UNIVERSITÉ DE LAUSANNE

DOCTORAL THESIS

Software and Numerical Tools for Paleoclimate Analysis

Author:
Philipp S. SOMMER

Supervisor:
Dr. Basil A. S. Davis

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Science

in the

The Davis Group
Institute of Earth Surface Dynamics (IDYST)

October 5, 2019

"The purpose of computing is insight, not numbers."

Richard Wesley Hamming

UNIVERSITÉ DE LAUSANNE

Abstract

Faculty of Geosciences and Environment (FGSE)
Institute of Earth Surface Dynamics (IDYST)

Doctor of Science

Software and Numerical Tools for Paleoclimate Analysis

by Philipp S. SOMMER

The Thesis Abstract is written here (and usually kept to just this page). The page is kept centered vertically so can expand into the blank space above the title too...

Acknowledgements

The acknowledgments and the people to thank go here, don't forget to include your project advisor...

For/Dedicated to/To my...

Contents

Chapter 1

Introduction

1.1 Motivation

Climate science and in particular the study of past climates face an increasing need for the analysis, standardization and sharing of data. Scientists made huge efforts to explore climate archives throughout the world to investigate the evolution of the Earths climate. In parallel, computational climate models grew in complexity and data ouput due to an increase of computational power and the availability of supercomputers. This generates new challenges for big data analysis that can only solved by high quality and flexible software packages.

1: Add reference.

2: Add reference.

The Neotoma Database, a global international database for paleoenvironmental proxies (Williams et al., 2018) currently lists XXX datasets with in total XXX samples for past 12'000 years, the Holocene. Such data collections enable large-scale reconstructions of past climates that however face considerable challenges. They mainly arise from the heterogeneity of the data and the necessity of further quality control and standardization. One key problem is the accessibility of data. A lot of data is not available in standardized relational databases and either held private, or is stored in less standardized archives such as PANGAEA, or is not available in a digital format at all. The latter often results in the need of digitizing the associated data from a published diagram, a tedious and imprecise task (Sommer et al., 2019). Additionally handling such a big heterogeneous data resource and analyzing its contents is a key challenge and requires flexible visualization resources that efficiently allow the querying of spatial data with heterogeneous time and meta data information.

3: Add reference. https://pangaea.de/

4: Add reference. EMPD paper

An additional challenge arises from the combination with numerical models that usually operate on a structured (Edwards, 2010; Treut et al., 2007) or unstructured grid with a fixed timestep. The development and analysis of such models requires visualization techniques that are interoperable with the specific data structure of the model (e.g. S. A. Brown et al., 1993; Rew and G. Davis, 1990) while still being flexible enough for general purposes and computations (Hoyer and Hamman, 2017; Sommer, 2017). Additionally it requires techniques to process observational data to make it comparable with climate models (Mauri et al., 2015) or to feed a model with the data using data assimilation of statistical models (Sommer and Kaplan, 2017).

5: Add reference. ICON

6: Add reference. POLNET-gridding paper

In the following section 1.2 I will lay down the interest in the study of paleoclimates, both from the observational and the modellers perspective. This is continued by a section 1.3 which highlights the specific requirements and the historical development of software in paleo-science and concludes with section 1.3.3 that provides an overview on the contents of this thesis.

1.2 Learning from the Past – Why we study paleo-climates

Mankind is facing large infrastructural challenges during this century, such as the loss of biodiversity, an exponentially growing world population and an acceleration of growth and globalization of markets. They all interact with a global climate change that may lead to a new environment none of us ever experienced. Any future global planning has to account highly diverse responses that range from regional to continental scales. The complex (climate) system will enter a state that is significantly different from everything we had since the beginning of the satellite era in the 19th century, the beginning of global meteorological data acquisition.

7: Add reference.

8: Add reference.

9: Add reference. cite World bank report?

10: Add reference.

Our knowledge about this new climate is therefore mainly based on computational Earth System Models (ESMs). They face the challenge of simulating a new climate based on our present knowledge of the interactions between the different compartments Ocean, Land and Atmosphere. Running such a model for the entire Earth with a reasonable resolution is therefore very cost-intensive and requires large computational resources. The validation of it becomes technically difficult considering the large amount of data output, and additionally conceptually difficult because of the aforementioned transition into a warmer world during the next century. We are entering a new state and it is questionable how well our models perform (Hargreaves et al., 2013; Karpechko, 2010; Ulden and Oldenborgh, 2006).

11: Add reference. check these references! taken from Achilles PhD thesis, there might be better ones

To evaluate their skill, we can only use our knowledge of the past climate from before the systematic measurement of temperature, precipitation, etc. These climates, also referred to as paleo-climates, provide the only opportunity to evaluate an ESM under conditions very different than today. paleo-climatic research has therefore been an integral part for climate sciences since the 80s (COHMAP Members, 1988; Joussaume and Taylor, 1995), particularly in the Paleoclimate Modelling Intercomparison Project (PMIP) (Braconnot et al., 2012, 2007a,b; Jungclaus et al., 2017; Kageyama et al., 2016; B. L. Otto-Bliesner et al., 2017).

The current geological period is the Quaternary. It is characterized by glacial-interglacial cycles mainly driven by orbital changes (Hays et al., 1976; Imbrie et al., 1993)that cause a varying insolation on the planet.

12: Add reference. Check these

The end of this period can be used for data-model comparisons due to the availability of paleo-climate archives. It started with the Last Interglacial (LIG) about 127'000 years ago and was followed by the Last Glacial Maximum (LGM) at 21'000 years ago. The warming of the atmosphere in the following interglacial has been interrupted by a rapid cooling, called the Younger Dryas, between 12'900 and 11'700 years ago, which then let to the onset of the current epoch, the Holocene (Walker et al., 2009) .

13: Add reference. check ibid.

Add some background on the Holocene. How did it change (global mean temperature estimate?), how was the insolation? CO_2 effects, impact of the ice sheets during the early holocene, changes in altitude, large-scale atmospheric circulation, human influences.

This epoch is of particular interest because the continental setup is comparable to nowadays while still having a climate that is significantly different from present day . Additionally we have a large set of proxies available to quantify the climate, independent from the model estimates, and for the entire globe (Wanner et al., 2008) .

14: Add reference. PMIP paper

15: Add reference. check ibid.

1.2.1 Proxy Data from the Holocene

Before 1850, there is almost no instrumental measurement of temperature. Instead we rely on archives such as lake sediments, glaciers, peat bogs, or speleothems that preserve climate proxies. The latter is a set of variables that are influenced by climate conditions and therefore allow an indirect measurement of climate parameters at ancient times, e.g. temperature, precipitation or sea-level. The most prominent proxies are isotopic compositions of $\delta^{18}O$ in glacial ice cores, marine sediments, peat bogsor speleothems; bio-ecologic assemblages such as pollen, chironomids or diatoms in lake sediments; foraminifera and alkenone in marine sediments; and the widths of tree rings.

The most abundant climate proxy, that I will also focus on in the next chapters, are pollen assemblages. It is the geographically most wide spread paleo-climate proxy (H. J. B. Birks and H. H. Birks, 1980) and has a long history in quantitative paleo-climatologic reconstructions (e.g. Bradley, 1985; Nichols, 1967, 1969).

The ability to serve as a proxy for the past arises from the chemically stable polymer sporopollenin, that allows it to be preserved over very long periods of time, in various environments such as lakes, wetlands or ocean sediments (Fægri et al., 1989; Havinga, 1967). Pollen are produced by seed-bearing plants (spermatophytes, Wodehouse, 1935) and as such have a high spatial continuity and prevalence. Their compositions (closely related to the surrounding vegetation) is highly dependent on the climate and allows the reconstruction of the latter through an inverse modelling approach .

The usefulness for large-scale data-model intercomparisons additionally arises from the existence of regional databases for fossil pollen assemblages. The earliest examples are the European Pollen Database (EPD) and North American Pollen Database (NAPD) that both started aroun 1990 and developed a similar structure in order to be compatible (Fyfe et al., 2009; Grimm, 2008) . This let to the development of other regional pollen databases, such as the Latin American Pollen Database (LAPD) (LAPD, Flantua et al., 2015; Marchant et al., 2002) in 1994 or the African Pollen Database (APD) (Vincens et al., 2007) in 1996, and others (see Grimm, 2008). These attempts finally let to the development of the Neotoma database (Williams et al., 2018), a global multiproxy database that incorporates many of the regional pollen databases.

The use of the above-mentioned proxies, particularly pollen, for paleo-climate reconstruction has a long academic tradition in geology (Bradley, 1985) and provides the source of large-scale paleo-climatic reconstructions in number of different studies (B. A. S. Davis et al., 2003; Marsicek et al., 2018; Mauri et al., 2015; Neukom et al., 2019a,b). They however have multiple uncertainties, that are often difficult to estimate and to consider. The main challenge for a data-model comparison are dating uncertainties, but often also about the influence of seasonality on the proxy (e.g. whether it represents summer, winter or annual temperature) and the quality of the record. Another challenge is the proper handling of uncertainties of the inverse modelling approach, spatial coverage of the proxy (see chapter 3), and, considering pollen assemblages, the various naming schemes for pollen taxa that need to be considered for large-scale reconstructions.

16: Add reference.
17: Add reference.
18: Add reference.
19: Add reference.
20: Add reference.
21: Add reference.
22: Add reference.
23: Add reference.
24: Add reference. Don't know about ibid., took it from Manus review paper...
25: Add reference. Manus review paper
26: Add reference. Don't know about ibid., took it from Manus review paper...
27: Add reference. cite some MAT, WAPLS, Bayesian, etc. papers
28: Add reference. add more..., Climate12K
29: Add reference. cite some MAT papers
30: Add reference. that North-US/South-US discrepancy...

1.2.2 Model Simulations of the Holocene

As mentioned in the earlier section 1.2, paleoclimate simulations of ESMs played in important role in previous intercomparisons. The Holocene analysis within past Paleoclimate Modelling Intercomparison Project (PMIP) versions focused mainly

on the mid-Holocene around 6000 years before present, a time period with a different latitudinal and seasonal distribution of incoming solar radiation (insolation) but greenhouse gas concentrations similar to the preindustrial period (B. L. Otto-Bliesner et al., 2017). The main findings from previous intercomparisons are an underestimation of polar amplification in PMIP2 and PMIP3 models due to sea ice and vegetational feedbacks, and an underestimation of the north-south temperature gradient over Europe (Brewer et al., 2007; Basil A. S. Davis and Brewer, 2009; Intergovernmental Panel on Climate Change, 2014; Masson-Delmotte et al., 2006; Zhang et al., 2010).

The focus on only a short time slice (mainly due to the high computational costs for running an ESM over a large simulation period) however has several complications. Climate changes, i.e. the shift into a different climatic state, cannot be simulated and high dating uncertainties hinder a credible comparison of models and proxies. Therefore multiple recent studies published and proposed increasing efforts for transient model simulations, i.e. simulations that cover multiple millenia during the last deglaciation (Ivanovic et al., 2016) and the Holocene (B. L. Otto-Bliesner et al., 2017). Several studies used Earth System Models of Intermediate Complexity (EMICs) for transient simulations that cover parts of the last 12'000 years (e.g. Gregoire et al., 2015; Menviel et al., 2011; Roche et al., 2011) and a few used a global coupled ESM (Bette L. Otto-Bliesner et al., 2014; Varma et al., 2012). As stated by Weitzel et al., 2019, those model results can clarify the role of internal climate variability for Holocene temperature trends and large-scale patterns.

31: Add reference.

Despite the computational costs for running these models, technical challenges arise from the size of the data that easily exceeds the size of multiple Gigabyte per climatic variable with a monthly resolution. It requires software that is able to deal with data too large to fit into memory (see section 1.3.2 and chapter 2) and automated techniques to identify patterns in the data.

1.3 Software for Paleoclimatology

The usage of software is crucial for the quantitative reconstruction of Earth's Climate. Paleoclimate research is facing an information overload problem and requires innovative methodologies in the realm of visual analytics, the interplay between automated analysis techniques and interactive visualization (Keim et al., 2008; Nocke, 2014). As such, a visual representation of the paleoclimate reconstruction has been essential for both, proxies (Bradley, 1985; Grimm, 1988; Nichols, 1967) and models (Böttinger and Röber, 2019; Nocke, 2014; Nocke et al., 2008; Phillips, 1956; Rautenhaus et al., 2018), although the visualization methods significantly differ due to the differences in data size and data heterogeneity.

The second important aspect for software and paleoclimate is the distribution of data to make it accessible to other researchers, the community and policy makers, which is commonly established through online accessible data archives and recently also through map-based web interfaces (Bolliet et al., 2016; Williams et al., 2018).

32: Add reference. cite some open-data publications

The following sections provide an overview on the different techniques used by modelers and palynologists to visualize and distribute their data and concludes with an introduction into Open-Source Software Development, which forms the basis of the software solutions that are presented later in this thesis (chapters 2, 3, and 4, and appendix C).

1.3.1 Sofware for Proxy Data Analysis, Visualization and Distribution

Due to the nature of stratigraphic data, proxies, especially pollen assemblages, are often treated as a collection of multiple time-series (one-dimensional arrays). The size of one dataset is generally small (in the range of kB) and can be treated as plain text files. Traditionally, numerical and statistical analysis are separated from the visualization.

In palynology, standard analytical tools are Microsoft Excel[1] and the R software for statistical computing (R Core Team, 2019). The latter also involves multiple packages for paleoclimatic reconstruction, such as `rioja` (Juggins, 2017) and `analogue` (Simpson, 2007; Simpson and Oksanen, 2019) or bayesian methods (Nolan et al., 2019; Tipton, 2017). Alternatively, desktop applications exist, such as Polygon[2] by Nakagawa et al., 2002 or the CREST software by M. Chevalier et al., 2014; Manuel Chevalier, 2019.

33: Add reference. add more?

It is a long-standing tradition to visualize stratigraphic data, and especially pollen data, in form of a stratigraphic (pollen) diagram (Bradley, 1985; Grimm, 1988). Especially during the 19th century, when it was not yet common to distribute data alongside a peer-reviewed publication, pollen diagrams where the only possibility to publish the entire dataset (see also chapter 3). The generation of these diagrams is usually based on desktop applications such as C2 (Juggins, 2007), Tilia[3] (Grimm, 1988, 1991). A more recent implementation into the psyplot framework (Sommer, 2017, chapter 2) is also provided with the psy-strat plugin[4] (Sommer, 2019).

Raw pollen data is at present made available through web archives, such as PANGAEA[5] or the National Climatic Data Center (NCDC) by the National Oceanic and Atmospheric Administration (NOAA)[6] where researchers can create a DOI for their raw data. Collections of data, such as regional pollen databases or project specific collections (e.g. B. A. S. Davis et al., 2013; Whitmore et al., 2005) are usually published in one of the above-mentioned archives or associated with a publication. A different approach has been developed by Bolliet et al., 2016 to develop a small web application as an interface into the data collection, the *ClimateProxiesFinder* (Brockmann, 2016, chapter 4).

Outstanding compared to the previous data interfaces is the new infrastructure for the Neotoma database (Williams et al., 2018). It consists of the map-based web interface, the Neotoma Explorer[7], a RESTful api[8] that allows an interaction with other web services, the neotoma R package (Goring et al., 2015) and an interface into the Tilia software for stratigraphic and map-based visualizations (Williams et al., 2018). This rich functionality is, however, bound to the structure of Neotoma and as such, different from the Javascript-based approach developed in chapter 4 cannot easily be transferred to other projects.

1.3.2 The Development of Computational Climate Model Analysis

Software and computational numerics play a crucial row for our understanding of climate since the first General Circulation Models (GCMs) by Phillips, 1956 after

[1] https://products.office.com/en/excel
[2] http://polsystems.rits-paleo.com
[3] https://www.tiliait.com/
[4] https://psy-strat.readthedocs.io
[5] https://pangaea.de/
[6] https://www.ncdc.noaa.gov/data-access/paleoclimatology-data
[7] https://apps.neotomadb.org/Explorer
[8] https://api.neotomadb.org

Visualize multiple grids on the same map, e.g. by using the grid specifications from Treut et al., 2007.

FIGURE 1.1: A selection of grid sizes and formats since the first IPCC report

world war II (Edwards, 2010; Lewis, 1998). The first simulations and analysis of GCMs where limited by the available computational facilities, the model by Phillips, 1956 for example operated on a 17×16 grid simulating a surface with the size of roughly one tenth of the Earth. The possibilities of climate modeling increased rapidly, mainly due to a drastic increase in computational capacity and the availability of supercomputers. This let to an increase in speed by a factor of roughly one million between 1970 and 2007, permitting an increase in model complexity, length of the simulations, and spatial resolution (Treut et al., 2007). In the past decade, unstructured grids raised more and more attention (Skamarock et al., 2012; Zängl et al., 2014) especially with the focus on *seamless prediction* (Bauer et al., 2015; Hoskins, 2012) that allows a refined grid resolution in selected regions of the earth (Rautenhaus et al., 2018).

The varying grids, multi-dimensionality and volume of the data requires visual analytic methodologies much different from what is used with proxy data (section 1.3.1) and much more diverse. In general, scientists tend to disentangle the numeric post-processing of climate model output (such as computing aggregated statistics in time or space dimension) and then visualize these aggregated results (Böttinger and Röber, 2019; Schulz et al., 2013). Common post-processing software are, for example Climate Data Operators (CDOs) (Schulzweida, 2019), netCDF Operators (NCOs) (Zender, 2008, 2016; Zender and Mangalam, 2007) and R (R Core Team, 2019), or more recently also python packages such as `xarray` (Hoyer and Hamman, 2017). The choice of method thereby depends on the scientiests preference but also on the size of the data. Especially the analysis of transient model runs (section 1.2.2) requires software that is able to deal with data that is too big to fit into memory and requires parallel computation. Such an analysis can be done pursued with CDOs and `xarray` through it's interface with the parallel computing package dask (Dask Development Team, 2016; Rocklin, 2015) (see also chapter 6).

Rautenhaus et al., 2018 provide a detailed overview about how such a high amount of data is visualized. Methods range from 2D projection on a map to 3D interactive visualizations, depending on the background and knowledge of the researcher (Nocke, 2014). Nocke et al., 2008 recognize preference for script-based solutions such as Python, R or domain-specific languages such as NCL (D. Brown et al., 2012) that is still persistent today (Rautenhaus et al., 2018). Nocke, 2014, Schulz et al., 2013 and Rautenhaus et al., 2018 attribute this to the importance of comparability and reproducibility for research, and the usability in peer-reviewed publications. Therefore, 3D visualization, e.g. with ParaView (Ayachit, 2015), VAPOR (Clyne et al., 2007) or Avizo, are mainly used by visualization experts rather than

34: Add reference.

scientists (Nocke et al., 2008). However, Nocke, 2014 note that young researchers are more open for 3D visualization and especially with the newly emerging unstructured grids, they become more prominent. Besides psyplot (Sommer, 2017, chapter 2) and UV-CDAT, there exists to my knowledge no scripted method that easily visualizes climate model data on unstructured grids, without the need for interpolation to a rectilinear grid. These have however been implemented for ParaView (Röber et al., 2015) and Vapor (Jubair et al., 2015) and new interfaces into VTK have also been developped by Sullivan and Trainor-Guitton, 2019.

Look into ibid.

35: Add reference.

1.3.3 Methods and Workflows in Open-Source Software Development

The importance and necessity of software for visualization and data analysis led to the development of the software packages presented in this thesis. Most of them are written in the programming language Python (Perez et al., 2011), on the one hand due to my personal preference, but mainly due to the recent developments in out-of-core computing with the establishment of xarray and dask (Dask Development Team, 2016; Hoyer and Hamman, 2017; Rocklin, 2015). Another important reason, especially for psyplot (chapter 2) and straditize (chapter 3) was the availability of a highly flexible and stable package for graphical user interfaces, PyQt, and the comparably simple possibility to implement an in-process python console into the PyQ5 application that allows to handle the software functionalities both, from the command line and from the GUI.

36: Add reference.

37: Add reference. jupyter qtconsole

Modern Free and Open-Source Software (FOSS) development is not only about making the source code available, but rather about providing a sustainable and maintainable package that allow continuous and transparent development under the aspect of rapidly evolving environment. In the following sections, I will introduce the most important FOSS development concepts (e.g. Shaw, 2018; Stodden and Miguez, 2014) and the necessary vocabulary. These concepts are used by many of the well-established software packages, such as matplotlib (Hunter, 2007), numpy (T. E. Oliphant, 2006), and scipy (Jones et al., 2001).

Version Control

Version control systems record changes to a file and enables the user to roll-back to previous versions of it. The usage of a such a system is inevitable for sustainable FOSS packages. It enables contributions by other FOSS developers and the usage through external packages.

The packages I present in the following chapters are hosted on Github[9], a freely available web platform for hosting projects that are managed with git (Chacon et al., 2019).

Version control with git has a specific terminology. Central aspects are *repositories* (project folders), *commits* (change of the project files), *issues* (bug reports), *branches* and *forks* (copies of the (main) project), and *pull requests* (contributions to a project). The following list explains this vocabulary in a bit more detail, a more complete list is provided by in Github, Inc., 2019.

Repositories are the most basic elements of git and Github. It can be compared to a folder that contains all the necessary files associated with a project (e.g. the source code and documentation of a software package). It also contains all the different versions (revisions) of the project files.

[9]The packages are available at `https://github.com/Chilipp`. Other potential platforms for version control are sourceforge (`https://sourceforge.net`) and Bitbucket (`https://bitbucket.org`)

Commits or revisions track the changes in the repository. Each commit is a change to a specific file (or a set of files) that is associated with a unique ID and a message of the author about to describe the changes.

Issues are suggested improvements, bug reports or any other question to the repository. Every issue has a associated discussion page the topic can be discussed between repository owners and the users.

Branches are parallel versions of a repository. Often one incorporates new developments into a separate branch that does not affect the main version of the repository (the *master* branch) and merge the two versions when the new developments are fully implemented.

Forks are copies of repositories. When someone wants to contribute to a software package (repository) that does not belong to him, he can *fork* (copy) it, implement it's changes, and then create a *pull request* to contribute to the official version. Different from a branch, that is a (modified) copy of another branch, forks are copies of the entire repository, i.e. all existing branches.

Pull request are the proposed changes to a repository. One can create a fork of the repository from someone else, implement changes in this fork and then create a pull request to merge it into the original repository. Every pull request has an associated discussion page that allows the repository owner to moderate and discuss the suggested changes.

Webhooks are general methods for web development. Github can trigger a hook to inform a different web service (such as a Continuous Integration (CI) (section 1.3.3)) that a repository has changed or that someone contributed in a discussion. In chapter 4 we use Github webhooks for a automated administration of a repository.

Automated Tests, Test Coverage and Continuous Integration

The most important aspect for FOSS development, especially considering the rapid evolution of this area, is the existence of automated tests. One distinguishes unit tests (test of one single routine) and integration tests (tests of one or more routines within the framework) (Shaw, 2018). The boundary between the two tests is rather vague and what is more appropriate highly depends on the structure of the software that is supposed to be tested. For complex frameworks (such as psyplot or straditize), integration tests are needed to ensure the operability within the framework. Other more simple software packages, (such as docrep or model-organization, see appendix C.2) go well with unit tests only.

Another good standard for such an test suite is to use an automated test discovery tool (e.g. the Python unittest package (Python Software Foundation, 2019) or pytest (Krekel et al., 2004)) that also reports the test coverage (i.e. the fraction of the code that is tested by the test suite). These functionalities are then implemented on a CI service, such as Travis CI[10], Appveyor[11] or CircleCi[12] that are integrated into the Github repository (section 1.3.3). Every commit to the Github repository or a new pull requests then triggers the tests. This transparently allows to ensure the

[10] https://travis-ci.org/
[11] https://appveyor.com
[12] https://circleci.com/

operability of the software, and the test coverage report ensures that the newly implemented functionality is properly tested. A software development concept that is build entirely on that is the test-driven development. Within this framework, new features are implemented by starting with the test that should be fulfilled by the new feature and then improving the software until this test pass (Beck, 2002).

Automated Documentation

Documentation is the key aspect of a sustainable software and much of the geoscientific code has a lack of proper documentation (based on personal experience). For the software in this thesis, for different levels of the documentation play an important role:

The Application programming interface (API) documentation is meant to document the major parts of the software code that is subject to be used by external scripts or packages. It is usually implemented in the code and documents the essential subroutines and methods of the software.

The graphical user interface (GUI) documentation provides help for the most high-level functionality for the software. The GUI is a user interface into the software through graphical elements (such as buttons, checkboxes, etc.). Unlike the API documentation, it should not require knowledge about programming.

The contributing and/or developers guide is targeting other software developers that might want to contribute to the software package. This document states how other software developers should contribute to the software and introduces the central structural aspects and frameworks of the software.

The manual (or also commonly referred to as *the* documentation) is the document that contains all necessary information for the software, such as installation instructions, tutorials, examples, etc.. It often includes some (or multiple) of the above parts.

The documentations for the software in this thesis have been automatically generated with Sphinx, a Python tool to generate documentations in various different formats (such as HTML, PDF, etc.) (Hasecke, 2019; Perez et al., 2011). It is also implemented as a webhook into the Github repository (see section 1.3.3) to automatically generate an up-to-date documentation of the software for each commit to the Github repository. This provides an additional automated test for the software, and especially it's high-level-interface, in addition to the automated test suite described above (section 1.3.3). Most of the manuals are hosted and build online with the free services offered by readthedocs.org.

Distribution through package managers and virtual environments

FOSS software is meant to be extensible and to build upon other FOSS packages. This requires an accurate and transparent handling of it's dependencies and requirements which is usually provided through the so-called packaging of the software (e.g. Torborg, 2016). There exists a variety of package managers and the choice most often depends on the framework of the software.

The software in this thesis is mainly distributed via two systems. The first one is python's own package manager *pip* which is based on the packages uploaded to pypi.org. The second one which got increasing importance during the recent past is

the open-source Anaconda Distribution[13]. Both work on multiple operating systems (Windows, Linux and Mac OS), but the Anaconda Distribution contains also non-python packages (e.g. written in C or C++) for which the Python packages rely on, and it contains a rich suite of r-packages.

One step further, compared to package managers, are the distribution of virtual environments. These systems do not only provide the software, but also a full operating system and the installed dependencies. A popular platform (used also for the Eurasian Modern Pollen Database (EMPD) database) is provided through so-called Docker containers[14]. Compared to package managers, this system has the advantage of simplifying the installation procedure for the user because he only has to download the corresponding docker image. The docker image itself then runs independent of the local file system in a separate isolated mode.

1.4 Challenges tackled in this thesis

I present several new tools in this thesis that tackle the aspects described in the previous sections. It is divided into two parts: Part I are the software chapters 2, 3 and 4 that introduce new packages developed for paleoclimate analysis. Part II consists of the analysis chapters 5 and

6 that address two use-cases tackling the combination of observations and models

for an informed paleoclimate understanding.

