Hyperparameters and Metrics in the Context of Optimization

Bachelor thesis
by

Kevin Sinigalia

Institute of Theoretical Informatics

Reviewer:	TT-Prof. Dr. Peer Nowack
Second Reviewer:	T.T.-Prof. Dr. Pascal Friederich
Advisor:	TT-Prof. Dr. Peer Nowack
Begin:	07.12.2023
Submission:	30.03.2024

Declaration of Authorship

Ich versichere wahrheitsgemäß, die Arbeit selbstständig verfasst, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde sowie die Satzung des KIT zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung beachtet zu haben.

Karlsruhe, Month dd, yyyy

Karlsruhe, 03.30.2024, Kevin Sinigalia

Kevin Sinigalia

Approved as examination copy:

Karlsruhe, Month dd, yyyy

Contents

Abstract

Reconstructing the causal relationships of observed phenomena through traditional methods, e.g. experiments in the real world, is not always feasible or too expensive. One alternative, due to the increasing availability of data, is the use of causal discovery methods, which infer causal relationships on the basis of the observed data. The focus of this work is on the application of the causal discovery methods PCMCI and PCMCI$^+$ in the context of climate data sets (Big Data Concept), such as model data from the Coupled Model Intercomparison Project phase 6 (CMIP6) and observational data. The goal of this work is to find the best combination of causal discovery methods (i.e., PCMCI, PCMCI$^+$) and their hyperparameter settings to most cleanly separate causally reconstructed networks, which we know should be similar or dissimilar. Established benchmarks are to find clear skill to separate CMIP6 model groups that share a common model development background (e.g., all models from the UK Met Office) compared to the rest.

PCMCI$^+$ offers the advantage over PCMCI of inferring contemporaneous causal links, thereby providing more links to distinguish between different climate models. On the other hand, PCMCI benefits from an additional hyperparameter to filter the links for significance at the end. For this project's methodology comparison, PCMCI proved more suitable, particularly demonstrating improved performance with higher volumes of data and number of components. Subsequent focus was placed on finding the hyperparameter settings (data-specific and method-specific settings) to maximize the disparity between the climate models. It was found that applying the method to more data yields better performance in this context. Hyperparameters need to be chosen notably small to filter the climate model-specific links, which form the basis for distinguishing between the climate models. Initially, only the F_1-score served as the comparison metric. After the most discriminative hyperparameter setting was established, the networks learned from CMIP6 data are compared with a network learned from observations, to evaluate which models better fit reality.

To further improve the discrimination between the climate models significantly, a subnetwork was inferred, where comparison of causal networks was then conducted solely within this subnetwork. Only causal links where choosed, that contribute most to climate model consistency and climate model discrepancy. In addition to the F_1-score, other metrics were used to compare the causal networks, aming for further increasing the discrimination between the climate models. The metrics used range from solely focusing on the resulting matrices of the PCMCI($^+$) methods to including both matrices and graphs, as seen in a modification of the F_1-score, or pure graph comparison metrics, with the modified F_1-score using a penalty term for wrong links performing best.

Zusammenfassung

Die Rekonstruktion der kausalen Zusammenhänge von beobachteten Phänomenen durch traditionelle Methoden, z. B. Experimente in der realen Welt, ist nicht immer durchführbar oder zu teuer. Eine Alternative ist aufgrund der zunehmenden Datenverfügbarkeit der Einsatz von Kausalerkennungsmethoden (Causal Discovery), welche auf der Grundlage der beobachteten Daten auf kausale Zusammenhänge schließen. Der Schwerpunkt dieser Arbeit liegt auf der Anwendung der Causal Discovery Methoden PCMCI und PCMCI$^+$ im Kontext von Klimadatensätzen (Big Data Konzept), wie z.B. Modelldaten aus dem Coupled Model Intercomparison Project Phase 6 (CMIP6) und Beobachtungsdaten. Das Ziel dieser Arbeit ist es, die beste Kombination von Causal Discovery Methode (d.h. PCMCI, PCMCI$^+$) und deren Hyperparameter Settings zu finden, um die rekonstruierten kausalen Netzwerke, bei denen klar ist, dass sie ähnlich oder unähnlich sein sollten, möglichst sauber zu trennen. Benchmarks sind die CMIP6-Modellgruppen, die einen gemeinsamen Hintergrund in der Modellentwicklung haben (z. B. alle Modelle des UK Met Office), die von den anderen Gruppen klar unterscheidbar sein sollen.

PCMCI$^+$ bietet gegenüber PCMCI den Vorteil, dass zeitgleiche kausale Zusammenhänge abgeleitet werden können und somit mehr Zusammenhänge zur Verfügung stehen, um zwischen verschiedenen Klimamodellen zu unterscheiden. Andererseits profitiert PCMCI von einem zusätzlichen Hyperparameter, mit dem die kausalen Links am Ende auf Signifikanz gefiltert werden. Für den Methodenvergleich im Rahmen dieses Projekts erwies sich PCMCI als geeigneter, da es insbesondere bei größeren Datenmengen und höhere Anzahl an Komponenten eine bessere Leistung zeigte. Anschließend wurde der Schwerpunkt darauf gelegt, die Hyperparameter-Einstellungen (daten- und methodenspezifische Einstellungen) zu finden, um die Diskrepanz zwischen den Klimamodellen zu maximieren. Es wurde festgestellt, dass die Anwendung der Methoden auf mehr Daten in diesem Zusammenhang eine bessere Leistung erbringt. Die Hyperparameter müssen besonders klein gewählt werden, um die klimamodellspezifischen kausalen Links herauszufiltern, welche die Grundlage für die Unterscheidung zwischen den Klimamodellen bilden. Ursprünglich diente nur der F_1-Score als Vergleichsmaßstab. Nachdem die Hyperparameter-Einstellung festgelegt wurde, welcher zur höchsten Unterscheidbarkeit der Klimamodelle führt, wurden die aus aus den CMIP6-Daten gelernten Netzwerke mit einem aus Beobachtungsdaten gelernten Netzwerk verglichen, um zu bewerten, welche Modelle besser zur Realität passen.

Um die Unterscheidung zwischen den Klimamodellen noch weiter zu verbessern, wurde ein Subnetzwerk abgeleitet und der Vergleich der kausalen Netzwerke wurde ausschließlich innerhalb dieses Subnetzwerkes durchgeführt. Es wurden nur kausale Links ausgewählt, die am meisten zur Selbstkonsistenz der Klimamodelle und zur Diskrepanz zwischen den Klimamodelle beitragen. Neben dem F_1-Score wurden auch andere Metriken zum Vergleich der kausalen Netzwerke verwendet, um die Unterscheidung zwischen den Klimamodellen weiter zu verbessern. Die verwendeten Metriken reichen von der ausschließlichen Verwendung der resultierenden Matrizen der PCMCI($^+$)-Methoden, bis hin zur gemeinsamen Einbeziehung von Matrizen und Graphen, einer Modifikation des F_1-Score, oder reinen

Graphen-Vergleichsmetriken, wobei der modifizierte F_1-Score mit einem Strafterm für falsche Links am besten abschnitt.

1. Introduction

What are the various teleconnections between globally distributed components and how strong are they? The first steps involved in solving a problem like this are to understand the underlying mechanisms and especially what it is caused by. Identifying these causal relationships of observed phenomena is a relevant topic in all fields of science and is still a challenge. The traditional approach is through interventional discovery, where cause-effect relationships are discovered through intentional interventions in a system. However, this is not always feasible, can involve high costs or is unethical, especially in large complex dynamical systems, such as our climate system. The improvements in data collection and increasing amounts of data over the last decade have created the opportunity for causal discovery, that aims to reconstruct the underlying causal relationships based only on the observed data. One approach is to use constraint-based methods, which infer causal relationships by conducting statistical tests [1].

The focus of this work is on the application of the causal discovery methods PCMCI and PCMCI$^+$ [2] in the context of climate data sets (Big Data Concept) such as model data from the Coupled Model Intercomparison Project phase 6 (CMIP6) and observational data. The methods are applied to PCA-Varimax dimension-reduced sea level pressure data, as a proxy of atmospheric dynamical couplings in CMIP6 models and observations and thus a number of causal networks for comparison are learned. The objective is to achieve more accurate projections for climate phenomena e.g. as climate change. Several models exist for this purpose, which capture reality to varying extents due to uncertainties in their physical process representations. The question here is whether, by applying the methods to the different model data, it is possible to identify which models are related to each other and generate similar results, as well as which results are closer to reality (comparison with the observation data as ground truth).

The question arises, which of the methods can reveal the discrepancies between various non-related climate models best. PCMCI$^+$ offers the advantage over PCMCI in inferring contemporaneous causal links. On the other hand, PCMCI benefits from an additional hyperparameter, allowing for further filtering of the significant links defined thereby. To distinguish between climate models, it is crucial to retain as many model-specific links as possible while filtering out noise to highlight discrepancies.

Further goal is to most cleanly separate causally reconstructed networks, which we know should be similar or dissimilar, where also the respective hyperparameter setting is needed. Established benchmarks are to find clear skill to separate CMIP6 model groups that share a common model development background (e.g. all models from the UK Met Office) compared to the rest [3]. Hyperparameters to be tuned include the number of principal components used to construct the networks (e.g. 20, 40, 60, 80 100), the number of years (i.e. samples) available to fit the networks (e.g. 10, 20, 30, 40 years), and the significance threshold used to define true causal links. An alpha significance threshold parameter is needed to define method-specific links in the resulting networks, separating detected links

as cleanly as possible from those generated by noise as mentioned above. After identifying most of the model-specific links through the appropriate hyperparameter settings, another question arises: which climate model is closest to the observational data? In this regard, the causal graphs resulting from the identified hyperparameter settings for the climate models will be compared against the causal graph computed from the observational data using the same settings.

Another question is the choice of metrics to distinguish between the various causal graphs of the climate models. The causal graphs that represent the causal relationships, the strength of the links, or the p-value of each link can be considered individually or in combination. The goal is to find a metric that strongly weights the differences in the networks to highlight them, without compromising the self-consistency within a climate model. In other words, the comparison metric between networks from non-related climate models should generate a clearly distinguishable value, unlike networks from the same or related climate models. After an introduction to causal discovery and the methodology, these questions will be addressed.

2. Background

2.1. Causal Discovery

Why is determining causality so important? Correlation or linear regression can be used to determine potential influences between variables. However, no explanation of the causes is given and it does not provide not sufficient information for solving many problems, since it only indicates that two or more variables vary in respect to each other. A more specific relationship between variables, which also helps to describe cause-effect relationships, is causation. Causation applies when, as a result of the change in one variable or the occurrence of an event, the value of another variable changes. Thus, V^i is the cause of an effect on V^j if V^j changes in response on changes in V^i. The notion of causality thus enables analysis of how a system would respond to an intervention, which is achieved by (randomly) assigning values to a single variable. Causality is represented mathematically via Structural Causal Models [1], that can be illustrated by a causal graph (Figure 2.1). The traditional approach is by using planned or randomized experiments. In these experiments, variables are randomized or intentionally modified and the resulting changes in other variables are used to reconstruct the structure of the causal relationships between them. These are obviously sometimes resource intensive, unethical or infeasible, especially in large complex climate systems. Recent data collection in the past decade enabled causal discovery that can be used where causal discovery through experiments fails. Causal discovery is a method to infer a causal structure from given observed data[4].

The considered approach to causal discovery is to learn a *directed graph* from (conditional) independencies in the given data, which are statistically tested. For time series data, a directed *time-series graph* is learned. The nodes in a time series graph represent the variables at different time lags and the edges a causal link. In a finite time series dataset, every causal discovery method faces the trade-off between too many false positives (FP), too many false negatives (FN) and too few true positives (TP). False positives are links that describe causal relationships that do not actually exist and are thus incorrectly inferred. False negatives represent causal links that exist but were not identified as such. True positives are the actual causal relationships correctly identified as corresponding links. A causal link $X^i_{t-\tau} \to X^j_t$ exists if $X^i_{t-\tau}$ is *not* conditionally independent of X^j_t given the past of all variables, formally defined by $X^i_{t-\tau} \not\perp\!\!\!\perp X^j_t \mid X^-_t \setminus \{X^i_{t-\tau}\}$. $\not\perp\!\!\!\perp$ denotes the absence of a (conditional) independency with $X^-_t = \{X^j_{t-\tau} : j = 1, ..., N, \tau = 1, 2, ...\}$. The causal discovery theory states that when testing for conditional independence, it is sufficient to condition only on the parents of the variable being tested. This follows from the causal Markov condition [5]. Conditioning on the parents also leads to a higher detection power of true causal links [6]. In causal discovery, the term "parents" refers to variables that are direct causes of a given target variable. In a causal model, it is assumed that each variable is influenced by its direct predecessors i.e. "parents". For example, in figure 2.1, $\mathcal{P}(X^4) = \{X^3, X^1\}$ or $X^3 \in \mathcal{P}(X^2)$ [7].

For the models used, in addition to other specific assumptions explained later to their

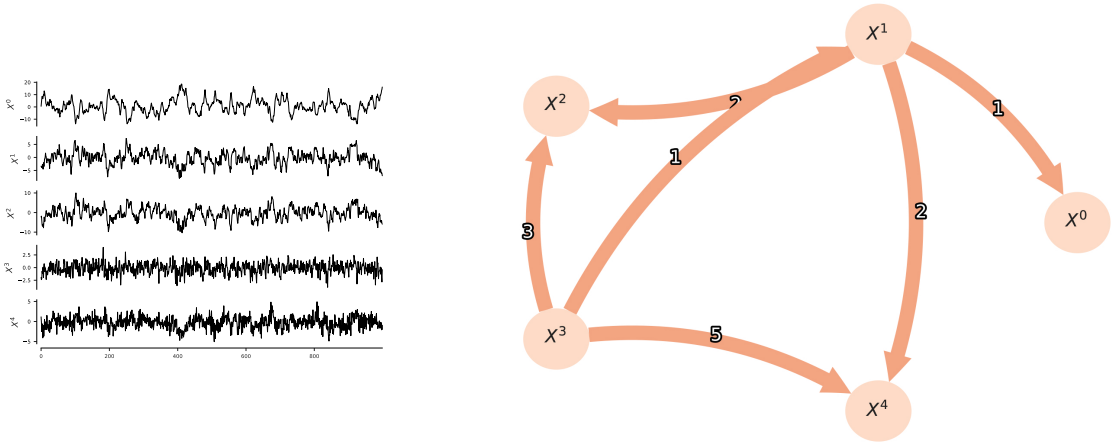


Figure 2.1: The left figure depicts a dataset with time series data from 5 variables X_i, where the data from variable X_i is in row i. The focus lies in right figure, that shows a causal graph inferred by using this dataset only. The nodes represent the different variables, and the edges represent the causal relationships. The edge labels describe the time lag τ between cause and effect. A link from X^i to Y^j with a time lag of τ represents a cause-effect relationship between X_t^i and Y_τ^j. Due to assumptions explained later and for simplicity, arbitrary values can be used for t without adding additional nodes for each specific t.

context, common assumptions include the *Causal Markov Condition* as already mentioned above, *Faithfulness* and causal *stationarity*. The Causal Markov Assumption states that the probability joint distribution of a set of variables satisfies the Markov property of the underlying causal graph, meaning it fulfills the conditional independence relationships implied by the causal graph's Markov property. In the other direction, the Causal Faithfulness Assumption states that all conditional independence relationships maintained by the joint distribution are implied by the causal graph [8]. Causal stationarity refers to the property of a system where the causality between variables remains constant over time. This means that the causal relationships between variables remain the same regardless of the duration or timing of the observation. Or in other words, if the causal link $X_t^i \to X_{t-\tau}^j$ is found, causal stationary implies that this causal link is also valid for all other available timestamps t for the same variables and same time-lag τ [6]. By applying the stationary assumption a the large time-series graph could be reduced to the smaller, clearer one in figure 2.1.

The conditional independence tests to determine the independence between two variables $X_{t-\tau}^i X_t^j \mid S$ given a conditioning set S, denoted as $CI(X_{t-\tau}^i, X_t^j, S)$, play an essential part of the methods used in this study. The focus was on the linear conditional independence test called ParCorr, although theoretically other conditional independence tests provided by the TIGRAMITE library could have been used [2]. ParCorr is based on partial correlations followed by a subsequent t-test to assess the dependencies between variables. The underlying assumption for the estimation of dependencies between variables is as follows [7]:

$$X^i = S\beta_{X^i} + \epsilon_{X^i}, \quad X^j = S\beta_{X^j} + \epsilon_{X^j}$$

with the estimated coefficients β and Gaussian noise ϵ, leading to the following residuals

with β estimated as $\hat{\beta}$:

$$r^{X^i} = X^i - S\hat{\beta}_{X^i}, \quad r^{X^j} = X^j - S\hat{\beta}_{X^j}$$

ParCorr measures the strength of potential relationships between two variables by applying partial correlations, while removing influences from all other variables in S by using least-squares regression(2.1). The independence of the resulting residuals (2.1) is determined using Pearson correlation and a t-test. If the variables are independent, the Pearson's correlation between the two residuals should be close to zero. The t-test determines whether the correlation of the two residuals significantly deviates from zero. A significant t-value would suggest that the residuals are not independent of each other, implying a possible causal relationship between the variables.

2.1.1. PCMCI

The name "PCMCI" for the algorithm comes from the combination of two main components, which are explained below: the PC algorithm and the MCI tests.

In this study, for PCMCI only PC-α and MCI-α were varied, as they are the main parameters (together with τ_{max}). PC-α represents a significance level used to determine the causal parents in the initial step, as explained in the following. MCI-α is then used at the end to perform thresholding, selecting the relevant links defined in this way. τ_{max} was set to 10, defining the maximum time delay assumed for causal relationships between two variables. The default settings were used for the remaining parameter values. The output of PCMCI consists of the *graph matrix*, the *p matrix*, and the *value matrix*. The focus of the output lies on *p matrix* and *value matrix*. The *p matrix* is three-dimensional, where an entry at index i, j, τ indicates the statistical significance of the causal link $X_i^t \to X_{t-\tau}^i$. The corresponding entry in the *value matrix* at the same index represents the strength of the link. In the three-dimensional *graph matrix*, the links are graphically represented, obtained by thresholding with MCI-α on *p matrix*.

To compute Conditional Independence of variables as efficient as possible, it is crucial to keep the conditioning set, denoted S, as small as possible. In S, only the relevant variables should be included, as they are conditioned on when testing for conditional independence (2.1). Unnecessary variables in S consume more computational resources and add noise. Following the Markov condition, it is sufficient to condition only on the causal parents of the variables that are being tested for independence. In PCMCI, the first step is thus the PC-step, aimed at determining these causal parents. The PC-step is a variation of the PC algorithm [9] adapted for PCMCI purposes.

In the first part of the PC-step, the preliminary parents of all variables are initialized as $\widehat{\mathcal{P}}(X_t^i) = X_t^-$, with $X_t^- = \{X_{t-\tau}^j : j = 1, ..., N, \tau = 1, ..., \tau_{max}\}$. In the first iteration with $p = 0$, all variables are removed from $\widehat{\mathcal{P}}(X_t^i)$ that have been tested as independent from $X_{t-\tau}^i$ without needing to condition on other variables (conditioning set of dimension $= p = 0$). In other words, all variables $X_{t-\tau}^j$ are removed from $\widehat{\mathcal{P}}(X_t^i)$ if the null hypothesis for the unconditional independence test $X_{t-\tau}^j \perp\!\!\!\perp X_t^i$ cannot be rejected at significance level $\alpha =$ PC-α (e.g. all uncorrelated variables). In the second iteration with $p = 1$, conditioning sets of dimension 1 are utilized to conduct conditional independence tests. In this process, the conditioning set $\mathcal{P}$ of the strongest p drivers with the highest dependency in P is selected by sorting the variables in $\widehat{\mathcal{P}}(X_t^i)$ by their absolute test statistic value. Variables $X_{t-\tau}^j$ identified as independent of X_t^i, when testing the null hypothesis $X_{t-\tau}^j \perp\!\!\!\perp X_t^i \mid \mathcal{P}$, are then removed from $\widehat{\mathcal{P}}(X_t^i)$. Subsequently, in the third iteration with $p = 2$, conditioning

sets of dimension 2 are utilized, conditioning on the two strongest drivers with the highest dependency from $\widehat{\mathcal{P}}(X_t^i)$ in the previous iteration. This process continues until no new conditioning sets are discovered. Typically, the loop converges and identifies the relevant conditioning sets for the variables. These conditioning sets are likely to contain the true causal parents but may also include false positives. Both aspects can be controlled through the parameter *pc-alpha* [6].

In the following MCI step, MCI tests are performed iterating through all pairs

$$(X_{t-\tau}^j, X_t^i), i = 1, ..., N,\ \tau = 0, ..., \tau_{max}$$

with MCI (momentary conditional independence) defined as:

$$\text{MCI: } X_{t-\tau}^j \not\perp\!\!\!\perp X_t^i \mid \widehat{\mathcal{P}}(X_t^i) \setminus \{X_{t-\tau}^j\}, \widehat{\mathcal{P}}(X_{t-\tau}^j)$$

with using the determined conditioning sets $\widehat{\mathcal{P}}(X_t^i)$ and $\widehat{\mathcal{P}}(X_{t-\tau}^j)$ from the previous PC step. In this process, $X_{t-\tau}^j$ must be removed from the conditioning set because it is tested whether $X_{t-\tau}^j$ is a causal parent of X_t^i. For $\tau = 0$, the conditional dependencies between $X_{t-\tau}^j$ and X_t^i are determined. Because both variables have the same timestamp, a dependency cannot be oriented. If $X_{t-\tau}^j$ is in the past of X_t^i, then logically $X_{t-\tau}^j$ would be the causal parent of X_t^i if a dependency is found and vice versa. Without this temporal ordering, it cannot be determined in this context. The MCI step assigns to each link a p-value and a value indicating its strength. The p-value denotes the significance of the link and is derived from the MCI test. The link strength is determined by the test statistic measure used in combination with MCI. In this case, since ParCorr is used as the CI test, the link strength corresponds to the partial correlation value $\in [-1, 1]$. The resulting *p-value* are stored in *p_matrix* and the link strength values in the *val_matrix*, which play a key role for the methods and results mentioned later [7][6] . With the MCI-α parameter, thresholding can be applied to the p-matrix, allowing only links that are statistically significant according to MCI-α to be retained.

2.1.2. PCMCI$^+$

The main advantage of PCMCI$^+$ over PCMCI lies in the orientation of contemporaneous links. Main free parameters of PCMCI$^+$ are, in addition to the assumed maximum time lag τ_{max}, only the significance threshold PC-α. Default values were used for the remaining parameters not mentioned here. The key output values of PCMCI$^+$ are the graph matrix, the val matrix, and the p matrix. Unlike PCMCI, the focus here is on the graph matrix. Similar to PCMCI, the p-matrix and the val-matrix represent the significance and strength of the links, respectively. However, in PCMCI$^+$, the p matrix does not indicate the directionality of contemporaneous links. Lagged links can always be oriented due to time-order considerations. Therefore, the graph matrix plays an important role in PCMCI$^+$ for contemporaneous links, unlike as in PCMCI.

PCMCI$^+$ consists of the PC phase, PC skeleton phase, PC collider orientation phase and PC rule orientation phase, which are explained below. The PC phase, also known as the "condition selection phase," is the same as in PCMCI and serves to estimate a superset of lagged parents for each variable.

In the following PC skeleton phase, the causal graph is reinitialized. The lagged links are derived from the causal parents calculated in the PC step. For all $X_{t-\tau}^j$ in the set of parents of $\mathcal{P}(X_t^i)$, it holds that $X_{t-\tau}^j \to X_t^i$. The contemporaneous links are initialized as "$\circ - \circ$". (undirected) for all pairs of varaibles(X_t^j, X_t^i). For all variables X_t^i, $A_t(X_t^i)$ denotes all

variables X_t^j that are adjacent to X_t^i (contemporaneuos links adjacency). Then, all adjacent pairs $(X_{t-\tau}^j, X_t^i)$ are tested for independence, iterating over the possible contemporaneous conditions $S \in A_t(X_t^i)$ using the MCI test:

$$X_{t-\tau}^j \perp\!\!\!\perp X_t^i \mid S, \widehat{\mathcal{P}}_{t-\tau}^-(X_{t-\tau}^j), \widehat{\mathcal{P}}_t^-(X_t^i) \setminus \{X_{t-\tau}^j\}$$

$\widehat{\mathcal{P}}_t^-(X_t^j)$ denotes the set of lagged adjacencies resulting from the previous PC step [10].

In the following PC collider orientation phase, *unshielded triples* $X_{t-\tau}^j \to X_t^k \circ\!\!-\!\!\circ X_t^i$, where $X_{t-\tau}^j$ and X_t^i are not adjacent, are oriented. In this phase, it is iterated over all unshielded triples. For each unshielded triple, a independence test is conducted using all subsets $S \in \{X_t^j \neq X_t^i \in \mathbf{X}_t, X_t^i \text{ and } X_t^j \text{ are adjacent}\}$ to determine if $X_{t-\tau}^j$ and X_t^i are independent, given conditioning set $Z = (S, \widehat{\mathcal{P}}_{t-\tau}^-(X_{t-\tau}^j), \widehat{\mathcal{P}}_t^-(X_t^i) \setminus \{X_{t-\tau}^j\}$. If the p-value of the conditional independence test is greater than PC-α, S is stored as a separating subset. If no separating subsets exist, the triple is marked as ambiguous. Otherwise, if all possible separating subsets S have been tested, the fraction n_k of separating subsets containing X_t^k is determined. Based on this value, the decision is made whether the triple is oriented or not. In this project, the orientation rule 'majority' is used. Under this rule, the unshielded triple is oriented as a collider $X_{t-\tau}^j \to X_t^k \leftarrow X_t^i$ if $n_k < 0.5$, unoriented if $n_k > 0.5$, and ambiguous if $n_k = 0.5$. If conflicting orientations arise, the corresponding links are marked as conflicting ("$\times-\times$") [10].

In the following PC rule orientation phase, attempts are made to orient unambiguous triples using rules R1-R3 [10].

2.2. Related work

This Bachelor thesis is motivated by a publication in *Nature Communications* (2020), which, however, only used PCMCI for the network comparisons [1], as mentioned in the introduction. In particular, PCMCI only computes causal links with a time lag $\tau > 0$, which excludes many links that could theoretically contribute further to the discrepancy between the climate models. Since the publication only utilized PCMCI, it raises the question of whether other causal discovery methods could increase the discrepancy between the models. Furthermore, the publication only employed limited variations in hyperparameters, thus no optimization for them was conducted. The correct hyperparameter setting has a significant impact on which links are retained or removed, with the best-case scenario being where all model-specific links are retained. Also, a modified version of the F_1-score was used as a comparison metric. The question of whether there exist additional metrics that focus more on the differences between the climate models to increase the discrepancy remained unanswered [1].

2.3. Used Data

The datasets used originate from CMIP6 climate model data and one observation dataset. The Coupled Model Intercomparison Project (CMIP) is a global initiative of climate researchers that compares different climate models in order to develop a better understanding of climate change. It includes a variety of models that simulate the Earth's complex climate system. CMIP6 builds on the previous phases, incorporating new models, improved simulations and expanded research questions. The used CMIP6 dataset consists of 3 to 4 ensembles from each of the climate models. Ensembles of a climate model are collections of simulations i.e. different 'realizations' that are created from the same model with slightly varying input parameters, since in reality measurements of these parameters can also show deviations. For example, a minimal deviation in temperature, as can also be the case in

real measurements, could lead to very different results in a climate model. One reason for creating ensembles is therefore to better understand and quantify the uncertainties in the climate projections. Another reason is that by analyzing different ensembles, researchers can determine which results are robust and which are sensitive to certain model parameters. Overall, different ensembles of a climate model enable a more comprehensive and robust analysis of climate projections [11][12][13].

One of the most important elements of CMIP6 is the large amount of air pressure data generated by a variety of climate models. The air pressure data collected and analyzed within CMIP6 provide valuable insights into various aspects of the climate system. Air pressure is a crucial parameter that influences both short-term weather phenomena and long-term climate patterns. By analyzing air pressure data, researchers can identify patterns and trends that provide important information about the dynamics of the atmosphere and help to understand climatic changes [12].

The methods are applied to PCA-Varimax dimension-reduced sea level pressure data, from 1979-2014 and the months June, July, August only, as a proxy of atmospheric dynamical couplings in CMIP6 models and observations and thus a number of causal networks for comparison are learned.

2.4. Hyperparameters

The selection of hyperparameters significantly influences the outcome and should therefore be chosen appropriately. Among the method-specific hyperparameters, MCI-alpha and PC-alpha, the varied hyperparameters included the number of years from which the data originates and the number of components used. The objective was to determine the optimal hyperparameters setting and how the selection of hyperparameters depends on each other to achieve a high discrepancy between climate models. In addition, a network of links or components can be seen as another hyperparameter on which the climate models are compared.

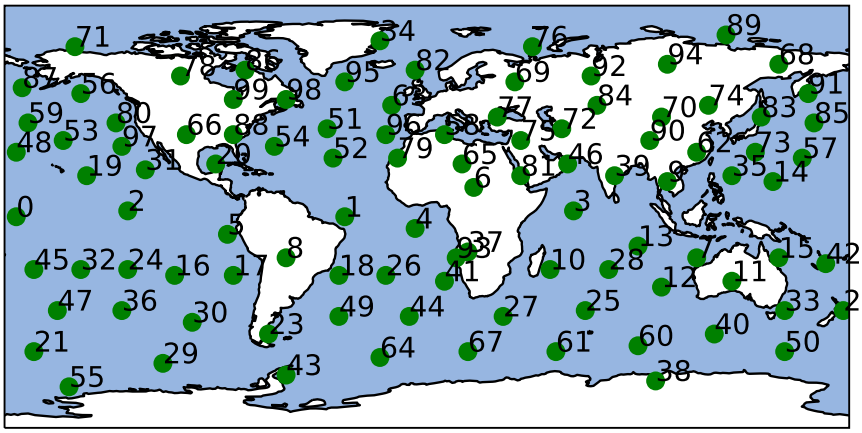


Figure 2.2: The figure displays the geographical locations of the 100 available components.

2.5. Metrics

The choice of metrics also strongly influences the discrimination between climate models. Graphs resulting from causal discovery, value matrices describing link strength, and p-matrices containing the p-value of potential links can all be compared. A combination of these three outputs can also be included into the metric. Depending on how much weight is placed on differences, the distance between climate models may increase or decrease accordingly.