The first part starts with the visualization framework psyplot in chapter 2, a suite of python packages that are designed for interactive visual analysis of data both from a GUI and the command line. The scope of this software is not limited to paleoclimate analysis and serves a more general purpose. As such, it serves as a base infrastructure for many of the topics described in the other chapters.

Straditize, described in the next chapter 3, addresses the problem of gathering paleo-climate information that has been collected during the pre-digital area. This software is a semi-automated digitization package for stratigraphic diagrams, particularly pollen diagrams. Straditize is built on top of psyplot and its GUI and as such provides a rich interactive documentation and visualization methods of the software tools.

Chapter 4 covers the last aspect of software usage for paleoclimate data: data distribution. In this chapter I describe new infrastructural tools for the sustainable management of a community-driven pollen database, the Eurasian Modern Pollen Database (EMPD). They consist of a flexible and lightweight map-based web interface, the EMPD-viewer, into the data and a webserver for an automated administration of the database. Within this section, I also present another use case for the EMPD-viewer that is adapted to a large northern-hemispheric database of fossil and modern pollen records.

The second part starts with the weather generator GWGEN in chapter 5, a statistical model that uses modern relationships in observational data to inform large-scale paleo climate models with temporally downscaled temperature, precipitation, cloud cover and wind speed records.

[13] https://www.anaconda.com
[14] https://www.docker.com

Finally, in chapter 6 I investigate the question to what extent large-scale atmospheric circulation features can be estimated from proxy data. In this analysis I analyze the long-term stability of spatial correlation patterns between surface temperature and northern hemispheric teleconnections based on three ESMs.

This thesis finishes with the conclusions in chapter 7 which summarizes the new tools and findings and provides an outlook for the further development of the methods presented. In the Appendix I present another work that has developed in a cooperation which is also based on the infrastructural tools from psyplot and GWGEN (appendix A), and I provide a list of the publications during my thesis (appendix B) and an overview about all the software packages that have been developed (appendix C).

References

Ayachit, Utkarsh (2015). *The paraview guide: a parallel visualization application*. Kitware, Inc.

Bauer, Peter, Alan Thorpe, and Gilbert Brunet (2015). "The quiet revolution of numerical weather prediction". In: *Nature* 525.7567, pp. 47–55. DOI: 10.1038/nature14956.

Beck, Kent (2002). *Test Driven Development. By Example*. Addison Wesley. 192 pp. ISBN: 978-0-321-14653-3. URL: https://www.ebook.de/de/product/3253611/kent_beck_test_driven_development_by_example.html.

Birks, Harry John Betteley and Hilary H Birks (1980). *Quaternary palaeoecology*. Edward Arnold London.

Bolliet, Timothé, Patrick Brockmann, Valérie Masson-Delmotte, Franck Bassinot, Valérie Daux, Dominique Genty, Amaelle Landais, Marlène Lavrieux, Elisabeth Michel, Pablo Ortega, Camille Risi, Didier M. Roche, Françoise Vimeux, and Claire Waelbroeck (2016). "Water and carbon stable isotope records from natural archives: a new database and interactive online platform for data browsing, visualizing and downloading". In: *Climate of the Past* 12.8, pp. 1693–1719. DOI: 10.5194/cp-12-1693-2016.

Böttinger, Michael and Niklas Röber (2019). "Visualization in Climate Modelling". In: *International Climate Protection*. Ed. by Michael Palocz-Andresen, Dòra Szalay, Andràs Gosztom, Làszlò Sìpos, and Tìmea Taligàs. Cham: Springer International Publishing, pp. 313–321. ISBN: 978-3-030-03816-8. DOI: 10.1007/978-3-030-03816-8_39. URL: https://doi.org/10.1007/978-3-030-03816-8_39.

Braconnot, P., Sandy P Harrison, Masa Kageyama, Patrick J Bartlein, Valerie Masson-Delmotte, Ayako Abe-Ouchi, Bette Otto-Bliesner, and Yan Zhao (2012). "Evaluation of climate models using palaeoclimatic data". In: *Nature Climate Change* 2.6, p. 417. DOI: 10.1038/nclimate1456. URL: https://www.nature.com/articles/nclimate1456.

Braconnot, P., B. Otto-Bliesner, S. Harrison, S. Joussaume, J.-Y. Peterchmitt, A. Abe-Ouchi, M. Crucifix, E. Driesschaert, Th. Fichefet, C. D. Hewitt, M. Kageyama, A. Kitoh, M.-F. Loutre, O. Marti, U. Merkel, G. Ramstein, P. Valdes, L. Weber, Y. Yu, and Y. Zhao (2007a). "Results of PMIP2 coupled simulations of the Mid-Holocene and Last Glacial Maximum – Part 2: feedbacks with emphasis on the location of the ITCZ and mid- and high latitudes heat budget". In: *Climate of the Past* 3.2, pp. 279–296. DOI: 10.5194/cp-3-279-2007. URL: https://www.clim-past.net/3/279/2007/.

Braconnot, P., Otto-Bliesner, S. P. Harrison, S. Joussaume, J.-Y. Peterchmitt, A. Abe-Ouchi, M. Crucifix, E. Driesschaert, Th. Fichefet, C. D. Hewitt, M. Kageyama, A. Kitoh, A. Laîné, M.-F. Loutre, O. Marti, U. Merkel, G. Ramstein, P. Valdes, S. L. Weber, Y. Yu, and Y. Zhao (2007b). "Results of PMIP2 coupled simulations of the Mid-Holocene and Last Glacial Maximum – Part 1: experiments and large-scale features". In: *Climate of the Past* 3.2, pp. 261–277. DOI: 10.5194/cp-3-261-2007. URL: https://www.clim-past.net/3/261/2007/.

Bradley, Raymond S (1985). *Quaternary paleoclimatology : methods of paleoclimatic reconstruction*. eng. Boston ; London [etc.]: Allen and Unwin. ISBN: 0045510679.

Brewer, Simon, J. Guiot, and F. Torre (2007). "Mid-Holocene climate change in Europe: a data-model comparison". In: *Climate of the Past* 3.3, pp. 499–512. DOI: 10.5194/cp-3-499-2007. URL: https://www.clim-past.net/3/499/2007/.

Brockmann, Patrick (2016). *ClimateProxiesFinder: dc.js + leaflet application to discover climate proxies*. Last accessed: 2019-08-30. URL: https://github.com/PBrockmann/ClimateProxiesFinder (visited on 10/06/2016).

Brown, Dave, Richard Brownrigg, Mary Haley, and Wei Huang (2012). "NCAR Command Language (NCL)". eng. In: DOI: 10.5065/d6wd3xh5.

Brown, Stewart A., Mike Folk, Gregory Goucher, Russ Rew, and Paul F. Dubois (1993). "Software for Portable Scientific Data Management". In: *Computers in Physics* 7.3, p. 304. DOI: 10.1063/1.4823180.

Chacon, Scott, Ben Straub, and Pro Git Contributors (2019). *Pro Git*. 2nd ed. Last accessed: 2019-08-31. URL: https://github.com/progit/progit2 (visited on 08/31/2019).

Chevalier, M., R. Cheddadi, and B. M. Chase (2014). "CREST (Climate REconstruction SofTware): a probability density function (PDF)-based quantitative climate reconstruction method". In: *Climate of the Past* 10.6, pp. 2081–2098. DOI: 10.5194/cp-10-2081-2014.

Chevalier, Manuel (2019). "Enabling possibilities to quantify past climate from fossil assemblages at a global scale". In: *Global and Planetary Change* 175, pp. 27–35. DOI: 10.1016/j.gloplacha.2019.01.016.

Clyne, John, Pablo Mininni, Alan Norton, and Mark Rast (2007). "Interactive desktop analysis of high resolution simulations: application to turbulent plume dynamics and current sheet formation". In: *New Journal of Physics* 9.8, pp. 301–301. DOI: 10.1088/1367-2630/9/8/301.

COHMAP Members (1988). "Climatic Changes of the Last 18,000 Years: Observations and Model Simulations". In: *Science* 241.4869, pp. 1043–1052. ISSN: 00368075, 10959203. URL: http://www.jstor.org/stable/1702404.

Dasgupta, Aritra, Jorge Poco, Enrico Bertini, and Claudio T. Silva (2016). "Reducing the Analytical Bottleneck for Domain Scientists: Lessons from a Climate Data Visualization Case Study". In: *Computing in Science & Engineering* 18.1, pp. 92–100. DOI: 10.1109/mcse.2016.7.

Dask Development Team (2016). *Dask: Library for dynamic task scheduling*. URL: https://dask.org.

Davis, B. A. S., S. Brewer, A. C. Stevenson, and J. Guiot (2003). "The temperature of Europe during the Holocene reconstructed from pollen data". In: *Quat. Sci. Rev.* 22.15-17, pp. 1701–1716. ISSN: 02773791. DOI: 10.1016/s0277-3791(03)00173-2.

Davis, B. A. S., M. Zanon, P. Collins, A. Mauri, J. Bakker, D. Barboni, A. Barthelmes, C. Beaudouin, A. E. Bjune, E. Bozilova, R. H. W. Bradshaw, B. A. Brayshay, S. Brewer, E. Brugiapaglia, J. Bunting, S. E. Connor, J. L. de Beaulieu, K. Edwards, A. Ejarque, P. Fall, A. Florenzano, R. Fyfe, D. Galop, M. Giardini, T. Giesecke, M. J.

Grant, J. Guiot, S. Jahns, V. Jankovska, S. Juggins, M. Kahrmann, M. Karpinska-Kolaczek, P. Kolaczek, N. Kuhl, P. Kunes, E. G. Lapteva, S. A. G. Leroy, M. Leydet, J. Guiot, S. Jahns, V. Jankovska, S. Juggins, M. Kahrmann, M. Karpinska-Kolaczek, P. Kolaczek, N. Kuehl, P. Kunes, E. G. Lapteva, S. A. G. Leroy, M. Leydet, J. A. L. Saez, A. Masi, I. Matthias, F. Mazier, V. Meltsov, A. M. Mercuri, Y. Miras, F. J. G. Mitchell, J. L. Morris, F. Naughton, A. B. Nielsen, E. Novenko, B. Odgaard, E. Ortu, M. V. Overballe-Petersen, H. S. Pardoe, S. M. Peglar, I. A. Pidek, L. Sadori, H. Seppa, E. Severova, H. Shaw, J. Swieta-Musznicka, M. Theuerkauf, S. Tonkov, S. Veski, W. O. van der Knaap, J. F. N. van Leeuwen, J. Woodbridge, M. Zimny, and J. O. Kaplan (2013). "The European Modern Pollen Database (EMPD) project". In: *Vegetation History and Archaeobotany* 22.6, pp. 521–530. ISSN: 0939-6314. DOI: 10.1007/s00334-012-0388-5. URL: http://link.springer.com/article/10.1007/s00334-012-0388-5.

Davis, Basil A. S. and Simon Brewer (2009). "Orbital forcing and role of the latitudinal insolation/temperature gradient". In: *Climate Dynamics* 32.2, pp. 143–165. ISSN: 1432-0894. DOI: 10.1007/s00382-008-0480-9. URL: https://doi.org/10.1007/s00382-008-0480-9.

Edwards, Paul N. (2010). "History of climate modeling". In: *Wiley Interdisciplinary Reviews: Climate Change* 2.1, pp. 128–139. DOI: 10.1002/wcc.95.

Fægri, K., P. E. Kaland, and K. Krzywinski (1989). *Textbook of pollen analysis*. Ed. 4. Chichester, UK: John Wiley & Sons Ltd.

Flantua, Suzette G.A., Henry Hooghiemstra, Eric C. Grimm, Hermann Behling, Mark B. Bush, Catalina González-Arango, William D. Gosling, Marie-Pierre Ledru, Socorro Lozano-Garcìa, Antonio Maldonado, Aldo R. Prieto, Valentì Rull, and John H. Van Boxel (2015). "Updated site compilation of the Latin American Pollen Database". In: *Review of Palaeobotany and Palynology* 223, pp. 104–115. DOI: 10.1016/j.revpalbo.2015.09.008.

Fyfe, Ralph M., Jacques-Louis de Beaulieu, Heather Binney, Richard H. W. Bradshaw, Simon Brewer, Anne Le Flao, Walter Finsinger, Marie-Josè Gaillard, Thomas Giesecke, Graciela Gil-Romera, Eric C. Grimm, Brian Huntley, Petr Kunes, Norbert Kühl, Michelle Leydet, Andrè F. Lotter, Pavel E. Tarasov, and Spassimir Tonkov (2009). "The European Pollen Database: past efforts and current activities". In: *Vegetation History and Archaeobotany* 18.5, pp. 417–424. DOI: 10.1007/s00334-009-0215-9.

Github, Inc. (2019). "GitHub glossary". In: Last accessed: 2019-08-31. URL: https://help.github.com/en/articles/github-glossary (visited on 08/31/2019).

Goring, Simon, Andria Dawson, Gavin L Simpson, Karthik Ram, Russell W Graham, Eric C Grimm, and Jack W. Williams (2015). "neotoma: A Programmatic Interface to the Neotoma Paleoecological Database". In: *Open Quaternary* 1.1, p. 2. URL: http://doi.org/10.5334/oq.ab.

Gregoire, Lauren J., Paul J. Valdes, and Antony J. Payne (2015). "The relative contribution of orbital forcing and greenhouse gases to the North American deglaciation". In: *Geophysical Research Letters* 42.22, pp. 9970–9979. DOI: 10.1002/2015gl066005. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1002/2015GL066005. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2015GL066005.

Grimm, Eric C. (1988). "Data analysis and display". In: *Vegetation history*. Ed. by B. Huntley, T. Webb, B. Huntley, and T. Webb. Dordrecht: Springer Netherlands, pp. 43–76. ISBN: 978-94-009-3081-0. DOI: 10.1007/978-94-009-3081-0_3. URL: https://doi.org/10.1007/978-94-009-3081-0_3.

— (1991). "Tilia and Tiliagraph". In: *Illinois State Museum, Springfield* 101.

Grimm, Eric C. (2008). "Neotoma: an ecosystem database for the Pliocene, Pleistocene, and Holocene". In: *Illinois State Museum Scientific Papers E Series* 1. URL: https://www.neotomadb.org/uploads/NeotomaManual.pdf.

Hargreaves, J. C., J. D. Annan, R. Ohgaito, A. Paul, and A. Abe-Ouchi (2013). "Skill and reliability of climate model ensembles at the Last Glacial Maximum and mid-Holocene". In: *Clim. Past* 9.2, pp. 811–823. ISSN: 1814-9332. DOI: 10.5194/cp-9-811-2013.

Hasecke, Jan Ulrich (2019). *Software-Dokumentation mit Sphinx: Zweite überarbeitete Auflage (Sphinx 2.0) (German Edition)*. Independently published. ISBN: 1793008779. URL: https://www.amazon.com/Software-Dokumentation-mit-Sphinx-%C3%83%C2%BCberarbeitete-Auflage/dp/1793008779?SubscriptionId=AKIAIOBINVZYXZQZ2U3A%5C&tag=chimbori05-20%5C&linkCode=xm2%5C&camp=2025%5C&creative=165953%5C&creativeASIN=1793008779.

Havinga, A.J. (1967). "Palynology and pollen preservation". In: *Review of Palaeobotany and Palynology* 2.1-4, pp. 81–98. DOI: 10.1016/0034-6667(67)90138-8.

Hays, J. D., J. Imbrie, and N. J. Shackleton (1976). "Variations in the Earth's Orbit: Pacemaker of the Ice Ages". In: *Science* 194.4270, pp. 1121–32. ISSN: 0036-8075 (Print) 0036-8075 (Linking). DOI: 10.1126/science.194.4270.1121. URL: http://www.ncbi.nlm.nih.gov/pubmed/17790893.

Hoskins, Brian (2012). "The potential for skill across the range of the seamless weather-climate prediction problem: a stimulus for our science". In: *Quarterly Journal of the Royal Meteorological Society* 139.672, pp. 573–584. DOI: 10.1002/qj.1991.

Hoyer, S. and J. Hamman (2017). "xarray: N-D labeled arrays and datasets in Python". In: *Journal of Open Research Software* 5.1. DOI: 10.5334/jors.148. URL: http://doi.org/10.5334/jors.148.

Hunter, J. D. (2007). "Matplotlib: A 2D Graphics Environment". In: *Computing in Science Engineering* 9.3, pp. 90–95. ISSN: 1521-9615. DOI: 10.1109/MCSE.2007.55.

Imbrie, J., A. Berger, E. A. Boyle, S. C. Clemens, A. Duffy, W. R. Howard, G. Kukla, J. Kutzbach, D. G. Martinson, A. McIntyre, A. C. Mix, B. Molfino, J. J. Morley, L. C. Peterson, N. G. Pisias, W. L. Prell, M. E. Raymo, N. J. Shackleton, and J. R. Toggweiler (1993). "On the structure and origin of major glaciation cycles 2. The 100,000-year cycle". In: *Paleoceanography* 8.6, pp. 699–735. DOI: 10.1029/93pa02751. URL: https://ui.adsabs.harvard.edu/abs/1993PalOc...8..699I.

Intergovernmental Panel on Climate Change (2014). "Evaluation of Climate Models". In: *Climate Change 2013 – The Physical Science Basis: Working Group I Contribution to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge University Press, pp. 741–866. DOI: 10.1017/CBO9781107415324.020.

Ivanovic, R. F., L. J. Gregoire, M. Kageyama, D. M. Roche, P. J. Valdes, A. Burke, R. Drummond, W. R. Peltier, and L. Tarasov (2016). "Transient climate simulations of the deglaciation 21–9 thousand years before present (version 1) – PMIP4 Core experiment design and boundary conditions". In: *Geosci. Model Dev.* 9.7, pp. 2563–2587. DOI: 10.5194/gmd-9-2563-2016. URL: https://www.geosci-model-dev.net/9/2563/2016/.

Jones, Eric, Travis Oliphant, Pearu Peterson, et al. (2001). *SciPy: Open source scientific tools for Python*. [Online; accessed 2017-02-18]. URL: http://www.scipy.org/.

Joussaume, S and KE Taylor (1995). "Status of the paleoclimate modeling intercomparison project (PMIP)". In: *World Meteorological Organization-Publications-WMO TD*, pp. 425–430.

Jubair, Mohammad Imrul, Usman Alim, Niklas Roeber, John Clyne, Ali Mahdavi-Amiri, and Faramarz Samavati (2015). "Multiresolution visualization of digital

earth data via hexagonal box-spline wavelets". In: *2015 IEEE Scientific Visualization Conference (SciVis)*. IEEE. DOI: 10.1109/scivis.2015.7429508.

Juggins, Steve (2007). "C2: Software for ecological and palaeoecological data analysis and visualisation (user guide version 1.5)". In: *Newcastle upon Tyne: Newcastle University* 77. URL: https://www.staff.ncl.ac.uk/stephen.juggins/software/C2Home.htm.

— (2017). *rioja: Analysis of Quaternary Science Data*. R package version 0.9-21. URL: http://www.staff.ncl.ac.uk/stephen.juggins/.

Jungclaus, J. H., E. Bard, M. Baroni, P. Braconnot, J. Cao, L. P. Chini, T. Egorova, M. Evans, J. F. Gonzàlez-Rouco, H. Goosse, G. C. Hurtt, F. Joos, J. O. Kaplan, M. Khodri, K. Klein Goldewijk, N. Krivova, A. N. LeGrande, S. J. Lorenz, J. Luterbacher, W. Man, A. C. Maycock, M. Meinshausen, A. Moberg, R. Muscheler, C. Nehrbass-Ahles, B. I. Otto-Bliesner, S. J. Phipps, J. Pongratz, E. Rozanov, G. A. Schmidt, H. Schmidt, W. Schmutz, A. Schurer, A. I. Shapiro, M. Sigl, J. E. Smerdon, S. K. Solanki, C. Timmreck, M. Toohey, I. G. Usoskin, S. Wagner, C.-J. Wu, K. L. Yeo, D. Zanchettin, Q. Zhang, and E. Zorita (2017). "The PMIP4 contribution to CMIP6 – Part 3: The last millennium, scientific objective, and experimental design for the PMIP4 *past1000* simulations". In: *Geosci. Model Dev.* 10.11, pp. 4005–4033. DOI: 10.5194/gmd-10-4005-2017. URL: https://www.geosci-model-dev.net/10/4005/2017/.

Kageyama, M., P. Braconnot, S. P. Harrison, A. M. Haywood, J. Jungclaus, B. L. Otto-Bliesner, J.-Y. Peterschmitt, A. Abe-Ouchi, S. Albani, P. J. Bartlein, C. Brierley, M. Crucifix, A. Dolan, L. Fernandez-Donado, H. Fischer, P. O. Hopcroft, R. F. Ivanovic, F. Lambert, D. J. Lunt, N. M. Mahowald, W. R. Peltier, S. J. Phipps, D. M. Roche, G. A. Schmidt, L. Tarasov, P. J. Valdes, Q. Zhang, and T. Zhou (2016). "PMIP4-CMIP6: the contribution of the Paleoclimate Modelling Intercomparison Project to CMIP6". In: *Geosci. Model Dev. Discuss.* 2016, pp. 1–46. DOI: 10.5194/gmd-2016-106. URL: https://www.geosci-model-dev.net/11/1033/2018/gmd-11-1033-2018.html.

Karpechko, A.Y. (2010). "Uncertainties in future climate attributable to uncertainties in future Northern Annular Mode trend. NAM AND FUTURE CLIMATE UNCERTAINTIES". In: *Geophysical Research Letters* 37. ISSN: 0094-8276. DOI: 10.1029/2010gl044717.

Keim, Daniel, Gennady Andrienko, Jean-Daniel Fekete, Carsten Görg, Jörn Kohlhammer, and Guy Melançon (2008). "Visual Analytics: Definition, Process, and Challenges". In: *Information Visualization: Human-Centered Issues and Perspectives*. Ed. by Andreas Kerren, John T. Stasko, Jean-Daniel Fekete, Chris North, Andreas Kerren, John T. Stasko, Jean-Daniel Fekete, and Chris North. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 154–175. ISBN: 978-3-540-70956-5. DOI: 10.1007/978-3-540-70956-5_7. URL: https://doi.org/10.1007/978-3-540-70956-5_7.

Krekel, Holger, Bruno Oliveira, Ronny Pfannschmidt, Floris Bruynooghe, Brianna Laugher, and Florian Bruhin (2004). *pytest 5.1*. URL: https://github.com/pytest-dev/pytest.

Lewis, J. M. (1998). "Clarifying the Dynamics of the General Circulation: Phillips's 1956 Experiment". In: *Bulletin of the American Meteorological Society* 79.1, pp. 39–60. DOI: 10.1175/1520-0477(1998)079<0039:ctdotg>2.0.co;2. URL: https://ui.adsabs.harvard.edu/abs/1998BAMS...79...39L.

Marchant, Robert, Lucia Almeida, Hermann Behling, Juan Carlos Berrio, Mark Bush, Antoine Cleef, Joost Duivenvoorden, Maarten Kappelle, Paulo De Oliveira, Ary Teixeira de Oliveira-Filho, Socorro Lozano-Garìia, Henry Hooghiemstra, Marie-Pierre Ledru, Beatriz Ludlow-Wiechers, Vera Markgraf, Virginia Mancini, Marta

Paez, Aldo Prieto, Olando Rangel, and Maria Lea Salgado-Labouriau (2002). "Distribution and ecology of parent taxa of pollen lodged within the Latin American Pollen Database". In: *Review of Palaeobotany and Palynology* 121.1, pp. 1–75. DOI: 10.1016/s0034-6667(02)00082-9.

Marsicek, Jeremiah, Bryan N. Shuman, Patrick J. Bartlein, Sarah L. Shafer, and Simon Brewer (2018). "Reconciling divergent trends and millennial variations in Holocene temperatures". In: *Nature* 554.7690, pp. 92–96. DOI: 10.1038/nature25464.

Masson-Delmotte, V., M. Kageyama, P. Braconnot, S. Charbit, G. Krinner, C. Ritz, E. Guilyardi, J. Jouzel, A. Abe-Ouchi, M. Crucifix, R. M. Gladstone, C. D. Hewitt, A. Kitoh, A. N. LeGrande, O. Marti, U. Merkel, T. Motoi, R. Ohgaito, B. Otto-Bliesner, W. R. Peltier, I. Ross, P. J. Valdes, G. Vettoretti, S. L. Weber, F. Wolk, and Y. YU (2006). "Past and future polar amplification of climate change: climate model intercomparisons and ice-core constraints". In: *Climate Dynamics* 26.5, pp. 513–529. ISSN: 1432-0894. DOI: 10.1007/s00382-005-0081-9. URL: https://doi.org/10.1007/s00382-005-0081-9.

Mauri, A., B. A. S. Davis, P. M. Collins, and J. O. Kaplan (2015). "The climate of Europe during the Holocene: a gridded pollen-based reconstruction and its multi-proxy evaluation". In: *Quat. Sci. Rev.* 112, pp. 109–127. ISSN: 0277-3791. DOI: 10.1016/j.quascirev.2015.01.013. URL: http://www.sciencedirect.com/science/article/pii/S0277379115000372.

Menviel, L., A. Timmermann, O. Elison Timm, and A. Mouchet (2011). "Deconstructing the Last Glacial termination: the role of millennial and orbital-scale forcings". In: *Quaternary Science Reviews* 30.9, pp. 1155–1172. ISSN: 0277-3791. DOI: 10.1016/j.quascirev.2011.02.005. URL: http://www.sciencedirect.com/science/article/pii/S0277379111000539.

Nakagawa, Takeshi, Pavel E. Tarasov, Kotoba Nishida, Katsuya Gotanda, and Yoshinori Yasuda (2002). "Quantitative pollen-based climate reconstruction in central Japan: application to surface and Late Quaternary spectra". In: *Quaternary Science Reviews* 21.18-19, pp. 2099–2113. DOI: 10.1016/s0277-3791(02)00014-8.

Neukom, Raphael, Luis A. Barboza, Michael P. Erb, Feng Shi, Julien Emile-Geay, Michael N. Evans, Jörg Franke, Darrell S. Kaufman, Lucie Lücke, Kira Rehfeld, Andrew Schurer, Feng Zhu, Stefan Brönnimann, Gregory J. Hakim, Benjamin J. Henley, Fredrik Charpentier Ljungqvist, Nicholas McKay, Veronika Valler, Lucien von Gunten, and P. A. G. E. S. 2k Consortium (2019a). "Consistent multidecadal variability in global temperature reconstructions and simulations over the Common Era". In: *Nature Geoscience* 12.8, pp. 643–649. ISSN: 1752-0908. URL: https://doi.org/10.1038/s41561-019-0400-0.

Neukom, Raphael, Nathan Steiger, Juan Josè Gòmez-Navarro, Jianghao Wang, and Johannes P. Werner (2019b). "No evidence for globally coherent warm and cold periods over the preindustrial Common Era". In: *Nature* 571.7766, pp. 550–554. DOI: 10.1038/s41586-019-1401-2.