3. Methods

The following section describes the methods used and how they were implemented

3.1. Implementation

The entire project was written in Python 3.10. The Tigramite library was installed and used for applying PCMCI and PCMCI$^+$ [2][1][10]. Prior to using Tigramite, the following most important libraries had to be installed: Cartopy [14]. Paralellization was done with MPI. For computationally intensive tasks that could not be run on a standard personal computer, the bwUniCluster was utilized. Jupyter Notebook was used for visualization, with Cartopy [14] for further visualization. The used versions of the libraries as well as used and developed code can be found on GitHub [15].

3.1.1. PCMCI

The final version of the PCMCI script evolved from its initial version, where parameters such as PC-α, number of components ($nVar$), and MCI-α could be adjusted. τ_{min} was set to 0 and τ_{max} to 10 under the assumption that causal relationships between components do not exist at $\tau > 10$. Partial correlation (ParCorr) was used as a test for statistical independence. For the number of components $nVAR \leq 50$, the first $nVAR$ elements from set

$$\{k_i \mid i \in [0, 100]\} \setminus \{k_{34}, k_{38}\}$$

were used for calculations, sorted in ascending order (in the context of index i) starting from component index 0. For $nVAR > 50$, the components from set

$$\{k_i \mid i \in [0, 100]\}$$

were used in the same manner.

The script iterated over all provided ensembles of climate models, storing the results of each run in a associated folder. Each iteration involved opening the file containing the ensemble's data and constructing the Dataframe required for PCMCI from the provided data and information. Parallelization using MPI facilitated faster runtime, particularly for the PC and MCI steps. In the PC step, components were evenly distributed among processes by using MPI, with each process computing the condition selection i.e. parents for its assigned components. The results were then gathered and sent by using MPI to all processes by the process with rank=0. Similarly, MCI tests were parallelized, with each process conducting tests only on outgoing links of its assigned components. The results were subsequently gathered on process 0 by using MPI again an summed up. The final outcome of applying PCMCI was stored in a dictionary containing the result and relevant parameter information.

In further phases, the script was subsequently extended to enable more efficiency. Instead of single PC-α and $nVAR$ values, the script were added two lists, to which different values

for PC-α and $nVAR$ could be passed. Initially, an additional outer loop was added to smoothly run the previous script for various PC-α values. Then an additional outer loop was added, in which the script was also executed with different $nVAR$ values as well. Results for different $nVAR$ or PC-α values continued to be stored in separate folders to maintain organization.

3.1.2. PCMCI$^+$

The setup and result saving process for PCMCI$^+$ were identical to the PCMCI script. PCMCI$^+$ was given as mode parameter *contemp_conds* and as collider orientation rule the rule *majority* was used. A issue with PCMCI$^+$ arose because in the official parallelized version, only Step 1, i.e., the condition setup step, was parallelized. Runtime analyses, however, revealed that the majority of runtime was consumed in Step 2 (PC skeleton phase) and Step 3 (PC collider orientation phase). Consequently, applying PCMCI$^+$ on a number of components ≥ 25 was not feasible even on the cluster. To enable the application of PCMCI$^+$ to more components, the question arose of how and which steps could be further parallelized, particularly regarding whether this was possible for the PC skeleton phase and the PC collider orientation phase.

The removal of edges in the PC skeleton step occurs in a loop, with the edges to be tested and possibly removed processed in a fixed order. In each iteration, information is computed for subsequent iterations, and edges may be removed. To determine if an edge should be removed, various combinations of potential separating sets are considered in an inner loop, also presented in a fixed order. If a separating set is found that implies no link or causal relationship, the edge is removed, and the respective separating set is stored [10]. It was not feasible to parallelize the outer loop because it would alter the order in which edges are considered and possibly removed, leading to different outcomes due to subsequent loop iterations being executed with different information than in the sequential version. Similarly, the inner loop could not be parallelized because rearranging the order of potential separating sets could result in a different separating set being found. While the sequential version always finds the first occurring separating set in the given list, in the parallelized version, the list would be divided, potentially altering the order in which potential separating sets are considered. Consequently, another separating set might lead to the removal of the edge. As the separating set is also required for further steps in PCMCI$^+$, parallelizing this step could also yield different results.

The PC collider orientation phase consists of several parts. In the first part, the unshielded triples in the graph matrix are collected. Parallelization seemed potentially feasible for this step but was not implemented due to time constraints and the complexity of the code in that step. In the code snippet, only iteration over the graph matrix was performed to determine the unshielded triples, which runs in linear time and thus contributes less to the overall runtime. In the next step, iteration was performed over the individual unshielded triples to attempt their orientation [10]. Since each iteration carried out the potential orientation of unshielded triples without information from previous iterations, there was an opportunity for parallelization using MPI. Unshielded triples and ambiguous triples to be oriented were scattered in separate lists per process, which were then gathered and merged after the step. For simplicity, the separating sets were not considered, as they were not needed for further calculations. Parallelization was only applied to the used orientation rule 'majority'. In the third step, iteration was performed over the unshielded triples to be oriented. If, after considering the current unshielded triple, orientation is to be set to $i \to k_0$ while $k \to i_0$ has already been oriented, both entries in the graph matrix are set to x-x to represent a conflict. Race conditions could occur during the division of unshielded triples to process if both links are considered simultaneously but not yet oriented. Since simple synchronization is not possible with MPI, the second part of third step was not

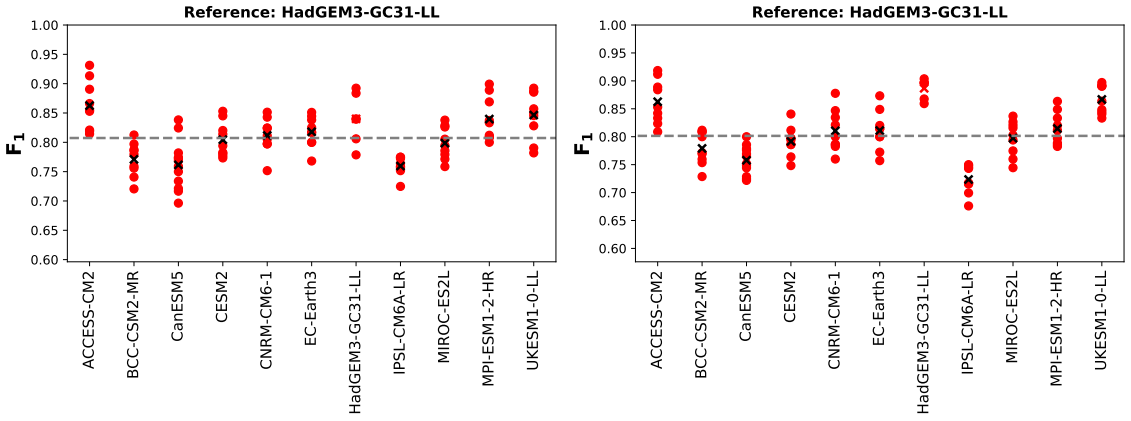


Figure 3.1: This figure shows two plots in which HadGEM3-GC31-LL is used as the reference model. Each ensemble of each model is compared to each ensemble of the reference model with respect to the F_1-score as a metric. the resulting F_1-score is labeled as a dot. The average F_1-score achieved for the ensembles of each model is labeled as a cross. The dashed line describes the average F_1-score across all models. For both plots, PCMCI was applied to different hyperparameters. In contrast to the left plot, the reference model and its two related models stand out from the other models in the right plot, as their networks are more similar and thus achieve a higher F_1-score in comparison. It is therefore clear that the selection of hyperparameters plays a major role in the discrimination of the models.

parallelized. However, if the triples are suitably divided, which was not done in this case due to already linear runtime, parallelization is also possible here.

3.2. Finding optimal hyperparameters

The first objective is to discover how the hyperparameters can affect the discriminability of the different climate models. A slightly modified version of the F_1-score is used as a comparison metric with the following definition:

$$F_1 = \frac{2 * P * R}{P + R}, P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN} \tag{3.1}$$

P stands for *Precision* and R stands for *Recall* (True Positive (TP), False Negative (FN), False Positive (FP)). The following points were used as a reference for greater discriminality:

- Ensembles of similar climate models should achieve a higher F_1-score when compared with each other than when compared with other climate models.
- The F_1-scores for different ensembles of the same model should not differ significantly when compared to each other.

In order to evaluate and compare the different results that arise due to different hyperparameters in the above mentioned regard, two new metrics were introduced ("model" refers to "climate model"):

$$M_1 = avg(\sum_{\text{reference model } i} \sum_{\text{model } j} \max_{\text{e1,e2}}(F1_{i,j,\ e1,\ e2}) - \min_{\text{e1',e2'}}(F1_{i,j,e1',\ e2'})) \tag{3.2}$$

$$M_2 = avg(\sum_{\text{reference model } i} \sum_{\text{non-related model j}} M(i,j)) \tag{3.3}$$

with

$$M(i,j) = avg(\sum_{i' \in [i]} |\overline{F}_{1_{i,j}} - \overline{F}_{1_{i,i'}}|) \tag{3.4}$$

where $F_{1_{i,\ j,\ ei,\ ej}}$ denotes the achieved F_1-score for climate model with index j and ensemble e_i on the ensemble e_j of the reference climate model i, $[i]$ the set of the climate models related to the reference climate model inclusive the reference climate model itself and $\overline{F}_{1_{i,j}}$ the average achieved F_1-score of model with index j on reference model i over all ensembles of both. With regard to the $F_1 - score$, M_1 describes the average of the "spread" within a climate model across all climate models (on fixed reference model) and M_2 the average distance of the climate models to the reference climate model and its related climate models (see Figure 3.2). In order to achieve the best possible discrimination between the models, it is essential to minimize M_1 and maximize M_2. Or in ather terms, the goal is to maximize the quotient $\frac{M_2}{M_1}$ for all reference models combined. A lower (left plot) and a higher (right plot) $\frac{M_2}{M_1}$ value can be seen in figure 3.2.

3.2.1. F_1-score

For fixed hyperparameters on running PCMCI, the F_1-score was computed for all possible combinations of $(reference\ model_i) \times (climate\ model_j) \times (ensemble_k) \times (ensemble_l)$. The $reference\ model_i$ represents the climate model with index i considered as ground truth, while $climate\ model_j$ represents any climate model with index j being compared against. Similarly, $ensemble_k$ ($ensemble_l$) represents the networks of ensemble k (l) of the reference model (climate model). If $i = j$, then $k = l$ is not taken into account, since two identical networks would be compared to each other. The resulting F_1-score for each possible combination as described above was stored in a 4-dimensional array, where the first two dimensions represent the climate model indices, and the next two dimensions represent the corresponding ensembles from both climate models. In that sense, the entry at index (i, j, e, e') is the resulting F_1-score from climate model i and its ensemble e as ground truth vs. climate model j and its ensemble e'.

When calculating the F_1-score, a parameter τ_{min} could be arbitrarily set. The options for τ_{min} were mainly $\tau_{min} = 1$ since links with $\tau = 0$ represent no causal links due to missing orientation. Further parameter was *same sign* to include or ignore the directionality of the link strength. During the calculation of the F_1-score, it was iterated over the two *p_val* matrices ("p-matrix") with thresholding applied by MCI-α to filter the links. If both p-matrices at index (i, j, τ) were $<$ MCI-α, for $same\ sign = True$, it was checked whether both value matrices at the same index had the same sign. If so, then TP was incremented by 1. if not, then FN was incremented by 1. The idea here was to consider links as false if the influence is in different directions. For $same\ sign = False$, TP was always increased by 1, although $same\ sign = False$ was not used. To avoid double counting of unoriented contemporaneous links, only $(i, j, 0)$ entries with $i < j$ were considered in the p-matrix.

For PCMCI$^+$, the function used to calculate the F_1-score was modified from PCMCI, as the interpretation of the p-matrix for contemporaneous links differs in PCMCI$^+$. In PCMCI, only a link $X^i_{t-\tau} \to X^j_t$ with $\tau > 0$ or $X^i_{t-\tau}$o$-$oX^j_t with $\tau = 0$ occurs if (i, j, τ) in the corresponding p-matrix is $<$ MCI-α. However, PCMCI+ does not use an MCI-α parameter for thresholding the p-matrix like PCMCI. Instead, iteration over the graph matrix was performed to calculate the F_1-score. In PCMCI$^+$, there can be three different types of contemporaneous links: $X^i_t \to X^j_t$ as lagged link oriented in one direction, $X^i_t \leftarrow X^j_t$ as lagged link oriented in the opposite direction, X^i_to$-$oX^j_t and $X^i_t \times - \times X^j_t$ where both were potentially interpreted as "unoriented". If only causal links were considered ($\tau_{min} > 0$ or oriented), the unoriented links were ignored, since they only represent correlation. If an oriented link occurred in one reference graph matrix and an unoriented one in the other, it

was counted as a False Negative (FN) since no corresponding causal link could be resolved. The remaining links $\tau > 0$ were treated as in PCMCI.

3.2.2. Varying single hyperparameters

In order to gain an initial understanding of how variations in the hyperparameters affect M_1 and M_2, each hyperparameter was systematically varied, with the remaining hyperparameters being fixed. As the runtime of PCMCI$^+$ takes too much time even on the cluster, the search for the optimal setup was conducted on PCMCI.

3.2.2.1. Varying PC-α

To understand the effects of variations on PC-alpha alone, PCMCI was running with only different PC-α values and fixed hyperparameter settings (MCI-α, number of components), due to long runtimes. The results of each PCMCI run at different PC-α values were stored separately in folders to be used for further research in later phases. Subsequently, for each folder, the F_1-score was computed and saved as a file for the same purpose. The various values for M_1 and M_2 for each PC-α were then visualized in Jupyter Notebook using Matplotlib (3.3, 3.4).

PCMCI and PCMCI+ have the capability to provide a list of PC-α values, allowing for the optimization of the PC-α used for each variable. In this process, the most optimal PC-α from the list is calculated and adopted. However, this functionality was not utilized for this project as it yielded poorer results for the project's objectives. The aim of this work, namely to achieve a high discrepancy between the climate models, has a different context compared to optimizing PC-α per ensemble. In this context, the mentioned optimization approach is not considered [2].

3.2.2.2. Varying MCI-α

In order to find a MCI-α that leads to the best results, the F_1-score was computed on several MCI-$\alpha \in [10^{-1}, 10^{-10}]$. Since the MCI-$\alpha$ value is only used as a threshold at the end to determine which links to retain based on the p-matrix, it was unnecessary to run PCMCI again multiple times for variations in MCI-α. Instead, the p-matrices already calculated for each ensemble under same hyperparameters were reused. These p-matrices were then used to compute the F_1-score, with the F_1-score function additionally using MCI-α as an input parameter to perform thresholding on the p-matrices before calculation. Since the calculations of the F_1-score for different MCI-alpha values are independent of each other, this provided a straightforward opportunity for parallelization. The list of MCI-alpha values to be used was divided among the processes. Subsequently, this result was stored in a dictionary, with the given MCI-α value used as key. For later use, the final result (dictionary), along with the values of other hyperparameters (components, PC-α) used, was saved as a file. For M_1 and M_2, all results were grouped according to same PC-α and the number of components, in order to visualize the effect when varying MCI-α only(Figure 3.3, 3.4).

3.2.2.3. Varying the number of components

To investigate how the number of components affects M_1 and M_2, PCMCI was running with fixed MCI-α and PC-α on the first N components. Due to high runtime, only a subset of $N \in [0, 99]$ were tested. The PCMCI result was then saved as a file for studies in later stages. The visualization followed the same principle as in 3.2.2.1, but for the number of components used (3.3, 3.4).

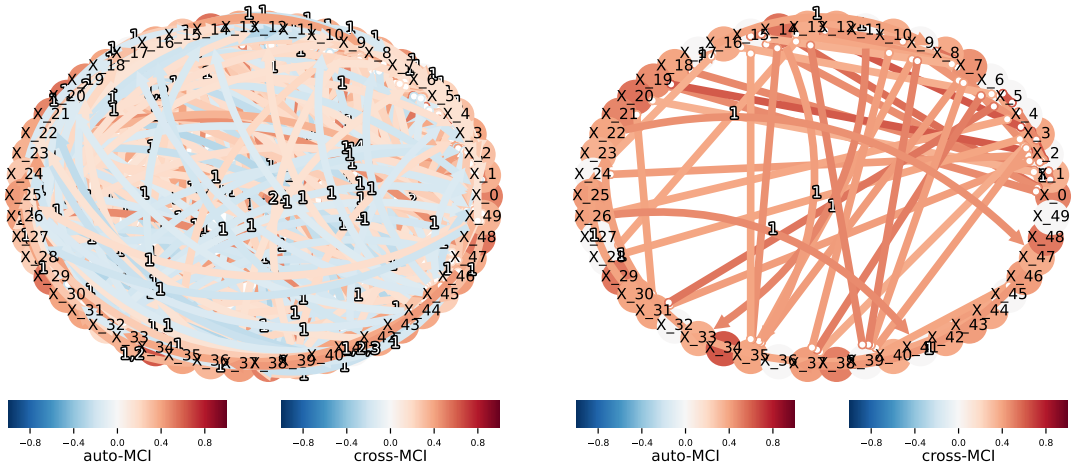


Figure 3.2: To understand the effect of MCI-α, the results from PCMCI (PC-$\alpha = 0.3$, full observed years) were visualized for MCI-$\alpha = 10^{-5}(left)$ and MCI-$\alpha = 10^{-30}$ (right). It is evident that with a low MCI-α, only a fewer links are included. This information allowed narrowing the search space for the optimal MCI-α.

3.2.2.4. Varying the number of observed years

To evaluate the impact of the selection of observed years on the discrepancy, for some given setting of MCI-α, PC-α, and number of components, PCMCI was running on the first $n * 365$ days with $n \in [1, 10]$ and maximum number of days. The F_1-score was then computed for each individual case and then compared. The optimal number of observed years was then adopted for further computations.

3.2.3. Varying multiple hyperparameters

Since searching for optimal hyperparameters in an N-dimensional space along the axes alone may not necessarily lead to the desired outcome, various hyperparameter configurations were tested in the next step. A classic grid search approach, in which all hyperparameter combinations are tested in a predetermined order, was not feasible due to high computational costs. The initial approach involved testing some combinations in a three-dimensional space (number of components, PC-α, MCI-α) to predict trends and thus restrict the number of possible hyperparameter combinations to test.

Due to testing various MCI-α values in the one-dimensional optimization, the search for the optimal MCI-α could be constrained to an interval $[10^{-1}, 10^{-10}]$, since too low MCI-α removes too many links.

The optimal value for $\frac{M_2}{M_1}$ mostly fell between 10^{-3} and 10^{-8}, which also makes sense, as too low MCI-α leads to excessive noise in the networks, while excessively high MCI-α causes the model-specific links to disappear. This can also be seen in figure 3.4, where beyond a certain MCI-α value, M_2 and thus the discrepancy between the climate models models, starts decreasing again. Figure 3.2 clearly demonstrates how with a too low MCI-α, too many links disappear.

For each given combination of PC-α and number of components, a folder was created. PCMCI was then running with these given combinations on all ensembles and the results were saved in the respective folder as files. Subsequently, the F_1-scores were computed as described in 3.2.2.2 for each folder and saved as files. Each file contains the corresponding 4-dimensional F_1-scores array for every MCI-α value in the given list, as well as information

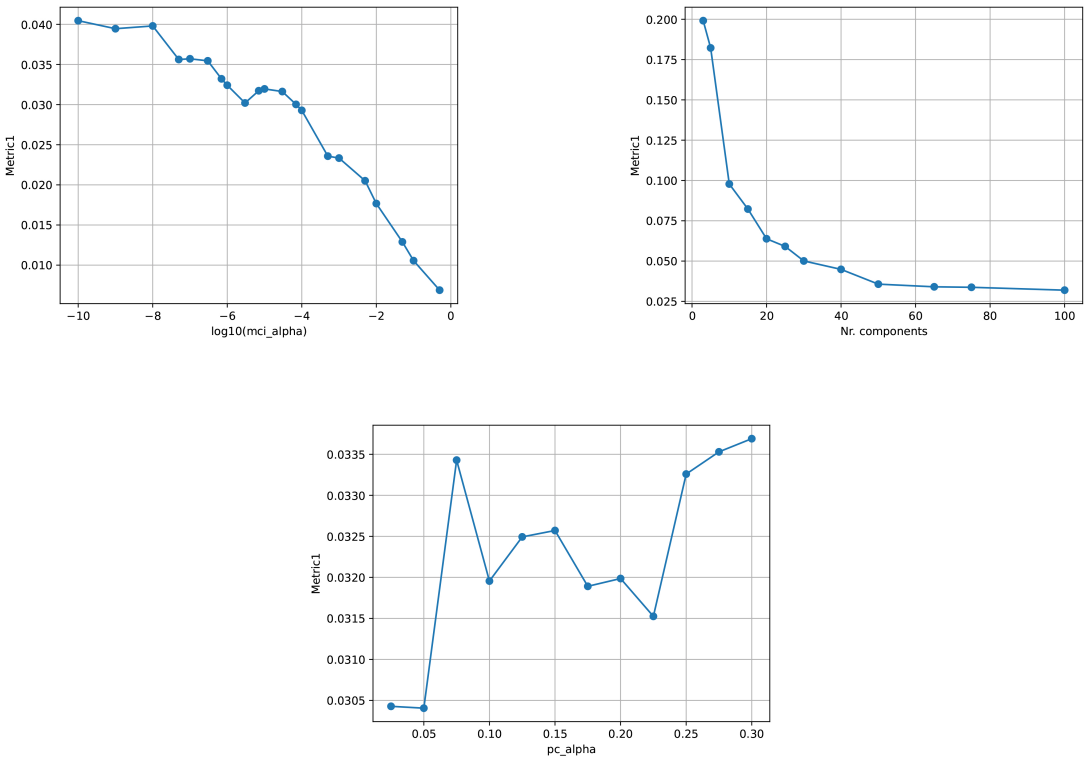


Figure 3.3: The upper left plot visualizes the effect of changes in MCI-α on M_1. For this purpose, PCMCI was running on all 100 components. The MCI-α values were chosen between 10^{-1} and 10^{-10}. The upper right plot visualizes the effect of the number of components used has on M_1. For N components, the first $N-1$ components were selected. Here, PCMCI was also run on these N components and fixed MCI-α value. The plot below visualizes the effect of changes in PC-α on M_1. PCMCI was run again on the first 100 components, as well as fixed MCI-α.

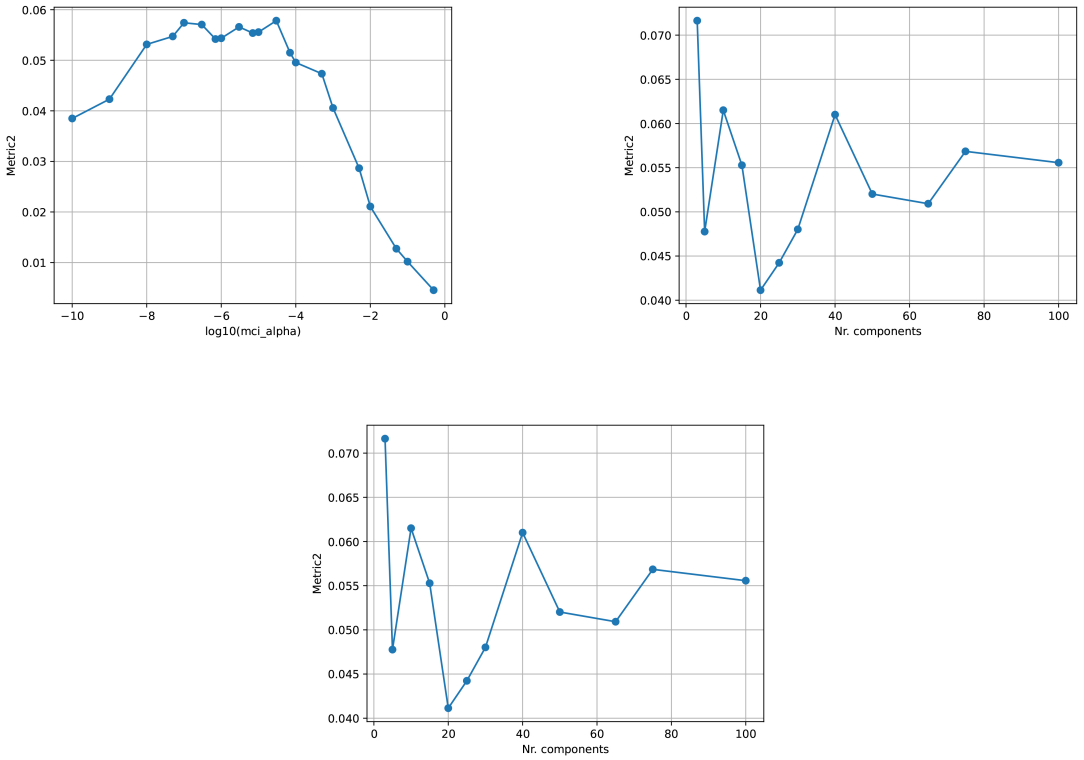


Figure 3.4: The first plot visualizes the effect of changes in MCI-α on M_2. For this purpose, PCMCI was running on all 100 components and PC-α = 0.1. The second plot shows how the number of selected components affects M_2. Here again the first N components were selected and PCMCI was running with PC-$\alpha = 0.1$, as well as a fixed MCI-α. As with M_1, the third plot visualizes the effect of different PC-α values on M_2 with PCMCI running on 100 components and a fixed MCI-α value.

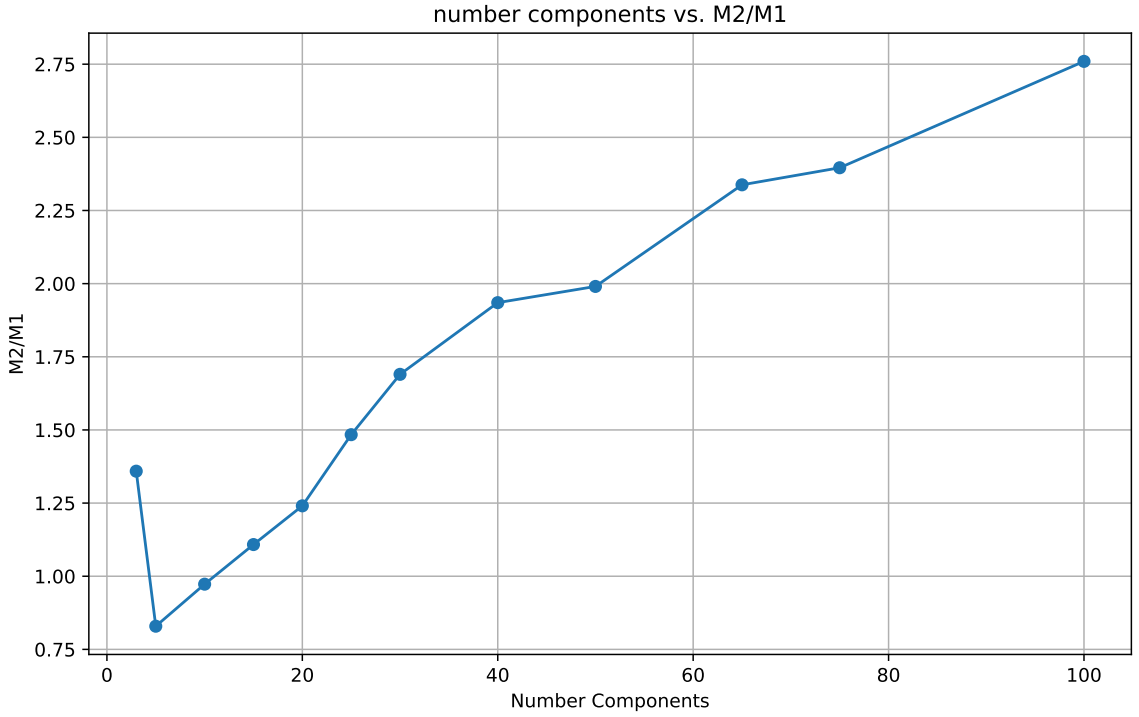
about the other hyperparameters. In terms of runtime considerations, MCI-α plays a relatively minor role, as it is only utilized at the end of PCMCI as a thresholding parameter for the p-matrix or significant links. Consequently, its impact on the final outcome is not that important, since it can be seen as fine tuning at the end which can also be adjusted afterwards. Thus, to improve the comprehensibility of the analysis, the three-dimensional hyperparameter space was reduced into a two-dimensional space. Following algorithm 1, for each combination of PC-α and number of components, the most optimal MCI-α value was computed in terms of $\frac{M_2}{M_1}$ ratio and saved in a dictionary. The discovered MCI-α values confirmed the assumption made above.

Algorithm 1: Calculate optimal M_2/M_1 ratio

```
/* F1-scores contains all F1-scores from the files described above   */
Input  : F1-scores
Output : M2divM1optimums
/* dictionary                                                        */
M2divM1optimum ← {};
for element in F1-scores do
    /* element = (number components, pc-α                           */
    M2divM1 ← None;
    F1-scores_local ← F1-scores[element];
    for MCI-α in F1-scores_local do
        M1 ← metric1(F1-scores_local[MCI-α]);
        M2 ← metric2(F1-scores_local[MCI-α]);
        if M2/M1 > M2divM1 then
            M2divM1 ← (MCI-α, M2/M1);
    M2divM1optimums[element] ← (M2divM1, MCI-α);
```

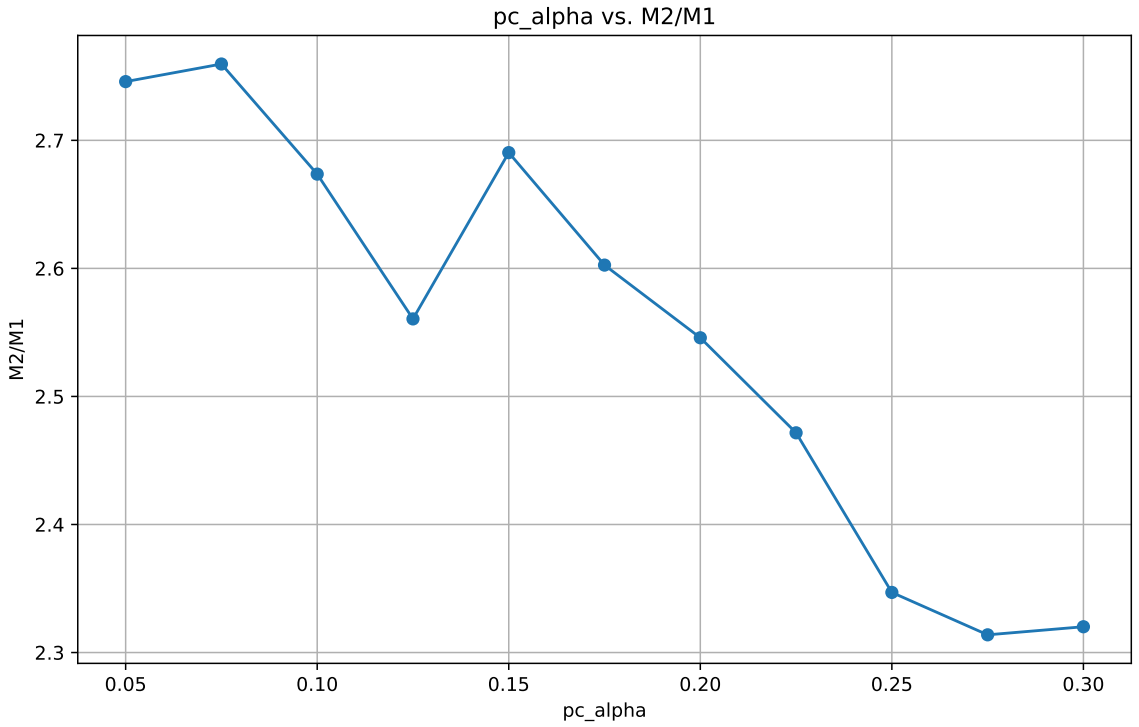
The resulting dictionary from Algorithm 1 was then used for further analysis to examine the interactions between the hyperparameters and to further restrict the search space. This Dictionary was then reduced to only the hyperparameter "number of components". For each number of components, the combinations with PC-α and MCI-α were then filtered to identify those achieving the highest value for $\frac{M_2}{M_1}$ (3.5). Furthermore, the same method was applied to reduce the dictionary (1) to only consider the best values achieved for $\frac{M_2}{M_1}$ for different PC-α. For each given PC-α, the optimal combinations with the other hyperparameters were identified (3.6). The same process was repeated for MCI-α as well (3.7). These three sets of results helped to further constrain the search space for the best-performing hyperparameter settings. From the visualization of the effect from the number of components hyperparameter on $\frac{M_2}{M_1}$, it was observed that a higher number of components was in correlation with a higher $\frac{M_2}{M_1}$ value. This aligns with the assumption that more components provide more information to the different climate models, allowing for more stable and model-specific results. When reducing to PC-α, this assumption was also true, as PCMCI almost always performed better with the full number of components used. Additionally, it was evident that even better results could be achieved with lower PC-α values than used to up to this point. When reducing to MCI-α, it was apparent that MCI-α, except for excessively high or low values, showed little correlation in combination with the other hyperparameters in the context of optimizing $\frac{M_2}{M_1}$. Furthermore, it was observed that in this reduction the best combinations with a number of components equal to 100 consistently yielded higher values in $\frac{M_2}{M_1}$ compared to other combinations.

Due to the reasons mentioned above, as well as the high runtime of PCMCI for many combinations of hyperparameter settings, the further search for the optimal hyperparameter



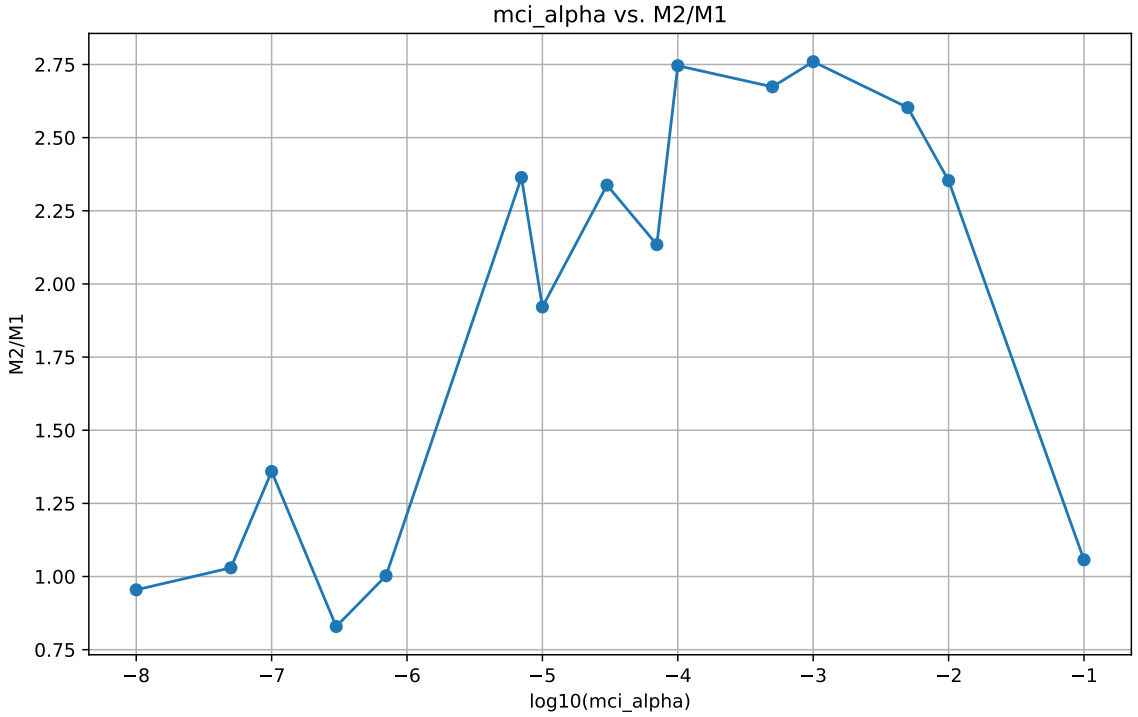
number components	pc_alpha	mci_alpha	M2/M1
3	0.2	1e-07	1.35902
5	0.075	3e-07	0.82918
10	0.05	1e-07	0.97288
15	0.1	0.0005	1.10825
20	0.075	0.001	1.24058
25	0.075	0.005	1.48381
30	0.075	0.01	1.68998
40	0.075	0.001	1.935
50	0.05	0.01	1.99044
65	0.05	3e-05	2.33773
75	0.225	0.001	2.39607
100	0.075	0.001	2.75961

Figure 3.5: The graph illustrates how the selection of the number of components affects the discrimination between the climate models, as described by the $\frac{M_2}{M_1}$ metric. For each investigated number of components, the best-performing combination, in terms of $\frac{M_2}{M_1}$, of PC-α and MCI-α values was used before plotting the corresponding value. The table lists, for each number of components, the corresponding optimal combination of PC-α and MCI-α. The resulting $\frac{M_2}{M_1}$ score was rounded up to the first five decimal places.



pc_alpha	number components	mci_alpha	M2/M1
0.3	100	0.001	2.32014
0.275	75	0.0005	2.31383
0.25	100	0.0005	2.34694
0.225	100	0.001	2.47159
0.2	100	0.001	2.54582
0.175	100	0.005	2.60257
0.15	100	0.001	2.69042
0.125	100	0.001	2.56055
0.1	100	0.0005	2.67363
0.075	100	0.001	2.75961
0.05	100	0.0001	2.74587

Figure 3.6: The graph illustrates how the selection of PC-α affects the discrimination between the climate models. For each PC-α, the most effective combination, in terms of $\frac{M_2}{M_1}$, was filtered from the given number of components and MCI-α, and the corresponding value was plotted. The table below displays, for each used PC-α, the respective combinations and the $\frac{M_2}{M_1}$ score rounded to the first five decimal places.



mci_alpha	number components	pc_alpha	M2/M1
1e-08	3	0.075	0.95454
5e-08	3	0.125	1.02982
1e-07	3	0.2	1.35902
3e-07	5	0.075	0.82918
7e-07	15	0.275	1.00272
7e-06	75	0.05	2.36379
1e-05	50	0.1	1.92128
3e-05	65	0.05	2.33773
7e-05	65	0.075	2.13423
0.0001	100	0.05	2.74587
0.0005	100	0.1	2.67363
0.001	100	0.075	2.75961
0.005	100	0.175	2.60257
0.01	75	0.1	2.35311
0.1	15	0.05	1.05699

Figure 3.7: The plot illustrates the maximum achievable M2/M1 value for each mci-alpha, attained through combinations with other hyperparameters, namely number components and pc-alpha. The table below indicates, for each mci-alpha value, the corresponding number of components and pc-alpha at which this value is reached. M_2 $\overline{M_1 scores have been rounded to the first 5 decimal places.}$

setting focused on the settings with the number of components set to 100. In the subsequent analysis, a series of PC-α values within the interval [10^{-3}, 10^{-45}] were tested until a significant decrease in $\frac{M_2}{M_1}$ was observed, aiming to determine an optimal value.

3.2.4. Searching for the best-performing subnetwork

Another approach to achieve a higher discrimination between the climate models was to identify a subnetwork consisting of a relatively large number of climate model-specific links or components.

3.2.4.1. Searching for the best-performing set of components

The initial strategy involved an iterative process to determine a specific set of components for creating a better performing subnetwork, in terms of maximizing $\frac{M_2}{M_1}$ resulting from PCMCI. The first approach for this strategy consisted of running PCMCI on the first 10 components $\in \{0, 1, ..., 10\}$, with a different component k from that set being omitted each time. Subsequently, for each resulting set of components, the optimal combination, in the context of maximizing $\frac{M_2}{M_1}$, of the given parameters PC-α and MCI-α was computed. The underlying assumption was that the top-performing combination in 10 components $\in \{0, 1, ..., 10\}$ without component k would similarly yield better performing results when scaling to a combination of 20 components $\in \{0, 1, ..., 20\}$, by excluding the same component k than when excluding one of the other components k', as in the lower performing sets of components. This process can be applied multiple times on a higher number of components for each iteration.

The second approach involved deciding which components to retain or discard based on their geographical location and the links between them. The initial concept was to identify the geographical locations of sets of components where PCMCI detects a relatively large number of links, which are implied by each of all ensembles. The objective was to remove components that do not contribute to the climate model discrepancy, thereby proportionally increasing the number of model-specific links.

For this purpose, in the graph matrices from all ensembles resulting from PCMCI (in this case: PC-$\alpha = 0.025$, $N = 100$, MCI-$\alpha = 10^{-4}$), all non-empty entries were replaced with 1, and all empty entries with 0. Subsequently, a new matrix $G_{overlaps}$ was created, where entry $[i, j, k]$ represents the sum of all entries at position $[i, j, k]$ in the converted graph matrices. Thresholding was then applied on $G_{overlaps}$ to identify links where greater than or equal to a with $0 \leq a \leq |ensembles|$ and less than or equal to b with $0 \leq a \leq b \leq |ensembles|$ different ensembles agree. To identify subnetworks that do not contribute to the discrepancy among the different climate models, $a, b = 35$ was chosen (35 ensembles in total). The 100 components were then visualized using the python library Cartopy (3.8).

In this process, PCMCI was applied to different sets of components in specific regions where none or few links were found that do not contribute for increasing discrepancy between the different climate models, such mainly as all components at a latitude ≥ 20. Another approach was to exclude regions, for example, mainly all components with a latitude ≥ -20 and ≤ 20.

The next approach involved computing a value $k_{discrepancy}$ for each component k that describes the ratio of involved links contributing significantly versus those contributing not at all to the discrepancy between the climate models. A link is considered involved on k if it is connected to component k. To compute the links contributing to the discrepancy, thresholding was applied on $G_{overlaps}$. For the links contributing to the discrepancy, a new array $G_{leastOverlaps}$ was created, with

$$G_{leastOverlaps}[i][j][\tau] = \begin{cases} 1 & \text{if } a \leq G_{overlaps}[i][j][\tau] \leq b \\ 0 & \text{otherwise} \end{cases}$$

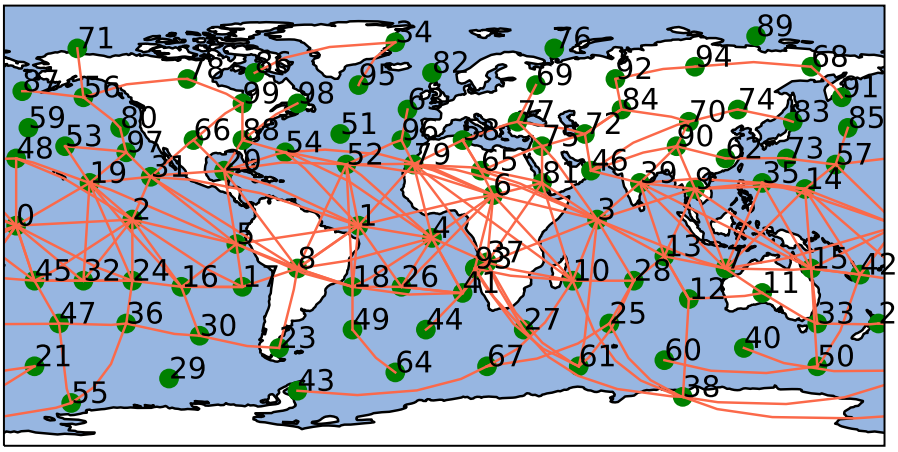


Figure 3.8: The plot visualizes the 100 different components and their geographical locations, represented as nodes. An edge between two components i and j exists if the following condition is fulfilled: $\exists \tau \in [0, \tau_{max}] : G_{overlaps}[i][j][\tau] \geq 35$. The image thus visualizes all edges where the resulting causal graphs of each of the 35 different ensembles agree.

For the links not contributing to the discrepancy, the array $G_{mostOverlaps}$ was created, with

$$G_{mostOverlaps}[i][j][\tau] = \begin{cases} 1 & \text{if } G_{overlaps}[i][j][\tau] \geq c \\ 0 & \text{otherwise} \end{cases}$$

$k_{discrepancy}$ calculates as follows:

$k_{discrepancy^+} = \sum_i \sum_\tau G_{leastOverlaps}[k][i][\tau]$

$k_{discrepancy^-} = \sum_i \sum_\tau G_{mostOverlaps}[k][i][\tau]$

$$k_{discprenacy} = \begin{cases} \frac{k_{discrepancy^+}}{k_{discrepancy^-}} & \text{if } k_{discrepancy^-} \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

a, b and c are further hyperparameters which can be varied. The idea was that with a well-chosen a, the links to be adopted are those that occur in enough graphs of the different ensembles, such as those of the same climate model, to create discrepancy. If a is chosen too small, it will also adopt climate model-unspecific links, which occur only sporadically in the various ensembles. Consequently, M_1 increases undesirably. b should not be chosen too high to avoid adopting links that do not contribute to the discrepancy. On the other hand, c should not be chosen too small to ensure that only non-contributing links to the discrepancy are adopted. With a too low c, undesired climate model-specific links can also be adopted, which then contribute to the discrepancy. To filter the components that contributed significantly to the discrepancy, selection based on $k_{discrepancy}$ could be applied. For instance, components k satisfying $k_{discrepancy} > d$, with $d = 5$ would be chosen. In this case, d was chosen to cover more of the upper range of the number of components, as previous attempts indicated a correlation between a higher number of components and a higher value in $\frac{M_2}{M_1}$. Another reason for limiting the number of components was the runtime of PCMCI combined with further optimization. After selecting the components, the further optimization process followed the procedure outlined in Chapter 3.2.3.

3.2.4.2. Searching for the best-performing subnetwork consisting of links

The idea here was to calculate the F_1-score only over a predefined subnetwork. The advantages of this approach are more realistic calculations, as all given conditions are used since no components are dropped. Another advantage lies in the runtime, as the subnetwork is passed to the F_1-score function, avoiding the need to run PCMCI again each time.

To construct the subnetwork, PCMCI was first applied with the best-performing combination of the hyperparameters (number of components, PC-α, MCI-α). The idea here was to further improve the result after selecting the best hyperparameters setting by performing fine-tuning on the resulting networks. The F_1-score should then be calculated only on discrepancy-generating links to achieve a higher value in $\frac{M_2}{M_1}$. To find the subnetwork, the Algorithms 2 and 3 were used. In a loop, iteration was performed over all climate models. In the first part of the loop, Algorithm 2 was used to compute the intersection, displayed as a new graph $G_\cap$, of links from the graphs of the climate model and its related climate models ensembles. In the second part of the loop, Algorithm 3 was applied on $G_\cap$ to retain only the links that also appear in at least $threshold1$ and at most $threshold2$ other non-related climate models. The result computed by Algorithm 3 was then added to a dictionary at the end of each loop iteration. This dictionary contained the retained links as keys and additional information about the links as values, thus representing the subnetwork. The links were represented as indices, where index $[i, j, k]$ denotes a link from component i to

component j with $\tau = k$. In a grid search, various combinations of $(treshold1, threshold2)$ were tested to find the best performing subnet in the context of maximizing $\frac{M_2}{M_1}$. In the F_1-score calculation, the dictionary was passed as further parameter. During the F_1-score calculation, all links that were not present in the dictionary were skipped.

Algorithm 2: The function **getModelLinks** aggregates all resulting graph matrices from the ensembles of a given *climate model* into a single graph matrix. In the resulting graph matrix, an index entry is non-empty exactly when all graph matrices to be aggregated have the same entry at that index.

```
Function getModelLinks(climate model):
    model_graphs ← all resulting graphs from the ensembles of climatemodel;
    result_array ← empty graph matrix;
    for index in result_array do
        if all graphs in model_graphs have the same entry at index then
            result_array[index] ← entry;
        else
            result_array[index] ← empty string;
    return result_array;
```

Algorithm 3: The function **thresholdModelGraph** applies thresholding on the graph *graph* (which is, in this case, the result from Algorithm 2) of a climate model. In this process, all links are retained if they exist in at least threshold1 and at most *threshold2* of all graphs from the non-related climate models.

```
Function thresholdModelGraph(graph, model_name, threshold1, threshold2):
    otherGraphs ← graphs from all non related climate models;
    relatedGraphs ← graphs from all related climate models;
    // dictionary
    linksDict ← {};
    for index, element in graph) do
        if element ≠ emptystring then
            // Count occurrences in otherGraphs
            count ← 0;
            count ←
                ∑_{graph' in otherGraphs} c // with c = { 1 if graph'[index] == element ; 0 otherwise }
                // Update linksDict if count is within thresholds
            if (count ≥ threshold1) and (count ≤ threshold2) then
                linksDict[index] ← ((count, model_name, element));
    return linksDict;
```

3.3. Metrics

To pursue even higher values in $\frac{M_2}{M_1}$, there is still the option to utilize additional metrics or adapt existing ones. The selected metrics are used to measure, similar to the F_1-score, the distance between two graphs.

3.3.1. Modifying the F_1-score

First step was to begin with modifications to the F_1-score. Instead of incrementing TP, FP and FN by a fixed constant of 1 when comparing links, the initial idea was to include additional information from both *value matrices* to achieve higher discrimination between the climate models. The operation was adjusted from

- $TP+=1$
- $FP+=1$
- $FN+=1$

to

- $TP+=f(refValMatrix[index], valMatrix[index])$
- $FP+=g(refValMatrix[index], valMatrix[index])$
- $FN+=h(refValMatrix[index], valMatrix[index])$.

The challenge was to find discrepancy-increasing functions f, g, and h. The initial idea was to use the sigmoid function. The idea behind using the sigmoid function was that for small differences in the *value matrix*, the networks can be seen as more similar to each other, and thus FN and FP should be incremented by a smaller value, while TP should be incremented by a larger one. Conversely, for large differences, the networks are less similar so FN and FP should be incremented by a higher value, while TP should be incremented by a lower one. A variation of the sigmoid funciton $\sigma(z) = \frac{1}{1+e^{-z}}$ was used:

$f(refValMat[index], valMat[index]) = 1 - \sigma(a * (z - b))$

and

$g, h(refValMat[index], valMat[index]) = \sigma(c * (z - d))$,

with $z = |refValMat[index] - valMat[index]|$, where $refValMat$ refers to the corresponding value matrix of the reference climate models ensemble and $valMat$ to the corresponding of the other ensemble which is evaluated.

The parameters a, b, c and d were chosen to yield values near 0 and 1 for the marginal values of $z \in [0, 2]$. As per the previous idea, the value for TP was subtracted from 1. The new comparison metric value, instead of the F_1-score, was then calculated for different versions of a, b, c and d.

The next function used was similar to $ReLU$. For TP, the idea was to allow a certain tolerance between the two value matrices, treating individual entries as equal, thereby incrementing TP by 1 as before. For larger differences, the incrementing value decreases linearly. For TP:

$$f(z) = \begin{cases} 1 & \text{if } z \leq a \\ -z + 1 + a & \text{otherwise} \end{cases}$$

For FN and FP, a similar version was used with the same reasoning. If the graphs differ in a link but the corresponding entry in the value matrix is similar, it increments by 0. For larger differences in the value matrix, it increments by a higher value. For FN and FP:

$$g, h(z) = \begin{cases} 0 & \text{if } z \leq a \\ -z + 1 + a & \text{otherwise} \end{cases}$$

One further modification was made, where $f(z) = 1 - z^2$ and $g(z), h(z) = z^2$, or $g(z), h(z) = 1$. The idea behind $f(z)$ was that larger differences in the value matrices would lead to

a smaller increase in TP, since z^2 increases quadratically with increasing difference in $refValMatrix$ and $valMatrix$. The idea behind using $g(z), h(z) = z^2$ was similar. When a link is missing and the difference in the value matrices is not large, FN or FP are incremented by a smaller amount. Using $g(z), h(z) = 1$ was expected to result in a lower F_1-score for very different graphs, since they are stronger weighted on average.

This led to the further modification with the idea of focusing more on the differences in the graphs rather than on the similarities, as ultimately the differences contribute to higher discrepancy. TP was increased by a constant of 1, while FN and FP were increased by a chosen constant a. When comparing two graphs, differences are weighted more heavily for $a > 1$ and less heavily for $a < 1$. To increase the discrepancy between the climate models, the focus was mainly on different values of $a > 1$. A is another parameter in which the objective was to find the value that maximizes the discrepancy among given graphs. A series of potential values for a was stored in an array, and the modified F_1-scores for each given a were computed. Each result for a specific a was then saved in its own directory. Penalty terms a where saved as a list, that was iteretated over for more efficency.

3.3.2. Edge Kernel

With the edge kernel, another metric was constructed to measure the distance between two graphs. In this approach, the corresponding $value\_matrix$ from each graph was transformed into new value matrix $value\_matrix'$ by setting all entries at index (i, j, k) to 0 if the entry in the corresponding $graph\_matrix$ at index (i, j, k) is empty. Subsequently, both value matrices were transformed into the Euclidean vector space using the function $t : value\_matrix' \xrightarrow{\text{numpy.flatten}} vector$. The angle between the two vectors was then computed and returned as distance.

3.3.3. Degree Centrality

As one of three different centrality metrics (betweenness, closeness, degree), a distance metric was also constructed from using the degree centrality [16]. Initially, the two graph matrices were transformed into a NetworkX graph with $V = \{(k, \tau) \mid k \in components,\ \tau \in [0, \tau_{max}]\}$ and $E = \{((k, 0), (k', \tau)) \mid graph\_matrix[k, k', \tau] \neq \text{empty string}\}$, where $graph\_matrix$ denotes the respective graph matrix, resulting in G_{ref} and G. The degree centrality was then computed for each node using the NetworkX centrality function. Subsequently, for all nodes v in G_{ref} and v' in G, with $v = v'$, the absolute difference of their degree centrality was computed. The distance was then determined as the average difference.

3.3.4. L_1 norm and variation of the L_1 norm

As an alternative approach, only the resulting value matrices were compared. One approach to evaluate the differences between matrices is by using the p-norms, which provide a measure of the size or distance of a matrix. By applying different p-values, different aspects of the differences between the matrices can be highlighted and analyzed. The p-norm of a matrix is defined as:

$$\|A\|_p = \left(\sum_{i,j} |a_{ij}|^p \right)^{\frac{1}{p}}$$

As first p-norm, the L_1 norm (adapted to the 3-dimensional matrix) with

$$\|A\|_1 = \sum_{ij\tau} |A_{ij\tau}|$$

was considered. In the context of this study where the two value matrices, $refValMat$ and $valMat$, that are being compared. The L_1 norm is utilized to quantify the discrepancies between corresponding elements of these matrices. Applied to $A = refValMat - valMat$, with $refValMat$ being the value matrix from one reference climate model ensemble and $valMat$ the value matrix from the other ensemble, results in:

$$\|refValMat - valMat\|_1 = \sum_{i,j,\tau} |refValMat[i,j,\tau] - valMat[index]| \text{ with } \tau > 0.$$

However, to formulate a distance metric where higher values indicate greater similarity between the matrices, the L_1 norm is adapted by introducing a parameter a:

$$\|refValMat - valMat\|_1 = \sum_{i,j,\tau} a - |refValMat[i,j,\tau] - valMat[index]| \text{ with } \tau > 0.$$

Here, a ensures that the resulting score is always greater than zero, regardless of the differences between the matrices. Moreover, a serves as an affine transformation, leaving the fundamental nature of the metric unchanged. a should not be chosen too small to ensure a score greater than 0 in the end if desired. However, with $a \geq 2$, this is always guaranteed since the value matrix entries are $\in [-1, 1]$. a acts here only as an affine transformation and thus does not alter the results. In the unmodified version, MCI-α does not play a role, as each entry is considered without additionally incorporating the links in the resulting graph.

In a further variation, a penalty term starting at $penalty = 0$ was introduced to include not only the value matrices but also the respective graph matrices. In this case, the L_1 norm was computed only over the value matrix entries at index $[i, j, \tau]$ if both corresponding graph matrices at index $[i, j, \tau]$ had the same entry. If the entry in the graph matrices at index $[i, k, \tau]$ was different, the penalty term $penalty$ was incremented by a value p. As a distance measure, the penalty term was then added to the restricted L_1 norm at the end. p acts as a hyperparameter that can be varied to affect the result. With a suitable choice of p, an even higher level of discrepancy should can be achieved than with the L_1 norm alone. To interpret higher return values as indicating greater dissimilarity, the same process as in 3.3.4 was repeated to modify the L_1 metric with parameter a. Instead of adding the penalty term at the end to the score of the modified L_1 metric, it was subtracted from the modified L_1 metric score. In contrast to the version without the penalty term, the hyperparameter a no longer acts as an affine transformation. When only a is increased while p remains fixed, the effect of the penalty term decreases. However, if only p is increased, the penalty term dominates. This also means that different values for a and p can lead to the same result, as long as they are in the same proportional relation to each other in the context of their influence on the computation of the metric score.

3.3.5. L_2 norm and variation of the L_2 norm

The next norm from the p-norms was the 2-norm, also known as the Euclidean norm or L_2 norm, defined by the following formula (again adapted to the 3-dimensional matrix):

$$\|A\|_2 = \sqrt{\sum_{i,j,\tau} |A_{ij\tau}|^2} \text{ with } \tau > 0 \text{ and } A = refValMat - valMat.$$

The 2-norm provides insights into the overall spread or variability of the elements from matrices. In the context if this study, matrix comparison between both value matrices, the

2-norm was used to quantify dissimilarities and similarities between the matrices. Again, only indices with $\tau > 0$ were considered to include on causal links and not correlations. Similarly to the L_1 norm, the L_2 norm was slightly modified. In the absence of a link in the modified version, the difference was valued as 4, as the entries in the value matrices are from the interval $[-1, 1]$, and the maximum squared difference is thus 4. This time, to simplify thing, no affine transformation and mirroring were applied, so lower achieved values indicate greater similarity, as the L_2 distance between the matrices is also lower. Again, in the unmodified version, MCI-α does not play a role, as each entry is considered without additionally including the links in the resulting graph.

4. Results & Discussion

4.1. Optimization of PC-α, MCI-α, number of components

As assumed in the Methods section, a low MCI-α value leads to a decrease in discrepancy, as too many model-specific links are considered insignificant. Conversely, setting MCI-α too high results in too much noise in the resulting graph, also leading to a decrease in discrepancy which can be seen in Figure 4.1.

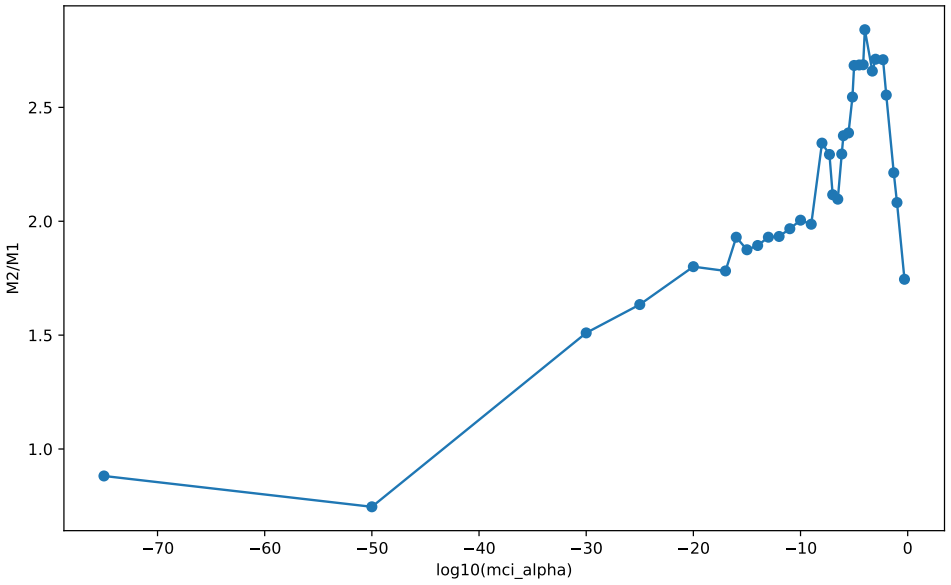


Figure 4.1: The plot displays the performance in the context of $\frac{M_2}{M_1}$ (y-axis) of PCMCI (100 components, PC-$\alpha = 0.0005$, full time-series data) for various MCI-α values. The MCI-α values were chosen in the interval $[0.5, 10^{-75}$ (x-axis).