Nichols, Harvey (1967). "The Post-glacial history of vegetation and climate at Ennadai Lake, Keewatin, and Lynn Lake, Manitoba (Canada)". In: E&G – Quaternary Science Journal 18.1. DOI: 10.23689/fidgeo-1124.

— (1969). "The Late Quaternary History of Vegetation and Climate at Porcupine Mountain and Clearwater Bog, Manitoba". In: *Arctic and Alpine Research* 1.3, p. 155. ISSN: 00040851. DOI: 10.2307/1550287. URL: http://www.jstor.org/stable/1550287.

Nocke, Thomas (2014). "Images for Data Analysis: The Role of Visualization in Climate Research Processes". In: *IMAGE POLITICS OF CLIMATE CHANGE: VISUALIZATIONS, IMAGINATIONS, DOCUMENTATIONS*. Ed. by Schneider, B and

Nocke, T. Vol. 55. Image-Series, 55–77. ISBN: 978-3-8394-2610-4; 978-3-8376-2610-0.

Nocke, Thomas, Till Sterzel, Michael Böttinger, Markus Wrobel, et al. (2008). "Visualization of climate and climate change data: An overview". In: *Digital earth summit on geoinformatics*, pp. 226–232.

Nolan, Connor, John Tipton, Robert K Booth, Mevin B Hooten, and Stephen T Jackson (2019). "Comparing and improving methods for reconstructing peatland water-table depth from testate amoebae". In: *The Holocene* 29.8, pp. 1350–1361. DOI: 10.1177/0959683619846969.

Oliphant, Travis E (2006). *A guide to NumPy*. Vol. 1. Trelgol Publishing USA. URL: http://www.numpy.org/.

Otto-Bliesner, B. L., P. Braconnot, S. P. Harrison, D. J. Lunt, A. Abe-Ouchi, S. Albani, P. J. Bartlein, E. Capron, A. E. Carlson, A. Dutton, H. Fischer, H. Goelzer, A. Govin, A. Haywood, F. Joos, A. N. LeGrande, W. H. Lipscomb, G. Lohmann, N. Mahowald, C. Nehrbass-Ahles, F. S. R. Pausata, J.-Y. Peterschmitt, S. J. Phipps, H. Renssen, and Q. Zhang (2017). "The PMIP4 contribution to CMIP6 – Part 2: Two interglacials, scientific objective and experimental design for Holocene and Last Interglacial simulations". In: *Geosci. Model Dev.* 10.11, pp. 3979–4003. DOI: 10.5194/gmd-10-3979-2017. URL: https://www.geosci-model-dev.net/10/3979/2017/.

Otto-Bliesner, Bette L., James M. Russell, Peter U. Clark, Zhengyu Liu, Jonathan T. Overpeck, Bronwen Konecky, Peter deMenocal, Sharon E. Nicholson, Feng He, and Zhengyao Lu (2014). "Coherent changes of southeastern equatorial and northern African rainfall during the last deglaciation". In: *Science* 346.6214, pp. 1223–1227. ISSN: 0036-8075. DOI: 10.1126/science.1259531. eprint: https://science.sciencemag.org/content/346/6214/1223.full.pdf. URL: https://science.sciencemag.org/content/346/6214/1223.

Perez, Fernando, Brian E. Granger, and John D. Hunter (2011). "Python: An Ecosystem for Scientific Computing". In: *Computing in Science & Engineering* 13.2, pp. 13–21. DOI: 10.1109/mcse.2010.119.

Phillips, Norman A. (1956). "The general circulation of the atmosphere: a numerical experiment". In: *Quarterly Journal of the Royal Meteorological Society* 82.352, pp. 123–164.

Python Software Foundation (2019). *unittest – Unit testing framework*. URL: https://docs.python.org/3.7/library/unittest.html (visited on 09/02/2019).

R Core Team (2019). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. URL: https://www.R-project.org/.

Rautenhaus, Marc, Michael Böttinger, Stephan Siemen, Robert Hoffman, Robert M. Kirby, Mahsa Mirzargar, Niklas Rober, and Rudiger Westermann (2018). "Visualization in Meteorology—A Survey of Techniques and Tools for Data Analysis Tasks". In: *IEEE Transactions on Visualization and Computer Graphics* 24.12, pp. 3268–3296. DOI: 10.1109/tvcg.2017.2779501.

Rew, R. and G. Davis (1990). "NetCDF: an interface for scientific data access". In: *IEEE Computer Graphics and Applications* 10.4, pp. 76–82. DOI: 10.1109/38.56302.

Röber, Niklas, P. Adamidis, and Jörn Behrens (2015). "Visualization and Analysis of Climate Simulation Performance Data". In: *EGU General Assembly Conference Abstracts*. Vol. 17. EGU General Assembly Conference Abstracts, p. 15318.

Roche, D. M., H. Renssen, D. Paillard, and G. Levavasseur (2011). "Deciphering the spatio-temporal complexity of climate change of the last deglaciation: a model

analysis". In: *Climate of the Past* 7.2, pp. 591–602. DOI: 10.5194/cp-7-591-2011. URL: https://www.clim-past.net/7/591/2011/.

Rocklin, Matthew (2015). "Dask: Parallel Computation with Blocked algorithms and Task Scheduling". In: *Proceedings of the 14th Python in Science Conference*. Ed. by Kathryn Huff and James Bergstra, pp. 130–136.

Schulz, Hans-Jorg, Thomas Nocke, Magnus Heitzler, and Heidrun Schumann (2013). "A Design Space of Visualization Tasks". In: *IEEE Transactions on Visualization and Computer Graphics* 19.12, pp. 2366–2375. DOI: 10.1109/tvcg.2013.120.

Schulzweida, Uwe (2019). *CDO User Guide*. DOI: 10.5281/zenodo.2558193. URL: https://doi.org/10.5281/zenodo.2558193.

Shaw, Anthony (2018). *Getting Started With Testing in Python*. Last accessed: 2019-09-02. URL: https://realpython.com/python-testing/ (visited on 10/22/2018).

Simpson, G. L. (2007). "Analogue Methods in Palaeoecology: Using the analogue Package". In: *Journal of Statistical Software* 22.2, pp. 1–29.

Simpson, G. L. and J. Oksanen (2019). *analogue: Analogue and weighted averaging methods for palaeoecology*. R package version 0.17-3. URL: https://cran.r-project.org/package=analogue.

Skamarock, William C., Joseph B. Klemp, Michael G. Duda, Laura D. Fowler, Sang-Hun Park, and Todd D. Ringler (2012). "A Multiscale Nonhydrostatic Atmospheric Model Using Centroidal Voronoi Tesselations and C-Grid Staggering". In: *Monthly Weather Review* 140.9, pp. 3090–3105. DOI: 10.1175/mwr-d-11-00215.1.

Sommer, Philipp S. (2017). "The psyplot interactive visualization framework". In: *The Journal of Open Source Software* 2.16. DOI: 10.21105/joss.00363. URL: https://doi.org/10.21105/joss.00363.

— (2019). *psy-strat v0.1.0: A Python package for creating stratigraphic diagrams*. DOI: 10.5281/zenodo.3381753. URL: https://doi.org/10.5281/zenodo.3381753.

Sommer, Philipp S. and Jed O. Kaplan (2017). "A globally calibrated scheme for generating daily meteorology from monthly statistics: Global-WGEN (GWGEN) v1.0". In: *Geosci. Model Dev.* 10.10, pp. 3771–3791. DOI: 10.5194/gmd-10-3771-2017.

Sommer, Philipp S., Dilan Rech, Manuel Chevalier, and Basil Davis (2019). "straditize: Digitizing stratigraphic diagrams". In: *Journal of Open Source Software* 4.34, p. 1216. DOI: 10.21105/joss.01216. URL: https://doi.org/10.21105/joss.01216.

Stodden, Victoria and Sheila Miguez (2014). "Best Practices for Computational Science: Software Infrastructure and Environments for Reproducible and Extensible Research". In: *Journal of Open Research Software* 2.1. DOI: 10.5334/jors.ay.

Sullivan, C. and Whitney Trainor-Guitton (2019). "PVGeo: an open-source Python package for geoscientific visualization in VTK and ParaView". In: *Journal of Open Source Software* 4.38, p. 1451. DOI: 10.21105/joss.01451.

Tipton, John (2017). *BayesComposition: Fit forward and inverse prediction Bayesian functional models for compositional data*. R package version 1.0. URL: https://github.com/jtipton25/BayesComposition.

Torborg, Scott (2016). *python-packaging: Tutorial on how to structure Python packages*. Revision 35daf993. URL: https://python-packaging.readthedocs.io (visited on 09/02/2019).

Treut, Hervé Le, Richard Somerville, Ulrich Cubasch, Yihui Ding, Cecilie Mauritzen, Abdalah Mokssit, Thomas Peterson, and Michael Prather (2007). "Historical Overview of Climate Change Science". In: *Climate Change 2007: The Physical Science Basis. Contribution of Working Group I to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change*. Ed. by S. Solomon, D. Qin, M. Manning, Z. Chen,

M. Marquis, K.B. Averyt, M. Tignor, H.L. Miller, S. Solomon, D. Qin, M. Manning, Z. Chen, M. Marquis, K.B. Averyt, M. Tignor, and H.L. Miller. Cambridge, United Kingdom and New York, NY, USA: Cambridge University Press. Chap. 1, pp. 93–123. URL: https://www.ipcc.ch/site/assets/uploads/2018/03/ar4-wg1-chapter1.pdf.

Ulden, A.P. van and G.J. van Oldenborgh (2006). "Large-scale atmospheric circulation biases and changes in global climate model simulations and their importance for climate change in Central Europe". In: *Atmos Chem Phys* 6, pp. 863–881. ISSN: 1680-7324. DOI: 10.5194/acp-6-863-2006.

Varma, V., M. Prange, U. Merkel, T. Kleinen, G. Lohmann, M. Pfeiffer, H. Renssen, A. Wagner, S. Wagner, and M. Schulz (2012). "Holocene evolution of the Southern Hemisphere westerly winds in transient simulations with global climate models". In: *Climate of the Past* 8.2, pp. 391–402. DOI: 10.5194/cp-8-391-2012. URL: https://www.clim-past.net/8/391/2012/.

Vincens, Annie, Anne-Marie Lézine, Guillaume Buchet, Dorothée Lewden, and Annick Le Thomas (2007). "African pollen database inventory of tree and shrub pollen types". In: *Rev. Palaeobot. Palynol.* 145.1-2, pp. 135–141. ISSN: 00346667. DOI: 10.1016/j.revpalbo.2006.09.004.

Walker, Mike, Sigfus Johnsen, Sune Olander Rasmussen, Trevor Popp, Jørgen-Peder Steffensen, Phil Gibbard, Wim Hoek, John Lowe, John Andrews, Svante Björck, Les C. Cwynar, Konrad Hughen, Peter Kershaw, Bernd Kromer, Thomas Litt, David J. Lowe, Takeshi Nakagawa, Rewi Newnham, and Jakob Schwander (2009). "Formal definition and dating of the GSSP (Global Stratotype Section and Point) for the base of the Holocene using the Greenland NGRIP ice core, and selected auxiliary records". In: *J. Quat. Sci.* 24.1, pp. 3–17. ISSN: 02678179 10991417. DOI: 10.1002/jqs.1227.

Wanner, Heinz, Jürg Beer, Jonathan Bütikofer, Thomas J. Crowley, Ulrich Cubasch, Jacqueline Flückiger, Hugues Goosse, Martin Grosjean, Fortunat Joos, Jed O. Kaplan, Marcel Küttel, Simon A. Müller, I. Colin Prentice, Olga Solomina, Thomas F. Stocker, Pavel Tarasov, Mayke Wagner, and Martin Widmann (2008). "Mid- to Late Holocene climate change: an overview". In: *Quaternary Science Reviews* 27.19-20, pp. 1791–1828. ISSN: 0277-3791. DOI: 10.1016/j.quascirev.2008.06.013. URL: http://www.sciencedirect.com/science/article/pii/S0277379108001479.

Weitzel, Nils, Sebastian Wagner, Jesper Sjolte, Marlene Klockmann, Oliver Bothe, Heather Andres, Lev Tarasov, Kira Rehfeld, Eduardo Zorita, Martin Widmann, Philipp Sommer, Gerd Schädler, Patrick Ludwig, Florian Kapp, Lukas Jonkers, Javier Garcìa-Pintado, Florian Fuhrmann, Andrew Dolman, Anne Dallmeyer, and Tim Brücher (2019). "Diving into the Past: A Paleo Data–Model Comparison Workshop on the Late Glacial and Holocene". In: *Bulletin of the American Meteorological Society* 100.1, ES1–ES4. DOI: 10.1175/bams-d-18-0169.1.

Whitmore, J., K. Gajewski, M. Sawada, J.W. Williams, B. Shuman, P.J. Bartlein, T. Minckley, A.E. Viau, T. Webb, S. Shafer, P. Anderson, and L. Brubaker (2005). "Modern pollen data from North America and Greenland for multi-scale paleoenvironmental applications". In: *Quaternary Science Reviews* 24.16-17, pp. 1828–1848. DOI: 10.1016/j.quascirev.2005.03.005.

Williams, John W., Eric C. Grimm, Jessica L. Blois, Donald F. Charles, Edward B. Davis, Simon J. Goring, Russell W. Graham, Alison J. Smith, Michael Anderson, Joaquin Arroyo-Cabrales, Allan C. Ashworth, Julio L. Betancourt, Brian W. Bills, Robert K. Booth, Philip I. Buckland, B. Brandon Curry, Thomas Giesecke, Stephen T. Jackson, Claudio Latorre, Jonathan Nichols, Timshel Purdum, Robert

E. Roth, Michael Stryker, and Hikaru Takahara (2018). "The Neotoma Paleoecology Database, a multiproxy, international, community-curated data resource". In: *Quaternary Research* 89.1, pp. 156–177. DOI: 10.1017/qua.2017.105.

Wodehouse, Roger Philip (1935). *Pollen grains: Their structure, identification and significance in science and medicine*. McGraw-Hill Book Co.

Zängl, Günther, Daniel Reinert, Pilar Rípodas, and Michael Baldauf (2014). "The ICON (ICOsahedral Non-hydrostatic) modelling framework of DWD and MPI-M: Description of the non-hydrostatic dynamical core". In: *Quarterly Journal of the Royal Meteorological Society* 141.687, pp. 563–579. DOI: 10.1002/qj.2378.

Zender, Charles S. (2008). "Analysis of self-describing gridded geoscience data with netCDF Operators (NCO)". In: *Environmental Modelling & Software* 23.10-11, pp. 1338–1342. DOI: 10.1016/j.envsoft.2008.03.004.

— (2016). "Bit Grooming: statistically accurate precision-preserving quantization with compression, evaluated in the netCDF Operators (NCO, v4.4.8+)". In: *Geoscientific Model Development* 9.9, pp. 3199–3211. DOI: 10.5194/gmd-9-3199-2016.

Zender, Charles S. and Harry Mangalam (2007). "Scaling Properties of Common Statistical Operators for Gridded Datasets". In: *The International Journal of High Performance Computing Applications* 21.4, pp. 485–498. DOI: 10.1177/1094342007083802.

Zhang, Q., H. S. Sundqvist, A. Moberg, H. Körnich, J. Nilsson, and K. Holmgren (2010). "Climate change between the mid and late Holocene in northern high latitudes – Part 2: Model-data comparisons". In: *Climate of the Past* 6.5, pp. 609–626. DOI: 10.5194/cp-6-609-2010.

Part I New Software Tools for Paleoclimate Analysis

Chapter 2

Psyplot

A flexible framework for interactive data analysis

2.1 Summary

From

Sommer, Philipp S. (2017e). "The psyplot interactive visualization framework". In: *The Journal of Open Source Software* 2.16. DOI: 10.21105/joss.00363. URL: https://doi.org/10.21105/joss.00363.

psyplot (Sommer, 2017e) is a cross-platform open source python project that mainly combines the plotting utilities of matplotlib (Hunter, 2007) and the data management of the xarray (Hoyer and Hamman, 2017) package and integrates them into a software that can be used via command-line and via a GUI.

The main purpose is to have a framework that allows a fast, attractive, flexible, easily applicable, easily reproducible and especially an interactive visualization of data.

The ultimate goal is to help scientists in their daily work by providing a flexible visualization tool that can be enhanced by their own visualization scripts.

The framework is extended by multiple plugins: psy-simple (Sommer, 2017d) for simple visualization tasks, psy-maps (Sommer, 2017a) for georeferenced data visualization and psy-reg (Sommer, 2017c) for the visualization of fits. It is furthermore extended by the optional graphical user interface psyplot-gui (Sommer, 2017b).

2.2 Introduction

The mathematical and statistical processing of climate data is closely related to its visualization and analysis. But in traditional visual analytics literature, these two aspects are commonly treated in separate manners. Keim et al., 2008 for instance, (following Wijk, 2005) distinguish two steps of visual analytics, the initial data processing with statistical or mathematical techniques, and a *sense-making loop* of visualization, exploration and the gain of new knowledge. Böttinger and Röber, 2019 distinguish the *filtering* step (data processing), and *mapping/rendering* step that describes the visualization. Also in the literature there is a clear division between the climate visualization (or visual analytic) papers and the standard statistical or climate literature that describes new methods for data processing. Visualization research focuses mainly on advanced visualization tools such as ParaView (Ayachit, 2015), VAPOR (Clyne et al., 2007) or Avizo[1] (e.g. Böttinger and Röber, 2019; Nocke

[1] https://www.fei.com/software/avizo3d/

et al., 2015; Rautenhaus et al., 2018; Wong et al., 2014) whereas statistical or climate literature commonly uses R (R Core Team, 2019), Python (Travis E Oliphant, 2006; Perez et al., 2011), CDOs (Schulzweida, 2019) or other command-line tools.

This separation, however, devalues the interplay between the new knowledge from the visualization step, that commonly raises the need for more statistical and mathematical processing of the initial data. This calls for integrated and flexible tools that tackle both steps: the data processing and the visualization, a requirement that is currently not fulfilled by the visualization tools described above. An example software that integrates data processing and data visualization is provided with the Earth System Model Evaluation Tool (ESMValTool) (Eyring et al., 2016). This framework provides common diagnostics for ESMs to enable model intercomparisons. The tool, however has limited interactivity and a slow learning curve for the implementation of new diagnostics.

This lack leads to large efforts of climate scientists to develop scripts for the data processing and visualization. They usually do not follow a systematic framework and as such need to be adapted every time a new project starts which also make them difficult to share with other researchers. The new *psyplot* framework wants to generalize this data processing and visualization by providing a framework that is highly flexible, interoperates with standard computational data processing tools and enables flexible visualizations and adaptations. The software is written in the programming language Python (Perez et al., 2011) and builds upon the visualization package *matplotlib* (Hunter, 2007) and the N-dimensional array processing package *xarray* (Hoyer and Hamman, 2017), that closely interoperates with the numeric packages numpy and scipy (Jones et al., 2001; Travis E Oliphant, 2006) and the parallel computing library dask (Dask Development Team, 2016). Due to the flexibility of Python, it can be used from the command-line, a GUI (section 2.3.3) or jupyter notebooks[2] (Kluyver et al., 2016). As such, it supports out-of-core computation (i.e. the processing of data too large to fit into memory), a rich set of visualization methods from matplotlib, and can be extended to other visualization packages, such as the 3D-visualization framework VTK (Sommer, 2019).

The next section 2.3 provide an overview of the framework with it's data model, plugins and GUI. Sections 2.4 and 2.5 finally discuss further usage and extensions to the software. For more information, usage and implementation examples I also refer to the online documentation `https://psyplot.readthedocs.io`.

2.3 The psyplot framework

The psyplot framework consists of three parts: The core structure that is built upon *xarray* and provides the general infrastructure (section 2.3.1), the plugins that use the plotting functionalities of *matplotlib* (section 2.3.2), and the GUI (section 2.3.3).

2.3.1 Data model

Psyplot and xarray

psyplot acts as a high-level interface into the packages *xarray* and *matplotlib*. The first one is a recent package for N-dimensional labeled arrays that adopts Unidata's self-describing Common Data Model on which the network Common Data Form (netCDF) is built (Brown et al., 1993; Hoyer and Hamman, 2017; Rew and Davis, 1990). The package integrates with standard python from the python environment,

[2] `https://jupyter.org/`

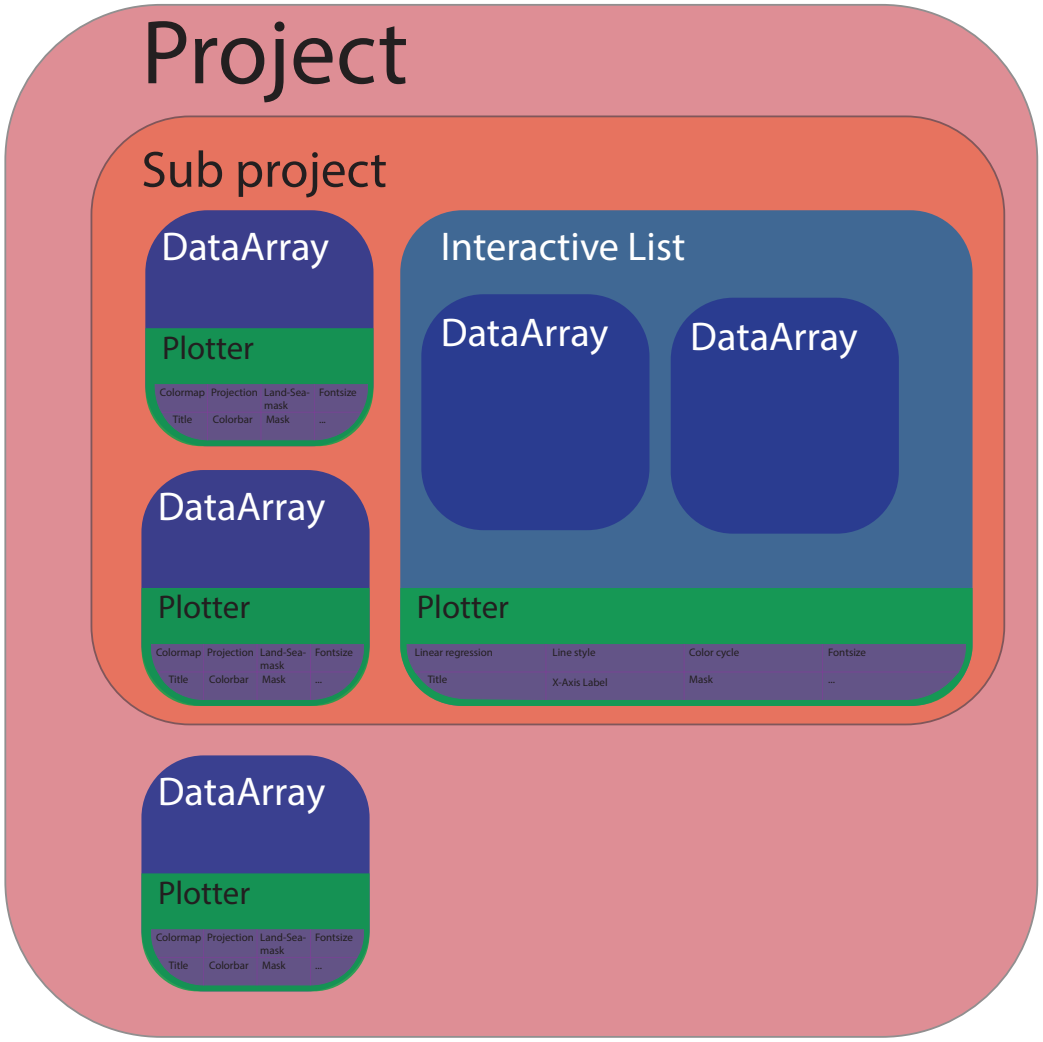


FIGURE 2.1: The psyplot core framework. A (sub) project consists of n-dimensional data arrays or a list of these that are each visualized by a plotter. Each plotter consists of a set of formatoptions that control the appearance of the plot or performs data manipulation.

such as the computing and analysis packages numpy (Travis E Oliphant, 2006), scipy (Jones et al., 2001; Travis E. Oliphant, 2007), pandas (McKinney, 2010) and statsmodels (Seabold and Perktold, 2010), but also offers intuitive interfaces for other packages, such as a package for empirical orthogonal functions (EOFs, Dawson, 2016), CDOs (Müller, 2019), fourier transforms (Uchida et al., 2019) any many more[3]. This large range of extensibility distinguishes psyplot from other high-level visualization software, such as ParaView or Vapor, and they can all be implemented as a formatoption (see below) or used in a pre-processing step.

Psyplot core structure

The core structure of psyplot consists of five base classes that interact with each other, the visualization objects *Plotter* and its *Formatoptions*, the data objects *DataArray* and an *InteractiveList* of them, and a collection of all of them, the psyplot *Project*. It is schematically visualized in figure 2.1.

The most high-level API object is the psyplot project that consists of multiple data objects that are (or are not) visualized. The main purpose is a parallel handling of multiple plots/arrays that may also interact with each other (e.g. through the sharing of formatoptions). It mainly spreads update commands to it's contained objects, but also serves as a filter for the data objects. Furthermore, one project may be split up into sub projects which then only control a specific part of the main project, e.g. for a specific formatting of only a small part of the data.

The next level is the *DataArray* from the xarray package (or more explicitly, it's accessor, the *InteractiveArray*[3]), that holds the data of one (or more) variables (e.g. temperature) and its corresponding coordinates (e.g. time, latitude, longitude, etc.). It may be one or multidimensional depending on the chosen visualization method. psyplot offers several methods to provide the coordinates for the plotting of different grids to make the visualization easier. The software can interprete CF Conventions[4] and UGRID conventions for unstructured grids (Jagers et al., 2018).

Multiple of these arrays can also be grouped together into an *InteractiveList* that shall be visualized by the same plot method (e.g. multiple lines or a scalar field with overlying vector field).

The visualization part in the framework is managed by the *Plotter* class, a collection of multiple *Formatoptions*. Each plotter subclass is designed to visualize the data in a specific manner (e.g. via line plots, violin plots, or map plots) and is completely defined through it's formatoptions.

Formatoptions are the core of the psyplot structure. The standard functionality of a formatoption is to control the visual appearance of one aspect of the plot (e.g. through the colormap, figure title, etc.). It is, however, completely unlimited and can also do data manipulations or calculations. The psy-reg plugin for example (see section 2.3.2) implements a formatoption that performs a regression through the data that is then visualized. As mentioned earlier, each plotter is set up through it's formatoptions where each formatoption has a unique formatoption key inside the plotter. This formatoption key (e.g. *title* or *cmap*) is what is used for updating the plot, manipulating the data, etc.. Formatoptions might also interact with other formatoptions inside the plotter or from other plotters. This concept of formatoptions allows to use the same formatoption with all different kinds of plotters and the interaction of multiple plots with each other. Common plot features, such as the figure title, colormap, etc., therefore don't have to be implemented explicitly for every plotter but can be used from existing implementations. This framework also allows a very easy integration and development of own formatoptions with a low or high level of complexity.

2.3.2 Psyplot plugins

The psyplot package provides the core of the data management described in the previous section 2.3.1. The real visualization is implemented in external plugins.

[3] several packages related to xarray are listed in the docs at `http://xarray.pydata.org/en/stable/related-projects.html` and psyplots integration (accessors) in particular is shown at `https://psyplot.readthedocs.io/en/latest/accessors.html`.

[4] `http://cfconventions.org`

The advantage of this approach is an increased flexibility of the entire framework (collaborations can evolve through dedicated plugins) and of managing the various dependencies of the packages. As such, the dependencies of psyplot are rather week (only xarray is needed), but the dependencies of the plugins can be more extensive (e.g. for geo-referencing or advanced statistics).

Each plugin defines new *Plotters* and *Formatoptions* that are specific to the purpose of the visualization/analysis task. The plotters can also be implemented as a plot method (see supplements 2.B to 2.E) and accessed through the psyplot core API (see supplements 2.A for an example).

The current lists of plugins include *psy-simple* for rather simple and standard visualization tasks, *psy-maps* for geo-referenced plots, *psy-reg* for statistical analysis visualization, and *psy-strat* for stratigraphic diagrams.

psy-simple: The psyplot plugin for simple visualizations

Much of the functionality that is used by other plugins is developed in the psy-simple plugin. This package targets simple visualizations and currently includes plot methods for one-dimensional data: line plots, bar plots and violin plots; for two-dimensional data: scalar plots, vector plots and combined scalar and vector plots; and plots that do not require complex data manipulation: a density plot and a plot of the weighted geographic mean. A full list of examples is provided in the supplementary material, section 2.B.

This package also implements most of the functionality to handle unstructured grids in 2D visualizations and defines most of the commonly used formatoptions. The latter include text manipulation (such as plot title, figure title, x- and y-axis labels, etc.), data masking, x- and y-axis tick labeling and positioning, as well as color coding for 2D plots (colormap, colormap sections, etc.).

psy-maps: The psyplot plugin for visualizations on a map

psy-maps builds on top of the psy-simple plugin and extends it's functionality for visualizations on a map using the functionalities of the cartopy package (Met Office, 2010 - 2015) (see supplements 2.C for examples). As such simplifies the automated generation of maps for climate model data through the flexibility of the psyplot framework.

psy-maps currently implements additional formatoptions for choosing the projection of the map, selecting the geographic region, drawing the contintens or shaded reliefs of land and ocean, and more. One feature that distinguishes psy-maps from other visualization software, even from pure cartopy, is the ability to visualize unstructured geo-referenced grids on the map. For this purpose, triangles are projected in a pre-processing step to the target projection, prior to the visualization with matplotlib. This drastically increases the performance and makes it possible to visualize even very large data sets. As such, psy-maps visualizes a global scalar field on a hexagonal grid of roughly 4.4 million grid cells ($\approx$ 13 km resolution) in roughly 3.5 minutes. The interactive usage of such a large dataset is however limited by the functionalities of matplotlib to handle such an immense amount of data.

psy-reg: The psyplot plugin for visualizing and calculating regression plots

psy-reg performs regression analysis on 1D variables using the methods of the statsmodels (Seabold and Perktold, 2010) and scipy (Jones et al., 2001; Travis E. Oliphant, 2007) packages, and visualizes the results with the functionalities of the psy-simple

plugin. As such, it implements formatoptions for univariate regressions, confidence intervals via bootstrapping, and combined plots of the data density and the fitted model (see also supplements 2.D). The necessity for this package arose from the need to visualize a regression model, compare it (visually) with the original data and to use it afterwards. Other python packages either focus only on the generation of the regressions (such as statsmodels or scipy), or on their visualization (such as seaborn (Waskom et al., 2018)). The psyplot plugin makes it possible to generate the visualization and to access the underlying regression model parameters and uncertainties.

psy-reg has been heavily used for the parameterization of the weather generator in chapter 5 which also gave the initial motivation for the package.

psy-strat: A psyplot plugin for stratigraphic plots

psy-strat is the latest plugin for psyplot that has been developed for stratigraphic diagram visualization. It is particularly designed for the straditize software (Sommer et al., 2019, chapter 3) and was motivated by the need for an automated creation of pollen diagrams. One example of such a diagram is provided in the supplementary material, section 2.E.

As the psy-reg and psy-maps plugins, psy-strat uses the functionalities of the psy-simple plugin for a visualization of multiple variables in separate diagrams that share one common vertical axis (usually age or depth)[5]. Additionally, besides the integration that is common for every psyplot plugin (see next section 2.3.3), psy-strat contains additional functionalities for the psyplot GUI. This implementation allows the user to select and reorder the variables (pollen taxa) that are shown in the stratigraphic diagram.

2.3.3 The psyplot Graphical User Interface

Psyplots objective of providing a platform for flexible and convenient data analysis is further approach with the *psyplot-gui* package. This extension to the framwork provides a GUI for simplified access to the plotting features in psyplot.

A strong focus of this interface is, again, the flexibility. psyplot-gui is based on the cross-platform PyQt5 library[6], a very flexible and frequently used package for graphical user interfaces. This enables other software to develop additional features for the package (see psy-strat in the previous section 2.3.2, for instance, or straditize in chapter 3) and to flexibly change the layout of the application. The GUI is complemented with an interactive console to provide a fully integrated python environment for data analysis.

The next paragraphs provide an overview on the various widgets, that are also displayed in figure 2.2 and 2.3.

Console

The central aspects to guarantee flexibility of the application is an in-process IPython console, based on the qtconsole package[7] that provides the possibility to communicate with the psyplot package via the command line and to load any other module or to run any other script or notebook, or even to run commands in different programming languages, such as R (R Core Team, 2019) or Julia (Bezanson et al., 2017).

[5] See psy-strat.readthedocs.io for an example of psy-strat.

[6] PyQt5 can be accessed via `https://riverbankcomputing.com/software/pyqt/intro`.

[7] `https://github.com/jupyter/qtconsole`

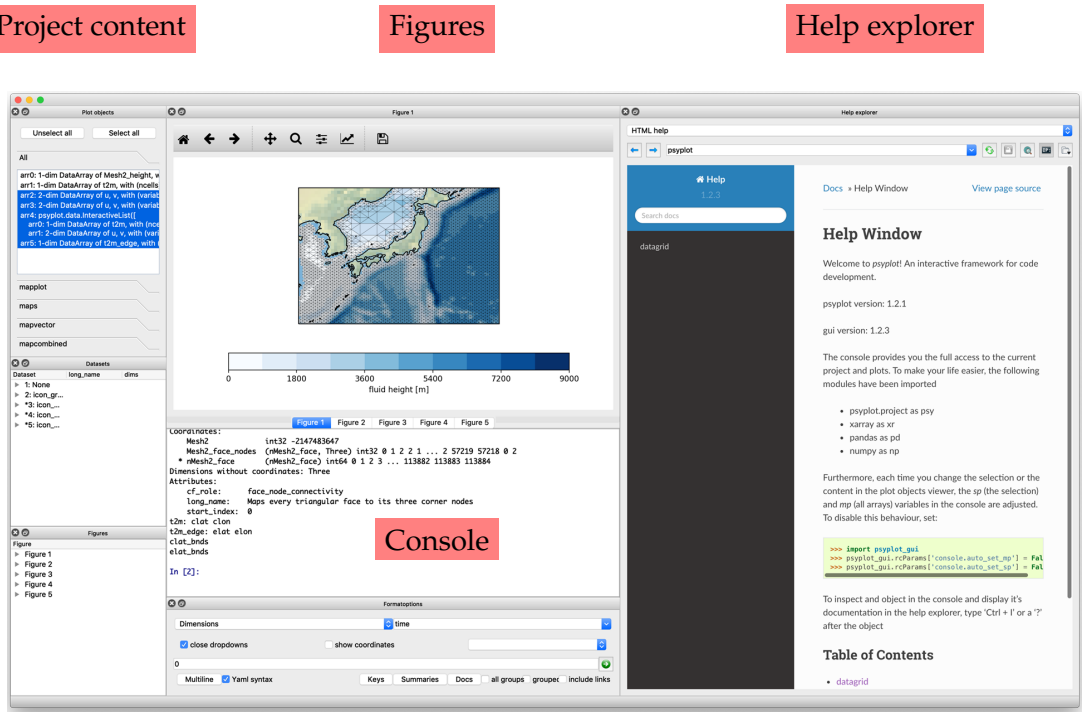


FIGURE 2.2: Screenshot of the psyplot GUI. The left part shows the content of the psyplot project, the upper center the plots, and the right part contains the help explorer. Below the plots, there is also the IPython console for the usage from the command line and a widget to update the formatoptions of the current project.

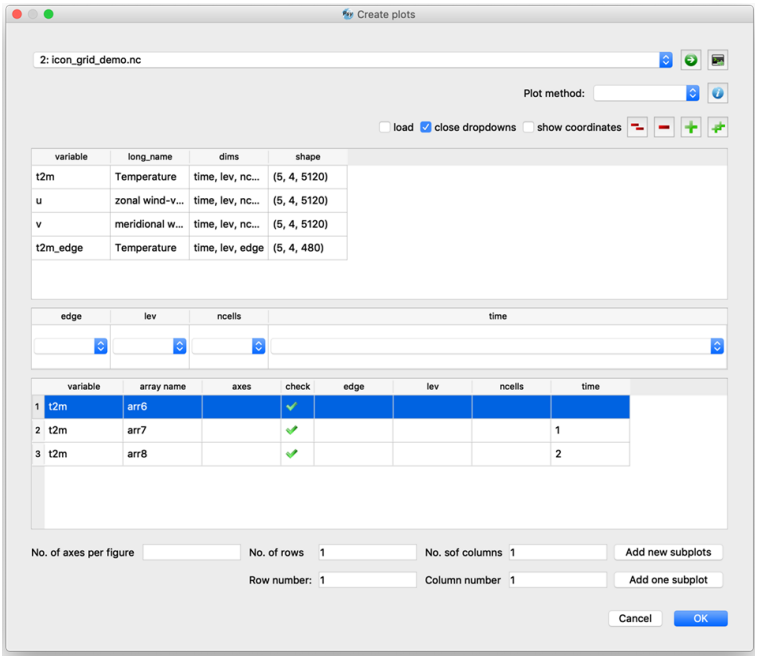


FIGURE 2.3: Plot creation dialog to generate new figures from an xarray dataset.

The console is fully integrated both ways into the GUI. The documentation of every python object in the terminal, for instance, can be viewed in the help explorer of the GUI. And vice versa: a change of the current project through the project content widgets, also changes the corresponding python variable in the shell.

Help explorer

As a complement to the console, the GUI contains a help explorer to provide immediate and dynamic access to the documentation of python objects in the console, rendered as an HTML webpage[8]. Furthermore, the help explorer is connected to multiple other widgets of the GUI in order to provide a dynamically generated documentation. The documentation of available formatoptions in the psyplot project, for instance, are rendered as HTML upon request, in order to make the various plot methods more accessible. The same principle works for the plot methods that are accessible in the plot creator.

Plot creator

The plot creator (figure 2.3) is the starting point of the GUI into the psyplot framework (at least, if one does not use the console or a script to generate the plots). It loads data from the disk or the in-process console, and essentially provides a wrapper around the psyplot plotting call (see suppl. section 2.A). It additionally displays the documentation of the method and its associated formatoptions. This widget creates new plots, that are appended to the psyplot project and are accessible through the console and the project content widgets.

Project content

The psyplot project is the most high-level API element in the psyplot framework (see section 2.3.1) and is displayed in the project content widgets of the GUI. All other elements, such as the formatoptions widget or the plot creator, are interfering with the project, and it is accessible as a variable in the console. The project content widget can be used to see the various items in the project, but it is also used to select the specific items for the so-called *current* sub-project. The latter is dynamically set in the console through the `sp` variable and it is used by the formatoptions widget to update the plotting parameters of the selected items.

Formatoptions

As mentioned in section 2.3.1, formatoptions are the core elements in psyplot that control the figure aesthetics of the plots and/or perform data manipulations. The generic formatoptions widget provides access to these parameters, in order to update them for the selected items in the current project. The formatoption itself (i.e. the python object) can in turn generate a widget that is implemented in the formatoptions widget, to make the available options more accessible. The *title* formatoption, for instance, generates a drop-down menu to select variable attributes (e.g. variable name, variable units, etc.) which is then embedded in the formatoptions widget. The modifications of the formatoptions via this widgets, updates the figures of the selected items.

[8] The help explorer widget has been originally motivated by the *Help* widget of the Scientific PYthon Development EnviRonment, Spyder (`https://www.spyder-ide.org/`) and uses the sphinx package (Hasecke, 2019) to convert restructured Text into HTML.

Figures and plots

The plots generated by the plotting methods are displayed in dedicated widgets inside the GUI and can be dynamically adjusted using the formatoptions widget or the console. The underlying library of the current implemented psyplot plugins, matplotlib, implements multiple backends to display the data interactively, or to export them as PDF, PNG, etc. The psyplot GUI has implemented a backend on top of the PyQt5 backend of matplotlib, which embeds the figures in the GUI. psyplot can, however, work with any backend of matplotlib and does not depend on the specific implementation.

2.4 Conclusions

psyplot (Sommer, 2017e) is a new data visualization framework that integrates rich computational and mathematical software into a flexible framework for visualization. It differs from most of the visual analytic software such that it focuses on extensibility in order to flexibly tackle the different types of analysis questions that arise in pioneering research. The design of the high-level API of the framework enables a simple and standardized usage from the command-line, python scripts or jupyter notebooks. A modular plugin framework enables a flexible development of the framework that can potentially go into many different directions. The additional enhancement with a flexible GUI makes it the only visualization framework that can be handled from the conveniently command-line, and via point-click handling. It also allows to build further desktop applications on top of the existing framework.

The plugins of psyplot currently provide visualization methods that range from simple line plots, to density plots, regression analysis and geo-referenced visualization in two dimensions. The software is currently entirely based on the visualization methods of matplotlib (Hunter, 2007), the most established visualization package in the scientific python community. However, the framework itself is agnostic to the underlying visualization method and can, as such, leverage a variety of existing analytical software.

2.5 Outlook

The possibilities for further development of the psyplot framework are numerous, due to it's intrinsic generality. The core of the psyplot framework will, in the future, be extended with a standardized algorithm for the generation of animations. Psyplot projects already have the functionality of being saved to a file and reloaded, but they will also be exportable as python scripts for a more flexible reusability and adaptability. The update process within a psyplot project (currently every item in the project is updated in parallel) also has potential for improvement by using a single-threaded scheduler approach that better reflects if one formatoption depends on the formatoptions of another plotter.

The GUI has especially high potential for further development, as it still lacks widgets to quickly and intuitively modify the visual appearance of the plots. The only possibility inside the GUI (besides the console) is to use the formatoptions widget whose main focus however is on flexibility, rather than usability and has, as such, limited possibilities for adaptation to specific use cases.

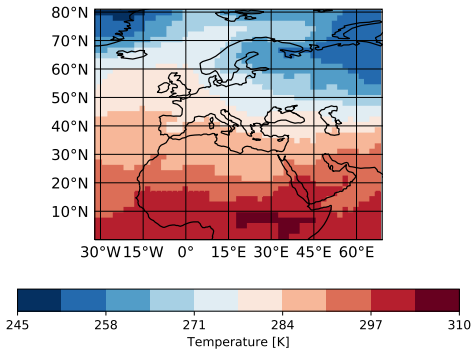
Another focus will be the development of new plot methods inside the psyplot framework. The major aspect will be on the development of 3D visualization methods of geo-referenced data, using recently published software that builds on top of

the visualization toolkit VTK (Sullivan and Kaszynski, 2019; Sullivan and Trainor-Guitton, 2019), see Sommer, 2019. psyplot has the unique potential to generate 3D visualizations conveniently from the command line, a distinguishing feature, compared to other visualization software packages, such as ParaView or Vapor. Further potential enhancements for visualizations can involve standard interactive visual analytic tools, e.g. such that the interactive selection of features in one plot affects the visualization in another plot (so-called brushing and linking).

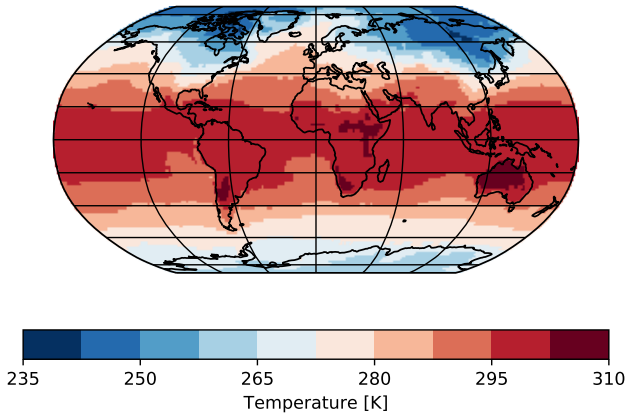
Supplementary material

2.A Example call of a plot method

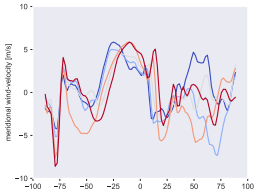
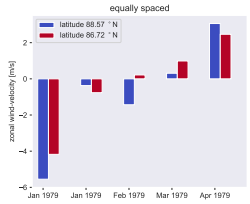
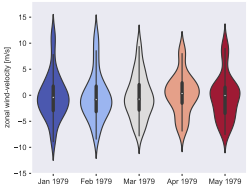
```
# example call for generating a map
import psyplot.project as psy

maps = psy.plot.mapplot(
    'psy-maps-demo.nc',  # input file name, can also be data in memory
    name='t2m',          # variable to plot (can also be multiples
    ### formatoptions
    # colorbar label uses meta attributes of netCDF variable
    clabel='%(long_name)s [%(units)s]',
    # select colormap
    cmap='RdBu_r',
    # focus on a specific lonlatbox given by [lonmin, lonmax, latmin, latmax]
    lonlatbox=['Europe', 'Europe', 0, 'Europe'])

maps.show()
```

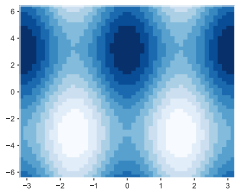
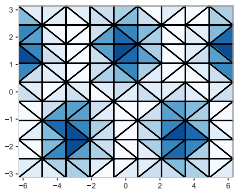
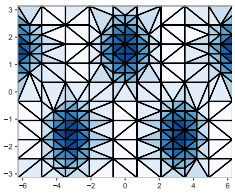


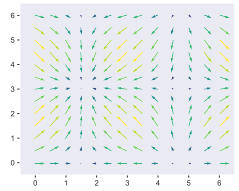
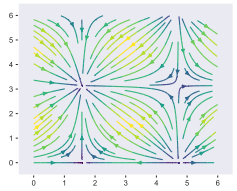
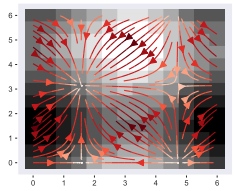

```
# Update the plot, e.g. change projection, plot global

maps.update(projection='robin', lonlatbox=None)
```

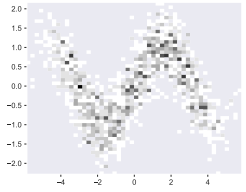
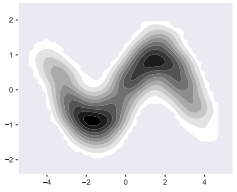
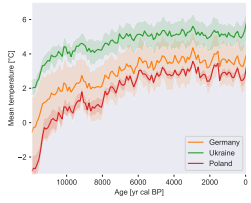


2.B psy-simple plot methods

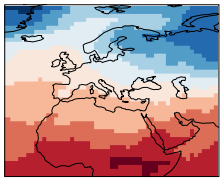
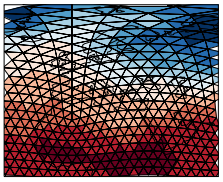
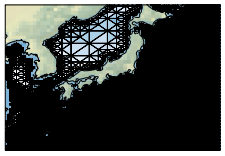
Plot method	lineplot	barplot	violinplot
Example			

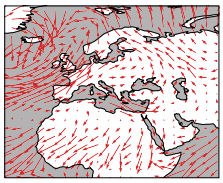
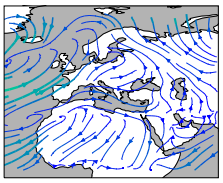
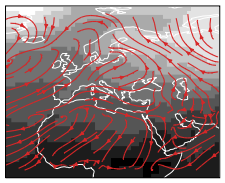
Plot method	plot2d	
Grid type	rectilinear	unstructured
Example		 

Plot method	vector	combined
Example	 	

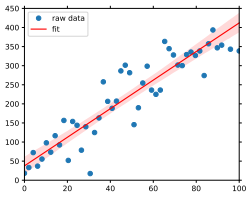
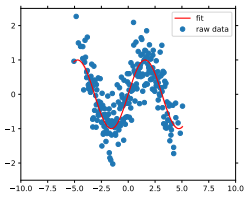
Plot method	density	fldmean
Example	 	

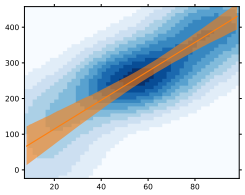
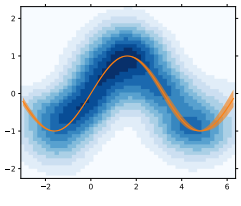
2.C psy-maps plot methods

Plot method	mapplot		
Grid type	rectilinear	unstructured	
Example			

Plot method	mapvector		combined
Example			

2.D psy-reg plot methods

Plot method	linreg	
Example		

Plot method	densityreg	
Example		

2.E psy-strat plot methods

References

Ayachit, Utkarsh (2015). *The paraview guide: a parallel visualization application*. Kitware, Inc.

Bezanson, Jeff, Alan Edelman, Stefan Karpinski, and Viral B. Shah (2017). "Julia: A Fresh Approach to Numerical Computing". In: *SIAM Review* 59.1, pp. 65–98. DOI: 10.1137/141000671. eprint: https://doi.org/10.1137/141000671. URL: https://doi.org/10.1137/141000671.

Böttinger, Michael and Niklas Röber (2019). "Visualization in Climate Modelling". In: *International Climate Protection*. Ed. by Michael Palocz-Andresen, Dòra Szalay, Andràs Gosztom, Làszlò Sìpos, and Tìmea Taligàs. Cham: Springer International Publishing, pp. 313–321. ISBN: 978-3-030-03816-8. DOI: 10.1007/978-3-030-03816-8_39. URL: https://doi.org/10.1007/978-3-030-03816-8_39.

Brown, Stewart A., Mike Folk, Gregory Goucher, Russ Rew, and Paul F. Dubois (1993). "Software for Portable Scientific Data Management". In: *Computers in Physics* 7.3, p. 304. DOI: 10.1063/1.4823180.

Clyne, John, Pablo Mininni, Alan Norton, and Mark Rast (2007). "Interactive desktop analysis of high resolution simulations: application to turbulent plume dynamics and current sheet formation". In: *New Journal of Physics* 9.8, pp. 301–301. DOI: 10.1088/1367-2630/9/8/301.

Dask Development Team (2016). *Dask: Library for dynamic task scheduling*. URL: https://dask.org.

Dawson, Andrew (2016). "eofs: A Library for EOF Analysis of Meteorological, Oceanographic, and Climate Data". In: *Journal of Open Research Software* 4. DOI: 10.5334/jors.122.

Eyring, Veronika, Mattia Righi, Axel Lauer, Martin Evaldsson, Sabrina Wenzel, Colin Jones, Alessandro Anav, Oliver Andrews, Irene Cionni, Edouard L. Davin, Clara Deser, Carsten Ehbrecht, Pierre Friedlingstein, Peter Gleckler, Klaus-Dirk Gottschaldt, Stefan Hagemann, Martin Juckes, Stephan Kindermann, John Krasting, Dominik

Kunert, Richard Levine, Alexander Loew, Jarmo Mäkelä, Gill Martin, Erik Mason, Adam S. Phillips, Simon Read, Catherine Rio, Romain Roehrig, Daniel Senftleben, Andreas Sterl, Lambertus H. van Ulft, Jeremy Walton, Shiyu Wang, and Keith D. Williams (2016). "ESMValTool (v1.0) – a community diagnostic and performance metrics tool for routine evaluation of Earth system models in CMIP". In: *Geoscientific Model Development* 9.5, pp. 1747–1802. DOI: 10.5194/gmd-9-1747-2016. URL: https://www.geosci-model-dev.net/9/1747/2016/.

Hasecke, Jan Ulrich (2019). *Software-Dokumentation mit Sphinx: Zweite überarbeitete Auflage (Sphinx 2.0) (German Edition)*. Independently published. ISBN: 1793008779. URL: https://www.amazon.com/Software-Dokumentation-mit-Sphinx-%C3%83%C2%BCberarbeitete-Auflage/dp/1793008779?SubscriptionId=AKIAIOBINVZYXZQZ2U3A%5C&tag=chimbori05-20%5C&linkCode=xm2%5C&camp=2025%5C&creative=165953%5C&creativeASIN=1793008779.

Hoyer, S. and J. Hamman (2017). "xarray: N-D labeled arrays and datasets in Python". In: *Journal of Open Research Software* 5.1. DOI: 10.5334/jors.148. URL: http://doi.org/10.5334/jors.148.

Hunter, J. D. (2007). "Matplotlib: A 2D Graphics Environment". In: *Computing in Science Engineering* 9.3, pp. 90–95. ISSN: 1521-9615. DOI: 10.1109/MCSE.2007.55.

Jagers, Bert, David Stuebe, Tom Gross, Chris Barker, Brian Zelenke, Rich Signell, Bob Oehmke, Alex Crosby, Karen Schuchardt, David Ham, Brian Blanton, Cristina Forbes, Charles Seaton, Dave Forrest, Bill Howe, Geoff Cowles, and Phil Elson (2018). *UGRID Conventions (v1.0)*. URL: http://ugrid-conventions.github.io/ugrid-conventions (visited on 09/05/2019).

Jones, Eric, Travis Oliphant, Pearu Peterson, et al. (2001). *SciPy: Open source scientific tools for Python*. [Online; accessed 2017-02-18]. URL: http://www.scipy.org/.

Keim, Daniel, Gennady Andrienko, Jean-Daniel Fekete, Carsten Görg, Jörn Kohlhammer, and Guy Melançon (2008). "Visual Analytics: Definition, Process, and Challenges". In: *Information Visualization: Human-Centered Issues and Perspectives*. Ed. by Andreas Kerren, John T. Stasko, Jean-Daniel Fekete, Chris North, Andreas Kerren, John T. Stasko, Jean-Daniel Fekete, and Chris North. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 154–175. ISBN: 978-3-540-70956-5. DOI: 10.1007/978-3-540-70956-5_7. URL: https://doi.org/10.1007/978-3-540-70956-5_7.