To capture as many true positive and as few false positive links as possible, the dimensions of conditioning sets and the determined effect size plays a crucial role, as explained below. For a high PC-α value, fewer variables $X^i_{t-\tau}, X^j_t$ are tested independent in the PC-step, since the threshold for rejecting the null hypothesis $X^i_{t-\tau} \perp\!\!\!\perp X^j_t$ is set too high. Since fewer links are removed, each variable tends to have more estimated causal parents. In the following MCI step, a higher proportion of partial correlation in the statistical tests (here ParCorr) can be falsely "explained away" by irrelevant correlations, as they lead to more noise within the test. Consequently, the calculated effect size of the link decreases, reducing the probability of being identified as true causal link and the probability of being mistakenly removed increases. By higher chance, true causal links are now removed, resulting into an increase in false negatives. Therefore, the identified links tend to be limited to the most

robust ones, as they cannot be explained away even with a considerable amount noise. These most robust links appear to become relatively more similar in their difference, both within the ensembles of a climate model and between the ensembles of different models, as the achieved F_1-score across all climate models is relatively low and because of the "similar dissimilarities", the discrepancy is also very low, which can be seen in Figure 4.2. Since the F_1-scores become very low for all models, M_1 and M_2 become also very low, which is visualized in Figure 4.4.

Conversely, a too low PC-α results in conditioning sets with too few variables for the MCI-step, increasing the chance of finding false causal parents. In statistical tests, portions of the partial correlation are often left unexplained, falsely attributing correlation between two variables due to some missing true causal parents. Ultimately, this leads to many false positives and thus increased noise in the found causal networks, including more and more correlations seen as causation. The false positives now mainly consist of correlations considered significant (see ParCorr), which, however, mostly appear to be more consistent within a climate model compared to the measured distance (with the F_1-score) between different climate models, as indicated by the relatively high and closely clustered F_1-scores within the reference climate model and loosely clustered F_1 scores from other climate models (Figure 4.5). The F_1-scores between different climate models thus exhibit greater variation, as also evident from M_1 in Figure 4.4. M_2 also increases, as the various climate models also seem to agree less on the significant correlations on average.

The described process is visualized in Figure 4.2, where $\frac{M_2}{M_1}$ represents discrepancy between the climate models. A higher value in $\frac{M_2}{M_1}$ indicates greater discrepancy. It becomes evident that the selection of PC-α must be within a certain range to obtain as many model-specific links as possible without getting into to excessive noise. Figure 4.3 illustrates the number of links found at different PC-α values. With a too high PC-α value, only the most robust links for each ensemble are found, leading to less discrepancy. Conversely, with a too low PC-α, networks contain too much specific noise, what can be seen in Figure 4.2 and Figure 4.3. The convergence of the found links is also noteworthy in this context, which, in combination with Figure 4.5, further suggests that the noise links or ensemble-specific correlations could be more similar within a model than across multiple models.

The number of observed years directly influences the amount of data being considered. Theoretically, PCMCI can conduct error-free independence tests on infinitely large datasets. Therefore, the expectation was that more data would lead to better results. With more data available, the conditional independence tests consist of less noise since the noise in the dataset in general is also decreased or can be recognized [10]. It was found that a higher number of observed years leads to a higher discrepancy, in line with expectations, which can be seen at figure 4.6. An interesting observation here is a linear increase in the achieved performance. The boundary values around the assumed optimum, namely the complete time series, were testen in smaller intervals.

In Figure 3.5, it is evident that a higher number of components also leads to increased discrepancy between the climate models. The reason behind this is similar to that of the argument for using the whole number of observed years as data. With more components available, theoretically, the true causal parents can be approximated to a greater extent. However, when there are too few components, especially if they are geographically distant, there is a higher chance of conditioning on false causal parents. This, as mentioned above, leads to more noise and consequently lower discrepancy.

As it emerged that applying PCMCI to the full dataset, as well as to the full number of components, leads to higher discrepancy, the focus was on the PCMCI-specific hyper-parameters PC-α and MCI-α, where the range of the optimal MCI-α could already be restricted. The drawback of using PCMCI on the full dataset, especially with the complete

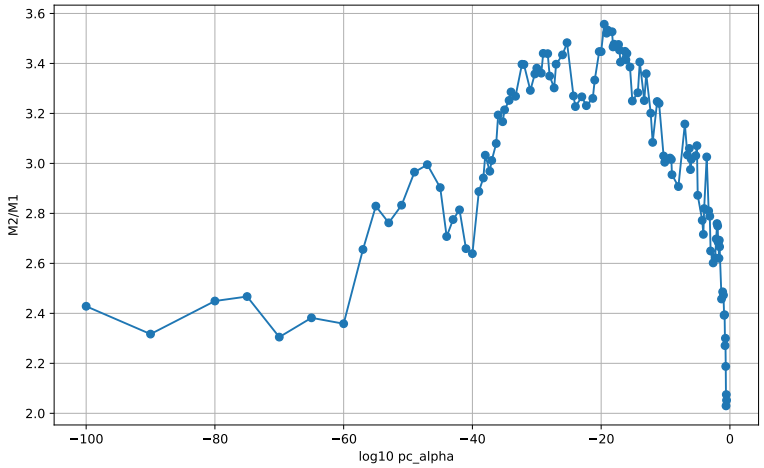


Figure 4.2: The plot illustrates the relationship between PC-$\alpha \in [3 * 10^{-1}, 1 * 10^{-100}]$ and $\frac{M_2}{M_1}$ in the context of the F_1-score. PCMCI was applied to the full dataset (all 100 components, full time series). For the calculation of $\frac{M_2}{M_1}$, a consistent series of MCI-α values was tested for each given PC-α. Only the best-performing MCI-α value was selected for visualization. It could also suggest that in this case, when evaluating the climate models, causal links are more suitable than mere correlations, since $\frac{M_2}{M_1}$ decreases at some point.

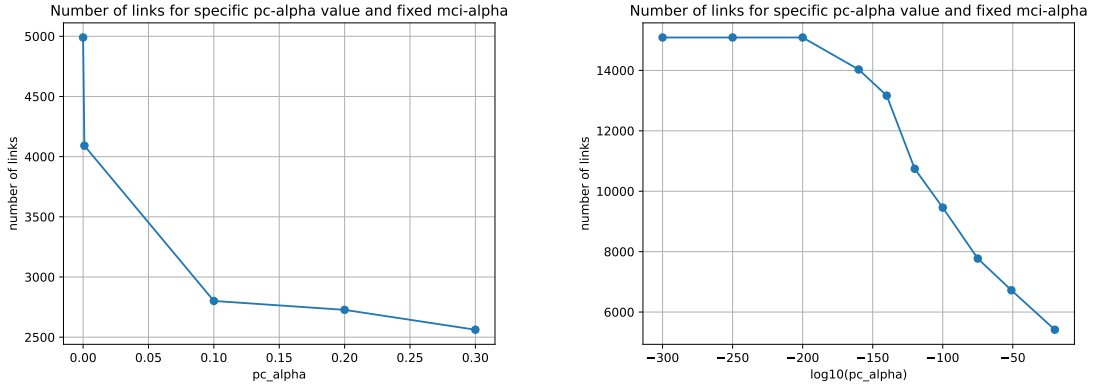


Figure 4.3: Both plots visualize the effect of the chosen PC-α on the resulting number of causal links (i.e. $X^i_{t-\tau} \to X^j_t, \tau > 0$ when applying PCMCI to all 100 components, along with the full time series data from ensemble$_1$ from ACCESS-CM2. The x-axis represents the used PC-α values $\in [0.3, 0.2, 0.1, 10^{-3}, 7.5 * 10^{-7}$ in the left plot and way smaller values in the right plot (represented in logarithmic style), with a fixed MCI-$\alpha = 0.01$. The y-axis represents the resulting number of causal links.

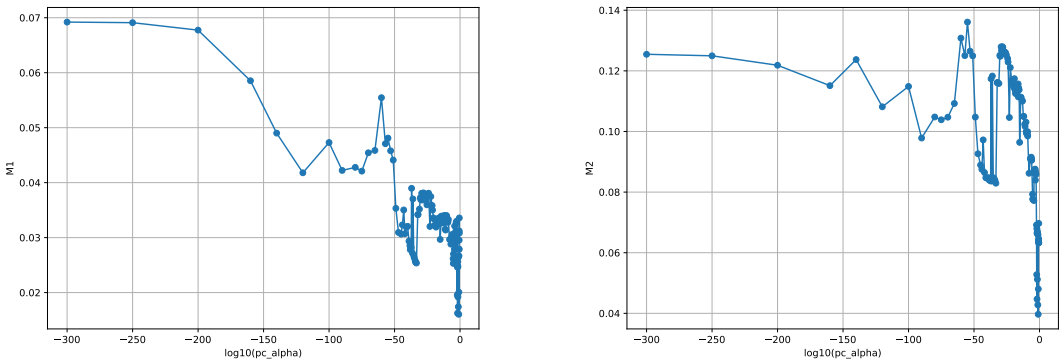


Figure 4.4: The left plot illustrates the effect of PC-α on M_1. Each dot corresponds to M_1 when applying PCMCI with the best performing MCI-α value selected from the MCI-α list in the context of maximizing $\frac{M_2}{M_1}$ and using the complete dataset (100 components, full time series). The right plot illustrates the same scenario with respect to M_2. More precisely, the same hyperparameters were used for the achieved M_1 and M_2 scores for a given PC-α.

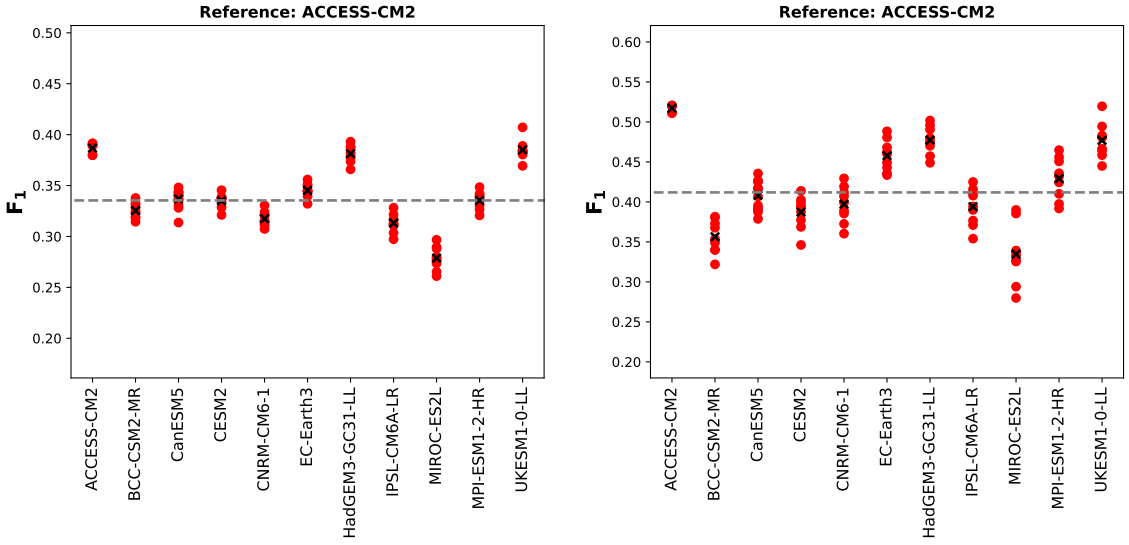


Figure 4.5: Both plots visualize the resulting F_1-scores with ACCESS-CM2 as reference model, when applying PCMCI to the full dataset (100 components, full time series), depending on the given PC-α. For the left plot, PC-α was set to 0.3, while in the right plot, PC-α was set to $1 * 10^{-300}$. The best-performing MCI-α value from a given list was used for each case. Both plots reveal a discrepancy between related and non-related climate models. With PC-α = 0.3, a discrepancy score of $\frac{M_2}{M_1} \approx 2.05$ was achieved, whereas with PC-$\alpha = 10^{-300}$, it was $\frac{M_2}{M_1} \approx 1.81$.

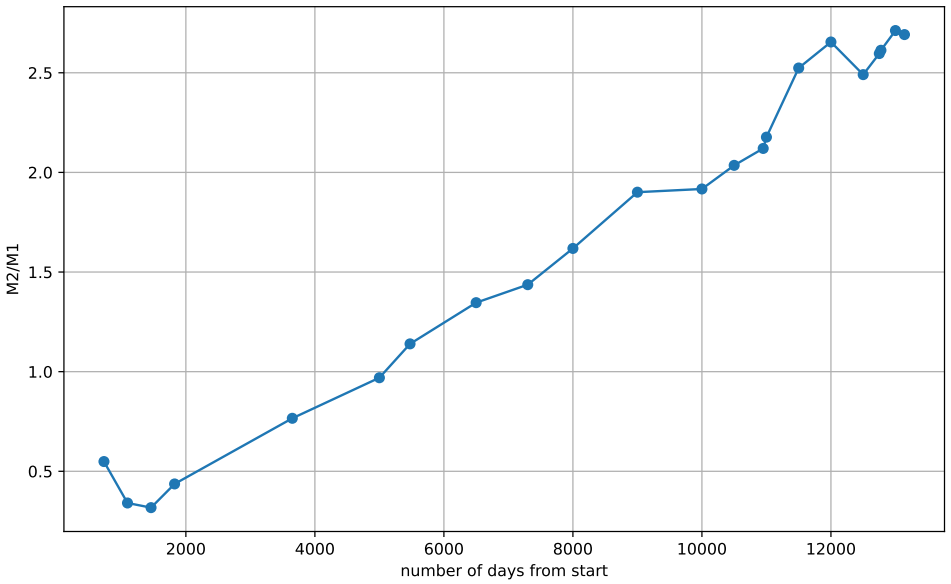


Figure 4.6: The plot illustrates the effect of the number of observed years on the climate model discrimination, starting from time stamp 0 (1979). PCMCI was applied on 100 components, with PC-$\alpha = 10^{-32}$, MCI-$\alpha = x$, and given observed years. For x, the best-performing MCI-α from the given values was adopted for the mentioned PC-α, number of components setting and observed years. Each point represents the achieved $\frac{M_2}{M_1}$ score at a given number of observed years (x-axis).

number of components, is the runtime. However, in this case it was still manageable due to parallelization. However, with the addition of further components, this may no longer be the case. The process of finding the optimal combination of PC-α and MCI-α involved computing the resulting $\frac{M_2}{M_1}$ score for PC-α values within the interval $[0.5, 10^{100}]$. The highest achieved value for $\frac{M_2}{M_1}$ was ≈ 3.56, attained with PC-$\alpha = 3 * 10^{-20}$ (Figure 4.2). The MCI-α values was selected as the best-performing value from the MCI-α list, resulting in MCI-$\alpha = 10^{-4}$4.2. However, this approach has limitations, as it only explores the near values around the true local optimum. The derivation of the optimal combination of hyperparameters is too complex and testing many more combinations to approximate the local optimum is not feasible due to the high runtime. Interestingly, when evaluated against one of the UK Met Office related climate models (Appendix A.1), EC-Earth's F_1 scores are quite close to those of the UK Met Office related climate models, despite not sharing an official common development background [17].

4.2. PCMCI vs. PCMCI+

In the context of evaluating climate model ensembles, the choice of methodology can significantly impact the ability to capture and quantify discrepancies between models effectively. In this regard, PCMCI and PCMCI+ emerged as two prominent techniques, each offering its own advantages and considerations, as mentioned in the following.

PCMCI has the advantage of an additional hyperparameter MCI-α, which is not present in PCMCI+. Another advantage of PCMCI is its runtime efficiency, as all steps are parallelized. In contrast, only the PC-step is officially parallelized in PCMCI+, which made it infeasible to apply it to a high number of components, limiting it to a maximum of 50 components for this project. Similarly, the selection of PC-α could not be set too high depending on the number of components.

During the PC-skeleton phase in PCMCI, the graph is reinitialized with fully connected contemporaneous links and the found links from the previous PC-step. In the non-parallelized PC-skeleton phase, all of these links are then tested for conditional independence. The runtime increases significantly with higher PC-α values, as fewer links are removed in the previous PC-step and in the PC-skeleton phase, iteration over these links can only be done sequentially due to the lack of parallelization.

For PCMCI, the result was, that especially with using a high number of components, a very low PC-α led to a higher discrepancy, with the MCI-α mostly ranging between 10^{-2} and 10^{-5}. Figure 4.2 shows a significant increase in $\frac{M_2}{M_1}$ as PC-α is chosen smaller in the range $[0, 10^{-10}]$. However, for small MCI-α values, the results worsen, as visualized in Figure 3.2. Therefore, PCMCI had the advantage of combining a small PC-α value with a higher MCI-α value, which led to a better discrepancy.

In PCMCI$^+$, in the PC-skeleton phase, MCI tests are used mainly when removing links. PC-α is used as a threshold for independence tests in both the PC step and the following PC-skeleton phase. However, the conditions of a very small α value for the PC steps and a rather larger α value for the subsequent MCI tests cannot be met simultaneously. Integration of contemporaneous links did not outweigh this disadvantage. Consequently, PCMCI performed better for this project in achieving a high discrepancy between the climate models. Furthermore, the F_1-scores from PCMCI$^+$ were significant lower compared to PCMCI.

Figure 4.7 shows the resulting $\frac{M_2}{M_1}$ score for 50 components and various PC-α values. With PCMCI$^+$, a $\frac{M_2}{M_1}$ score ≈ 2.32 was achieved at PC-$\alpha = 2 * 10^{-2}$. PCMCI, on the other hand, attained this result only from a PC-alpha value of 10^{-5} onwards. Even with further optimization by computing the subnetwork leading to an increase in discrepancy (in this case, $\frac{M_2}{M_1} \approx 2.78$), no significantly better result than PCMCI ($\frac{M_2}{M_1} \approx 2.72$) could be achieved.

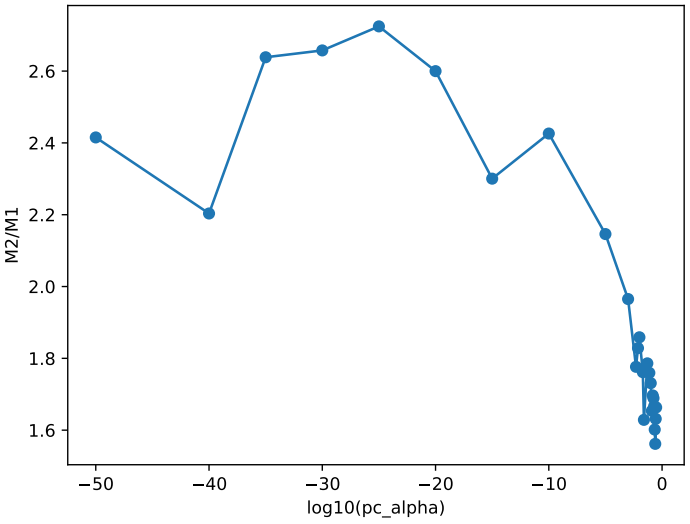


Figure 4.7: The figure visualizes the effect of different PC-α values on $\frac{M_2}{M_1}$. PCMCI was applied on 50 components using the full time series data. Each dot corresponds to a setting of the specified PC-α (x-axis) and the best-performing MCI-α value chosen from a given list, in the context of optimizing for $\frac{M_2}{M_1}$ (y-axis).

In this study, PCMCI emerged as the preferred methodology due to its ability to fine-tune link selection using the MCI-α hyperparameter and its parallelized implementation, enabling faster computations. The adaptability of PCMCI in selected PC-α values (due to

runtime) enabled the discovery of best configurations to improve discrepancy. Conversely, PCMCI$^+$ faced challenges in balancing the choice of PC-α and MCI-α values (where MCI-α is the same as PC-α) for PCMCI$^+$, leading to inferior performance compared to PCMCI. Therefore, PCMCI proved to be more effective in capturing and quantifying discrepancies within climate model ensembles in this study. The best-performing hyperparameter setting for PCMCI was found to be PC-$\alpha = 3 * 10^{-20}$ and MCI-$\alpha = 10^{-4}$, applied to all 100 components and the full time series data. All resulting F_1-scores for all reference climate models, resulting from this initial best hyperparameter setting, can be seen in Figure A.1 (Appendix).

4.3. Subnetwork

The following section presents the respective results of the subnetwork search using the two approaches. The approaches are described in the methods section, one based on components and the other on causal links.

4.3.1. Components

The initial approach involved iterative removal of components that did not contribute to the climate models discrepancy. Beginning with the first 11 components and utilizing full time series data, each iteration involved the removal of a different component. The expectation was that if the removal of the component n from the first N used components leads to improved results, then removing component n from the first $M > N$ used components would also yield better outcomes. An example of this is illustrated in Figure 4.8 for $N = 11$ and $M = 21$. It can be observed that PCMCI applied on the first 11 components without component 7 outperforms PCMCI applied on all other first 11 components without component $k \neq 7$. However, contrary to expectations, the same component $k \in [0, 10]$ from the first 21 components was removed in the second half. It is evident that the expectation did not hold true. PCMCI performed better on the first 11 components without component 7 than PCMCI on the other first 11 components without component $k \neq 7$, but PCMCI on the first 21 components without component 7 performed worse than most applications of PCMCI on the first 21 components without component $k \neq 7$. It is also evident that all specified combinations of 20 components performed worse than the best component configuration derived from the provided 10 components. There are several reasons that can explain why this iterative approach does not work in this way. The effect of removing a particular component k may depend on the context in which it is present. In the dataset with the first 11 components, other components may play a supporting or not supporting role that affects the result, when it comes to maximizing $\frac{M_2}{M_1}$. In the data set with the first 21 components, these context effects may be different, leading to different results, since other conditioning sets are used that produce different statistical (in)dependencies. A similar reasoning is that the data set of the first 21 components can be more complex overall than a data set of only the first 11 components. In a more complex system, the interactions between the components may be more diverse and less predictable, making the transferability of results between the two data sets more difficult. When it comes to the conditional (in)dependence tests, these tests are often sensitive to random variation, especially when the sample size is limited (as in this case, only a dataset of 35 years). The differences observed when removing a particular component k may be partly due to random variation, especially in the conditioning sets, which may be stronger or weaker in a larger data set of the first 21 components compared to the data set of the first 11 components. Therefore, the result does not necessarily imply that removing the same component in a data set with 21 components will yield the best performance. Furthermore, the components were not selected based on their geographical location. Consequently, important components may be removed, even those with significant geographical relevance

that could substantially influence the results. Removing these components could worsen the effects mentioned before.

nr. components	missing components	pc_alpha	mci_alpha	M2/M1
10	[5]	0.15	7e-05	0.6252886208800711
10	[8]	0.15	5e-08	0.6689156675844949
10	[10]	0.15	1e-09	0.7506948070758988
10	[0]	0.15	7e-07	0.7855628268665569
10	[1]	0.15	3e-05	0.8164081683950938
10	[9]	0.15	0.0001	0.8166169870698509
10	[2]	0.15	3e-05	0.842511420567941
10	[6]	0.15	7e-05	0.8585745546632403
10	[3]	0.15	7e-06	0.8951557733308689
10	[4]	0.15	3e-05	0.902508479444135
10	[7]	0.15	3e-05	1.012683072940788
20	[5]	0.15	1e-09	0.7461241840294314
20	[10]	0.15	1e-09	0.7860202240837727
20	[0]	0.15	1e-05	0.7892295505023246
20	[7]	0.15	1e-09	0.8293664533982578
20	[8]	0.15	1e-10	0.8398929433949558
20	[2]	0.15	7e-05	0.8801018296601586
20	[6]	0.15	3e-05	0.8856522572924317
20	[1]	0.15	3e-05	0.9161363903077195
20	[4]	0.15	0.0001	0.9365961115243215
20	[3]	0.15	7e-05	0.9523728416618805
20	[9]	0.15	0.0001	0.9564210882939165

Figure 4.8: The figures shows various combinations of 10 and 20 components. In the first half, combinations of 10 components are presented. These combinations each include the first 11 components, with a different component k removed in each case. The combinations of 20 components consist of the first 21 components, with the same component k removed as in the previous step. The number of components is provided in the first column, the removed component is noted in the second column. The third column displays the PC-α used, with PCMCI consistently applied with the same PC-alpha and on full time series data of the given components. The fourth column indicates the best performing MCI-α, with the context of maximizing $\frac{M_2}{M_1}$. The achieved score in $\frac{M_2}{M_1}$ is provided in the last column.

The second approach encountered similar challenges as the first. On one hand, only the 47 components above latitude 18 were considered, and on the other hand, only the components with $latitude < -15$ were included. The problem resulted from the interaction between the optimization of PC-alpha and MCI-alpha and the geographical selection of the components, which focused exclusively on regions with few non-discrepancy generating connections. The problem arises from the fact that different PC-α and MCI-α values lead climate models or their ensembles to identify different links. This effect is observed by comparing Figure 3.8 and Figure 4.9, resulting from PCMCI with different hyperparameters. Removing components, as described in the first approach, distorts statistical tests and alters results. Additionally, by removing components, the discrepancy-generating links, or links that do not contribute to the discrepancy, may now be found between other components. Furthermore, optimizing PC-α and MCI-α for each set of components leads to different components contributing to the discrepancy depending on the PC-α and MCI-α values. For these reasons, the geographical interpretation also failed to yield the desired results.

The third approach involved computing $k_{discrepancy}$ (3.2.4.1) for each component, repre-

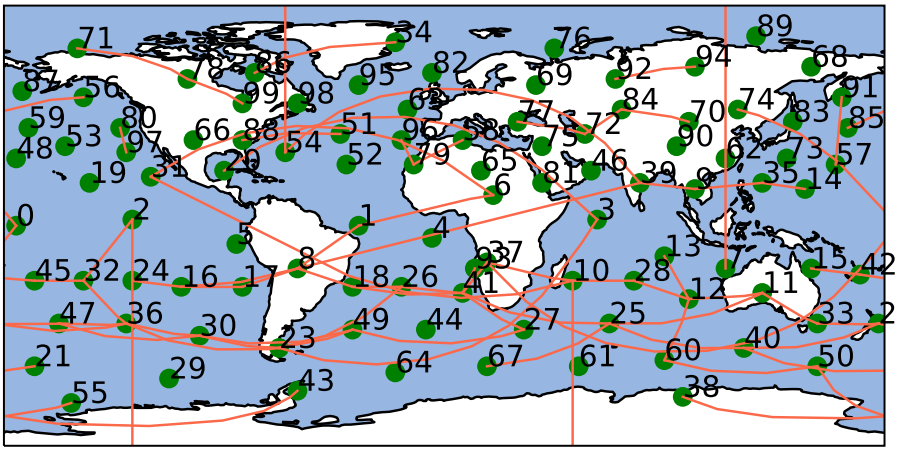


Figure 4.9: The plot illustrates the geographical distribution of the 100 components and the links between them. These links were computed using PCMCI with the initial best hyperparameter setting (full time series data, 100 components, PC-$\alpha = 3 * 10^{-20}$, MCI-$\alpha = 10^{-4}$). Only links where all climate models agreed were considered, meaning they appeared in every ensemble after applying PCMCI.

senting the ratio of involved links contributing significantly versus those not contributing to discrepancy and then to remove the components which do not. That approach faced the same problems as the first and second approach. The process of removing specific components introduces a layer of complexity, primarily because statistical tests operate differently due to differences in conditioning sets when components are dropped, as mentioned above. The application of the $k_{discrepancy}$ method, which is designed to calculate elements that are expected not to worsen discrepancies, did not produce the expected results and showed poor performance. This could be attributed to the selection of components that generate discrepancies, although not necessarily in the context of the discrimination between different climate models. In addition, reducing the number of components used reduces the potential benefits of a more comprehensive set of components and thus again worsens the shortcomings of the mentioned approaches.

To conclude, the iterative or specific removal of components that do not contribute to the discrepancy between climate models proved insufficient, as dropping certain components led to unexpected outcomes. This occurred because potentially crucial components and their influences on the results were not observed. Additionally, the geographical selection of components resulted in biased statistical tests and unpredictable outcomes. The application of $k_{discrepancy}$ also fell short, as excluding components containing potentially significant information led to erroneous conclusions and deteriorated results. Ultimately, reducing the number of components not only meant that potentially important conditions were neglected, but also that the methodology was not suitable for effectively capturing the complex relationships within the dataset.

4.3.2. Links

The new method was designed to compute the F_1-score specifically within a predefined subnetwork consisting of links, utilizing all available conditions without excluding any elements. This approach allows for more accurate calculations and decreases runtime by not needing to run PCMCI again. To construct the subnetwork, PCMCI was initially applied with optimized hyperparameters, followed by fine-tuning to identify discrepancy-generating links. Algorithms 2 and 3 were employed to compute the subnetwork, with a grid search exploring different combinations of threshold values. The subnetwork was computed using the best-performing setting identified thus far to further improve the results (maximize $\frac{M_2}{M_1}$). It was assumed that PCMCI settings leading to poorer performance in terms of climate model discrimination, consist of fewer discrepancy-generating links. Consequently, it was expected that the subnetwork derived from low performing settings would consist of fewer discrepancy generating links, potentially leading to less stable results.

In the search for the subnetwork consisting of links (on which the F_1-score is then computed), it was found that using $(threshold1, threshold2) = (0, 0)$ leads to the highest $\frac{M_2}{M_1}$ score of ≈ 16.73. Here, $M_2 \approx 0.88$ and $M_1 \approx 0.5$ (Figure 4.10). For $(threshold1, threshold2) = (0, 0)$, only the links are selected per climate model when they occur in each of its ensembles and these links must not appear in any of the other non-related climate models and their ensembles. When comparing two non-related models using the F_1-score, $TP = 0$ since, due to the definition of the threshold, there must be no overlaps. Consequently, the term above the fraction in $F_1 = \frac{2*Precision*Recall}{Precision+Recall}$ with $Recall = \frac{TP}{TP+FN}$ and $Precision = \frac{TP}{TP+FP}$ is 0, resulting in an F_1-score of 0. A problem occurred when comparing the models ACCESS-CM2, HadGEM3-GC31-LL, and UKESM1-0-LL because Algorithm 2 calculates the links for each climate model that are consistent across all its ensembles, but when removing the links in Algorithm 3, when counting the overlaps of links with the other ensembles of climate models, the related climate models are skipped. This results in links that have no guarantee of appearing in each of the ensembles of the three related climate models (Figure 4.12). The ensembles of the three different models now consist of either no same

links, only to a certain extent, or completely. Thus, the F_1-score when comparing them is neither always 0 nor 1 (Figure 4.12). When excluding the three related models, M_2 would constantly be at 1, as long as for each climate model, its specific and consistent links were found according to Algorithm 2 and Algorithm 3. The network with the used threshold of $(0, 0)$ consisted of 68 links and can thus be considered as overfitting. The geographic location of the links is visualized in Figure 4.11. An interesting aspect here is that these links are not primarily explained by ENSO. As illustrated in Figure 3.8, the climate models predominantly agree on links near the equator, which can be explained by ENSO. However, the discrepancy-generating links are mostly located outside this region as visualized in Figure 4.11. If another new climate model is added for evaluation in this case, this can most likely lead to false discrepancy. The higher *threshold*2 was set, the more links appeared in the subnet at a linear increase, which can be seen in Figure 4.10. With an increase in *threshold*2, M_2 decreased, as the models overlapped in more links. The rising F_1-score in the comparisons was another indication for that. However, M_1 also decreased, as more stable results can be generated on more links. With few links, due to a lower choice of *threshold*2, the subnetwork consisted mostly of discrepancy-generating links. The discrepancy-generating links were calculated and collected for each climate model and ultimately, the union of these links also led to more inconsistency within the climate models, since the ensembles of a model were relatively less in agreement in the strong discrepancy-generating links of other climate models. The resulting F_1-scores from all reference climate models and the subnetworks with $threshold = (0, 2)$ and $threshold = (0, 30)$ can be seen in Figure A.4 and A.5 (Appendix).