Kluyver, Thomas, Benjamin Ragan-Kelley, Fernando Pérez, Brian Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica Hamrick, Jason Grout, Sylvain Corlay, Paul Ivanov, Damián Avila, Safia Abdalla, and Carol Willing (2016). "Jupyter Notebooks – a publishing format for reproducible computational workflows". In: *Positioning and Power in Academic Publishing: Players, Agents and Agendas*. Ed. by F. Loizides and B. Schmidt. IOS Press, pp. 87–90. DOI: 10.3233/978-1-61499-649-1-87.

McKinney, Wes (2010). "Data Structures for Statistical Computing in Python". In: *Proceedings of the 9th Python in Science Conference*. Ed. by Stéfan van der Walt and Jarrod Millman, pp. 51–56.

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

Müller, Ralf (2019). *cdo-bindings: Ruby/Python bindings for CDO*. Last accessed: 2019-09-05. URL: https://github.com/Try2Code/cdo-bindings (visited on 09/05/2019).

Nocke, T., S. Buschmann, J. F. Donges, N. Marwan, H.-J. Schulz, and C. Tominski (2015). "Review: visual analytics of climate networks". In: *Nonlinear Processes in Geophysics* 22.5, pp. 545–570. DOI: 10.5194/npg-22-545-2015. URL: https://www.nonlin-processes-geophys.net/22/545/2015/.

Oliphant, Travis E (2006). *A guide to NumPy*. Vol. 1. Trelgol Publishing USA. URL: http://www.numpy.org/.

Oliphant, Travis E. (2007). "Python for Scientific Computing". In: *Computing in Science & Engineering* 9.3, pp. 10–20. DOI: 10.1109/mcse.2007.58.

Perez, Fernando, Brian E. Granger, and John D. Hunter (2011). "Python: An Ecosystem for Scientific Computing". In: *Computing in Science & Engineering* 13.2, pp. 13–21. DOI: 10.1109/mcse.2010.119.

R Core Team (2019). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. URL: https://www.R-project.org/.

Rautenhaus, Marc, Michael Böttinger, Stephan Siemen, Robert Hoffman, Robert M. Kirby, Mahsa Mirzargar, Niklas Rober, and Rudiger Westermann (2018). "Visualization in Meteorology—A Survey of Techniques and Tools for Data Analysis Tasks". In: *IEEE Transactions on Visualization and Computer Graphics* 24.12, pp. 3268–3296. DOI: 10.1109/tvcg.2017.2779501.

Rew, R. and G. Davis (1990). "NetCDF: an interface for scientific data access". In: *IEEE Computer Graphics and Applications* 10.4, pp. 76–82. DOI: 10.1109/38.56302.

Schulzweida, Uwe (2019). *CDO User Guide*. DOI: 10.5281/zenodo.2558193. URL: https://doi.org/10.5281/zenodo.2558193.

Seabold, Skipper and Josef Perktold (2010). *Statsmodels: Econometric and statistical modeling with python*.

Sommer, Philipp S. (2017a). "Chilipp/psy-maps: v1.0.0: First official and stable release". In: DOI: 10.5281/zenodo.845712. URL: https://doi.org/10.5281/zenodo.845712.

— (2017b). "Chilipp/psyplot-gui: v1.0.1: Graphical User Interface for the psyplot package". In: DOI: 10.5281/zenodo.845726. URL: https://doi.org/10.5281/zenodo.845726.

— (2017c). "Chilipp/psy-reg: v1.0.0: First official and stable release". In: DOI: 10.5281/zenodo.845717. URL: https://doi.org/10.5281/zenodo.845717.

— (2017d). "Chilipp/psy-simple: v1.0.0: First official and stable release". In: DOI: 10.5281/zenodo.845705. URL: https://doi.org/10.5281/zenodo.845705.

— (2017e). "The psyplot interactive visualization framework". In: *The Journal of Open Source Software* 2.16. DOI: 10.21105/joss.00363. URL: https://doi.org/10.21105/joss.00363.

— (2019). *psy-vtk: A VTK plugin for psyplot*. Last accessed: 2019-05-27. URL: https://github.com/Chilipp/psy-vtk (visited on 05/27/2019).

Sommer, Philipp S., Dilan Rech, Manuel Chevalier, and Basil Davis (2019). "straditize: Digitizing stratigraphic diagrams". In: *Journal of Open Source Software* 4.34, p. 1216. DOI: 10.21105/joss.01216. URL: https://doi.org/10.21105/joss.01216.

Sullivan, C. and Alexander Kaszynski (2019). "PyVista: 3D plotting and mesh analysis through a streamlined interface for the Visualization Toolkit (VTK)". In: *Journal of Open Source Software* 4.37, p. 1450. DOI: 10.21105/joss.01450.

Sullivan, C. and Whitney Trainor-Guitton (2019). "PVGeo: an open-source Python package for geoscientific visualization in VTK and ParaView". In: *Journal of Open Source Software* 4.38, p. 1451. DOI: 10.21105/joss.01451.

Uchida, Takaya, Ariel Rokem, Tom Nicholas, Ryan Abernathey, Jake Vanderplas, Yaroslav Halchenko, Andreas Mayer, Greg Wilson, Kai Pak, and Aurélien Ponte (2019). *xgcm/xrft v0.2.0*. DOI: 10.5281/zenodo.1402635.

Waskom, Michael, Olga Botvinnik, Drew O'Kane, Paul Hobson, Joel Ostblom, Saulius Lukauskas, David C Gemperline, Tom Augspurger, Yaroslav Halchenko, John

B. Cole, Jordi Warmenhoven, Julian De Ruiter, Cameron Pye, Stephan Hoyer, Jake Vanderplas, Santi Villalba, Gero Kunter, Eric Quintero, Pete Bachant, Marcel Martin, Kyle Meyer, Alistair Miles, Yoav Ram, Thomas Brunner, Tal Yarkoni, Mike Lee Williams, Constantine Evans, Clark Fitzgerald, , Brian, and Adel Qalieh (2018). *mwaskom/seaborn: v0.9.0 (July 2018)*. DOI: 10.5281/zenodo.592845.

Wijk, J. J. van (2005). "The Value of Visualization". In: *VIS 05. IEEE Visualization, 2005.* IEEE, pp. 79–86. DOI: 10.1109/visual.2005.1532781.

Wong, Pak Chung, Han-Wei Shen, Ruby Leung, Samson Hagos, Teng-Yok Lee, Xin Tong, and Kewei Lu (2014). "Visual analytics of large-scale climate model data". In: *2014 IEEE 4th Symposium on Large Data Analysis and Visualization (LDAV)*. IEEE, pp. 85–92. DOI: 10.1109/ldav.2014.7013208.

Chapter 3

Straditize

A digitization software for pollen diagrams

Need to write chapter 3

- quaternary paper
- straditize builds upon the psyplot GUI
- fill the gaps for large-scale climate reconstructions

Chapter 4

The EMPD and POLNET web-interfaces

4.1 Summary

The Eurasian (née European) Modern Pollen Database (EMPD) was established in 2013 as a public database of quality controlled and standardized modern pollen surface sample data to compliment the European Pollen Database (EPD) for fossil pollen (B. A. S. Davis et al., 2013). The first version of the EMPD (referenced herein as the EMPD1) contained almost 5000 samples, submitted by over 40 individuals and research groups from all over Europe. Over the last 6 years more data has continued to be submitted, and more efforts have been made to incorporate more data held in open data repositories such as PANGAEA, and as supplementary information in published studies. This data is now released as the Eurasian Modern Pollen Database, version 2 (Basil A. S. Davis et al., in prep) with an increase of 80 percent to 8663 samples (see figure 4.1).

The EMPD remains the only public and open access database of modern pollen samples covering the Eurasian continent and is entirely driven by the community of its data contributors. This effort of creating an open and accessible database led to the development of new open source data management tools that we present in this article. The EMPD2 is now hosted on the version control platform Github, with a dedicated web viewer at EMPD2.github.io and a automated administration app, the EMPD-admin (see table 4.1 for a list of the web resources). The new framework provides a simplified and transparent administration of multiple contributions from different sources and people to the database. All web components are hosted without any additional costs and as such, they have the potential to be applied for other community-based (regional) pollen databases, such as the Latin American Pollen

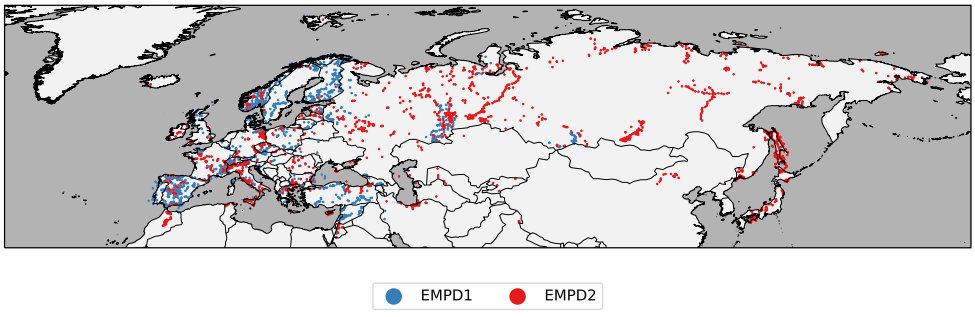


FIGURE 4.1: Modern calibration samples in the EMPD.

TABLE 4.1: EMPD Web resources

	Description	Online Access
EMPD2	Github Organization	github.com/EMPD2
EMPD-Viewer	Map-based web interface to the EMPD database	github.com/EMPD2/EMPD-Viewer empd2.github.io
EMPD-Data	Version controlled data repository of the EMPD	github.com/EMPD2/EMPD-data
EMPD-Admin	Automated administration web app for the EMPD	github.com/EMPD2/EMPD-admin empd-admin.herokuapp.com EMPD2.github.io/EMPD-admin

Database (LAPD) or African Pollen Database (APD), for instance. Especially the light-weight EMPD-viewer web interface can be ported to other database (as shown in section 4.3) to make diverse data accessible to the broad public.

4.2 The EMPD web framework

The EMPD web framework is built on very common open source software development tools that have been adopted for a transparent data management, in favor of open science. The EMPD is now hosted on the web platform Github (github.com/EMPD2). This web platform, free of charge, hosts the source code for many popular open source software packages but can also be used to host a diverse, but small database (in terms of megabytes), such as the EMPD. Github builts upon the version control system *git* that transparently manages changes to documents by providing a full history of their revisions. The web platform is intrinsically designed for community-based projects that focus on collaboration and contains many features for a transparent communication between users, maintainers and contributors of a project. Besides others, the platform provides repository (i.e. project) specific discussion pages, so-called issues, where users can provide feedback, report bugs, or discuss any other aspect of the project. These issues are often linked to so-called pull requests, where each pull request is a proposal for a change in the source files of the project. This is then discussed between project maintainer and contributor in a dedicated discussion/review page.

Another common feature for Github repositories are integrations with so-called Continuous Integration (CI) services, e.g. for automated testing and/or packaging the software. These services run predefined scripts (for example test scripts) every time someone contributes to the repository, or creates a pull request.

The following sections describe how these software development tools are implemented in the three components of the EMPD web framework, the EMPD-viewer (section 4.2.1), the EMPD data repository (section 4.2.2) and the EMPD-admin (section 4.2.3).

4.2.1 The EMPD viewer

The main public interface into the EMPD is an interactive web viewer accessible from EMPD2.github.io. This JavaScript-based application (see figure 4.2 for a screenshot) provides an intuitive interface into the database without requiring any particular computer expertise. It enables the user to view the data on a map and select and

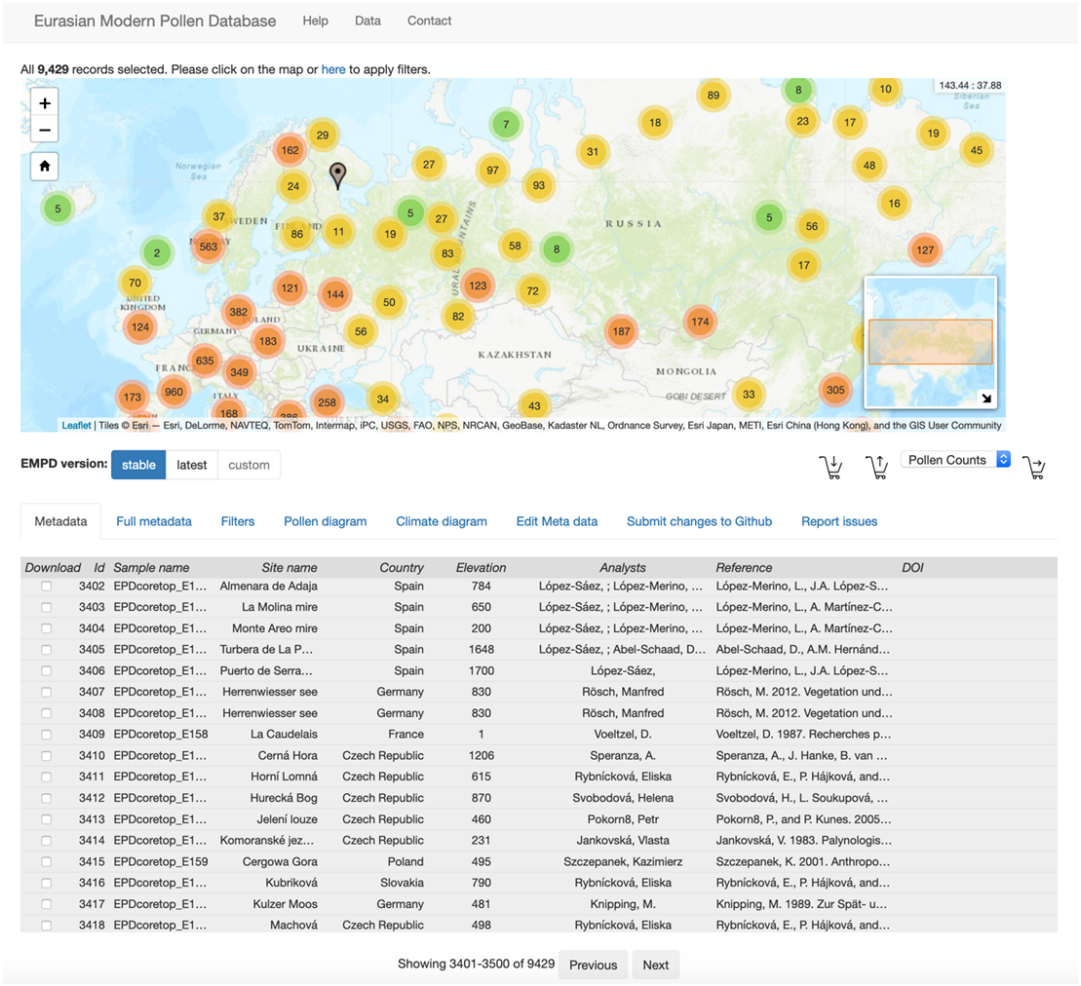


FIGURE 4.2: Screenshot of the EMPD viewer

download subsets of the database. The webpage involves no server-side processing and such it can be hosted for free using the service provided by Github Pages (pages.github.com). This provides a stable access to the database, independent of funding availabilities.

The Web Interface

The EMPD-Viewer has been initially based on the climate proxies finder (Bolliet et al., 2016; Brockmann, 2016) which can still be seen in it the layout and design of its graphical interface (i.e. its front-end). The code base, however, has been changed entirely, updated to the latest available versions of the underlying JavaScript dependencies and extended with multiple additional tools, shown in table 4.2. The central element of the viewer is a map to show the sample locations. It also allows to intuitive access to the essential meta data of every sample through the popup of the corresponding marker on the map. The detailed meta data can also be seen in the meta data table, together with all the other samples. Another key element of the viewer are the meta data filters, that subset the data using efficient and intuitive filtering tools. This allows to search the database, or to select specific countries, climatic regimes, sample types, samples of a specific data contributor/analyst, and more.

needs implementation

Additional information on the sample is revealed through a bar diagram of the associated pollen data, as well as a diagram showing the monthly, seasonal and annual precipitation and temperature at the side, based on the WorldClim dataset, version 2 (Fick and Hijmans, 2017).

Finally, the viewer contains elements that allow scientists to contribute to the database, even without dedicated knowledge about the Github framework. The meta data editor allows to edit a sample and then submit it via the data submission form. The request is handled by the EMPD-admin webapp (see section 4.2.3) that pushes the data to the corresponding pull request on Github that is then reviewed by the core database maintainers. Another implemented element is an issue report form that allows the user to highlight erroneous sample information which is then, again through the EMPD-admin, submitted as a Github issue to the data repository.

The web app is fully integrated into the Github framework of the EMPD and loads the displayed data from the online repository. As such, it also provides a further quality control check and allows the data contributors/maintainers to review and edit new contributions before they are merged into the database.

Implementation details

The viewer itself is very light-weight and can be flexibly adapted to other database systems (see for example section 4.3). As the climate proxies finder (Bolliet et al., 2016; Brockmann, 2016), the EMPD-viewers main viewing/filtering functionality it is built upon the *dc* (Zhu and the dc.js Developers, 2019), *crossfilter* (Square, Inc. and crossfilter contributors, 2019) and *leaflet* (Agafonkin and leaflet contributors, 2019) open source JavaScript libraries. We ported the app to the *npm* package manager (npmjs.com) which enables a better and more secure monitoring of the app dependencies. This package manager is also used for an automated testing of the viewer on a CI service, prior to deployment on the official web page. Due to time constrains, the viewer is not yet fully adapted to mobile devices.

4.2.2 The EMPD2 data repository

The raw data of the EMPD2 is accessible as plain text files in the *EMPD-data* Github repository (see table 4.1). The software development framework of Github (see introductory part of section 4.2) is adopted such that issues in the data repository can highlight errors in the database, or provide room for the discussion of potential new efforts that should be considered within the community-database. Pull requests into the repository are new data contributions that can be reviewed by the maintainers before being merged into the official database.

This methodology allows a fully transparent traceback of changes made to the EMPD through version control. The online access to the raw data files through Github also allows the EMPD viewer to interface with different versions of the database (see previous section).

The EMPD-data repository additionally uses the CI services from Travis CI (travis-ci.org) for automated tests of the meta data in each sample.

4.2.3 The EMPD-admin

In addition to the standard CI services, we developed the EMPD-admin webapp. Inspired by the web management tools of the conda-forge community (conda-forge.org), this tool provides an automated handling of data contributions from within Github

TABLE 4.2: Tools in the EMPD viewer

Map interface

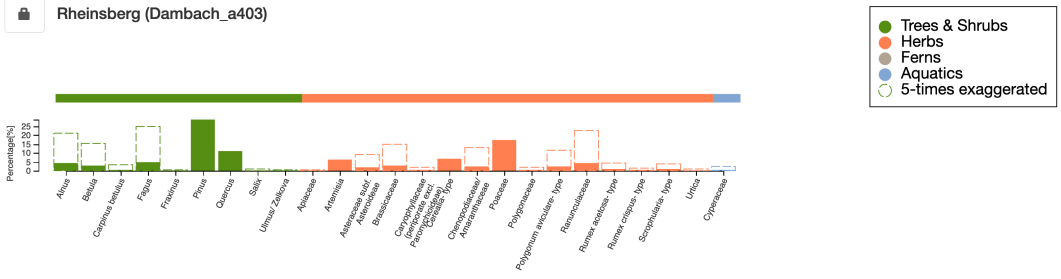
Meta data table

Metadata | Full metadata | Filters | Pollen diagram | Climate diagram | Edit Meta data | Submit changes to Github | Report issues

Download	Id	Sample name	Site name	Country	Elevation	Analysts	Reference	DOI
☐	3402	EPDcoretop_E1…	Almenara de Adaja	Spain	784	López-Sáez, ; López-Merino, …	López-Merino, L., J.A. López-S…	
☐	3403	EPDcoretop_E1…	La Molina mire	Spain	650	López-Sáez, ; López-Merino, …	López-Merino, L., A. Martínez-C…	
☐	3404	EPDcoretop_E1…	Monte Areo mire	Spain	200	López-Sáez, ; López-Merino, …	López-Merino, L., A. Martínez-C…	
☐	3405	EPDcoretop_E1…	Turbera de La P…	Spain	1648	López-Sáez, ; Abel-Schaad, D…	Abel-Schaad, D., A.M. Hernánd…	
☐	3406	EPDcoretop_E1…	Puerto de Serra…	Spain	1700	López-Sáez,	López-Merino, L., J.A. López-S…	
☐	3407	EPDcoretop_E1…	Herrenwiesser see	Germany	830	Rösch, Manfred	Rösch, M. 2012. Vegetation und…	
☐	3408	EPDcoretop_E1…	Herrenwiesser see	Germany	830	Rösch, Manfred	Rösch, M. 2012. Vegetation und…	
☐	3409	EPDcoretop_E158	La Caudelais	France	1	Voeltzel, D.	Voeltzel, D. 1987. Recherches p…	
☐	3410	EPDcoretop_E1…	Cerná Hora	Czech Republic	1206	Speranza, A.	Speranza, A., J. Hanke, B. van …	
☐	3411	EPDcoretop_E1…	Horní Lomná	Czech Republic	615	Rybnícková, Eliska	Rybnícková, E., P. Hájková, and…	
☐	3412	EPDcoretop_E1…	Hurecká Bog	Czech Republic	870	Svobodová, Helena	Svobodová, H., L. Soukupová, …	

Pollen Data

Metadata | Full metadata | Filters | Pollen diagram | Climate diagram | Edit Meta data | Submit changes to Github | Report issues



Climate Data

Metadata | Full metadata | Filters | Pollen diagram | Climate diagram | Edit Meta data | Submit changes to Github | Report issues

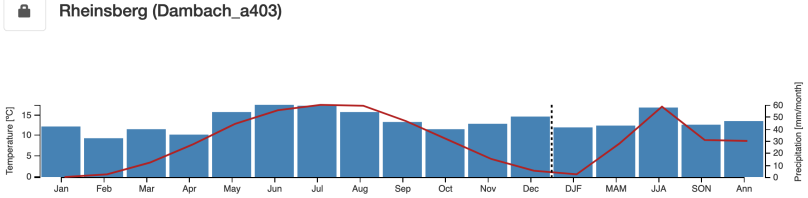
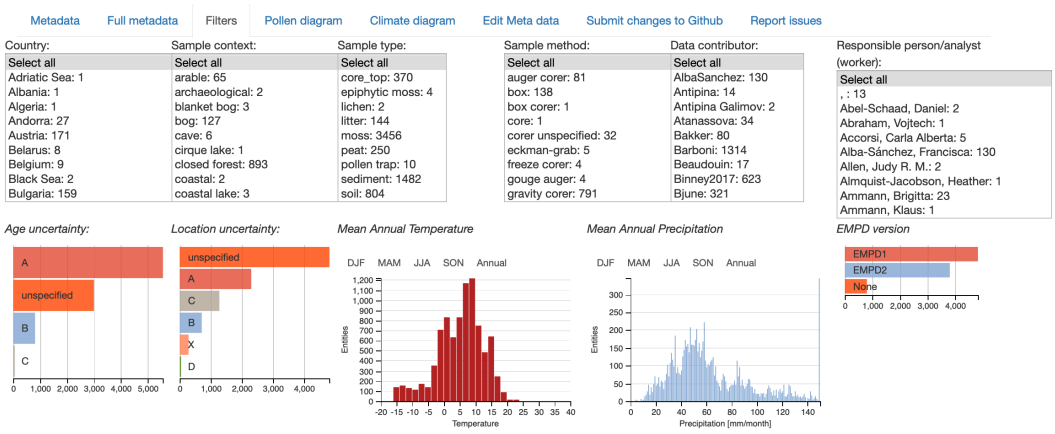
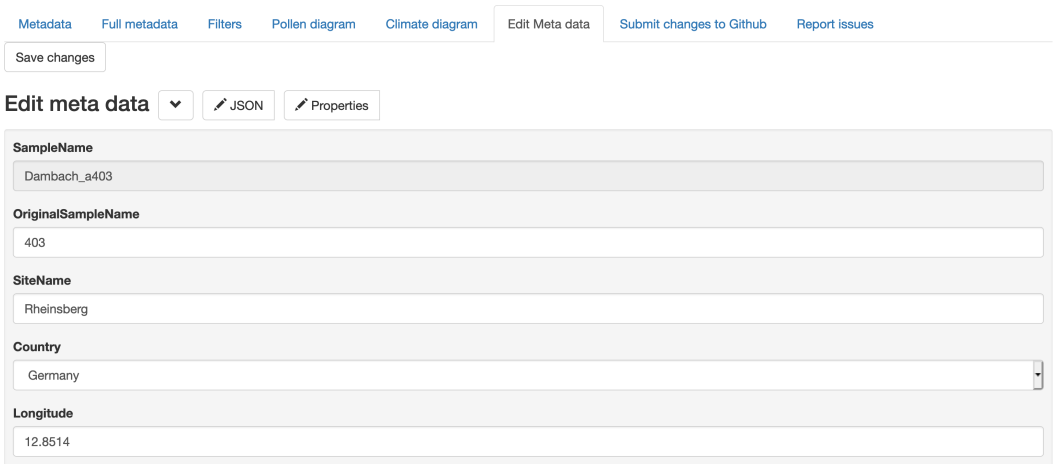


TABLE 4.2: Tools in the EMPD viewer (continued)

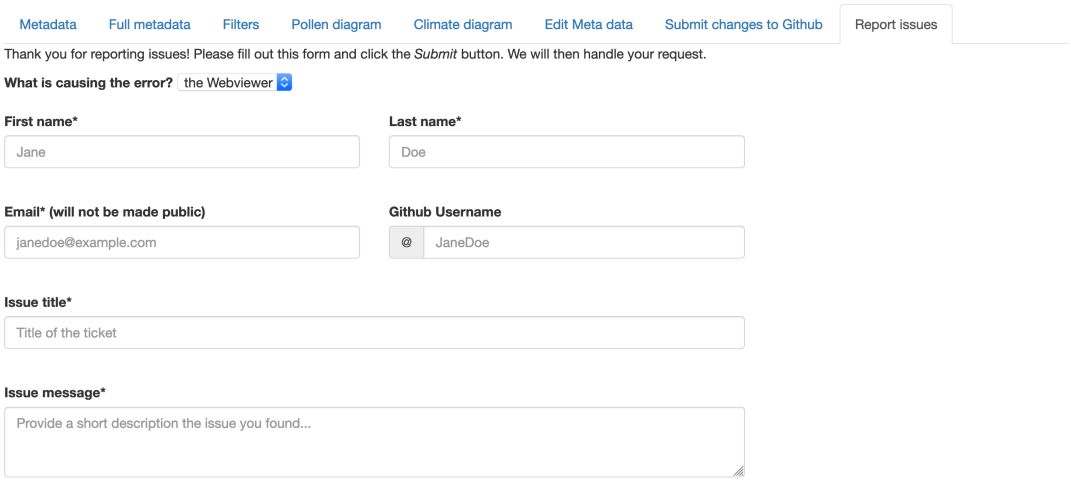
Meta data filter



Meta Data Editor



Issue submission form



Pull Requests. It acts like standard CI service and runs tests on the data contribution, every time changes have been made to the pull request.

But the main purpose of the EMPD-admin is to provide a web tool for an automated administration of the database, which is helpful for a community-project with changing maintainers. Hence, the EMPD-admin web app acts like a bot that reacts on comments within a pull request (i.e. data contribution). Maintainers and contributors can use this functionality and directly contact the bot, for instance, to subset the data, run specific tests on subsets of the data, or automatically fix certain meta data issues, such as wrong countries or missing elevation.

The bot is also integrated in the EMPD-viewer (see previous section 4.2.1). Bug reports or edited data are processed by the EMPD-admin and put online as an issue in the github repository, or it updates the corresponding data contribution.

As such, the administration of the database can be done entirely remotely, without having to install dedicated software on a local computer.

Implementation details

The EMPD-admin webapp is hosted for free at Heroku (https://www.heroku.com) at empd-admin.herokuapp.com with a software package documentation hosted at EMPD2.github.io/EMPD-admin. This, again, allows stability independent on the availability of funding. The package can, also be installed locally and used from the command-line, independent of Github and Heroku, which is sometimes helpful for very large data contributions..

The Python library is based on the tornado web framework www.tornadoweb.org, as well as pandas (McKinney, 2010), a tabular data analysis library for Python, and sqlalchemy (Bayer, 2012), a Python SQL toolkit.

4.2.4 Distribution of the tools

The EMPD is hosted within the EMPD2 Github organization (github.com/EMPD2) at github.com/EMPD2/EMPD-data. The source files of the viewer are accessible at github.com/EMPD2/EMPD-viewer, and for the EMPD-admin in the EMPD2/EMPD-admin repository (see also table 4.1).

The EMPD-data and the EMPD-admin are additionally both available as so-called Docker container image at `https://hub.docker.com/u/empd2`. These containers are lightweight, standalone, executable packages of software that include everything needed to run an application: code, runtime, system tools, system libraries and settings. As such, they extend standard software packaging systems by providing an entire operating system that contains the target application. This makes it especially useful for web applications (such as the EMPD-admin) that can, as such, operate in a well-defined and portable environment.

The EMPD-admin can, however, also be installed through the standard python package manager `pip`.

4.3 The POLNET viewer

The adaptability of the EMPD-viewer gave the motivation for an application with the POLNET database. This database, currently in development status, is a northern hemispheric, sub-tropical collection of modern and fossil pollen assemblages (Basil A. S. Davis and Kaplan, 2017; Sommer et al., 2019). The purpose of this database is to generate the source for large-scale climate reconstruction during the Holocene (past 12'000 years) that can be used for model-data comparisons. It contains about 3'300 fossil pollen sites and about 13'200 modern surface samples (see figure 4.3) and

Check this number

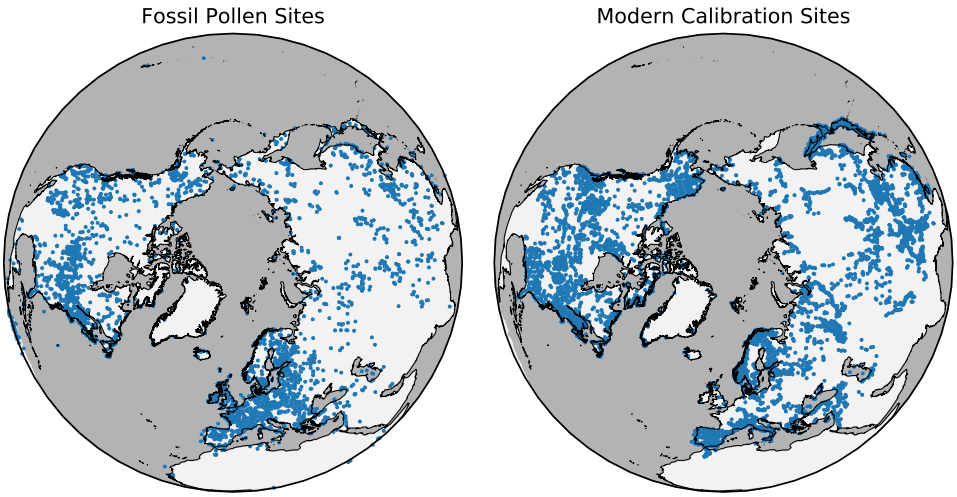


FIGURE 4.3: Maps of (left) fossil and (right) modern pollen sites in the POLNET database.

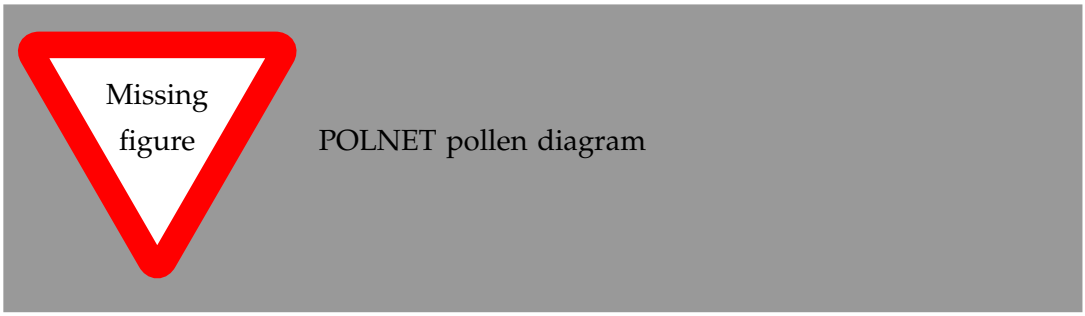


FIGURE 4.4: An exemplary pollen diagram from the POLNET viewer

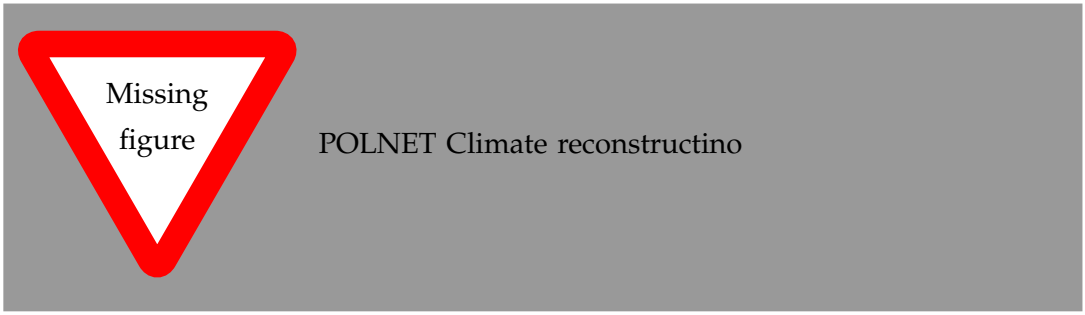


FIGURE 4.5: The climate reconstruction visualized in the POLNET viewer.

is at present the largest existing collection of fossil and modern pollen samples. The database will soon be made publicly available through a dedicated web interface, the POLNET viewer. We present it here as a sample application of the EMPD-viewer to demonstrate how this web interface can be extended and applied to other datasets, in order to make them more accessible.

Like its core application, the EMPD-viewer, the POLNET-viewer is a map-based interface with implemented meta data filters. As it is a data exploration and distribution tool only, we did not include the functionalities to edit the meta data or to submit issues. Instead we implemented new features to visualize the essential aspects of this database: fossil pollen data and climate reconstructions.

The fossil pollen data is loaded upon request from the dedicated Github repository. It is afterwards visualized in form of a stratigraphic pollen diagram, with the age of the samples on the vertical y-axis, and the pollen taxa organized as vertically aligned diagram columns (see figure 4.4).

Climate reconstructions are displayed in two different manners: Climate reconstructions are displayed in two different manners: The site-based reconstructions are visualized as line plots in a separate diagram, together with their associated uncertainties. The gridded temperature reconstruction, i.e. the final product of the database (see also chapter 6) is visualized as an overlay on the map of the web application. This results in a combined visualizations of site-based and gridded reconstructions (see figure 4.5) which enables an intuitive regional analysis of the reconstruction method.

References

Agafonkin, Vladimir and leaflet contributors (2019). *Leaflet - an open-source JavaScript library for mobile-friendly interactive maps*. URL: http://crossfilter.github.io/crossfilter/ (visited on 05/14/2019).

Bayer, Michael (2012). "SQLAlchemy". In: *The Architecture of Open Source Applications Volume II: Structure, Scale, and a Few More Fearless Hacks*. Ed. by Amy Brown and Greg Wilson. aosabook.org. URL: http://aosabook.org/en/sqlalchemy.html.

Bolliet, Timothé, Patrick Brockmann, Valérie Masson-Delmotte, Franck Bassinot, Valérie Daux, Dominique Genty, Amaelle Landais, Marlène Lavrieux, Elisabeth Michel, Pablo Ortega, Camille Risi, Didier M. Roche, Françoise Vimeux, and Claire Waelbroeck (2016). "Water and carbon stable isotope records from natural archives: a new database and interactive online platform for data browsing, visualizing and downloading". In: *Climate of the Past* 12.8, pp. 1693–1719. DOI: 10.5194/cp-12-1693-2016.

Brockmann, Patrick (2016). *ClimateProxiesFinder: dc.js + leaflet application to discover climate proxies*. Last accessed: 2019-08-30. URL: https://github.com/PBrockmann/ClimateProxiesFinder (visited on 10/06/2016).

Davis, B. A. S., M. Zanon, P. Collins, A. Mauri, J. Bakker, D. Barboni, A. Barthelmes, C. Beaudouin, A. E. Bjune, E. Bozilova, R. H. W. Bradshaw, B. A. Brayshay, S. Brewer, E. Brugiapaglia, J. Bunting, S. E. Connor, J. L. de Beaulieu, K. Edwards, A. Ejarque, P. Fall, A. Florenzano, R. Fyfe, D. Galop, M. Giardini, T. Giesecke, M. J. Grant, J. Guiot, S. Jahns, V. Jankovska, S. Juggins, M. Kahrmann, M. Karpinska-Kolaczek, P. Kolaczek, N. Kuhl, P. Kunes, E. G. Lapteva, S. A. G. Leroy, M. Leydet, J. Guiot, S. Jahns, V. Jankovska, S. Juggins, M. Kahrmann, M. Karpinska-Kolaczek, P. Kolaczek, N. Kuehl, P. Kunes, E. G. Lapteva, S. A. G. Leroy, M. Leydet, J. A. L. Saez, A. Masi, I. Matthias, F. Mazier, V. Meltsov, A. M. Mercuri, Y.

Miras, F. J. G. Mitchell, J. L. Morris, F. Naughton, A. B. Nielsen, E. Novenko, B. Odgaard, E. Ortu, M. V. Overballe-Petersen, H. S. Pardoe, S. M. Peglar, I. A. Pidek, L. Sadori, H. Seppa, E. Severova, H. Shaw, J. Swieta-Musznicka, M. Theuerkauf, S. Tonkov, S. Veski, W. O. van der Knaap, J. F. N. van Leeuwen, J. Woodbridge, M. Zimny, and J. O. Kaplan (2013). "The European Modern Pollen Database (EMPD) project". In: *Vegetation History and Archaeobotany* 22.6, pp. 521–530. ISSN: 0939-6314. DOI: 10.1007/s00334-012-0388-5. URL: http://link.springer.com/article/10.1007/s00334-012-0388-5.

Davis, Basil A. S. and Jed O. Kaplan (2017). *HORNET Holocene Climate Reconstruction for the Northern Hemisphere Extra-tropics*. SNF-Research-Plan. last accessed Jan, 30th, 2018. URL: http://p3.snf.ch/project-169598#.

Davis, Basil A. S., Manuel Chevalier, Philipp S. Sommer, et al. (in prep). "The Eurasian Modern Pollen Database (EMPD), Version 2". In: *Earth System Science Data ESSD*.

Fick, Stephen E. and Robert J. Hijmans (2017). "WorldClim 2: new 1-km spatial resolution climate surfaces for global land areas". In: *International Journal of Climatology* 37.12, pp. 4302–4315. DOI: 10.1002/joc.5086. eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/joc.5086. URL: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/joc.5086.

McKinney, Wes (2010). "Data Structures for Statistical Computing in Python". In: *Proceedings of the 9th Python in Science Conference*. Ed. by Stéfan van der Walt and Jarrod Millman, pp. 51–56.

Sommer, Philipp S., Basil A. S. Davis, Manuel Chevalier, Jian Ni, and John Tipton (2019). "The HORNET project: applying 'big data' to reconstruct the climate of the Northern Hemisphere during the Holocene". In: *20th Congress of the International Union for Quaternary Research (INQUA)*. International Union for Quaternary Research. URL: https://app.oxfordabstracts.com/events/574/program-app/submission/94623.

Square, Inc. and crossfilter contributors (2019). *Crossfilter - Fast Multidimensional Filtering for Coordinated Views*. URL: http://crossfilter.github.io/crossfilter/ (visited on 05/14/2019).

Zhu, Nick and the dc.js Developers (2019). *dc.js - Dimensional Charting Javascript Library*. URL: https://dc-js.github.io/dc.js/ (visited on 05/14/2019).

Part II Numerical Analysis of Paleoclimate Data

Chapter 5

GWGEN v1.0

A globally calibrated scheme for generating daily meteorology from monthly statistics

From

Abstract. While a wide range of earth system processes occur at daily and even sub-daily timescales, many global vegetation and other terrestrial dynamics models historically used monthly meteorological forcing, both to reduce computational demand and because global datasets were lacking. Recently, dynamic land surface modeling has moved towards resolving daily and subdaily processes, and global datasets containg daily and sub-daily meteorology have become available. These meteorological datasets, however, cover only the instrumental era of the last ca. 120 years at best, are subject to considerable uncertainty, and represent extremely large data files with associated computational costs of data input/output and file transfer. For periods before the recent past or into the future, global meteorological forcing can be provided by climate model output, but the quality of these data at high temporal resolution is low, particularly for daily precipitation frequency and amount. Here we present GWGEN, a globally applicable statistical weather generator for the temporal downscaling of monthly climatology to daily meteorology. Our weather generator is parameterized using a global meteorological database and simulates daily values of five common variables: minimum and maximum temperature, precipitation, cloud cover, and wind speed. GWGEN is lightweight, modular, and requires a minimal set of monthly mean variables as input. The weather generator may be used in a range of applications, for example, in global vegetation, crop, soil erosion, or hydrological models. While GWGEN does not currently perform spatially autocorrelated multi-point downscaling of daily weather, this additional functionality could be implemented in future versions.

5.1 Introduction

The development of the first global vegetation models in the 1970's (e.g., Helmut Lieth, 1975) brought about the demand for meteorological forcing datasets with global extent and relatively high spatial resolution, e.g., $1^\circ \times 1^\circ$. While a global weather station-based monthly climate dataset was available at this time (Walter and H Lieth, 1967), limitations in computers and storage allowed only the simplest treatment of these data. The first global simulations of the net primary productivity of the terrestrial biosphere (Helmut Lieth, 1975), thus used rasterized polygons of annual meteorological variables that had been crudely interpolated from the station-based

climatology. A decade later saw the development of better computers and more sophisticated global vegetation models (I. C. Prentice et al., 1992; I. Prentice, 1989) that recognized the need for forcing at a sub-annual timestep and development of these models was done in parallel with the first global, gridded high resolution (0.5°) monthly climatology (Leemans and Cramer, 1991). At the time, monthly meteorological data was the only feasible global data that could be produced, in terms of the raw station data available to feed the interpolation process, the processing time required to produce gridded maps, and the data storage and transfer capabilities of contemporary computer systems and networks. Global gridded monthly climate data thus became the standard for not only large-extent vegetation modeling (A. Haxeltine and I. C. Prentice, 1996; Alex Haxeltine et al., 1996; J. O. Kaplan et al., 2003; Kucharik et al., 2000; Woodward et al., 1995), but also for a wide range of studies on biodiversity and species distribution (e.g., Elith et al., 2006), vegetation trace gas emissions (e.g., Guenther et al., 1995), and even the geographic distribution of human diseases (e.g., Bhatt et al., 2013)

Over subsequent years, the global gridded monthly climate datasets were improved (New et al., 1999, 2002), developed with very high spatial resolution (Hijmans et al., 2005), and expanded beyond climatological mean climate to cover continuous timeseries over decades (Harris et al., 2014; Mitchell and P. D. Jones, 2005; New et al., 2000). The latter was an essential requirement for forcing dynamic global vegetation models (DGVMs) (e.g., Sitch et al., 2003). However, despite increasing quality, spatial resolution, and temporal extent in these datasets, the basic time step remained monthly, partly for legacy reasons — models had been developed in an earlier era subject to computational limitations and therefore used a monthly timestep for efficiency even if this was no longer strictly a constraint — and partly because of the challenge in developing a global, high-resolution climate dataset with a daily or shorter timestep still presented a major data management challenge.

On the other hand, there was increasing awareness that accurate simulation of many earth surface processes required representation of processes at a shorter-than-monthly timestep. Global simulation of surface hydrology (Gerten et al., 2004), crop growth (Bondeau et al., 2007), or biogeophysical processes (Krinner et al., 2005) needed sub-monthly forcing to produce reliable results. To address this need for better forcing data, two main approaches were taken: either monthly climate data were downscaled online using a stochastic weather generator (e.g., Pfeiffer et al., 2013), or a sub-daily, high-resolution, gridded climate timeseries was generated directly by merging high-temporal-resolution reanalysis data (e.g., NCEP, 6h, 2.5°) with high-spatial-resolution monthly climate data (e.g., CRU, 0.5°). The latter process resulted in the CRUNCEP dataset (Viovy and Ciais, 2016; Wei et al., 2014), which, while global, is large even by modern standards (ca. 350 Gb), is not available at spatial resolution greater than 0.5°, and covers only the period 1901-2014.

Forcing data for global vegetation and other models with shorter than monthly resolution at higher spatial resolutions than 0.5°, or for any other period than the last ca. 120 years, e.g., for the future or the more distant past, may therefore only be available through downscaling techniques. One approach to overcome the limitations of currently available datasets could be to use GCM output directly, however, most GCM output currently available does not have greater than 0.5° spatial resolution, with the current generation of GCMs typically approaching ca. $1° \times 1°$. Furthermore, there is a general observation that daily meteorology produced by GCMs is not realistic, particularly for precipitation (Dai, 2006; Stephens et al., 2010; Sun et al., 2006). An alternative approach is, therefore, to perform temporal downscaling on monthly meteorological data using a statistical weather generator.

Statistical weather generators were first developed primarily for crop and hydrological modeling at the field to catchment scale (Richardson, 1981; Woolhiser and Pegram, 1979; Woolhiser and Roldan, 1982). The weather generator was parameterized using daily meteorological observations at one or more weather stations close to the area of interest, although some attempts were made to generalize the parameterization over larger, sub-continental regions (e.g., D. S. Wilks, 1998, 1999b; Woolhiser and Roldán, 1986). Locally parameterized weather generators have been applied to a very wide range of studies (D. S. Wilks and Wilby, 1999; Daniel S. Wilks, 2010), and enhanced to include additional meteorological variables beyond the original precipitation, temperature, and solar radiation (e.g., Parlange and Katz, 2000). Applications of a weather generator at continental to global scales was still limited, however, because of the need to perform local parameterization.

The need to simulate daily meteorology in regions of the world with short, unreliable, or unavailable daily meteorological timeseries brought about the realization that certain features of weather generator parameterization might be generalized across a range of climates (S. Geng et al., 1986; Shu Geng and Auburn, 1987). This ultimately led to the development of globally applicable weather generators (Friend, 1998), and their incorporation in DGVMs (Bondeau et al., 2007; Gerten et al., 2004; Pfeiffer et al., 2013). The original global parameterization (S. Geng et al., 1986) of these weather generators was, however, limited to seven weather stations, mostly in the temperate latitudes. Friend, 1998 does not publish the parameters used in his global weather generator, but we assume these were the same as the original Shu Geng and Auburn, 1987 and S. Geng et al., 1986 models. Given the availability of 1) large datasets of daily meteorology, and 2) computers powerful enough to process these data, we therefore decided that it would be valuable to revisit these parameterizations, perform a systematic and quantitative evaluation of the resulting downscaled meteorology, and potentially improve our ability to perform monthly-to-daily downscaling of common meteorological variables with a single, globally applicable parameterization.

In the following sections we describe Global-WGEN (GWGEN), a weather generator parameterized using more than 50 million daily weather observations from all continents and latitudes. We demonstrate how updated schemes for simulating precipitation occurrence and amount, and for bias correcting wind speed, further improve the quality of the model simulations. We perform an extensive model evaluation and parameter uncertainty analysis in order to settle on a parameter set that provides the most accurate, globally applicable results. We comment on the limitations of the model and priorities for future research. GWGEN is an open-source, stand-alone model that may be incorporated into any number of models designed to work at global scale, including, e.g., vegetation, hydrology, climatology, and animal distribution models.

5.2 Model description

GWGEN requires the following six monthly summary values as input: 1) total monthly precipitation, 2) the number of days in the month with measurable precipitation (i.e., wet days), 3-4) monthly mean daily minimum and maximum temperature, 5) mean cloud fraction, and 6) wind speed. The model outputs are the same variables at daily resolution. This section summarizes the basic workflow in the model which is also shown schematically in Figure 5.1 and Algorithm 1.

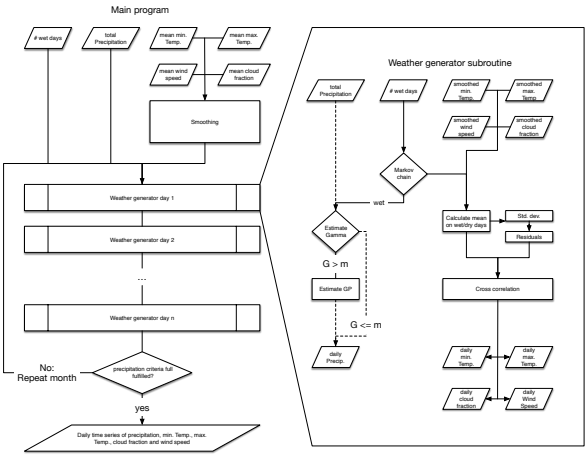


FIGURE 5.1: Schematic workflow of GWGEN. After smoothing the monthly input, the Markov Chain is used to decide, whether it is a dry or a wet day. If it is a wet day, we draw a random number from the Gamma-GP distribution. Furthermore, the other means of the variables ($\bar{T}_{\min/\max}, \bar{c}, \bar{w}$) are adjusted and their daily values are calculated using the estimated standard deviations and residuals. The wind speed furthermore undergoes a square root transformation before applying the cross correlation and in the end is corrected using the bias correction. A quality check in the end restricts our model to be within a 5% range of the observed total precipitation and to replicate the number of wet days from the input.

The first approximation of the daily variables comes from smoothing the monthly time series using a mean-preserving algorithm (Rymes and Myers, 2001).

For precipitation we then first use the Markov Chain approach (section 5.3.2) to decide the wet/dry state of the day. If it is a wet day, we calculate the gamma parameters using the equations (5.7) and (5.8). The resulting distribution allows us to draw a random number, the precipitation amount of the currently simulated day. If we are above the threshold μ, we draw a second random number from the GP distribution parameterized via equation (5.9) and the chosen GP shape.

The next step modifies the means of temperature, wind speed and cloud fraction depending on the wet/dry state of the day (lines 11 and 15 in algorithm 1). After that, we use the cross-correlation approach described in Richardson, 1981 (lines 18 - 20 and Equation 5.3.2) and calculate the daily values of these variables. Finally we use the quantile-based bias correction described in section 5.3.4 to correct the simulated wind speed.

We restrict the weather generator to reproduce the exact number of wet days (± 1) as the input and to be within a 5% range of the total monthly precipitation (with a maximum allowed deviation of 0.5 mm). If the program cannot produce these results, the procedure described above is repeated (see line 4).