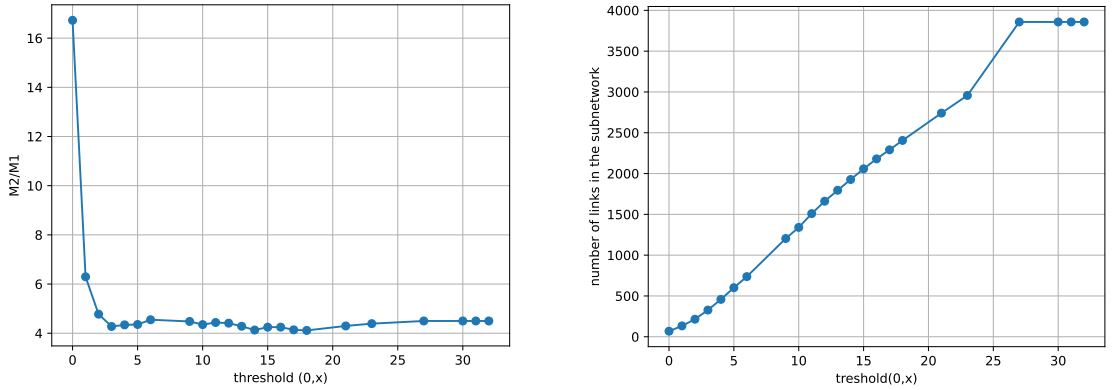


Figure 4.10: The left plot illustrates the effect of the parameter *threshold*2 from Algorithm 3 on the $\frac{M_2}{M_1}$ score. The subnetwork was constructed based on PCMCI using the current best hyperparameter setting (100 components, full time series, PC-*alpha* = $3 * 10^{-20}$, MCI-$\alpha = 0.0001$). The x-axis represents the threshold (*threshold*1, *threshold*2), where *threshold*1 is fixed at 0 while only *threshold*2 varies. The y-axis shows the number of links in the resulting subnetwork.

4.4. Metrics

In this section, an examination of different metrics, as mentioned in the implementation part, that used to measure the differences among climate model ensembles is discussed. The aim is to investigate how different parameters and modifications affect the computed scores in terms of $\frac{M_2}{M_1}$ on different metric. It is crucial to choose suitable metrics and parameters when trying to optimize the discrepancy between different climate models. As mentioned in the implementation, metrics ranging from conventional p-norms, such as the L_1 and L_2 norms, to modified scoring approaches that include penalty terms, are considered. Each metric brings its unique perspective to the distance between two climate models

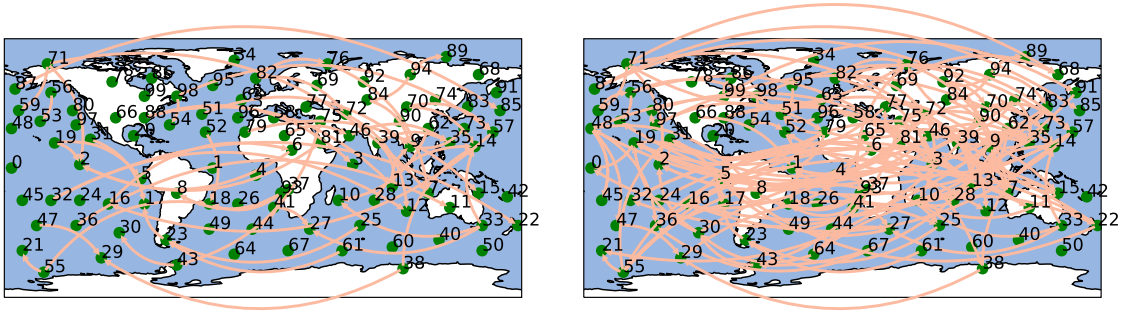


Figure 4.11: Both plots visualize the discovered subnetwork for $(threshold1, threshold2) = (0, 1)$ (left) and $(threshold1, threshold2) = (0, 3)$ (right), with a total link count of 133 (left) and 327 (right), respectively.

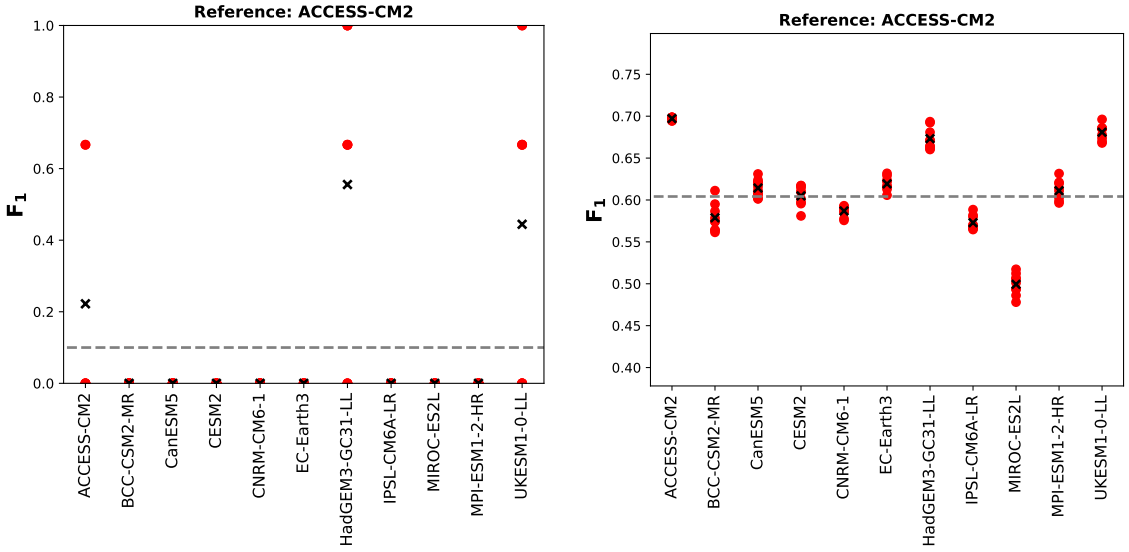


Figure 4.12: The left plot displays the achieved F_1-scores for the resulting subnetwork with $threshold = (0, 0)$. The subnetwork was found by using PCMCI with PC-$\alpha = 3{*}10^{-20}$ and MCI-$\alpha = 10^{-4}$, applied to 100 components and the full time series data. The right plot shows the achieved F_1-scores with the same PCMCI and dataset settings, but with $threshold = (0, 32)$.

and their ensembles (the resulting graph when applying PCMCI). Mainly, the impact of the parameter settings of PC-α and penalty terms on the climate model discrepancy is explored. These parameters influence the sensitivity of the metrics to variations in the PCMCI results, thus shaping the scores achieved in $\frac{M_2}{M_1}$. The modified F_1-score was also extended to the subnetwork network.

4.4.1. Modifications of the F_1-score

The use of the sigmoid function, applied to the difference in the value matrices when incrementing TP, FP and FN did not yield convincing results ($\frac{M_2}{M_1} < 3$), but it had a different optimum at PC-$\alpha = 10^{-6}$ and MCI-$\alpha = 10^{-5}$, with the same hyperparameters otherwise. The utilization of the ReLU-like function, with $a = 0.05$, performed even worse ($\frac{M_2}{M_1} < 2$) with an optimum at PC-$\alpha = 5 * 10^{-26}$ and MCI-$\alpha = 5 * 10^{-8}$. The poorer performance with the sigmoid function could be explained by the fact that, at a certain point, similarly distant differences in the value matrices are heavily weighted differently, which could disadvantage related ensembles compared to unrelated ones. Furthermore, the use of the sigmoid function could reduce the weight between FN and FP together compared to TP. As a result, false links are not as heavily weighted, leading to a decrease in discrepancy of the different climate models. The same argument could apply when using the ReLU-like function. Both the sigmoid function and the ReLU function must be called with suitable parameters, where these parameters for FN, FP and TP must not be equal. The goal should be to weigh false links more heavily. This requires further investigation into the differences in the value matrix entries and at what point TP should be weighted weaker/stronger or FP and FN should be weighted stronger/weaker to achieve a higher discrepancy score in $\frac{M_2}{M_1}$.

When using the penalty term on FN and TP only, the value of $\frac{M_2}{M_1}$ could be significantly increased, as shown in Figure 4.13. It can be observed that the effect of the penalty term decreases as it becomes larger. The reason for this is that at some point only the penalty dominates and the true positives (TP) hardly contribute. However, with higher penalty terms, the F_1-score also decreases significantly because both recall and precision become smaller, and so $F_1 = \frac{2*Precision*Recall}{Precision+Recall}$. In this case, there is a significant trade-off between achieving high F_1-scores and maximizing discrepancy (resulting F_1-scores with *penalty*=50 in Appendix A.1).

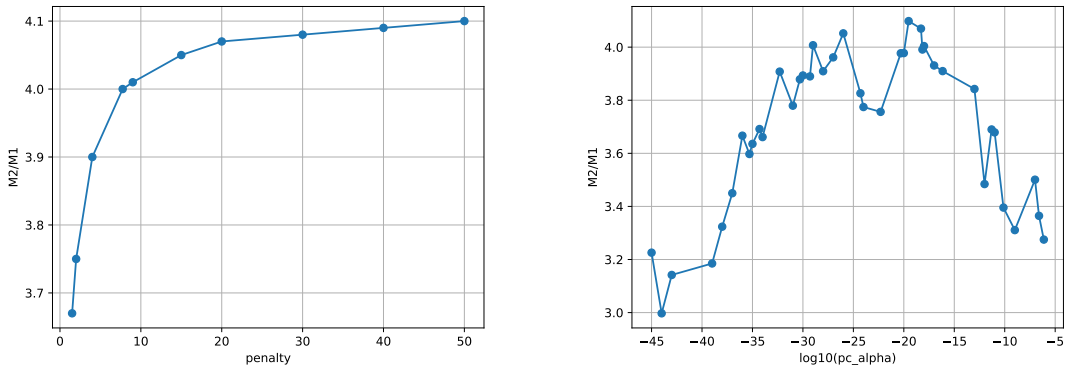


Figure 4.13: For both figures, PCMCI was executed on the full dataset (100 components, full time series). The left plot illustrates the effect of the penalty term (x-axis) on the modified F_1-score and the resulting score in $\frac{M_2}{M_1}$ (y-axis) for a fixed PC-$\alpha = 10^{-20}$ and MCI-$\alpha = 10^{-4}$. The right plot shows the best-performing value in $\frac{M_2}{M_1}$ (y-axis) for each PC-α (x-axis) using penalty term $penalty = 50$ and varying MCI-α values, with only the best-performing MCI-α being retained.

The modified F_1-score was also utilized on the computed subnetwork network to further improve the achieved score in $\frac{M_2}{M_1}$. Significant improvement in $\frac{M_2}{M_1}$ was observed across all thresholding parameters, with $threshold = (0, x)$ (Figure 4.14). The highest $\frac{M_2}{M_1}$ score was again achieved at $threshold = (0, 0)$, reaching $\frac{M_2}{M_1} \approx 19.20$. However, in this case, the results appeared more indicative of overfitting. A more stable outcome was obtained through thresholding with $threshold = (0, 5)$ (Appendix, Figure A.2 and A.7), where an $\frac{M_2}{M_1}$ score of ≈ 8.25 was achieved (Figure 4.14, second plot). In contrast to the unmodified version of the F_1-score, a tendency toward decreased performance with increasing upper thresholds can be observed. One reason for the significantly improved performance of the modified F_1-score function could be attributed to the stronger weighting of false positives (FP) and false negatives (FN). The subnetwork is made up of as many discrepancy-generating links as possible, with a focus shifting toward links where ensembles within a climate model agree as $threshold2$ decreases. Using the discrepancy-generating links alongside greater weighting of false links, a further increase in $\frac{M_2}{M_1}$ was achieved. As $threshold2$ increases, more links are included, leading to more stable results (reduced M_1, as models become "similarly dissimilar" and thus perform similarly "poorly"). However, with the higher number of links, there is an increased likelihood that related ensembles also become dissimilar. Consequently, they attain relatively lower F_1 scores more rapidly due to the stronger weighting of FP and FN, shifting them towards the scores of non-related climate models.

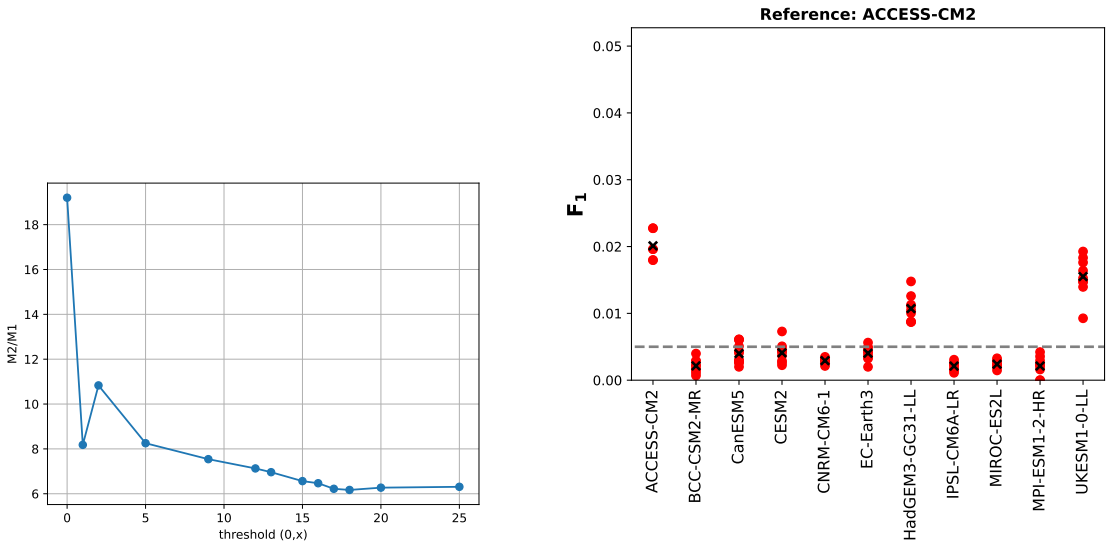


Figure 4.14: The plot on the left shows the subnet parameter threshold (0,x), with $threshold1 = 0$ and $threshold2 = x$ (on the x-axis), displaying the obtained $\frac{M_2}{M_1}$ ratio (on the y-axis). The subnet was constructed based on PCMCI on full time series and all components, with PC-$\alpha = 3 * 10^{-20}$ and MCI-$\alpha = 10{-}4$. The right plot showcases the resulting F_1-scores of various climate model ensembles when evaluated against the ensembles of the reference model ACCESS-CM2 on the calculated subnetwork with $threshold = (0, 5)$. The modified version of the F_1 score with $penalty = 50$ was utilized as a comparative metric.

4.4.2. p-norms and modifications

In this study, the 1-norm and 2-norm have been employed to measure the distances between the two matrices $refValMatrix$ and $valMatrix$, with a particular focus on the effects of PC-α, since MCI-α only plays a role when selecting the edges and in this case, it is iterated over the entire value matrix. The $refValMatrix$ represents the value matrix of the ensemble from the reference climate model, while the $valMatrix$ denotes the matrix from other climate models ensemble. The investigation aims to explore how varying PC-α values

impact the distances computed using the 1-norm and 2-norm. In Figure 4.15, the influence of PC-α on the L_1 and L_2 norms, along with the resulting score in $\frac{M_2}{M_1}$, is visualized. It can be observed that the optimum is achieved at about the same PC-α values compared to when considering the F_1-score alone (Figure 4.4). The results can be explained by the following: When PC-α values are too low, the conditioning sets become smaller, and the link strength indicated in the value matrix consists of more noise, as many influences from other variables are not taken into account. On the other hand, when PC-α is too high, the conditioning sets become too large. This leads to a larger part of the link strength being falsely explained away, resulting in less accurate results. When PC-α is either too high or too low, the value matrix entries are likely to be less climate model-specific, leading to a decrease in discrepancy. The best performing PC-α were for L_1 and the L_2 nearly the same, with the optimum in the L_2 norm ($\frac{M_2}{M_1} \approx 3.56$) being higher than that in the L_1 norm ($\frac{M_2}{M_1} \approx 3.31$). The better performance of the L_2 norm can be explained by considering quadratic differences. When differences between the two value matrices are considered, the distance increases quadratically, meaning that larger differences are weighted more heavily. Since the value matrices of related climate models appear to be more similar, this results in an even better discrepancy.

In addition to conventional L_1 and L_2 norms, in this study a modified approach has been introduced to include the graph matrix and a penalty term in distance measurements between the matrices $refValMatrix$ and $valMatrix$, where MCI-α primarily influences the selection of links. In contrast to the unmodified version, its impact is now significant when iterating over the value matrix and graph matrix. In Figure 4.16, the various results obtained are visualized. It is notable that the best-performing PC-α values are located at different positions compared to those observed with the L_1 or L_2 norms alone. In the modified L_1 norm, a penalty term of penalty=0.05 was employed, where an increase in the penalty term led to a linear deterioration in $\frac{M_2}{M_1}$. This degradation could be explained by the penalty term linearly "pulling apart" the different value matrices being compared, compared to the penalty term of penalty=0. With the linear increase in distance, related ensembles may relatively move further apart than non-related ensembles. This could account for the linear deterioration observed. Nonetheless, this modified version of the L_1 norm achieves a higher value in $\frac{M_2}{M_1}$ compared to the conventionally used L_1 norm (see Figure 4.15).

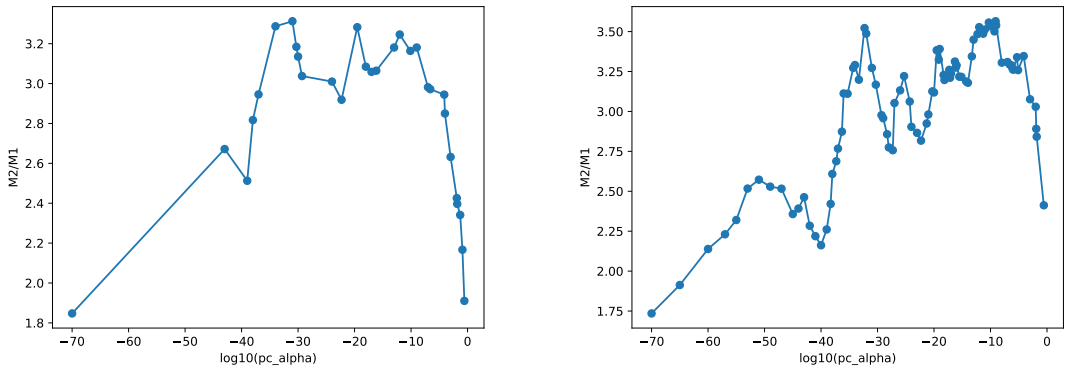


Figure 4.15: The left plot illustrates the impact of PC-α (x-axis) on the attained $\frac{M_2}{M_1}$ score using the L_1 norm, which is applied to the difference between the resulting value matrices of the compared ensembles. The value matrices are the result of PCMCI applied to the full dataset (100 components, full time series) of each ensemble. Only the value matrix entries corresponding to links with $\tau > 0$ were considered. Similarly, the left plot shows the effect of PC-α on the achieved $\frac{M_2}{M_1}$ score using the L_2 norm, which is used in the same way as the L_1 norm. PCMCI was applied with identical hyperparameter settings for both cases. In both figures, PC-α ranges from 0.275 to 10^{-70}.

In the modified L_2 norm, a similar trend is observed. Contrary to expectations, in both cases, a penalty term of $penalty = 0$ performs better (4.17). A neutral penalty term implies that differences are only measured in entries where both graph matrices agree. The expectation was that increasing the penalty term would lead to non-related climate models achieving a lower score (in the modified L_1 norm, a higher score is "better" due to the affine transformation, whereas in the modified L_2 norm, a lower score is better due to the absence of this transformation), as the differing links (whether present or absent) would be weighted more heavily. The influence of the quadratic value matrix differences likely dominates, and an increasing penalty term reduces its influence, potentially worsening the outcome.

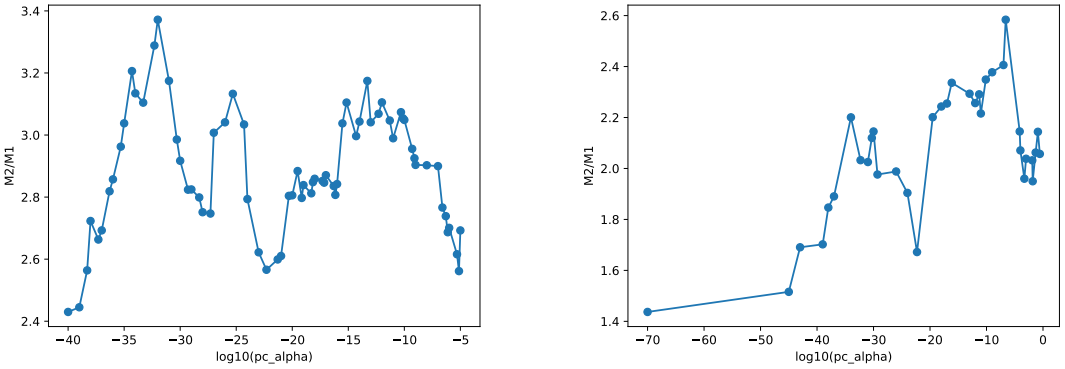


Figure 4.16: Both figures are based on the application of PCMCI to all 100 components and full time series data. The left figure illustrates the effect of PC-α (x-axis) on the resulting score in $\frac{M_2}{M_1}$ (y-axis) using the modified L_1 norm. A penalty term of $penalty = 0.05$ was used in this case. The right figure describes the same scenario using the modified L_2 norm with $penalty = 4$. For each PC-α, the best performing MCI-α was used.

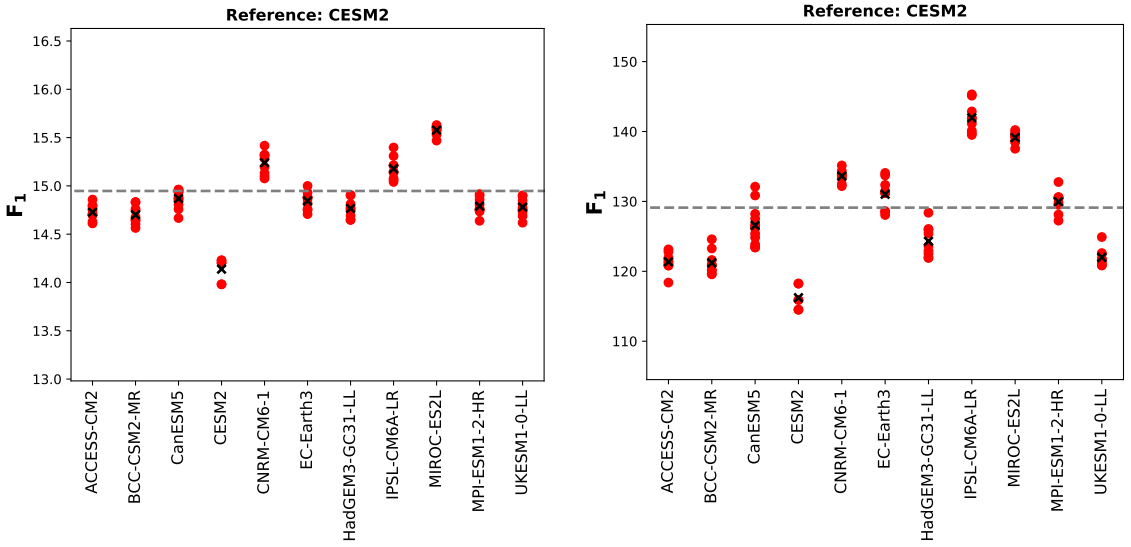


Figure 4.17

For completeness, the L_2 matrix norm was also applied to the respective p-matrices when evaluating an ensemble from the reference climate model against an ensemble from another climate model. Similar to the application on the value matrices, these comparisons are independent of MCI-α, as MCI-α is only used as a thresholding parameter to filter out the links defined as significant by MCI-α from the p-matrix. Due to the relatively poor

performance in maximizing $\frac{M_2}{M_1}$, smaller PC-α values, as seen in Figure 4.18, were not tested. A noticeable trend is the significant increase in $\frac{M_2}{M_1}$ when decreasing PC-α, with the rate of change seeming to converge up to a certain point. It is possible that with too many conditions found by a large PC-α, the MCI tests produce similar and thus more stable results across all climate models.

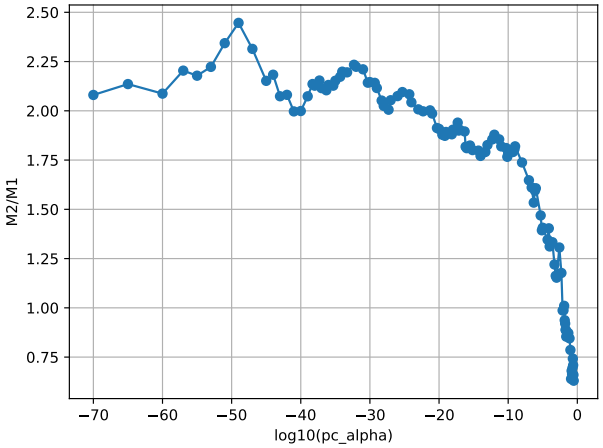


Figure 4.18: The figure illustrates the effect of PC-α (x-axis) on $\frac{M_2}{M_1}$ (y-axis) when the L_2 norm is used as the comparison metric over the two p-matrices. PCMCI was applied with the different given PC-α on all 100 components and full time series data. MCI-α has no effect in this context, as the p-matrix is considered as a whole.

4.4.3. Other metrics

In the analysis of the degree centrality comparison metric, PC-$\alpha \in [0.3, 10^{-12}]$ were tested, with 100 components, full time series data and similarly, the best-performing MCI-α value was selected as before. Due to the comparatively poor performance and the lack of improvement trend, no further PC-α values were tested. In comparison to the F_1-score, there was no significant improvement in $\frac{M_2}{M_1}$ beyond PC-$\alpha > 10^{-6}$. Also in contrast to the F_1-score, the optimal MCI-α values were mostly in the interval $[10^{-6}, 10^{-8}]$. For all PC-α values, a very low score in M_2 and $M_1 < 10^{-4}$ was achieved. The reason for this is likely that the degree centrality of the nodes in the resulting networks does not vary significantly, as there are no large differences in the number of links found (Figure 4.19).

With the edge kernel, a better discrepancy was achieved. Here, a different optimum was reached compared to the F_1 score. Both in PC-α (best performing value PC-$\alpha = 5^{-10}$) and MCI-α (best performing values $< 10^{-7}$), different optimums were observed. The metric scores achieved, after transformation into Euclidean space, were mostly relatively high ($cos(reference\ graph - other\ graph) > 0.7$) for all climate models and not only for the reference climate model. To achieve higher discrepancies, more links would need to differ both in their direction (sign in the value matrix) and in their occurrence. However, the cosine function flattens out at small angles, making it suboptimal for capturing discrepancies between the climate models, as differences should be highlighted more (Figure 4.19).

The investigation into various metrics for measuring differences among climate model ensembles reveals that the modified F_1-score with using a penalty term showed the best performance. This could be attributed to its ability to capture more complex relationships between two different graphs by combining $Presicion$, $Recall$, and the penalty term $penalty$, thereby amplifying the discrepancies between different graphs as they become less similar. The penalty term contributes to a stronger weighting of false links, consequently generating

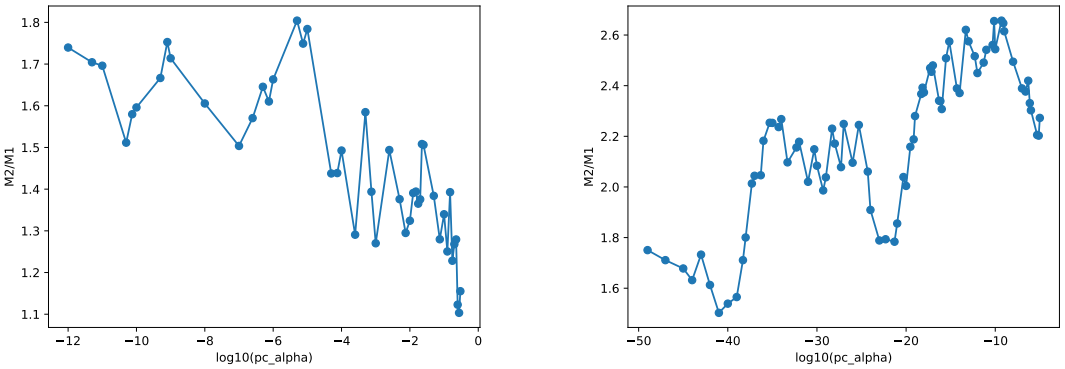


Figure 4.19: The left plot depicts the resulting scores in $\frac{M_2}{M_1}$ for degree centrality as comparison metric, depending on the chosen PC-α value. The right plot illustrates the same context, but with the edge kernel as used comparison metric.

a higher level of discrepancy between the models. In that case, only a linear penalty term was employed, without incorporating additional information from the provided graphs, value matrices, and p-matrices. A more complex penalty term could potentially further enhance the climate model discrepancy.

5. Summary, Conclusions & Outlook

The goal of the bachelor's thesis was to perform hyperparameter optimization for parameters such as number of components, number of years considered, PC-alpha, possibly MCI-alpha, and to identify a metric that maximizes the discrepancy between the various inferred causal graphs of climate models. Furthermore, the question arose as to which of the methods, PCMCI and PCMCI$^+$, is better suited for this purpose.

In conclusion, while PCMCI$^+$ offers advantages such as the collider orientation phase and the ability to handle contemporaneous conditions, its practical implementation faced challenges in achieving parallelization efficiency. Despite efforts to parallelize certain steps, such as the collider orientation phase, limitations arose due to the intricate dependencies within the algorithm. Moving forward, further optimization of parallelization strategies, particularly in steps 3 and 4, could enhance runtime performance. However, it's important to note that the parallelization of step 2, the PC skeleton phase, presents inherent difficulties due to its sequential nature. Thus, while PCMCI$^+$ shows promise, ongoing efforts in optimization are crucial for its effective application in large-scale analyses. Due to runtime constraints, PCMCI$^+$ could not be applied across various parameter settings, particularly in regions with higher PC-α values.

When comparing PCMCI to PCMCI$^+$, the evaluation of comparing different climate models revealed that PCMCI was the superior methodology for capturing and quantifying discrepancies between models effectively. PCMCI showed several advantages over PCMCI$^+$, including the presence of the additional hyperparameter MCI-α for fine-tuning link selection and its fully parallelized implementation, which enabled faster computations. Further optimization of parallelization, particularly in steps 3 and 4, holds potential for improving PCMCI$^+$ performance in future studies, while step 2 remains inherently unparallelizable due to its sequential nature, where each iteration's outcome is dependent on the previous one, preventing successful parallelization without impacting the accuracy of the outcomes. The flexibility of PCMCI in choosing PC-α values enabled the identification of best setups to improve the discrimination between the climate models. In contrast, PCMCI$^+$ faced challenges in finding the best performing PC-α values, leading to inferior performance compared to PCMCI. Therefore, PCMCI emerged as the preferred method for this study, with the best-performing hyperparameter setting identified as PC-$\alpha = 3 * 10^{-20}$ and MCI-$\alpha = 10^{-4}$, applied to all 100 components and the full time series data. The best hyperparameter was found using a grid search method, where trends could be observed, such as the consistent range within which the optimal MCI-α value resided and the observation that using the full number of components generally yielded better performance. Consequently, the focus narrowed to finding the optimal PC-α parameter, streamlining the search process.

In conclusion, utilizing the identified best-performing hyperparameter settings, the various climate models can be compared against observational data using the F_1-score. The identified hyperparameter setting should aim to retain as many climate model-specific links

as possible, as this results in a higher discrepancy between the climate models compared to using other tested hyperparameters. The question arises as to which climate model, with its model-specific links, is closest to the observational data. It is observed that most climate models perform 'similarly poorly' with their model-specific links, with HadGEM3-GC31-LL achieving the highest score. MIROC-ES2L, on the other hand, achieves by far the lowest score. When applied to the identified subnetwork, a similar pattern emerges (Figure 5.2, 5.1). The subnetwork with the most links ($threshold = (0, 30)$ or $threshold2 >= 27$) was utilized here to reduce overfitting. Higher values in $\frac{M_2}{M_1}$ were only achieved through very low $threshold2$ variants, which can be seen as overfitting. Additionally, evaluation was conducted based on the resulting subnetwork with a $threshold = (0, 0)$. The subnetwork comprises only model-specific links that occur solely within the respective model and not in the non-related climate models. It is evident that only IPSL-CM6A-LR intersects with observational data with a few links.

When it comes to the best performing subnetwork, two approaches were explored. The first approach involved determining a specific set of components, aiming to create a better-performing subnetwork consisting of a subet of the set of all 100 components. This iterative process used the idea that excluding certain components could increase the discrepancy between the climate models. Furthermore, the geographical location and connectivity between components were considered to identify and retain only those contributing significantly to the discrepancy. However, this method of iteratively excluding components did not yield satisfactory results. Despite considering geographical location and connectivity between components, excluding certain components did not consistently enhance the discrepancy between climate models as expected. This could be attributed to the intricate interactions and dependencies among components, where excluding one component may not necessarily lead to a proportional increase in discrepancy. Additionally, the complexity of climate systems might have made it challenging to isolate specific components that significantly contribute to model discrepancies solely based on their geographical or network properties. As a result, this approach did not effectively achieve the desired optimization of the subnetwork. However, it is possible that more complex approaches can be used to specifically remove components that do not contribute to climate model discrimination. Due to the complexity of the underlying mechanisms, this was not considered in more detail in this work. The second approach focused on constructing a subnetwork consisting of links that were deemed crucial for maximizing $\frac{M_2}{M_1}$. This involved calculating the F_1-score over a predefined subnetwork, allowing for more realistic calculations and reduced runtime compared to re-running PCMCI. Both approaches aimed to refine the subnetwork to improve the discrimination between climate models. These methodologies represent valuable strategies for optimizing the selection and configuration of components and links within a network, thereby enhancing the effectiveness of PCMCI in capturing and quantifying discrepancies within climate model ensembles.

In conclusion, the exploration of various metrics for measuring differences among climate model ensembles has provided valuable insights into optimizing the discrimination between different models. Among the metrics investigated, the modified F_1-score with the inclusion of a penalty term emerged as the top performer. This modified version effectively captures complex relationships between graphs by combining precision, recall, and the penalty term, thereby amplifying discrepancies between different models as they become less similar. The penalty term plays a crucial role in weighting false links more heavily, contributing to a higher level of discrepancy between the models. Additionally, the p-norms, particularly the L_1 and L_2 norms, showed promising performance and potential for further improvement. These norms measure distances between matrices representing ensemble values, providing a straightforward approach to quantify differences between climate models. It's worth noting that the modified F_1-score with a simple linear penalty term was employed in this study.

There is potential for further enhancement by incorporating more complex penalty terms that leverage additional information from provided graphs, value matrices, and p-matrices. Such refinements could potentially yield even better results in maximizing the discrepancy between climate models. Overall, the findings suggest that a combination of the modified F_1-score with an appropriate penalty term and p-norms holds promise for effectively measuring and optimizing the discrimination between climate model ensembles. Further research and experimentation in refining these metrics could lead to more accurate assessments of model differences, ultimately enhancing our understanding of climate variability and improving climate predictions.

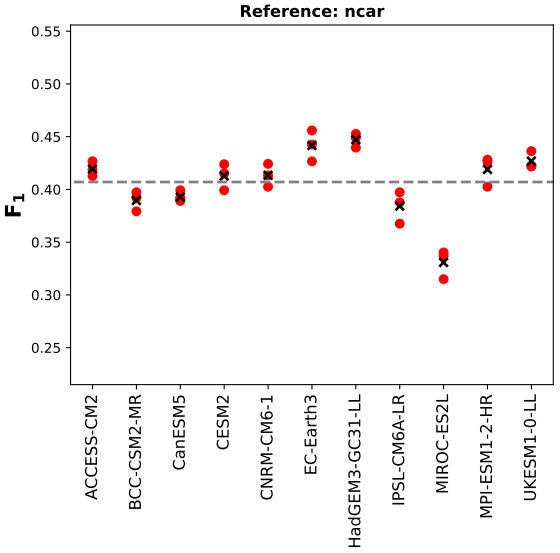


Figure 5.1: The plot visualizes the achieved F_1-scores of the ensemble graphs generated by various climate models when applying PCMCI to the entire dataset with PC-α set to $3 * 10^{-20}$ and MCI-α set to 10^{-4}.

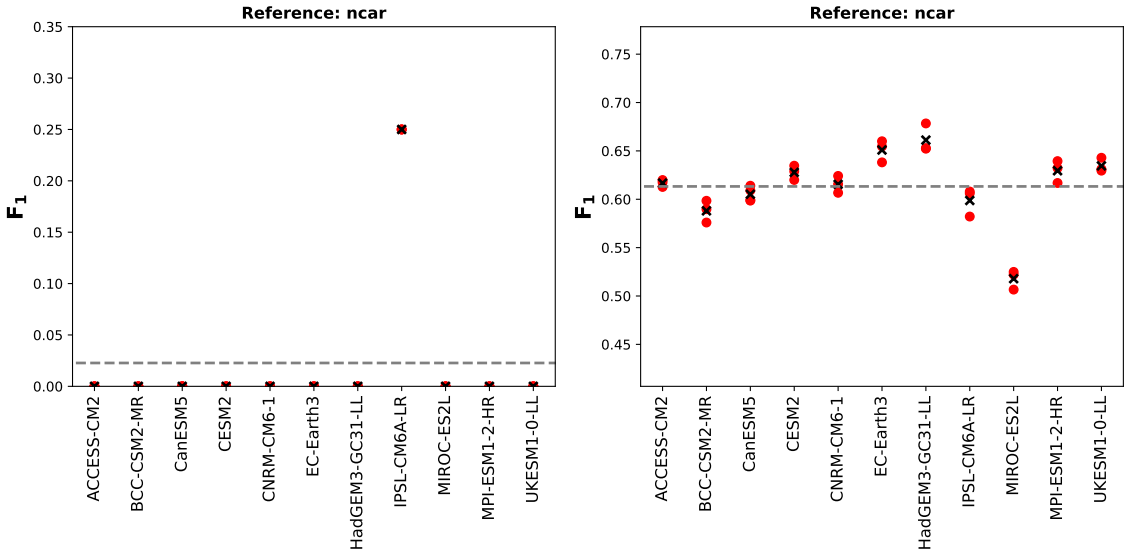
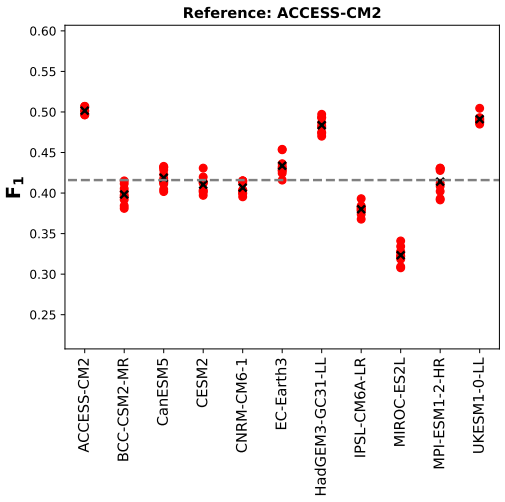
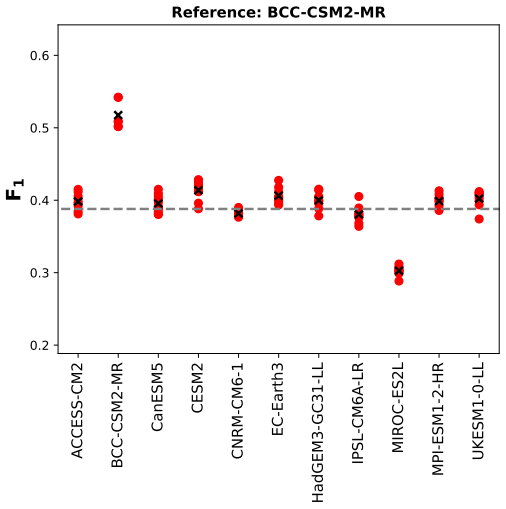


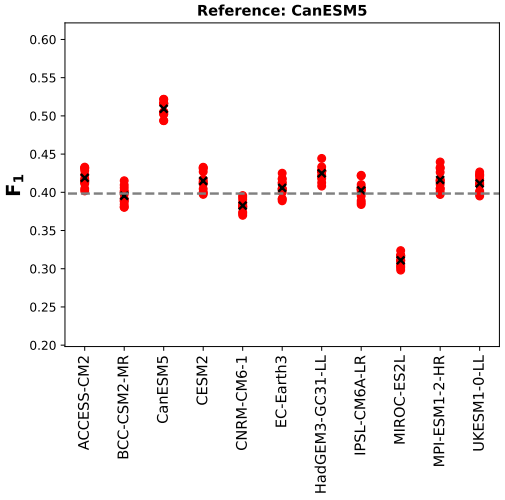
Figure 5.2: Both plots visualize the same comparisons with identical hyperparameters (optimal found hyperparameter setting), but on the computed subnetwork with a $threshold = (0, 0)$(left) $threshold = (0, 30)$ (right)

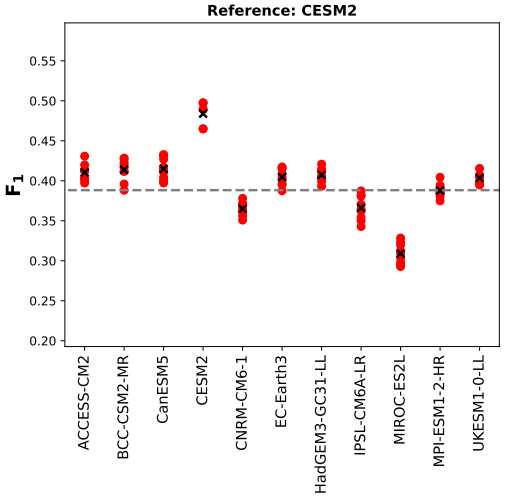
Appendix

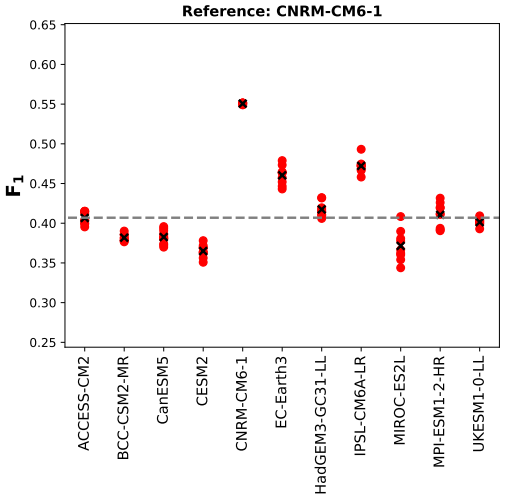
A. Appendix: F_1-scores

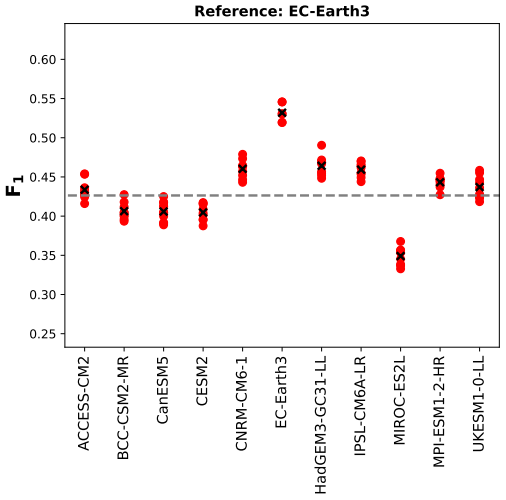

Reference: ACCESS-CM2
F1
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: BCC-CSM2-MR
F1
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CanESM5
F1
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CESM2
F1
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CNRM-CM6-1
F1
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: EC-Earth3
F1
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL

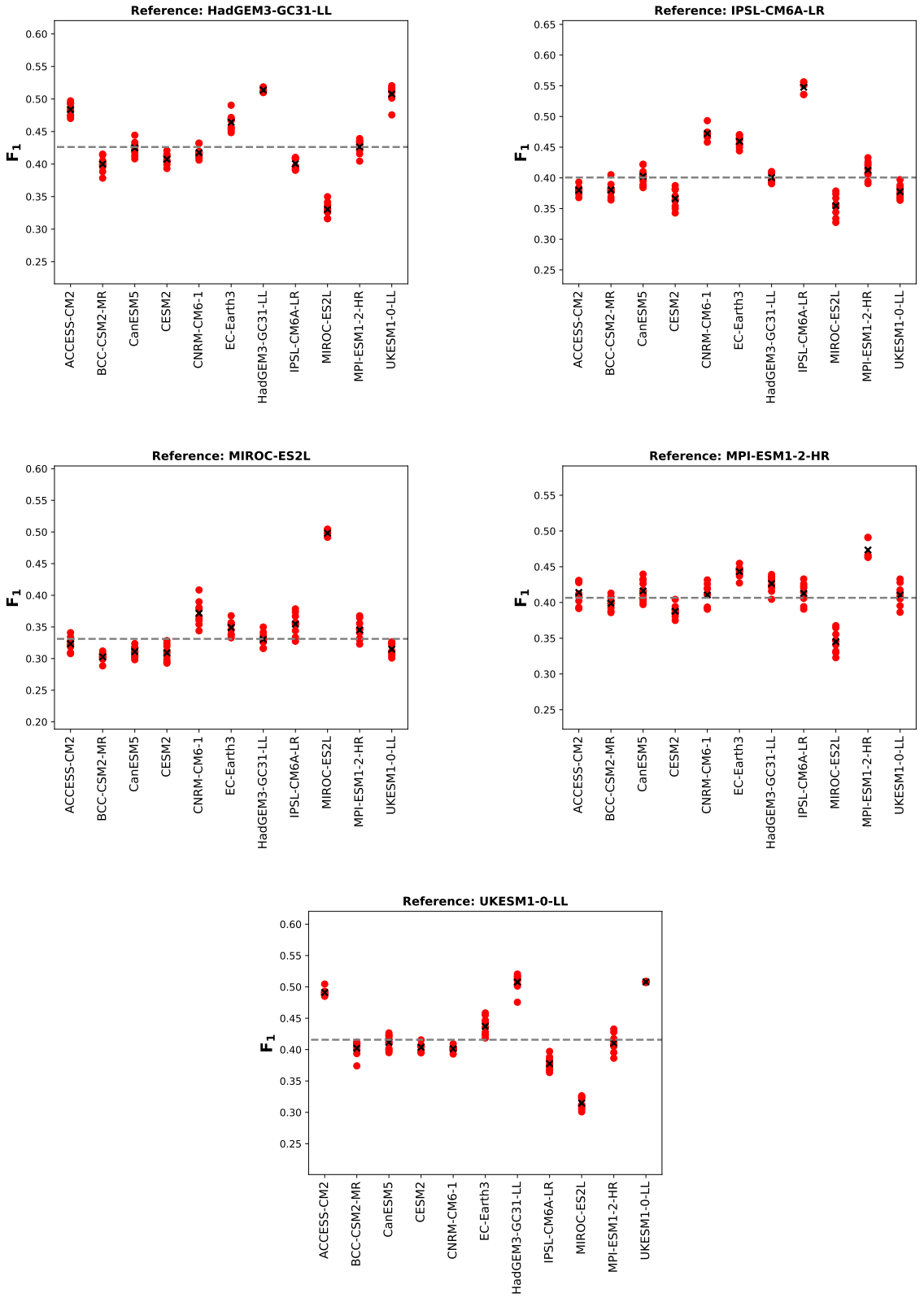
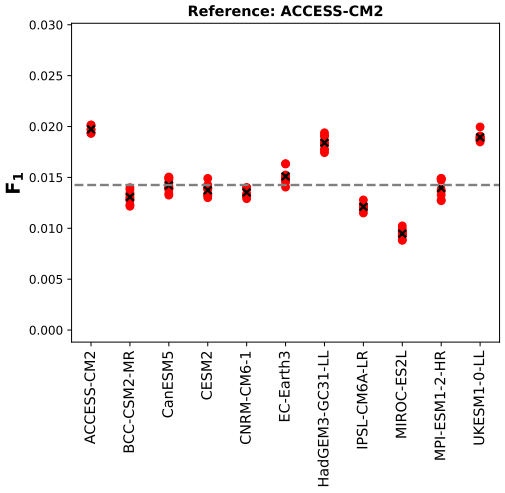
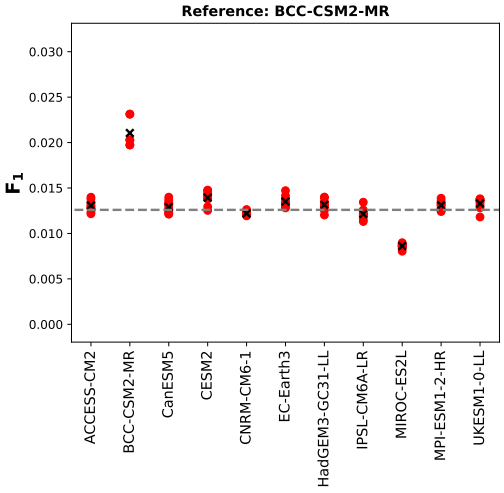
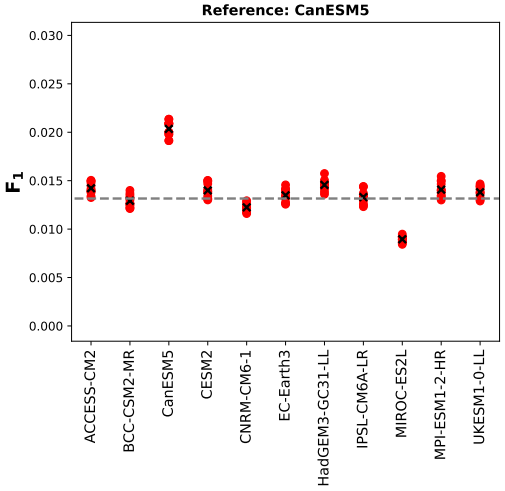
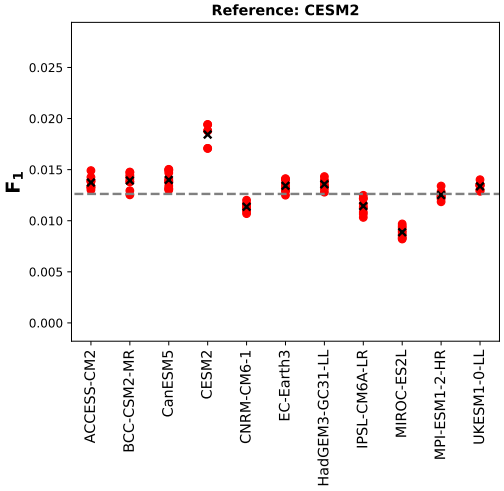


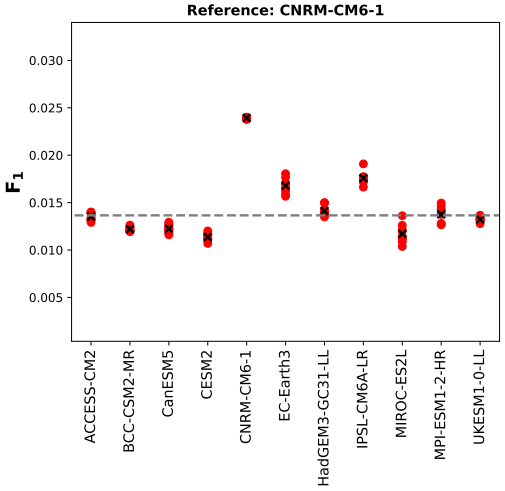
Figure A.0: The figures visualizes the various achieved F_1-scores of the resulting graphs of the climate models by applying PCMCI to the full dataset, including all 100 components, with PC-$\alpha = 3{*}10^{-20}$ and MCI-$\alpha = 10^{-4}$. The reference model is provided as the heading in each case. A red dot corresponds to the resulting F_1-score (y-axis) when an ensemble of a climate model (x-axis) is evaluated against an ensemble of the reference model.

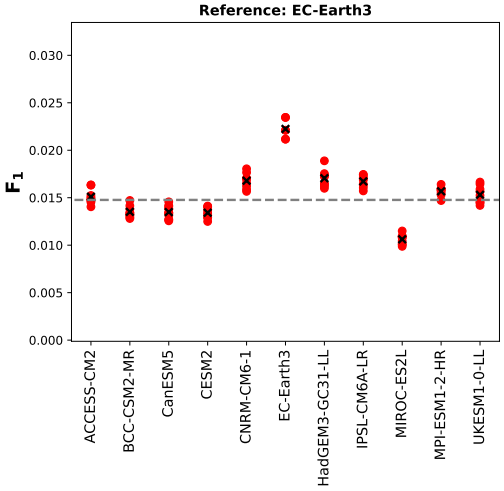

Reference: ACCESS-CM2
F1
0.030
0.025
0.020
0.015
0.010
0.005
0.000
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: BCC-CSM2-MR
F1
0.030
0.025
0.020
0.015
0.010
0.005
0.000
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CanESM5
F1
0.030
0.025
0.020
0.015
0.010
0.005
0.000
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CESM2
F1
0.025
0.020
0.015
0.010
0.005
0.000
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CNRM-CM6-1
F1
0.030
0.025
0.020
0.015
0.010
0.005
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: EC-Earth3
F1
0.030
0.025
0.020
0.015
0.010
0.005
0.000
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL

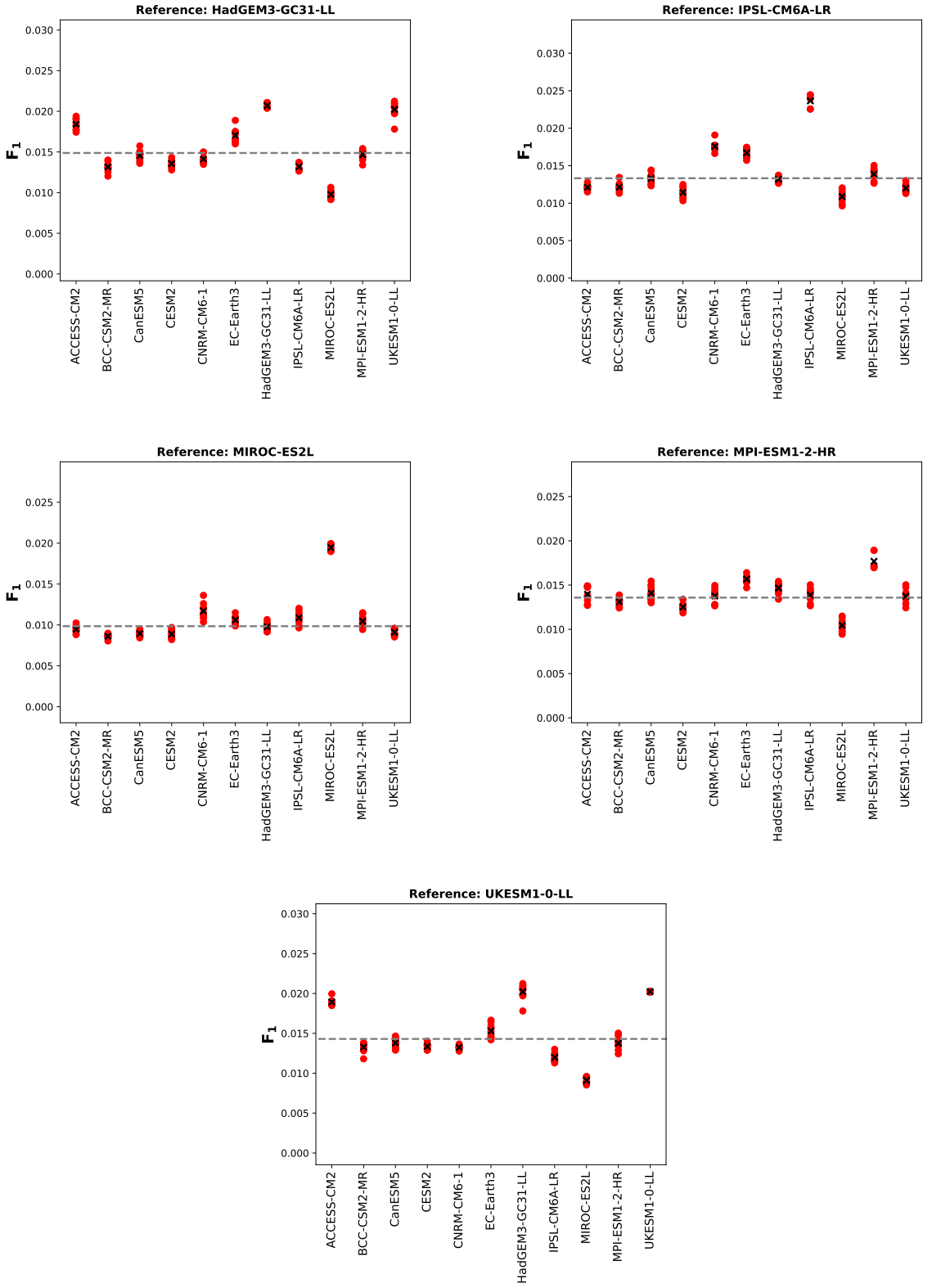
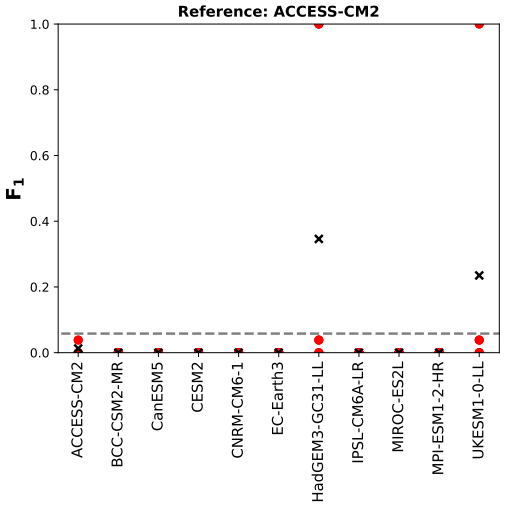
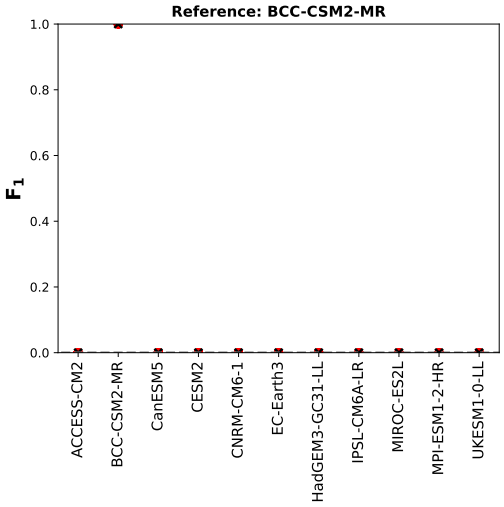
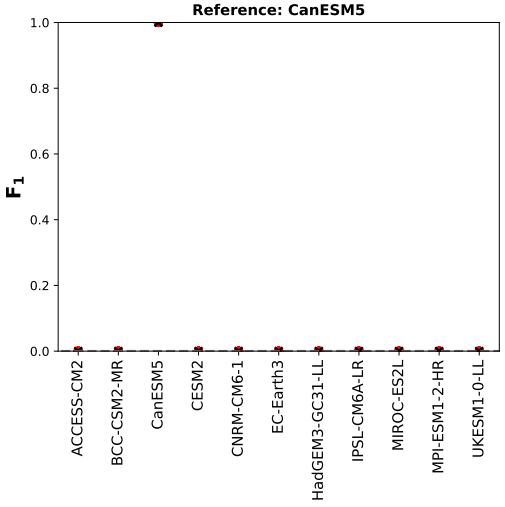
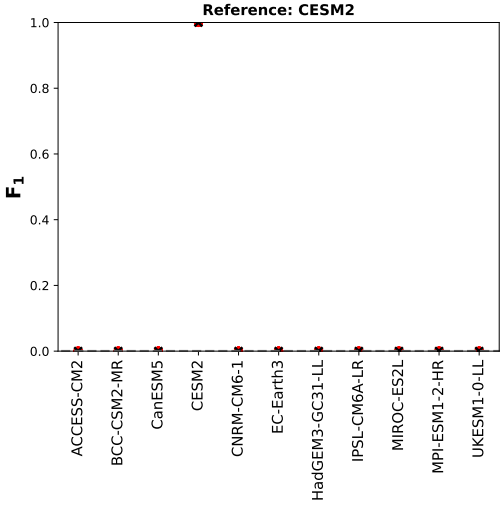


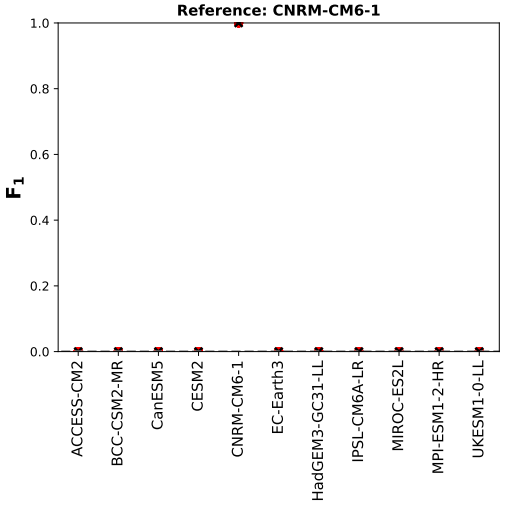
Figure A.1: The figures visualizes the various achieved F_1-scores (in this case, the modified F_1-score with penalty term $penalty = 50$) of the resulting graphs of the climate models by applying PCMCI to the full dataset, including all 100 components, with PC-$\alpha = 3 * 10^{-20}$ and MCI-$\alpha = 10^{-4}$. The reference model is provided as the heading in each case. A red dot corresponds to the resulting F_1-score (y-axis) when an ensemble of a climate model (x-axis) is evaluated against an ensemble of the reference model.