5.3 Model development

GWGEN is based on the WGEN weather generator (ibid.), using the method of defining the model parameters based on monthly summaries described by S. Geng et al., 1986 and Shu Geng and Auburn, 1987. GWGEN diverges from the original WGEN by using a hybrid-order Markov chain to simulate precipitation occurrence

Algorithm 1 Basic workflow of GWGEN

Require: monthly precipitation P_{in} [mm], cloud cover fraction c_{in}, minimum ($T_{\text{min,in}}$ [°C]) and maximum ($T_{\text{max,in}}$ [°C]) temperature, wind speed w_{in} [m/s], number of wet days n_{in}

Output: daily P_i [mm/d], c_i, T_i [°C], w_i [m/s] and the wet/dry state $s_i \in \{0, 1\}$

for month m in $input$ **do**
  smooth the monthly data using Rymes and Myers, 2001
  Set $j = 0, \chi = 0$
  while $j \equiv 0$ or $\left|\sum_{d_i \in m} P_i - P_{\text{in}}\right| > \min\left(5\% \cdot P_{\text{in}}, 0.5mm\right)$ or $|n_{\text{sim}} - n_{\text{in}}| > 1$ **do**
    for day d_i in m **do**
      Calculate p_{11}, p_{101}, p_{001} after equations (5.1) - (5.3) using n {Precipitation occurence after D. S. Wilks, 1999a}
      Use the Markov chain to determine whether d_i is wet ($s_i = 1$) or dry ($s_i = 0$)
      if $s_i = 1$ **then**
        Calculate θ, α and σ via eq. (5.7)-(5.9) {Precipitation amount after Neykov et al., 2014}
        Draw a random number P_i from the Gamma-GP distribution, eq. (5.6)
        Set $T_{\text{min},i} = T_{\text{min,wet}}, T_{\text{max},i} = T_{\text{max,wet}}, c_i = c_{\text{wet}}, w_i = w_{\text{wet}}$ from eq. (5.10) and (5.12) and tables 5.1, 5.3
        Set $\sigma_{T_{\text{min}},i} = \sigma_{T_{\text{min}},\text{wet}}, \sigma_{T_{\text{max}},i} = \sigma_{T_{\text{max}},\text{wet}}, \sigma_{w,i} = \sigma_{w,\text{wet}}, \sigma_{c,i} = \sigma_{c,\text{wet}}$ from eq. (5.11), (5.13) and (5.14) and tables 5.1, 5.2, 5.3
      else
        Set $P_i = 0\,\text{mm/d}$
        Set $T_{\text{min},i} = T_{\text{min,dry}}, T_{\text{max},i} = T_{\text{max,dry}}, c_i = c_{\text{dry}}, w_i = w_{\text{dry}}$ from eq. (5.10) and (5.12) and tables 5.1, 5.3
        Set $\sigma_{T_{\text{min}},i} = \sigma_{T_{\text{min}},\text{dry}}, \sigma_{T_{\text{max}},i} = \sigma_{T_{\text{max}},\text{dry}}, \sigma_{w,i} = \sigma_{w,\text{dry}}, \sigma_{c,i} = \sigma_{c,\text{dry}}$ from eq. (5.11), (5.13) and (5.14) and tables 5.1, 5.2, 5.3
      end if
      Draw 4 normally distributed random numbers $\epsilon \in \mathbb{R}^4$ {Cross correlation after Richardson, 1981}
      Set the residuals $\chi_i = \begin{pmatrix} \chi_{T_{\text{min}}} & \chi_{T_{\text{max}}} & \chi_c & \chi_w \end{pmatrix} = A\chi_{i-1} + B\epsilon \in \mathbb{R}^4$ with A and B from eq. (5.17)
      Calculate daily variables via
      $T_{\text{min},i} = \chi_{T_{\text{min}}} \cdot \sigma_{T_{\text{min}},i} + T_{\text{min},i} \qquad c_i = \chi_c \cdot \sigma_{c,i} + c_i$
      $T_{\text{max},i} = \chi_{T_{\text{max}}} \cdot \sigma_{T_{\text{max}},i} + T_{\text{max},i} \qquad w_i = \left(\chi_w \cdot \sqrt{\sigma_{w,i}} + \sqrt{w_i}\right)^2$
      Apply bias correction w (eq. (5.23))
      $j = j + 1$
    end for
  end while
end for

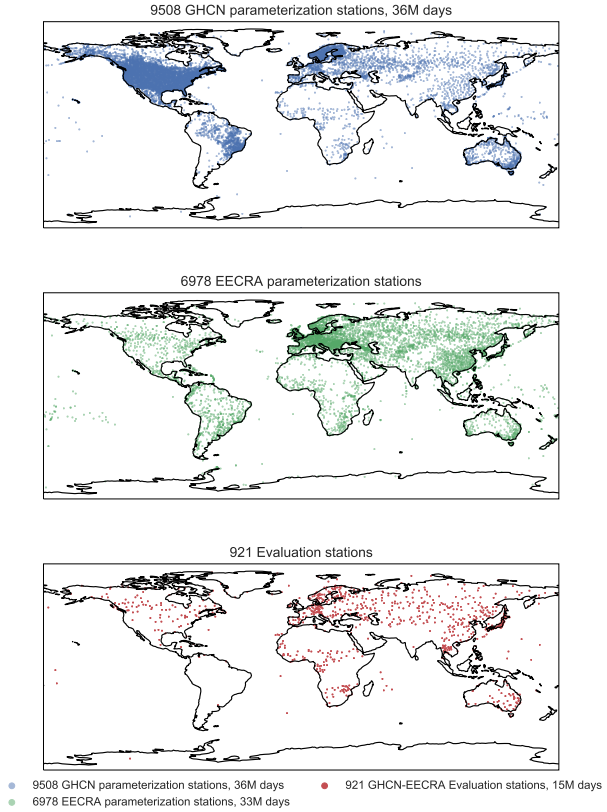


FIGURE 5.2: Weather stations used for parameterization and evaluation of the weather generator. The uppermost panel shows the locations of the stations used for parameterizing precipitation and temperature, the middle panel shows the stations for cloud fraction and wind speed, as well as for calculating the cross correlations between temperature, cloud fraction, and wind speed. The lower plot shows the location of the stations used to evaluate the model, which were excluded from the parameterization stations.

(D. S. Wilks, 1999a), and a hybrid Gamma-GP distribution (Furrer and Katz, 2008; Neykov et al., 2014) to estimate precipitation amount. Temperature, cloud cover, and wind speed are calculated following (Richardson, 1981), using cross correlation and depending on the wet/dry-state of the day. We further add a quantile-based bias correction for wind speed and minimum temperature, which improves the simulation results significantly.

In the following subsections, we first describe the global weather station database used to develop and evaluate the model, then describe the underlying relationships that we use to define GWGEN's parameters.

5.3.1 Development of a global weather station database

To parameterize GWGEN, we assembled a global dataset of daily meteorological observations. Precipitation and minimum and maximum daily temperature come from the daily Global Historical Climatology Network (GHCN-Daily) database (Menne et al., 2012a,b). The GHCN-Daily consists of observations collected at ca. 100'000 weather stations on all continents and many oceanic islands. As the GHCN-Daily

stations are highly concentrated in some parts of the world, particularly in the conterminous United States, we selected stations for our study using a geographic anti-aliasing filter to avoid an especially strong geographic bias in the generation of the model parameters. Dividing the world up into a 0.5° grid, we selected the single station with the longest record in each cell, if one was present. While the GHCN-Daily units for precipitation have a nominal precision of 0.1 mm, several of the stations in the United States reported precipitation in fractions of an inch, which were later converted to mm. To ensure uniform precision across all of our calibration stations — this was particularly important when generating the probability density functions for precipitation amount — we selected only those GHCN-Daily stations where all precipitation amounts between 0.1 and 1.0 mm d^{-1} were reported in the record. This resulted in 9508 stations covering all continents, although the distribution is strongly heterogenous, with the majority of the stations in North America, despite our geographic filter (Figure 5.2, top panel). For cloud cover, windspeed, and to calculate cross-correlations between temperature, cloud cover, and windspeed, we used the Extended Edited Cloud Report Archive (EECRA) database (Hahn and Warren, 1999). The geographic distribution of the 6978 EECRA stations we selected is different than the GHCN-Daily, with more stations in Europe (Figure 5.2, middle panel), but overall a relatively similar number of stations were used from both datasets. For the observations from both GHCN-Daily and EECRA, we made one additional filtering step, selecting only complete months, i.e., months with no days having missing observations, for further processing. In total, our database of daily meteorological observations used in the model parameterization contains ca. 69 million individual records.

Finally, we reserved some weather station records for model evaluation that were not used for model parameterization. These were individual stations, or two stations separated by a maximum distance of 1 km, where all of the daily meteorological variables that GWGEN simulates (P, T_{min}, T_{max}, c, w) were recorded on the same dates in the EECRA database. This merged selection from EECRA and GHCN resulted in a set of 921 stations representing ca. 15 million daily records, with observations on all continents, although the geographic distribution is once again highly heterogenous, with a particularly high density of stations in Japan and Germany (Figure 5.2, bottom panel).

5.3.2 Parameterization

Precipitation occurrence

Following S. Geng et al., 1986, we expect to find a good relationship between the fraction of days in a month with measurable precipitation and the probability that any given day will be wet. Following D. S. Wilks, 1999a we use a hybrid-order model that retains first-order Markov dependence for wet spells but allows second-order dependence for dry sequences; this hybrid-order scheme has been shown to be a good compromise between performance and simplicity. To parameterize the precipitation occurrence part of the model, we thus calculated transition probabilities for a wet day being followed by a wet day (p_{11}), for a wet day being followed by a dry day being followed by a wet day (p_{101}) and for two dry days being followed by a wet day (p_{001}). We perform this analysis on a station and month-wise basis, i.e., we first extract each of the (complete) Januaries, Februaries, etc. for a given station, and then merge all of the Januaries (Februaries, Marches, etc...) for this station into a single series representing each month. Merging months over several years is

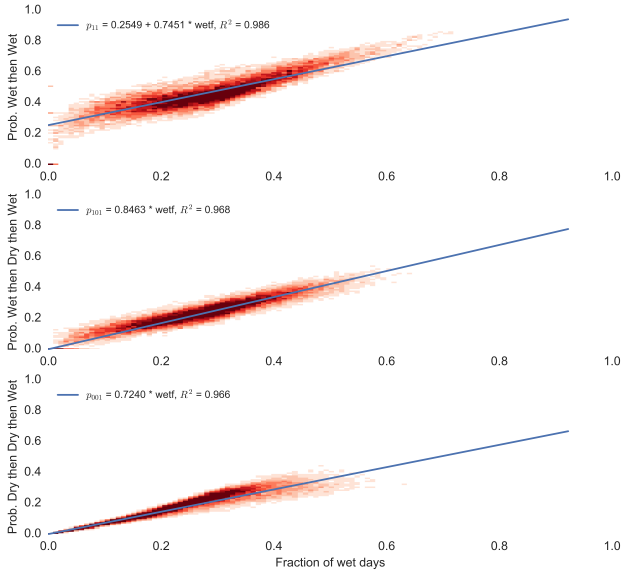


FIGURE 5.3: Transition probabilities vs. wet fraction. The red density plot in the background shows the density of the observations, and the blue lines are the linear regression line of the probability against the wet fraction. The fit for the p_{11} transition probability was forced to the point $(1, 1)$, the others were forced to $(0, 0)$. The underlying data for the fits correspond to the means of the the multi-year series for each month for each station.

particularly important for stations that have relatively little precipitation in a given month; for example, it could take several years of observations to observe a single (p_{101}) event. The final transition probabilities were then regressed against the fraction of days in the month with precipitation, which show the characteristic linear relationship described by S. Geng et al., 1986 (Figure 5.3).

Because the transition probabilities (p_{001}) and (p_{101}) must be zero by definition when the fraction of wet days (f_{wet}) is zero, i.e., a completely dry month, we force the linear regression between these quantities to pass through the origin. Likewise, we require the regression line for (p_{11}) to equal 1 when f_{wet} is 1. One has to note, however, that this methodology artificially increases the R^2 coefficient for the fit because we fix the intercept (see for example Gordon, 1981).

The analysis results in the the following relationships:

$$p_{11} = 0.2549 + 0.7451 \cdot f_{\text{wet}} \tag{5.1}$$

$$p_{101} = 0.8463 \cdot f_{\text{wet}} \tag{5.2}$$

$$p_{001} = 0.7240 \cdot f_{\text{wet}}. \tag{5.3}$$

In the weather generator (see line 6 in algorithm 1) we determine if any given day will have precipitation by calculating the appropriate probability density function selected from equations (5.1)-(5.3) on the basis of the precipitation state of the previous day (or two). Comparing the calculated probability from the selected equation with a random number $u \in [0, 1]$, a precipitation day is simulated if u is greater than its corresponding probability.

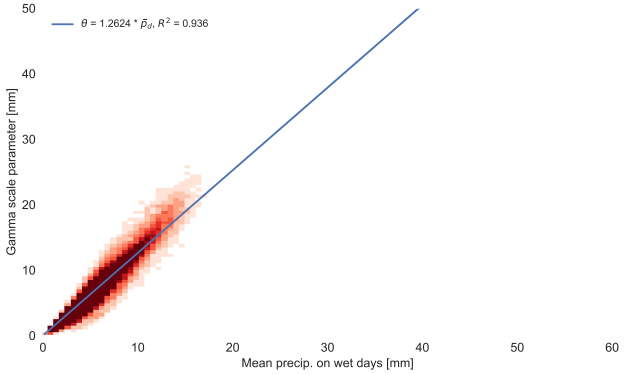


FIGURE 5.4: Mean precipitation - Gamma scale relationship. The blue line represents the best fit line of the mean precipitation on wet days to the estimated gamma scale parameter of the corresponding distribution. Each data point corresponds to one multi-year series of one month for one station.

Precipitation amount

Following the original WGEN (Richardson, 1981), GWGEN disaggregates precipitation amount using a statistical distribution. A number of different probability density functions have been used to estimate precipitation amount in weather generators including, e.g., single exponential or mixed exponential, one or two parameter gamma, or Weibull distribution (D. S. Wilks and Wilby, 1999). The strong relationship between the gamma scale parameter and the mean precipitation on wet days noted by S. Geng et al., 1986 makes generation of precipitation amounts with only monthly input data feasible. It is based upon the fact that the expected value of a gamma random variable equals the product of its two parameters. i.e $E(\Gamma) = \alpha\theta$. The gamma distribution, however, shows poor performance in simulating high-precipitation events consistent with observations. Furrer and Katz, 2008 and Neykov et al., 2014 suggest that a hybrid probability density function, based on both gamma and the generalized pareto (GP) distribution, has superior accuracy in simulating extreme precipitation events when compared to gamma alone. Because of its superior accuracy and ease of implementation, we therefore adopt the hybrid gamma-GP distribution for simulating precipitation amount in GWGEN.

The probability density function (pdf) of the gamma distribution is defined as

$$f(x) = \begin{cases} \frac{x^{\alpha-1}e^{-\frac{x}{\theta}}}{\theta^{\alpha}\Gamma(\alpha)} & \text{for } x > 0 \\ 0 & \text{for } x = 0 \end{cases} \tag{5.4}$$

where $\alpha > 0$ is the shape, and $\theta > 0$ the scale parameter. The pdf of the generalized pareto (GP) distribution is defined via

$$g(x) = \begin{cases} \frac{1}{\sigma}\left(1 + \frac{\xi(x-\mu)}{\sigma}\right)^{-\frac{1}{\xi}-1} & \text{for } \xi \neq 0 \\ \frac{1}{\sigma}e^{-\frac{x-\mu}{\sigma}} & \text{for } \xi = 0 \end{cases} \tag{5.5}$$

with $\sigma > 0$ being the scale parameter and $\xi \in \mathbb{R}$ the shape parameter. μ is the location parameter.

Following Furrer and Katz, 2008, we define the hybrid gamma-GP pdf as

$$h(x) = \begin{cases} f(x) & \text{for } x \leq \mu \\ (1 - F(\mu))\, g(x) & \text{for } x > \mu \end{cases}, \tag{5.6}$$

where $F(\mu)$ describes the cumulative gamma distribution function at the threshold μ. In our weather generator however, we first draw a random number from the gamma distribution and, if we are above the threshold, we draw another random number from the GP distribution. Thus, the frequency of precipitation events larger than μ is determined by the gamma distribution, but the actual amount of precipitation simulated when above the threshold μ is determined by the GP distribution (ibid.).

To determine the parameters of the hybrid distribution for precipitation, we started with the simple strategy by S. Geng et al., 1986. As above when calculating the Markov chain parameters, we created multi-year series for each of the parameterization stations for each month and extracted the days with precipitation. If a series contained more than 100 entries, we fit a gamma distribution using maximum likelihood to it in order to estimated the α and θ parameters.

Following ibid., we then fit a regression line of the gamma scale parameter against the mean precipitation on wet days $\bar{p}_d$ (see figure 5.4) and found the relationship

$$\theta = 1.262\, \bar{p}_d. \tag{5.7}$$

As proposed by ibid., we use this relationship in our model to estimate the scale parameter of the distribution. Using this approach, the gamma shape parameter α is a constant, given via

$$\alpha = \frac{\bar{p}_d}{\theta} = \frac{1}{1.262}. \tag{5.8}$$

The GP scale parameter σ on the other hand is calculated during the simulation following Neykov et al., 2014 via

$$\sigma = \frac{1 - F(\mu)}{f(\mu)}. \tag{5.9}$$

The other parameters of the GP distribution are obtained through a sensitivity analysis described in section 5.3.5.

Temperature

Following the standard WGEN methodology (Richardson, 1981) and S. Geng et al., 1986, daily temperature is determined through 2 processes: First, the wet/dry state of the day, and second the cross correlation (Equation 5.3.2).

In the weather generator, we know from the Markov chain (section 5.3.2), whether the current simulated day is a wet or dry day. Based upon the simple linear relationships

$$\begin{aligned} \bar{x}_{\text{wet}} &= c_{0,x,\text{wet}} + c_{1,x,\text{wet}} \cdot \bar{x} \\ \bar{x}_{\text{dry}} &= c_{0,x,\text{dry}} + c_{1,x,\text{dry}} \cdot \bar{x} \end{aligned} \tag{5.10}$$

we adjust the monthly mean $\bar{x}$ of the variable $x \in \{T_{\text{min}}, T_{\text{max}}\}$.

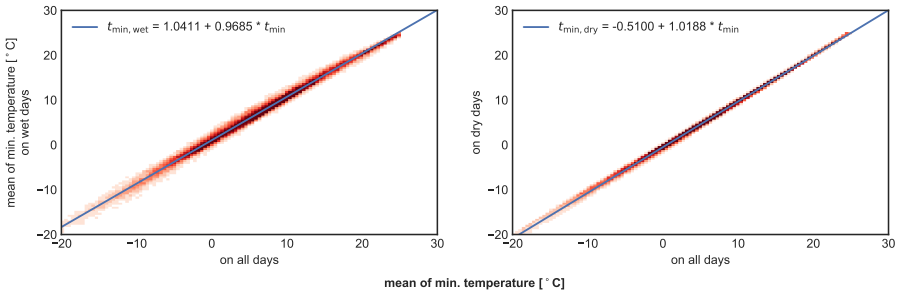


FIGURE 5.5: Correlation of minimum temperature on wet and dry days to the monthly mean. The y-axes show the mean minimum temperature on wet or dry days respectively, the blue line corresponds to the best fit line. Parameters of the fits are also shown in table 5.1.

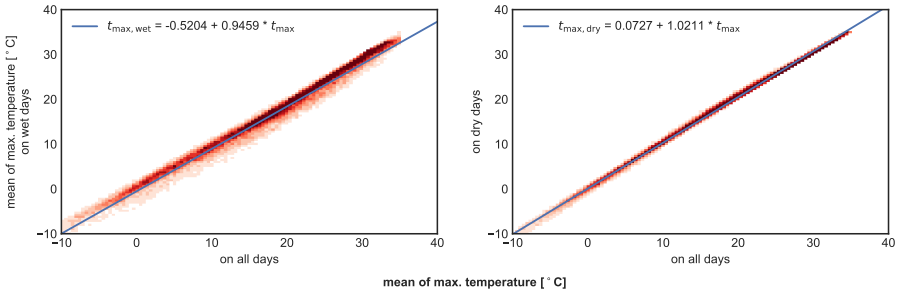


FIGURE 5.6: Correlation of maximum temperature on wet and dry days to the monthly mean. The y-axes show the mean maximum temperature on wet or dry days respectively, the blue line corresponds to the best fit line. Parameters of the fits are also shown in table 5.1.

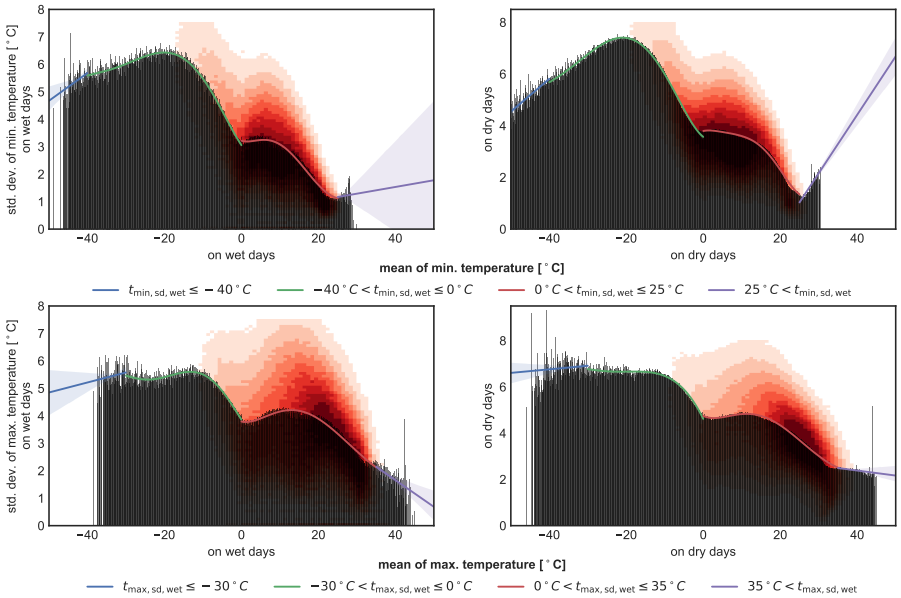


FIGURE 5.7: Correlation of standard deviation of the minimum and maximum temperature on wet and dry days to the monthly mean. The y-axes show the standard deviation, the x-axes the mean on wet or dry days respectively. The bars have a width of $0.1^{\circ}C$ (the data accuracy) and indicate the mean standard deviation for a given mean minimum temperature in one month. The lines are fitted to these bars, where the green and red polynomials of order 5 are the use all the data below or above $0^{\circ}C$ respectively and the blue and violet lines are a linear extrapolation of the data below $-40^{\circ}C$ (or $-30^{\circ}C$ for T_{max}) or above $25^{\circ}C$ (or $35^{\circ}C$) respectively. The red density plot in the background indicates the spread of the data. The bars and the density plot are based on the single month for each station (i.e. not the multi-year monthly series as for, e.g. mean temperature (figure 5.5 and 5.6)). Parameters of the fits are also shown in table 5.1.

TABLE 5.1: Fit results of temperature correlation for wet and dry days for figures 5.5, 5.6, 5.10 and 5.11. The coefficients c_0 to c_3 correspond to the coefficients used in equations (5.10) and (5.14).

plot	variable	R^2	c_0	c_1	c_2	c_3
5.6	$T_{\text{max,dry}}$	0.9969	0.0727	1.0211	0	0
5.6	$T_{\text{max,wet}}$	0.9752	-0.5204	0.9459	0	0
5.5	$T_{\text{min,dry}}$	0.9972	-0.5100	1.0188	0	0
5.5	$T_{\text{min,wet}}$	0.9840	1.0411	0.9685	0	0
5.11	$w_{\text{sd,dry}}$	0.4243	0	1.0860	-0.2407	0.0222
5.11	$w_{\text{sd,wet}}$	0.5003	0	0.8184	-0.1263	0.0093
5.10	w_{dry}	0.9930	0	0.9437	0	0
5.10	w_{wet}	0.9723	0	1.0937	0	0

To estimate the values of the parameters c_0 and c_1 in the above equations, we follow the same procedure as for the parameters of the Markov chain (section 5.3.2). We extracted the complete months for T_{min} and T_{max} from the GHCN-Daily dataset and created a multi-year series for each month and station. We then regressed the mean on wet and dry days separated against the overall mean of each month (Figures 5.5 and 5.6). Through this procedure, we estimate the parameters necessary for equations (5.10) (see table 5.1).

To estimate residual noise, we also need an estimate of the standard deviation of the variable (see Equation 5.3.2). Figure 5.7 shows the correlation between standard deviation on wet and dry days and the corresponding mean. The means of the standard deviations (black bars in figure 5.7) indicate a strong but non-linear relationship between the standard deviation and the corresponding mean. The correlation changes particularly at $0^\circ C$. We therefore use two different polynomials of order 5 for the values below and above the freezing point. Furthermore, to account for the sparse data below $-40^\circ C$ and above $25^\circ C$ for minimum temperature (or $-30^\circ C$ and $35^\circ C$ for maximum temperature), we use an extrapolation for the extremes as indicated by the blue and violet lines in figure 5.7. The formulae for the standard deviations σ of minimum and maximum temperature are therefore a combination of 4 polynomials:

$$\sigma_{T_{\text{min}},\text{wet/dry}} = \begin{cases} p_1(\bar{T}_{\text{min,wet/dry}}), & \text{for } \bar{T}_{\text{min,wet/dry}} \leq -40^\circ C \\ p_5(\bar{T}_{\text{min,wet/dry}}), & \text{for } -40^\circ C < \bar{T}_{\text{min,wet/dry}} \leq 0^\circ C \\ p_5(\bar{T}_{\text{min,wet/dry}}), & \text{for } 0^\circ C < \bar{T}_{\text{min,wet/dry}} \leq 25^\circ C \\ p_1(\bar{T}_{\text{min,wet/dry}}), & \text{for } 25^\circ C < \bar{T}_{\text{min,wet/dry}} \end{cases}$$

$$\sigma_{T_{\text{max}},\text{wet/dry}} = \begin{cases} p_1(\bar{T}_{\text{max,wet/dry}}), & \text{for } \bar{T}_{\text{max,wet/dry}} \leq -30^\circ C \\ p_5(\bar{T}_{\text{max,wet/dry}}), & \text{for } -30^\circ C < \bar{T}_{\text{max,wet/dry}} \leq 0^\circ C \\ p_5(\bar{T}_{\text{max,wet/dry}}), & \text{for } 0^\circ C < \bar{T}_{\text{max,wet/dry}} \leq 35^\circ C \\ p_1(\bar{T}_{\text{max,wet/dry}}), & \text{for } 35^\circ C < \bar{T}_{\text{max,wet/dry}} \end{cases} . \quad (5.11)$$

p_1 in eq. (5.11) denotes a polynomial of order 1, p_5 a polynomial of order 5. The coefficients of the different polynomials are shown in table 5.2.

These coefficients are based on the means of the standard deviation (black bars in figure 5.7). We chose this procedure to give the same weight to all temperatures.

TABLE 5.2: Fit results of the correlation of temperature standard deviation with the corresponding mean on wet/dry days for figure 5.7. The underlying equations are shown in equation (5.11).

variable	interval	R^2	c_0	c_1	c_2	c_3	c_4	c_5
$T_{max,sd,dry}$	(-∞, -30]	0.0125	7.3746	0.0154	0	0	0	0
	(-30, 0.0]	0.6721	4.6170	-0.3387	-0.0188	-0.0003	0.000003	0.0000001
	(0.0, 35]	0.9744	4.7455	-0.0761	0.0189	-0.0013	0.00003	-0.0000002
	(35, ∞)	0.0390	3.2554	-0.0218	0	0	0	0
$T_{max,sd,wet}$	(-∞, -30]	0.0366	6.6720	0.0364	0	0	0	0
	(-30, 0.0]	0.7362	3.8601	-0.2186	0.0039	0.0015	0.00006	0.0000007
	(0.0, 35]	0.9508	3.7919	-0.0313	0.0161	-0.0012	0.00003	-0.0000002
	(35, ∞)	0.2530	5.5529	-0.0973	0	0	0	0
$T_{min,sd,dry}$	(-∞, -40]	0.6006	10.8990	0.1271	0	0	0	0
	(-40, 0.0]	0.9509	3.5676	-0.1154	0.0282	0.0020	0.00004	0.0000003
	(0.0, 25]	0.9825	3.7941	0.0330	-0.0150	0.0019	-0.0001	0.000002
	(25, ∞)	0.7784	-4.6194	0.2261	0	0	0	0
$T_{min,sd,wet}$	(-∞, -40]	0.1661	9.7272	0.1011	0	0	0	0
	(-40, 0.0]	0.9285	3.0550	-0.2116	0.0137	0.0014	0.00004	0.0000003
	(0.0, 25]	0.9633	3.2187	-0.0451	0.0209	-0.0026	0.00010	-0.000001
	(25, ∞)	0.0089	0.5571	0.0244	0	0	0	0

Otherwise the fit would be dominated by the temperature values around the freezing points.