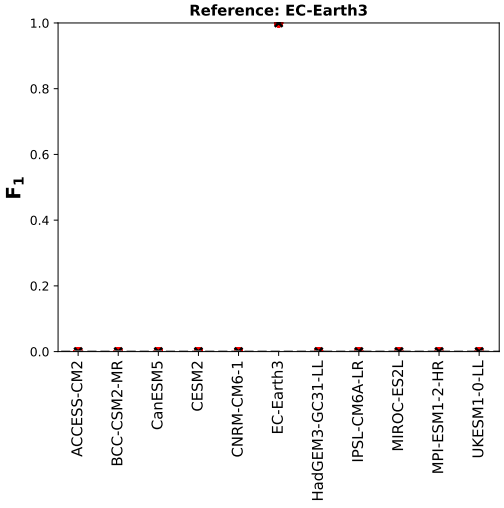

Reference: ACCESS-CM2
F1
1.0
0.8
0.6
0.4
0.2
0.0
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: BCC-CSM2-MR
F1
1.0
0.8
0.6
0.4
0.2
0.0
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CanESM5
F1
1.0
0.8
0.6
0.4
0.2
0.0
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CESM2
F1
1.0
0.8
0.6
0.4
0.2
0.0
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CNRM-CM6-1
F1
1.0
0.8
0.6
0.4
0.2
0.0
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: EC-Earth3
F1
1.0
0.8
0.6
0.4
0.2
0.0
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL

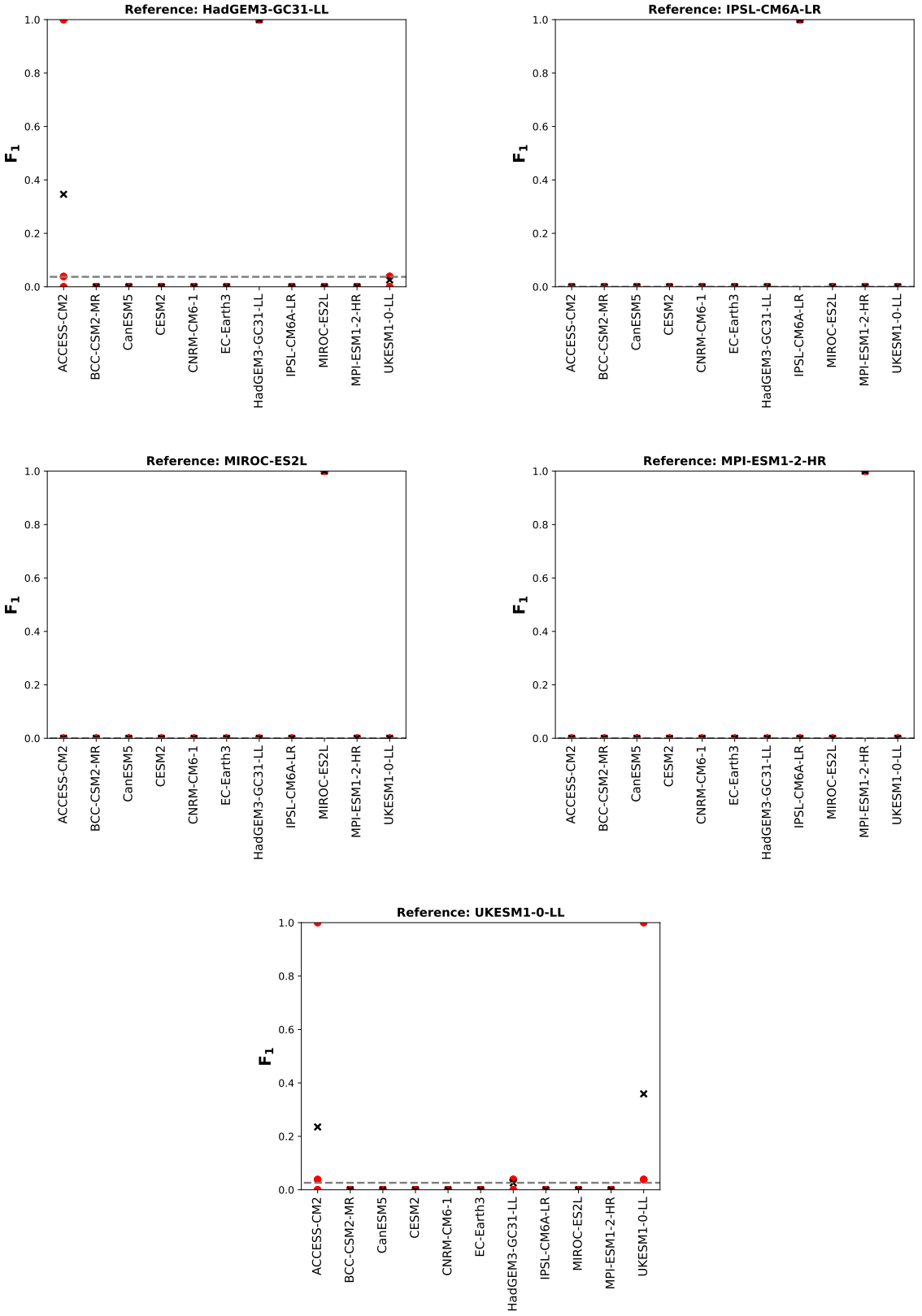
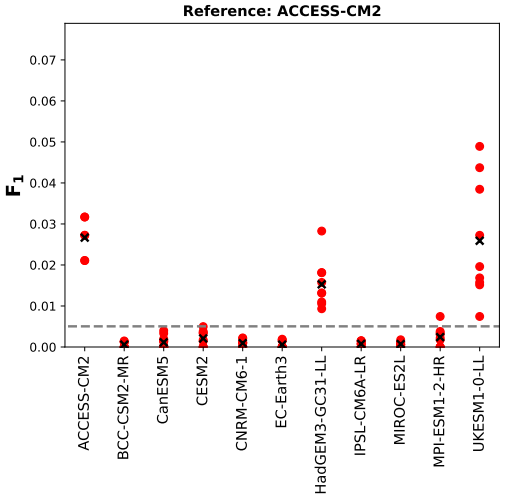
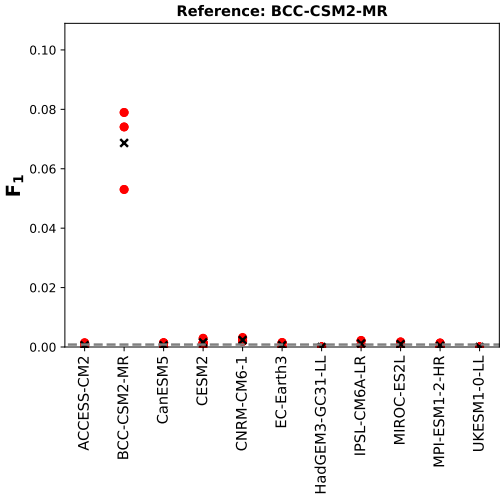
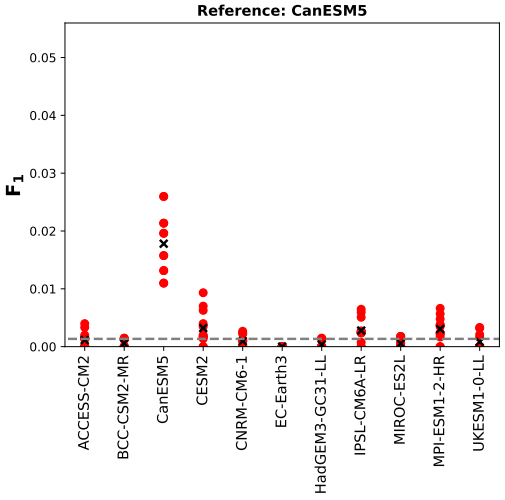
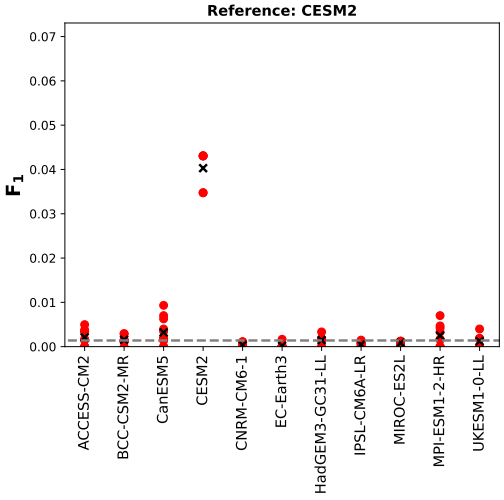


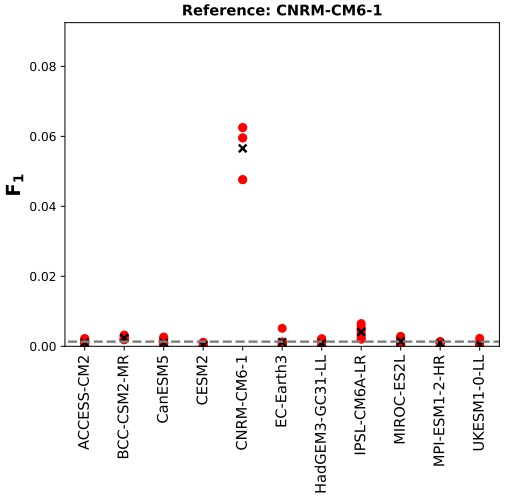
Figure A.2: The figures visualizes the various achieved F_1-scores (in this case, the modified F_1-score with penalty term $penalty = 50$) of the resulting graphs of the climate models by applying PCMCI to the full dataset, including all 100 components, with PC-α = $3 * 10^{-20}$ and MCI-$\alpha = 10^{-4}$, applied to the resulting subnetwork with $threshold = (0, 0)$. The reference model is provided as the heading in each case. A red dot corresponds to the resulting F_1-score (y-axis) when an ensemble of a climate model (x-axis) is evaluated against an ensemble of the reference model.

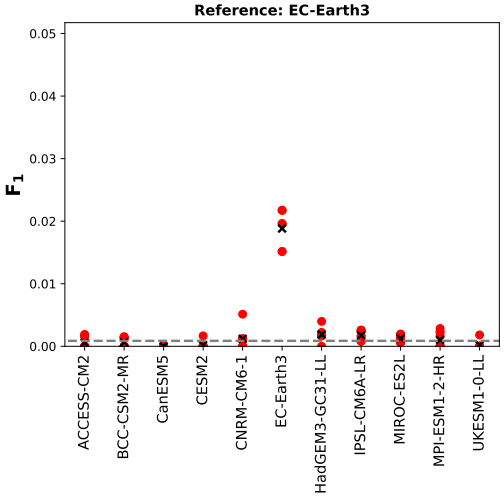

Reference: ACCESS-CM2
F1
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: BCC-CSM2-MR
F1
0.00
0.02
0.04
0.06
0.08
0.10
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CanESM5
F1
0.00
0.01
0.02
0.03
0.04
0.05
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CESM2
F1
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CNRM-CM6-1
F1
0.00
0.02
0.04
0.06
0.08
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: EC-Earth3
F1
0.00
0.01
0.02
0.03
0.04
0.05
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL

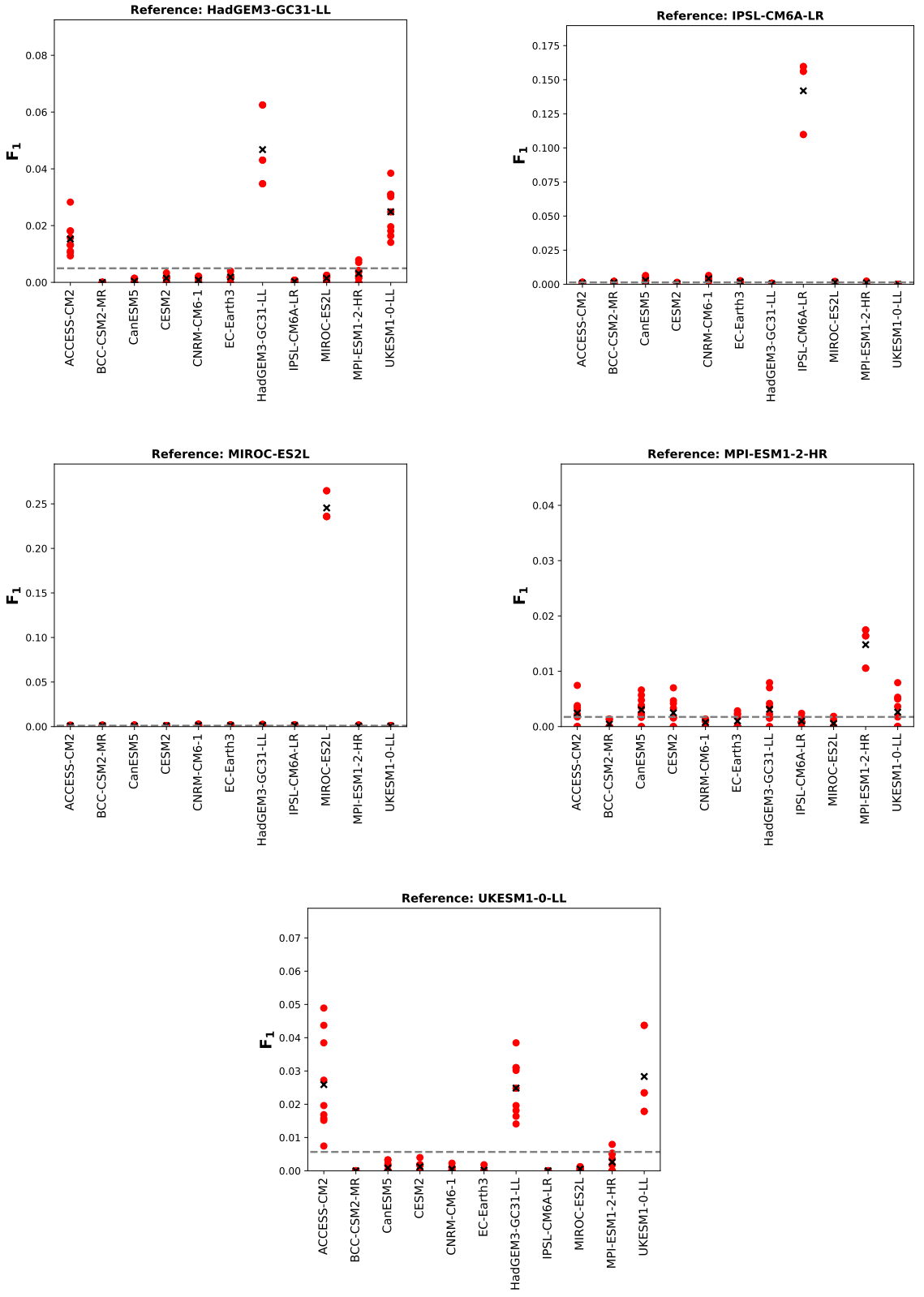
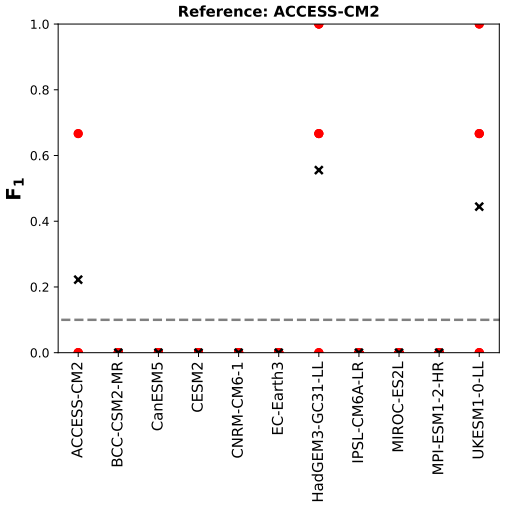
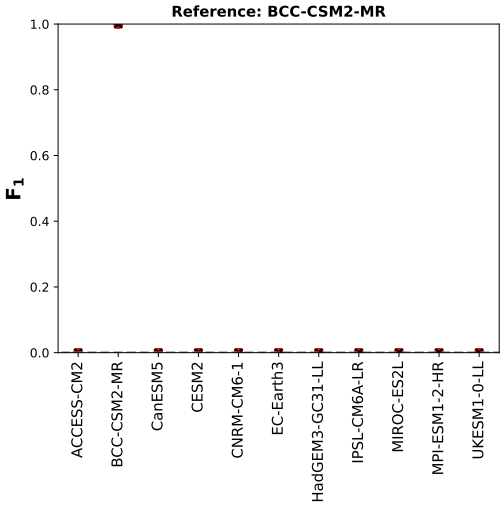
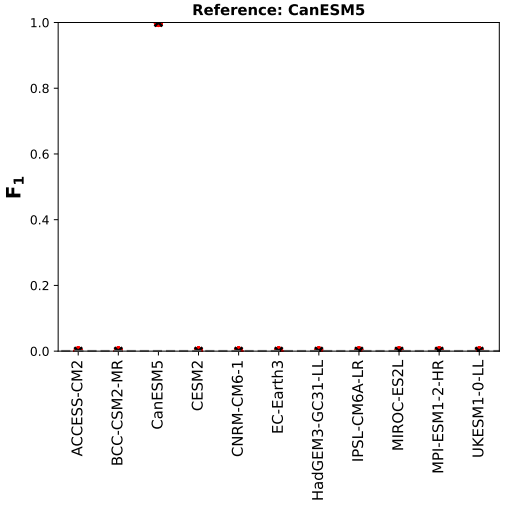
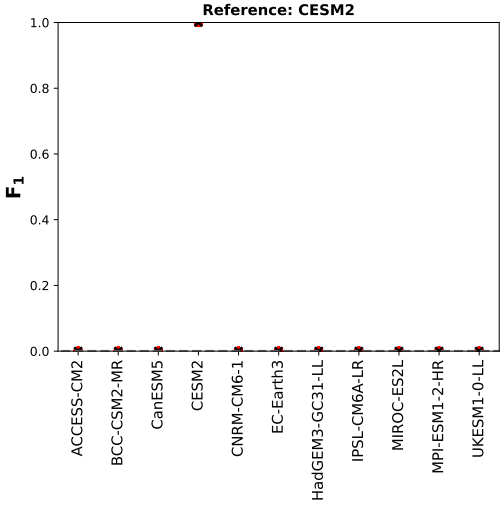


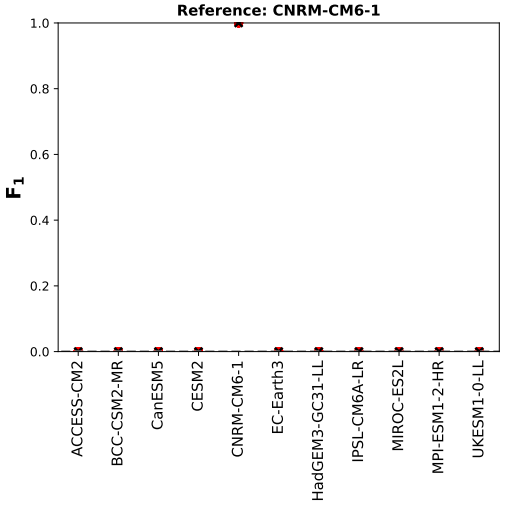
Figure A.3: The figures visualizes the various achieved F_1-scores (in this case, the modified F_1-score with penalty term $penalty = 50$) of the resulting graphs of the climate models by applying PCMCI to the full dataset, including all 100 components, with PC-α = $3 * 10^{-20}$ and MCI-$\alpha = 10^{-4}$, applied to the resulting subnetwork with $threshold = (0, 2)$. The reference model is provided as the heading in each case. A red dot corresponds to the resulting F_1-score (y-axis) when an ensemble of a climate model (x-axis) is evaluated against an ensemble of the reference model.

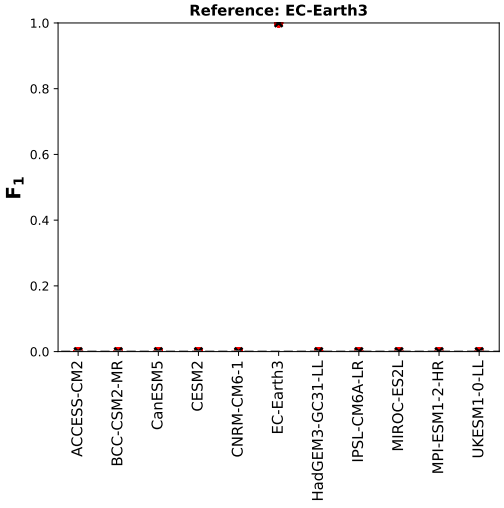

Reference: ACCESS-CM2
F1
1.0
0.8
0.6
0.4
0.2
0.0
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: BCC-CSM2-MR
F1
1.0
0.8
0.6
0.4
0.2
0.0
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CanESM5
F1
1.0
0.8
0.6
0.4
0.2
0.0
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CESM2
F1
1.0
0.8
0.6
0.4
0.2
0.0
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CNRM-CM6-1
F1
1.0
0.8
0.6
0.4
0.2
0.0
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: EC-Earth3
F1
1.0
0.8
0.6
0.4
0.2
0.0
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL

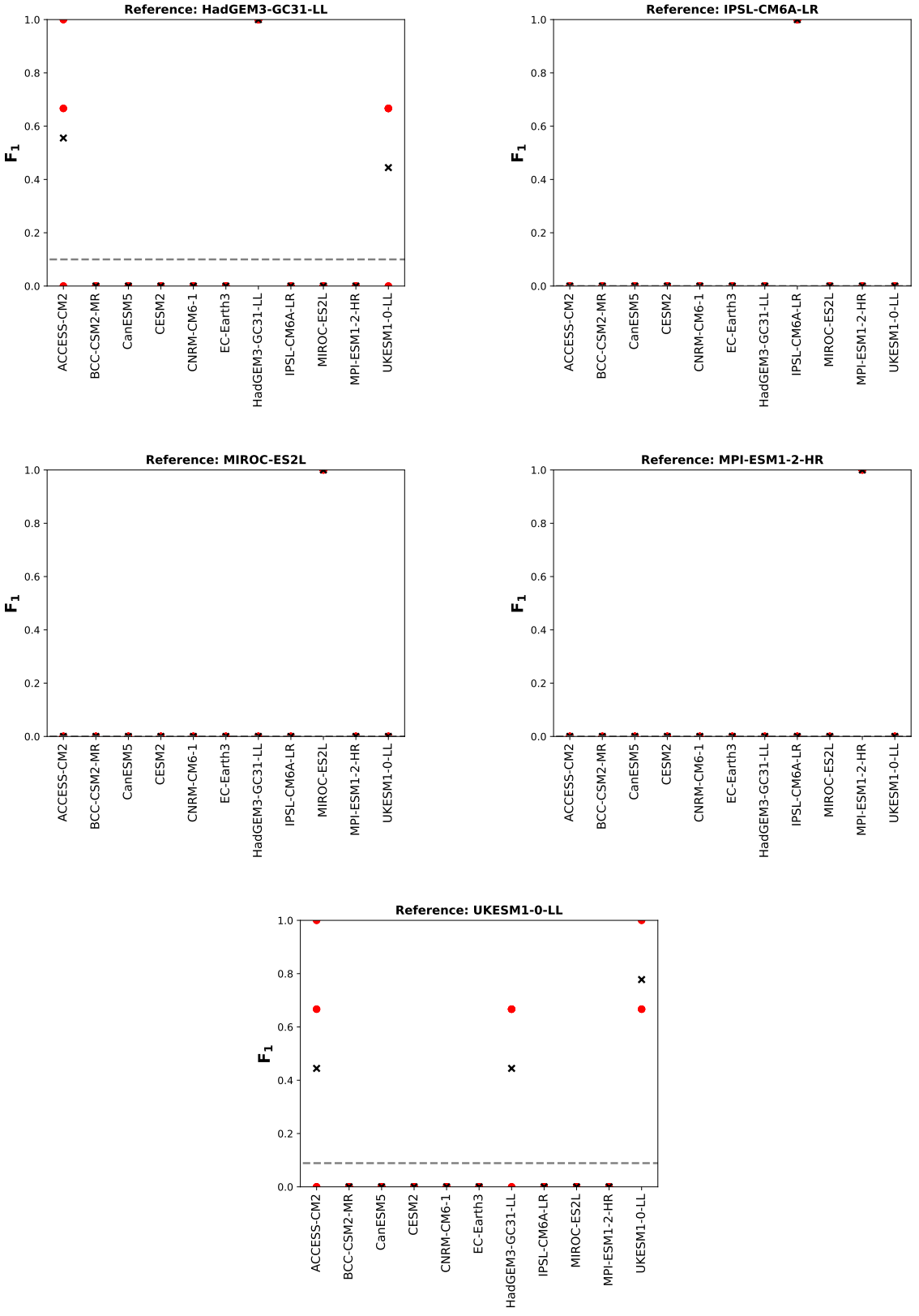
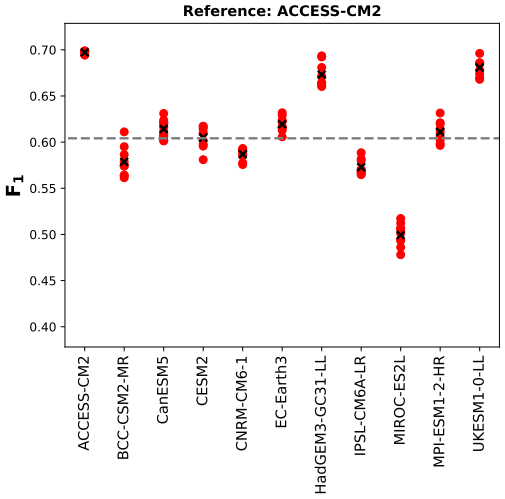
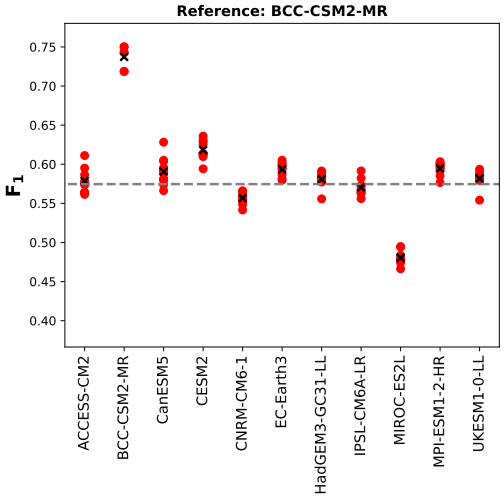
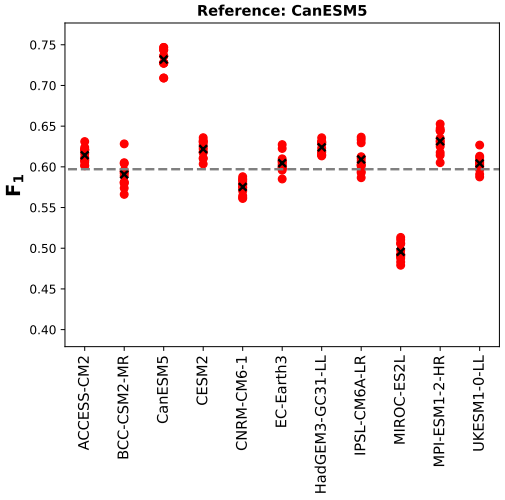
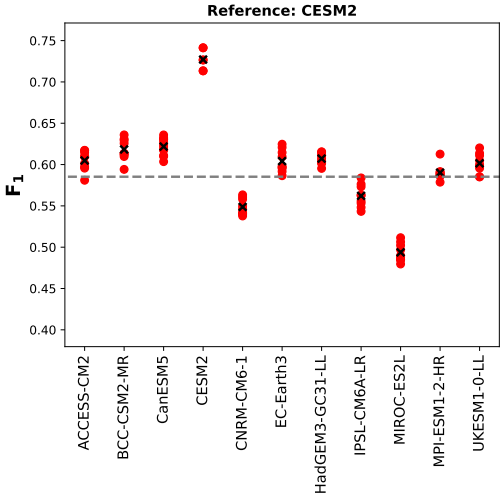


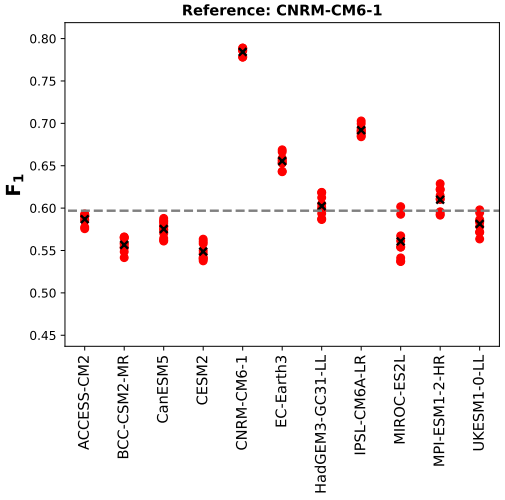
Figure A.4: The figures visualizes the various achieved F_1-scores (again not modified) of the resulting graphs of the climate models by applying PCMCI to the full dataset, including all 100 components, with PC-$\alpha = 3 * 10^{-20}$ and MCI-$\alpha = 10^{-4}$, applied to the resulting subnetwork with $threshold = (0, 2)$. The reference model is provided as the heading in each case. A red dot corresponds to the resulting F_1-score (y-axis) when an ensemble of a climate model (x-axis) is evaluated against an ensemble of the reference model.

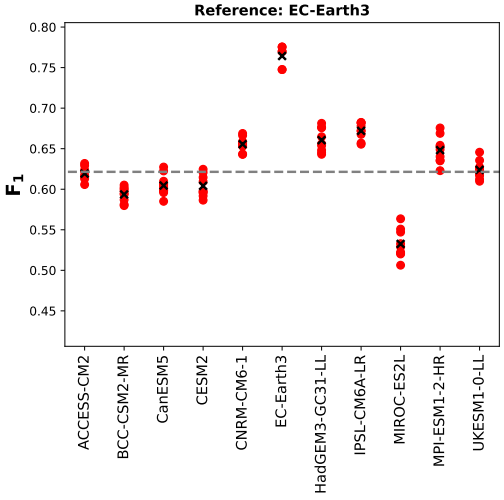

Reference: ACCESS-CM2
F1
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: BCC-CSM2-MR
F1
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CanESM5
F1
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CESM2
F1
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CNRM-CM6-1
F1
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: EC-Earth3
F1
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL

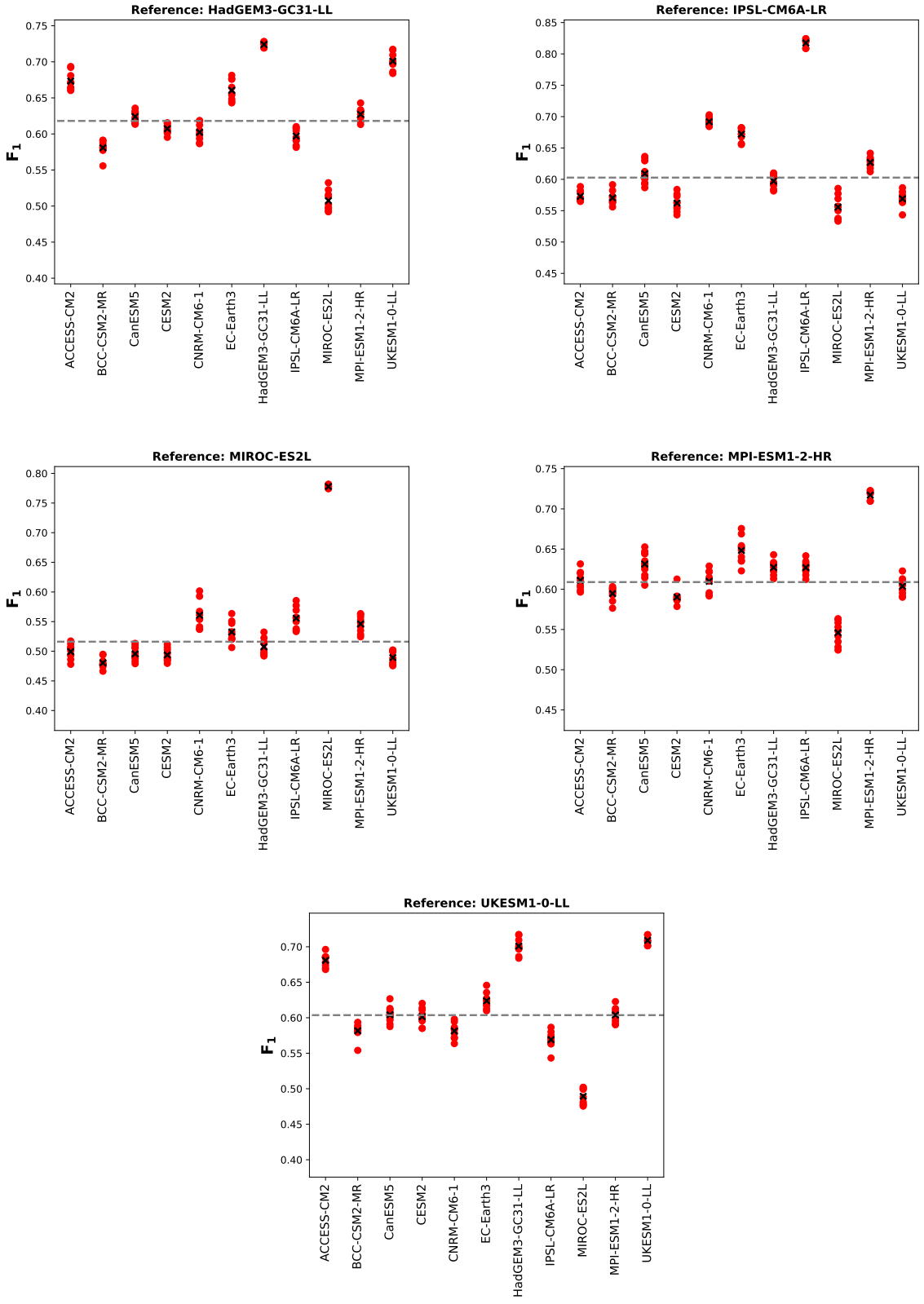
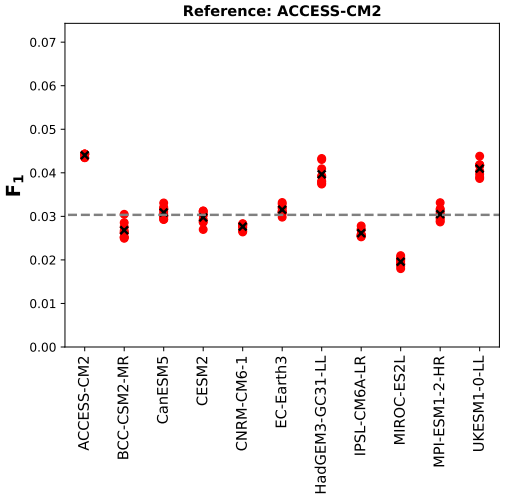
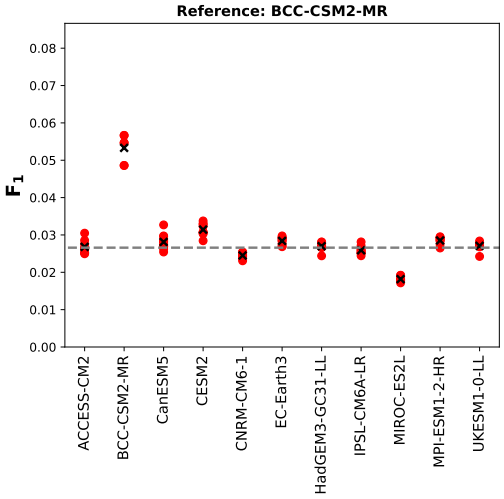
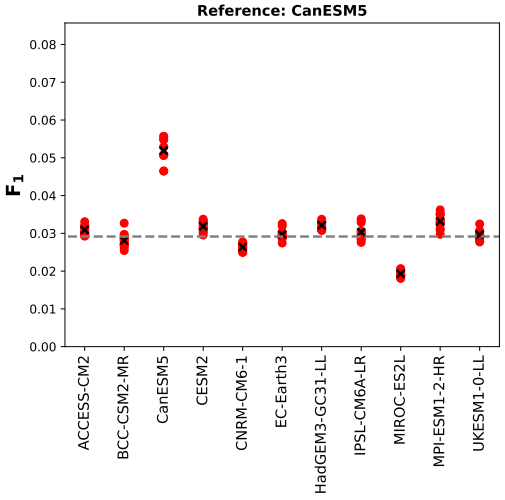
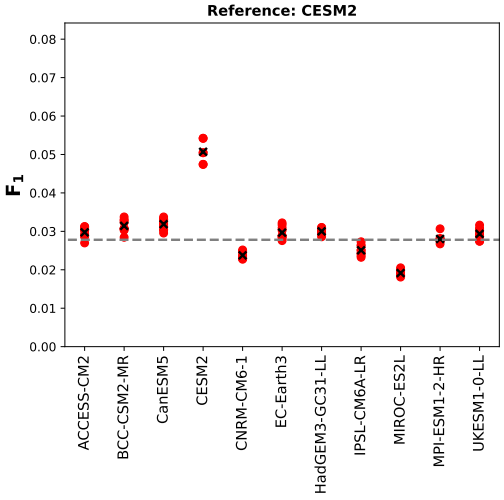


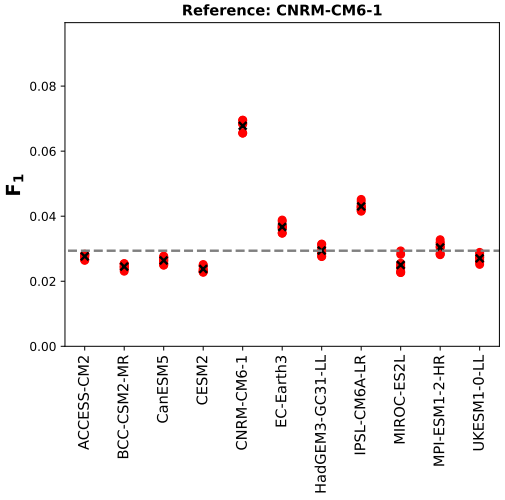
Figure A.5: The figures visualizes the various achieved F_1-scores (again not modified) of the resulting graphs of the climate models by applying PCMCI to the full dataset, including all 100 components, with PC-$\alpha = 3 * 10^{-20}$ and MCI-$\alpha = 10^{-4}$, applied to the resulting subnetwork with $threshold = (0, 30)$. The reference model is provided as the heading in each case. A red dot corresponds to the resulting F_1-score (y-axis) when an ensemble of a climate model (x-axis) is evaluated against an ensemble of the reference model.

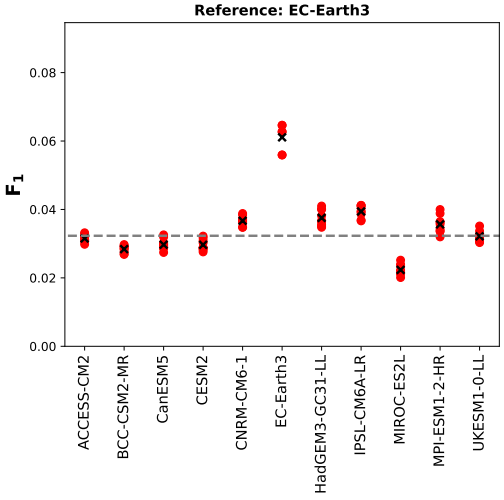

Reference: ACCESS-CM2
F1
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: BCC-CSM2-MR
F1
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CanESM5
F1
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CESM2
F1
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CNRM-CM6-1
F1
0.00
0.02
0.04
0.06
0.08
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: EC-Earth3
F1
0.00
0.02
0.04
0.06
0.08
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL

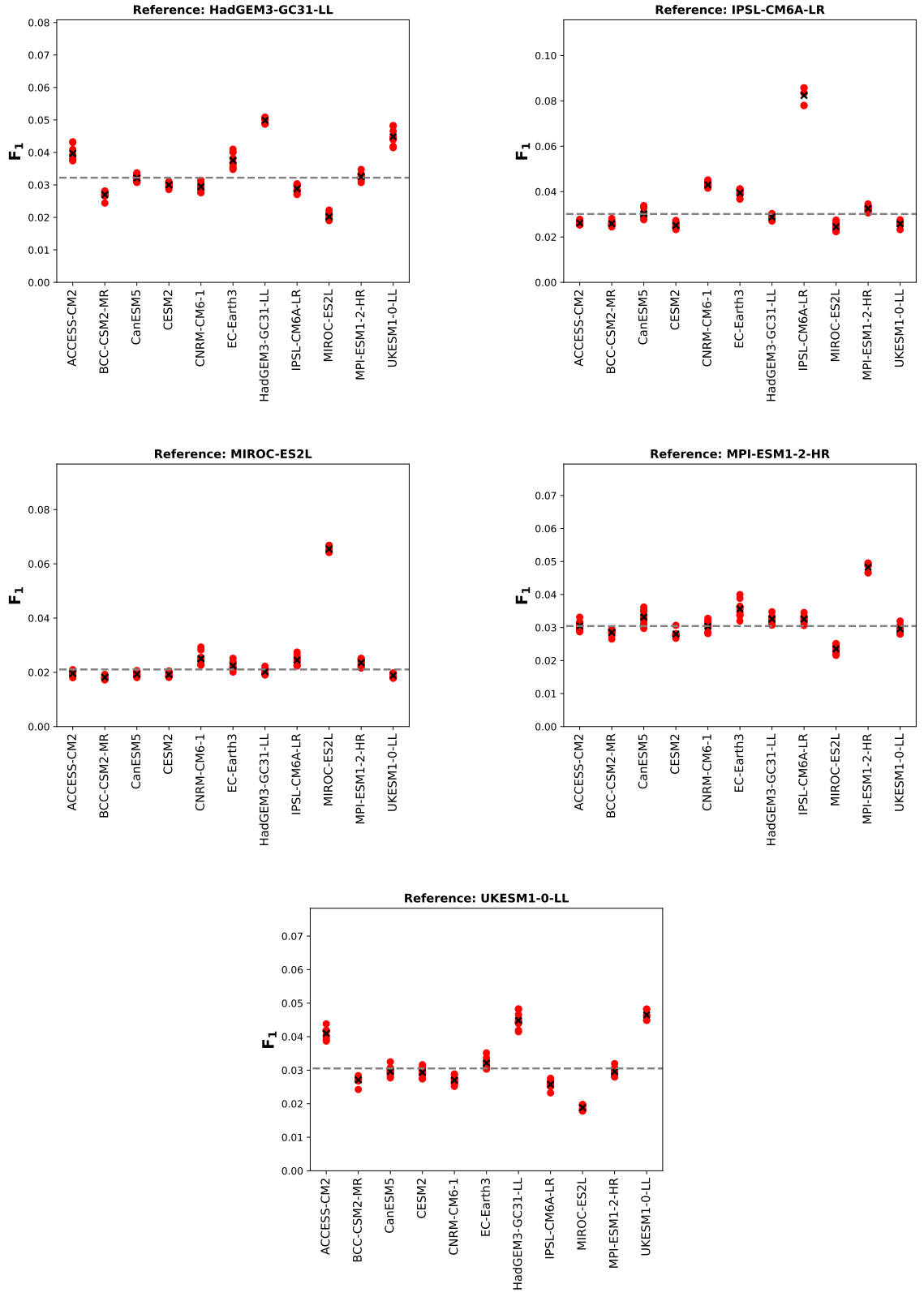
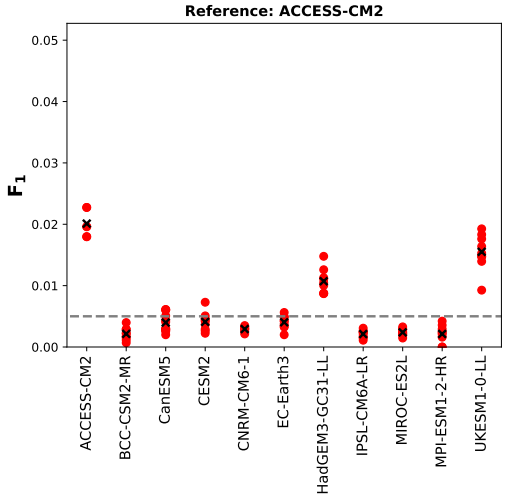
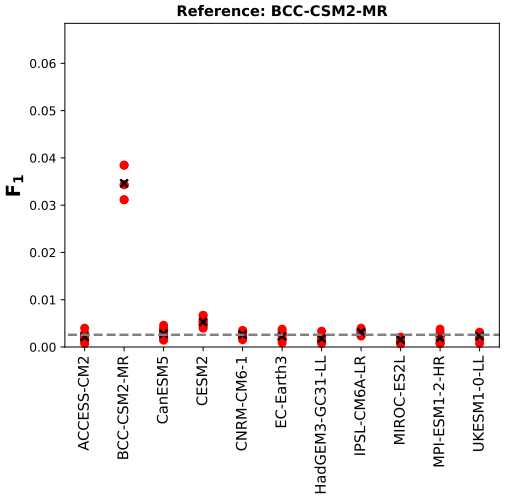
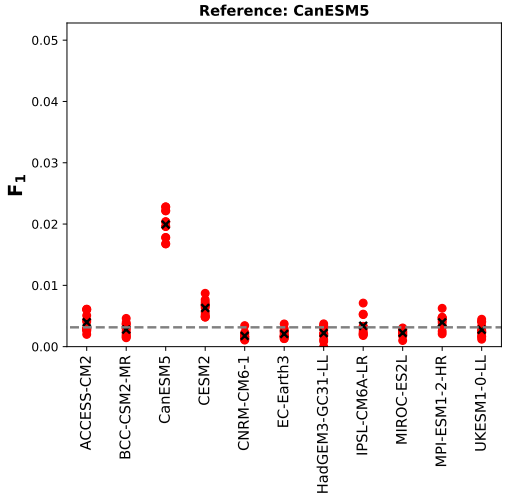
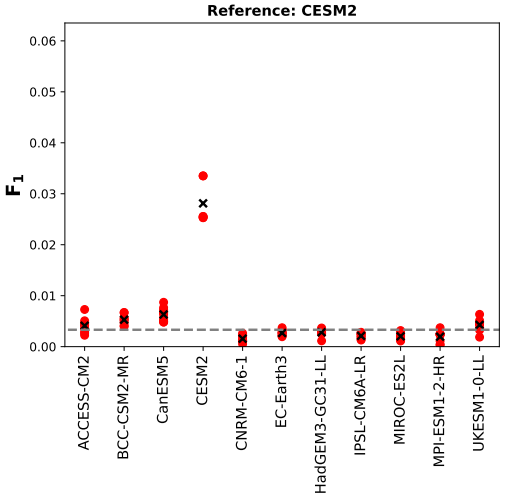


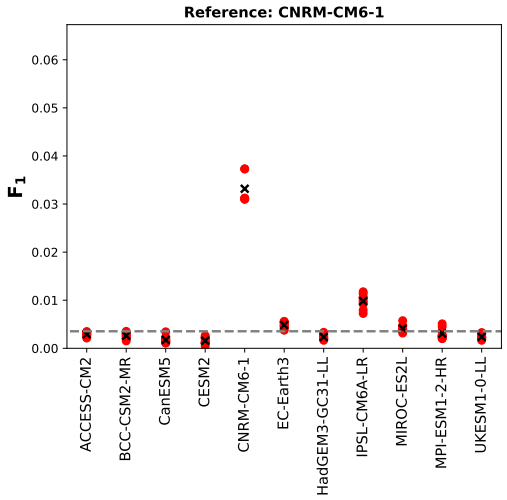
Figure A.6: The figures visualizes the various achieved F_1-scores (in this case, the modified F_1-score with penalty term $penalty = 50$) of the resulting graphs of the climate models by applying PCMCI to the full dataset, including all 100 components, with PC-α = $3 * 10^{-20}$ and MCI-$\alpha = 10^{-4}$, applied to the resulting subnetwork with $threshold = (0, 30)$. The reference model is provided as the heading in each case. A red dot corresponds to the resulting F_1-score (y-axis) when an ensemble of a climate model (x-axis) is evaluated against an ensemble of the reference model.

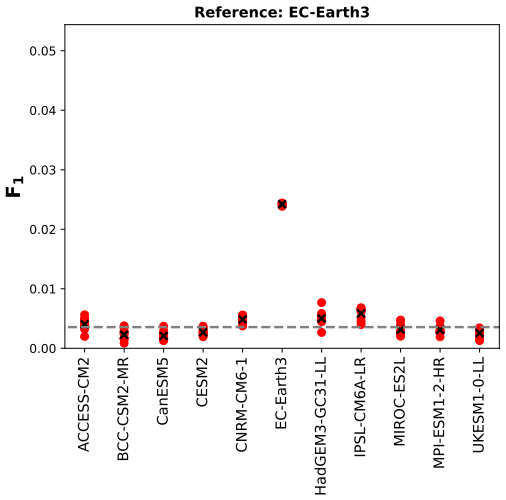

Reference: ACCESS-CM2
F1
0.00
0.01
0.02
0.03
0.04
0.05
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: BCC-CSM2-MR
F1
0.00
0.01
0.02
0.03
0.04
0.05
0.06
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CanESM5
F1
0.00
0.01
0.02
0.03
0.04
0.05
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CESM2
F1
0.00
0.01
0.02
0.03
0.04
0.05
0.06
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: CNRM-CM6-1
F1
0.00
0.01
0.02
0.03
0.04
0.05
0.06
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL


Reference: EC-Earth3
F1
0.00
0.01
0.02
0.03
0.04
0.05
ACCESS-CM2
BCC-CSM2-MR
CanESM5
CESM2
CNRM-CM6-1
EC-Earth3
HadGEM3-GC31-LL
IPSL-CM6A-LR
MIROC-ES2L
MPI-ESM1-2-HR
UKESM1-0-LL

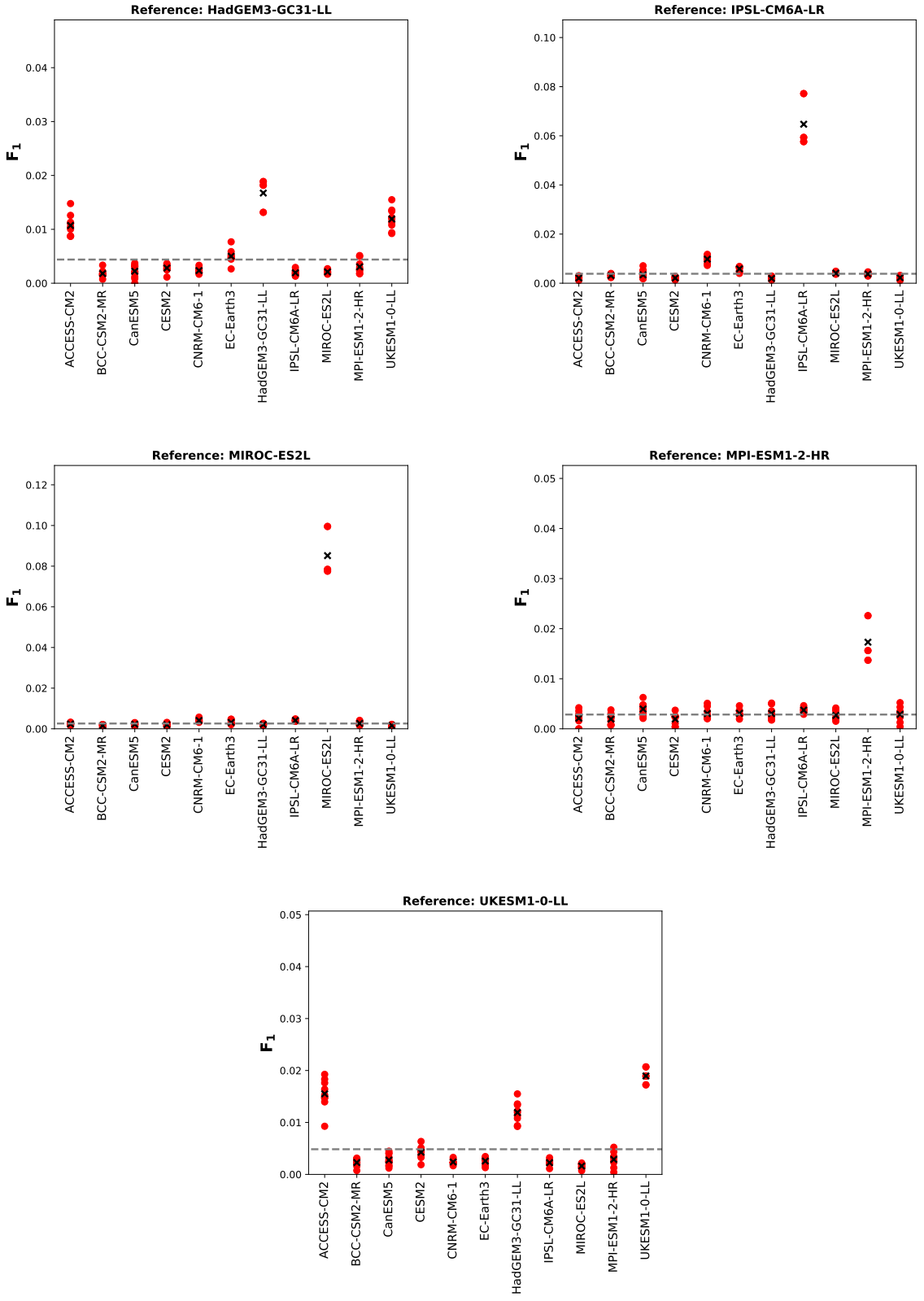


Figure A.7: The figures visualizes the various achieved F_1-scores (in this case, the modified F_1-score with penalty term $penalty = 50$) of the resulting graphs of the climate models by applying PCMCI to the full dataset, including all 100 components, with PC-α = $3 * 10^{-20}$ and MCI-$\alpha = 10^{-4}$, applied to the resulting subnetwork with $threshold = (0, 5)$. The reference model is provided as the heading in each case. A red dot corresponds to the resulting F_1-score (y-axis) when an ensemble of a climate model (x-axis) is evaluated against an ensemble of the reference model.

Bibliography

1. Runge, J. e. a. Causal inference for time series. *Nature Reviews Earth & Environment* **4,** 487–505. doi:10.1038/s43017-023-00431-y (2023).
2. Runge, J. *Tigramite* version 5.2. 2024. https://github.com/jakobrunge/tigramite.
3. Nowack, P. e. a. Response of stratospheric water vapour to warming constrained by satellite observations. *Nature Geoscience* **16,** 577–583. doi:10.1038/s41561-023-01183-6 (2023).
4. Runge, J. Causal network reconstruction from time series: From theoretical assumptions to practical estimation. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **28,** 075310. doi:10.1063/1.5025050 (2018-07).
5. Janzing, D. & Schölkopf, B. Causal inference using the algorithmic Markov condition. *IEEE Transactions on Information Theory* **56,** 5168–5194 (2010).
6. Nowack, P. e. a. Detecting and quantifying causal associations in large nonlinear time series datasets. *Science Advances* **5,** eaau4996. doi:10.1126/sciadv.aau4996 (2019).
7. Krich, C. e. a. Estimating causal networks in biosphere–atmosphere interaction with the PCMCI approach. *Biogeosciences* **17,** 1033–1061. doi:10.5194/bg-17-1033-2020 (2020).
8. Zhang, J. e. a. Weakening faithfulness: some heuristic causal discovery algorithms. *International Journal of Data Science and Analytics* **3,** 93–104. doi:10.1007/s41060-016-0033-y (2017-03).
9. Harris, N. & Drton, M. PC algorithm for nonparanormal graphical models. *Journal of Machine Learning Research* **14.** https://jmlr.org/papers/v14/harris13a.html (2013).
10. Runge, J. *Discovering contemporaneous and lagged causal relations in autocorrelated nonlinear time series datasets* in *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)* (eds Peters, J. & Sontag, D.) (PMLR, 2020), 1388–1397. https://proceedings.mlr.press/v124/runge20a.html.
11. D., B. *CiteDrive brings reference management to Overleaf* https://www.futurelearn.com/info/courses/climate-intelligence-using-climate-data-to-improve-business-decision-making/0/steps/300835. Accessed: (08.03.2024). n.d.
12. Tebaldi, C. e. a. Climate model projections from the Scenario Model Intercomparison Project (ScenarioMIP) of CMIP6. *Earth Syst. Dynam.* **12,** 253–293. doi:10.5194/esd-12-253-2021 (2021).
13. Wilcke, R. A. & Bärring, L. Selecting regional climate scenarios for impact modelling studies. *Environmental Modelling Software* **78,** 191–201. doi:https://doi.org/10.1016/j.envsoft.2016.01.002 (2016).

14. Met Office. *Cartopy: a cartographic python library with a matplotlib interface* version 0.21.1 (Exeter, Devon, 2010 - 2015). https://scitools.org.uk/cartopy.

15. Sinigalia, K. *Thesis* version 1. 2024. https://github.com/KevinSinigalia/bachelorThesis.

16. Zhang, J. & Luo, Y. *Degree Centrality, Betweenness Centrality, and Closeness Centrality in Social Network* in *Proceedings of the 2017 2nd International Conference on Modelling, Simulation and Applied Mathematics (MSAM2017)* (Atlantis Press, 2017/03), 300–303. doi:10.2991/msam-17.2017.68.

17. EC-Earth Consortium. *EC-Earth* https://ec-earth.org/ec-earth/ (2024-03-29).

Acknowledgement

The authors acknowledge support by the state of Baden-Württemberg through bwHPC.