Cloud fraction

Monthly mean cloud fraction is disaggregated, as for temperature, using the standard WGEN procedure of adding statistical noise to a wet- or dry-day mean and accounting for cross-correlation among the different weather variables. For the parameterization of the cloud fraction equations, we used the EECRA dataset. The original dataset contains eight measurements per day of the total cloud cover in

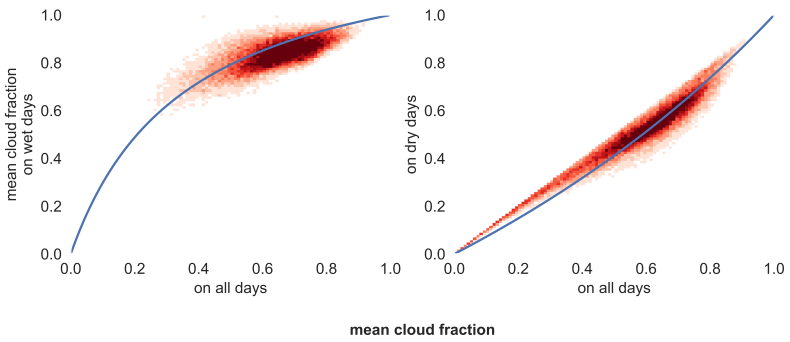


FIGURE 5.8: Correlation of cloud fraction on wet and dry days to the monthly mean.The y-axes show the mean cloud fraction on wet or dry days respectively, the blue line corresponds to the best fit line. Parameters of the fits are also shown in table 5.3.

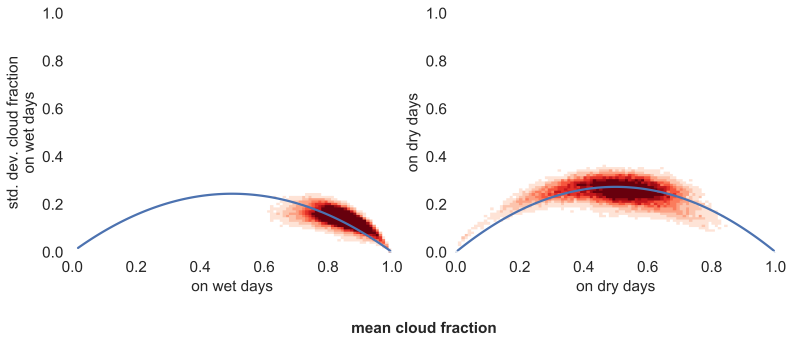


FIGURE 5.9: Correlation of standard deviation of the cloud fraction on wet and dry days to the corresponding monthly mean. The y-axes show the standard deviation, the x-axes the mean on wet or dry days respectively. The blue line corresponds to the best fit line. Parameters of the fits are also shown in table 5.3.

TABLE 5.3: Fit results of cloud correlation for wet and dry days for figure 5.8

plot	variable	a	std. err. of a	R^2
5.8	c_{dry}	0.4302	0.0013	0.8745
5.8	c_{wet}	-0.7376	0.0006	0.3881
5.9	$c_{\text{sd,dry}}$	1.0448	0.0004	0.2803
5.9	$c_{\text{sd,wet}}$	0.9881	0.0006	0.0802

units of octas, i.e., values ranging from 0 (clear sky) to 8 (overcast). Hence, to calculate the daily cloud fraction, those values were averaged and divided by 8 to produce a daily mean.

To adjust the monthly mean depending on the wet/dry state of the day, we could not use a simple linear relationship as we used for temperature because cloud fraction is bounded by a lower limit 0 and an upper limit of 1. Furthermore, we observed that cloud cover on wet days is usually greater or equal to the monthly mean cloud cover, whereas the cloud cover on dry days is usually less or equal to the monthly mean cloud cover. This results in a concave curve for the wet case and a convex curve for dry days. We used a qualitative graphical analysis to develop "best guess" equations that had the desired shape and propose the following formulae for the regression linking cloud cover on wet or dry days to the overall mean:

$$\begin{aligned} \bar{c}_{\text{wet}} &= \frac{-a_{\text{c,wet}} - 1}{a_{\text{c,wet}}^2 \cdot \bar{c} - a_{\text{c,wet}}^2 - a_{\text{c,wet}}} - \frac{1}{a_{\text{c,wet}}} \\ \bar{c}_{\text{dry}} &= \frac{-a_{\text{c,dry}} - 1}{a_{\text{c,dry}}^2 \cdot \bar{c} - a_{\text{c,dry}}^2 - a_{\text{c,dry}}} - \frac{1}{a_{\text{c,dry}}} \end{aligned} \tag{5.12}$$

with $a_{\text{c,wet}} < 0$ and $a_{\text{c,dry}} > 0$.

The standard deviation of cloud cover fraction becomes 0 when the mean monthly cloud fraction reaches both the minimum or maximum limits of 0 and 1. Hence, for $c_{\text{sd,dry}}$ and $c_{\text{sd,wet}}$ we have an concave parabola with the formula

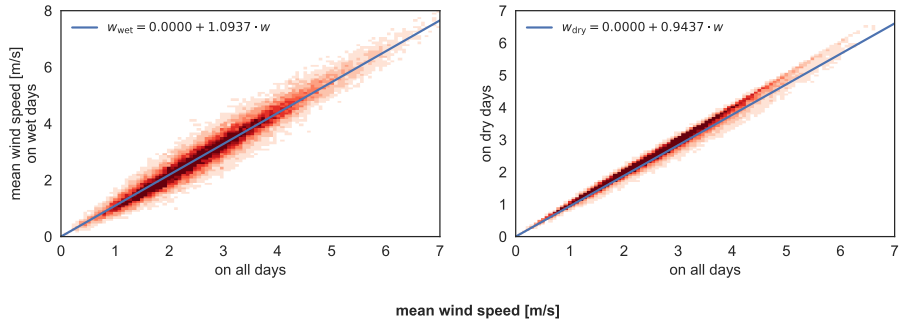


FIGURE 5.10: Correlation of wind speed on wet and dry days to the monthly mean.The y-axes show the mean cloud fraction on wet or dry days respectively, the blue line corresponds to the best fit line. Parameters of the fits are also shown in table 5.1.

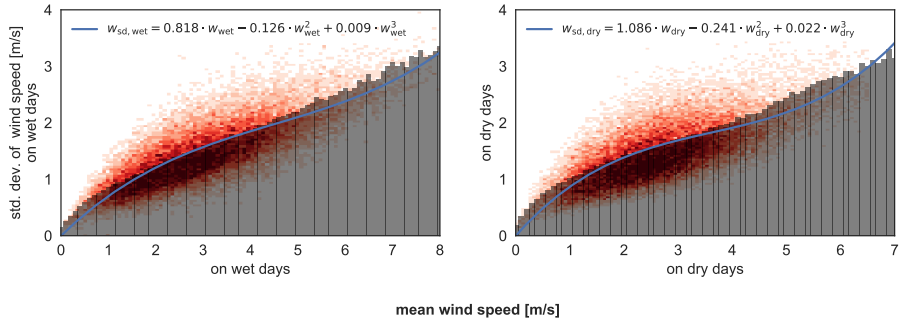


FIGURE 5.11: Correlation of standard deviation of wind speed on wet and dry days to the corresponding monthly mean. The y-axes show the standard deviation, the x-axes the mean on wet or dry days respectively. The blue line corresponds to the best fit line, a third order polynomial to the underlying red density plot. The black bars have a width of 0.1 m s^{-1}, the accuracy of the input data, and indicate the mean standard deviations for the given interval range. Parameters of the fits are also shown in table 5.1.

$$\begin{aligned} \sigma_{c,wet} &= a_{c,wet}^2 \cdot \bar{c}_{wet} \cdot (1 - \bar{c}_{wet}) \\ \sigma_{c,dry} &= a_{c,dry}^2 \cdot \bar{c}_{dry} \cdot (1 - \bar{c}_{dry}) \end{aligned} \tag{5.13}$$

with $a_{c,wet}, a_{c,dry} \geq 0$. Results of the fits can be seen in figure 5.8, 5.9 and the parameters in table 5.3.

Wind speed

The parameterization of the mean wind speed is based upon the same linear equation (5.10) as temperature. For the standard deviation however, we use a third-order polynomial given that is forced through the origin, given via

$$\begin{aligned} \sigma_{w,wet}(\bar{w}_{wet}) &= c_{1,w,wet}\, \bar{w}_{wet} + c_{2,w,wet}\, \bar{w}_{wet}^2 + c_{3,w,wet}\, \bar{w}_{wet}^3 \\ \sigma_{w,dry}(\bar{w}_{dry}) &= c_{1,w,dry}\, \bar{w}_{dry} + c_{2,w,dry}\, \bar{w}_{dry}^2 + c_{3,w,dry}\, \bar{w}_{dry}^3. \end{aligned} \tag{5.14}$$

This better resolves the complex behavior close to $0\,\mathrm{m\,s^{-1}}$ compared to a linear fit. The plots are shown in the figures 5.10 and 5.11 and the parameters for the fits are shown in table 5.1.

Cross correlation

Following Richardson, 1981 we use cross correlation to add additional residual noise to the simulated meteorological variables, which provides more realism in the daily weather result. This methodology, based on Matalas, 1967 preserves the serial and the cross correlation between the simulated variables. It implies that the serial correlation of each variable may be described by a first-order linear autoregressive model

Given the cross correlation matrix $M_0 \in \mathbb{R}^4 \times \mathbb{R}^4$ and the lag-1 correlation matrix $M_1 \in \mathbb{R}^4 \times \mathbb{R}^4$, we calculate

$$A = M_1 M_0^{-1} \qquad BB^T = M_0 - M_1 M_0^{-1} M_1^T. \tag{5.15}$$

The matrices A, B, M_0 and M_1 are calculated using the stations from the EECRA database in figure 5.2. The results are

$$M_0 = \begin{pmatrix} 1. & 0.565 & 0.041 & 0.035 \\ 0.565 & 1. & -0.089 & -0.043 \\ 0.041 & -0.089 & 1. & 0.114 \\ 0.035 & -0.043 & 0.114 & 1. \end{pmatrix} \qquad M_1 = \begin{pmatrix} 0.933 & 0.55 & 0.016 & 0.03 \\ 0.557 & 0.417 & -0.066 & -0.043 \\ 0.004 & -0.095 & 0.599 & 0.093 \\ 0.011 & -0.063 & 0.061 & 0.672 \end{pmatrix}. \tag{5.16}$$

leading to

$$A = \begin{pmatrix} 0.916 & 0.031 & -0.018 & 0.001 \\ 0.485 & 0.135 & -0.069 & -0.047 \\ 0.004 & -0.043 & 0.592 & 0.023 \\ 0.012 & -0.043 & -0.02 & 0.672 \end{pmatrix} \qquad B = \begin{pmatrix} 0.358 & 0. & 0. & 0. \\ 0.112 & 0.809 & 0. & 0. \\ 0.142 & -0.06 & 0.785 & 0. \\ 0.077 & -0.016 & 0.061 & 0.733 \end{pmatrix}. \tag{5.17}$$

The columns and rows in the two matrices correspond to min. and max. temperature, cloud fraction and square root of wind speed, respectively.

In the weather generator, the variables $T_{\min}, T_{\max}, c$ and w are then calculated using a combination of residual noise χ_i (where i denotes the current simulated day) and the mean of the variables. χ_i is determined by the other variables and the previous day using A and B from above (Matalas, 1967; Richardson, 1981). Hence, χ_i is given via

$$\chi_i = \begin{pmatrix} \chi_{T_{\min}} & \chi_{T_{\max}} & \chi_c & \chi_w \end{pmatrix} = A\chi_{i-1} + B\epsilon \in \mathbb{R}^4. \tag{5.18}$$

The daily values for the variables are then calculated via

$$T_{\min,i} = \chi_{T_{\min}} \cdot \sigma_{T_{\min},\mathrm{wet/dry}} + \bar{T}_{\min,\mathrm{wet/dry}} \qquad c_i = \chi_c \cdot \sigma_{c,\mathrm{wet/dry}} + \bar{c}_{\mathrm{wet/dry}} \tag{5.19}$$

$$T_{\max,i} = \chi_{T_{\max}} \cdot \sigma_{T_{\max},\mathrm{wet/dry}} + \bar{T}_{\max,\mathrm{wet/dry}} \qquad w_i = \left(\chi_w \cdot \sqrt{\sigma_{w,\mathrm{wet/dry}}} + \sqrt{\bar{w}_{\mathrm{wet/dry}}}\right)^2 \tag{5.20}$$

with $\sigma_{T_{\min},\mathrm{wet/dry}}, \sigma_{T_{\max},\mathrm{wet/dry}}$ from eq. (5.11), $\sigma_{c,\mathrm{wet/dry}}$ from eq. (5.13), $\sigma_{w,\mathrm{wet/dry}}$ from eq. (5.14), $\bar{T}_{\min,\mathrm{wet/dry}}, \bar{T}_{\max,\mathrm{wet/dry}}, \bar{w}_{\mathrm{wet/dry}}$ from eq. (5.10) and $\bar{c}_{\mathrm{wet/dry}}$ from eq. (5.12).

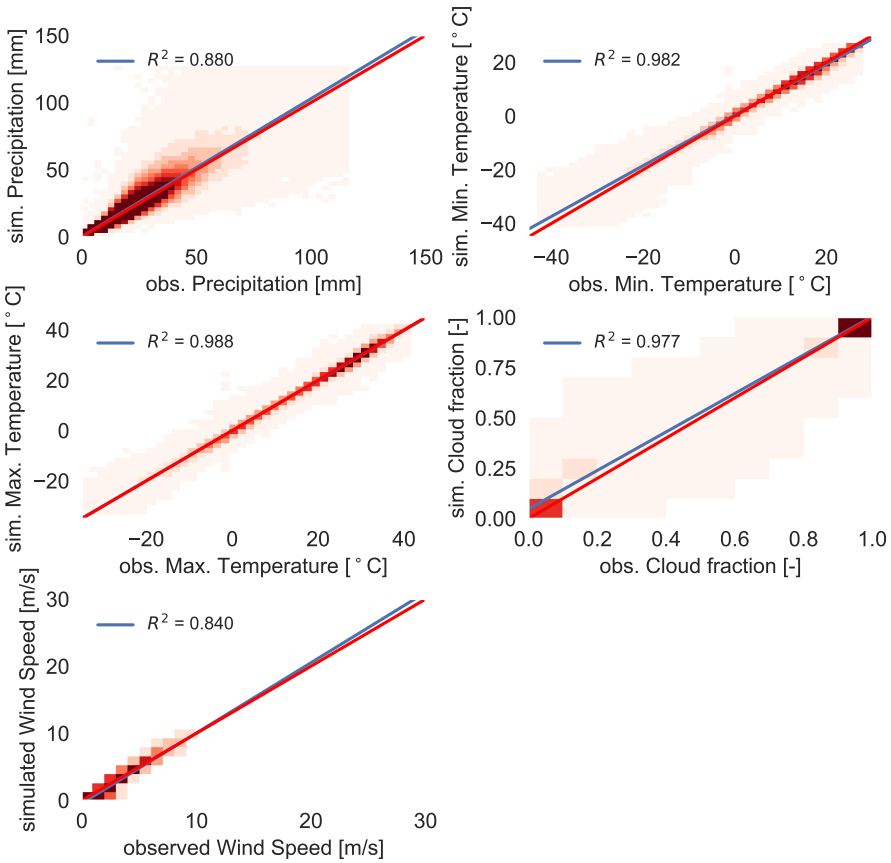


FIGURE 5.12: QQ-plots for all variables with all quantiles (1, 5, 10, 25, 50, 75, 90, 95 and 99) for $\mu = 5.0\,\text{mm}\,\text{mm}$, $\xi = 1.5$. The blue lines are linear regression from simulation to observation. The red line shows the ideal fit (the identity line). Blue shaded areas represent the 95% confidence interval. The plots compares the simulated quantile from the list above of one year of one station to the corresponding observed quantile of the same year and station. The plot for wind speed underwent used the bias correction from subsection 5.3.4.

Since this procedure always requires the residuals from the previous day, χ_{i-1}, we initialize χ_0 with 0, simulate the month and then simulate it again.

Note that, through the entire procedure, wind speed is subject to a square-root transformation (also when calculating M_0 and M_1) to account for the fact that it is not normally distributed.

5.3.3 Model Evaluation

To evaluate GWGEN, we started with the daily meteorology at the evaluation stations described above and calculated monthly summaries. We used this monthly data to drive the model and simulate daily meteorology. The resulting daily series now has the same length as the observed meteorology from the GHCN and EECRA database. Because we cannot expect the weather generator to reproduce the weather exactly as observed, for example the number of rainy days in a month may be the same as observed but they may not occur in precisely the same order, our evaluation is restricted to comparing the statistical properties of the input observed versus the output simulated daily meteorology.

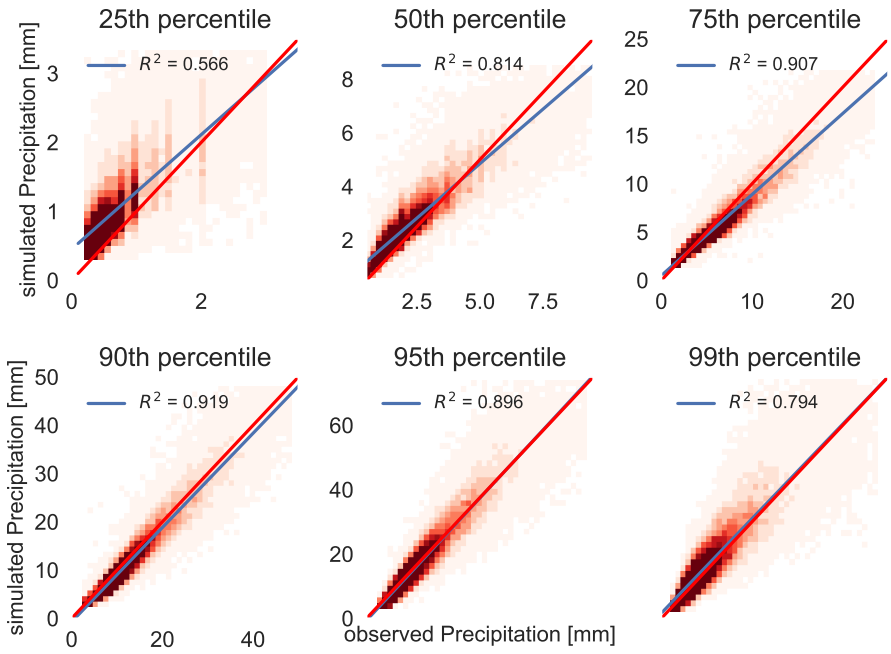


FIGURE 5.13: QQ-plot for different quantiles for precipitation for $\mu = 5.0$mm, $\xi = 1.5$. The blue lines are linear regression from simulation to observation. The red line shows the ideal fit (the identity line). Blue shaded areas represent the 95% confidence interval. The plots compare the simulated quantile of one year of one station to the corresponding observed quantile of the same year and station.

Figure 5.12 shows the comparison of simulated versus observed values for each of the five meteorological variables handled by GWGEN. For temperature, wind, and cloud fraction, the model does an excellent job of downscaling monthly input to daily resolution[1]. The comparison between precipitation amounts looks good when considering all of the data, however a closer look into the results (Fig. 5.13) shows that while the higher precipitation percentiles are well captured using the hybrid Gamma-GP distribution, the lower percentiles show somewhat worse results. This observation of poor performance for very low values also holds true for wind speed (not shown here). The lower values of the two variables, however, are very close to the precision of the observation (0.1 mm for precipitation and 0.1 m s^{-1} for wind speed). Very small precipitation amounts and low wind speeds are also less biophysically and ecologically important compared to the higher percentiles. We therefore consider the results of the evaluation largely acceptable.

In table 5.4 we also compare the simulated versus the observed frequencies. For very light rain (<=1mm), light rain (1-10mm), heavy rain (10-20mm) and very heavy rain (>20mm). As we can see, our model underestimates the occurrence of very light rain events (28.6% instead of 36.4%) and overestimates the light rain events (58.3% instead of 48.6%) but generally performs much better than GCMs (Dai, 2006; Sun et al., 2006), especially when it comes to the heavy rain events.

TABLE 5.4: Simulated and observed precipitation frequencies for certain ranges. The frequency is defined as the number of precipitation occurences in the specified range, divided by the total number of precipitation occurences.

Precip. range [mm]	Simulated	Observed
(0, 1]	0.285688	0.364014
(1, 10]	0.583330	0.486415
(10, 20]	0.074063	0.090178
(20, ∞]	0.056920	0.059392

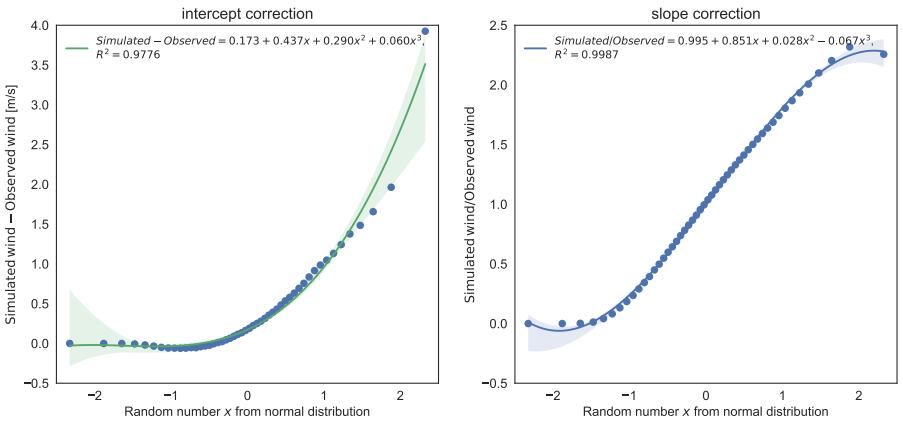


FIGURE 5.14: Basis for the wind bias correction. For the left plot, each data point corresponds to the difference of a simulated percentile to the observed percentile. For the right plot (wind speed), each data point corresponds to the fraction of simulated to the observed wind speed for a given percentile. The random number on the x-axis represents the residual value from a normal distribution centered at 0 with standard deviation of unity, as it is used in the cross correlation approach (Richardson, 1981).

5.3.4 Bias correction

After evaluating the results of GWGEN for wind speed for the different quantiles (see previous subsection 5.3.3) we found a strong, systematic bias between the simulated and the observed values. This observation led us to adopt a further measure to improve the quality of the model output by implementing a quantile-based bias correction.

We use an empirical distribution correction approach (quantile-mapping) (Lafon et al., 2012) to a posteriori correct the simulated data. In the quantile evaluation (previous subsection 5.3.3) we saw that the simulated wind speed is a linear function of the observed wind speed, i.e. $w_{sim} = \text{intercept} + \text{slope} \cdot w_{obs}$ (best fit line in figure 5.12). Therefore, we use two steps here, one is for the difference between simulation and observation (ideally 0), the other one is the fraction of observation and simulation (ideally 1). The first one corresponds to the intercept with the y-axis in figure 5.12, the second one to the slope of the best fit line. The analysis is based on every second percentile between 1 and 100 (i.e. $1, 3, 5, \ldots$) and mapped to it's corresponding random number $u \in \mathbb{R}$ from a normal distribution as it is used for the cross correlation in the weather generator (section 5.3.2, x-axis in figure 5.14 and Richardson, 1981).

Regarding the intercept (fig. 5.14, left) we see that it strongly follows an exponential function given through

$$f_{exp}(u) = \mathrm{e}^{a\,u+b}, \qquad a, b, u \in \mathbb{R}. \tag{5.21}$$

The slope (fig. 5.14, right) on the other hand can be described by a simple third-order polynomial given by

$$p3(u) = c_0 + c_1\,u + c_2\,u^2 + c_3\,u^3, \qquad c_0, c_1, c_2, c3, u \in \mathbb{R} \tag{5.22}$$

Hence, given the best fit lines in figure 5.14, the simulated wind speed is corrected via

$$w'_{sim} = \frac{w_{sim} - f_{exp}(u)}{p3(u)} \tag{5.23}$$

with $a = 1.1582, b = -1.3359, c_0 = 0.9954, c_1 = 0.8508, c_2 = 0.0278, c_3 = -0.0671$.

5.3.5 Sensitivity analysis

The Generalized-Pareto part of the hybrid Gamma-GP distribution, which we used to simulate precipitation amount, has two parameters: the GP shape, and the threshold parameter. Unlike the gamma parameters, we were unable to relate these GP parameters to any of the monthly summary data we use as input to GWGEN. Hence, we decided to set fixed values for these parameters, and determine them through a sensitivity analysis.

To select the "best" values of the GP parameters, we compared simulated with observed precipitation amounts, running GWGEN with a wide range of realistic parameter values. To quantitatively assess the model performance, we used two metrics: 1) direct comparison of the quantiles (see previous section), and 2) a Kolmogorov-Smirnov (KS) test that evaluates whether two data samples come from significantly different distributions. Our criteria were

1. The R^2 correlation coefficient between simulated and observed quantiles

[1] Note that the plot for wind speed has been bias corrected using the approach in subsection 5.3.4.

2. The fraction $\frac{\text{simulated precipitation}}{\text{observed precipitation}}$ from the slopes in figure 5.13 and it's deviation from unity

3. the fraction of simulated (station specific) years that are significantly different (KS test) from the observation

4. The mean of the above values

We tried two different approaches to select the gamma-GP crossover threshold: first we tried a fixed crossover point, second we used a quantile-based crossover point. For the latter, the model chooses to use the GP distribution if the quantile of the random number drawn from the gamma distribution is above a certain quantile threshold. This introduces a flexible crossover point in our hybrid distribution which, however, did not improve the results significantly. We therefore show here only the results using the fixed crossover point.

The values of the crossover point for our sensitivity analysis were 2, 2.5, 3, 4 and from 5 to 20 in steps of 2.5 and 20 to 100 in steps of 5. Furthermore we varied the GP shape parameter from 0.1 to 3 in steps of 0.1 (810 experiments in total). The results of this sensitivity analysis are shown in the supplementary material, figure 5.15.

In general we found that the three criteria 1, 2 and 3 could not be optimized all together at the same time. The R^2 is best for high thresholds and low GP shape parameters, the slope is best for low to intermediate thresholds and a low GP shape and the KS statistic is best for low threshold and intermediate GP shape parameters.

However, R^2 did not vary that much (from 0.68 to 0.74) and from a visual evaluation of the corresponding quantile plots we saw that the higher quantiles (>90) were much better represented for a better KS result. Hence we chose to follow the KS test criteria, which is also the strictest of our evaluation methods but again compared the different quantile plots to get good results for the higher quantiles. Finally, we chose a threshold of 5 mm and a GP shape parameter of 1.5. For this setting, 81.7% of the simulated years do not show a significant difference compared to the observation, the mean R^2 of the plots in figure 5.13 is 0.81 and the mean deviation of the slope from unity is 0.10 and for the upper quantiles (90 to 100), 0.017.

Nevertheless, in total the results seem to be fairly independent of the two parameters since even the amount of years without significant differences vary from 73% to only 83%. It is however better than the gamma distribution alone which still has 78.6% of station years not differing significantly but with a slope deviation from unity for the upper quantiles of 0.16. Thus using the hybrid Gamma-GP distribution improves the simulation of high-amount precipitation events by roughly factor 10 compared to a standard Gamma approach.

5.4 Limitations

As demonstrated above, GWGEN successfully downscales monthly to daily meteorology with good correlation and low bias when compared to observations. However, there are a few limitations of the model as currently described that should be noted. Importantly, this version of GWGEN neither downscales all conceivable meteorological variables, nor does it provide a mechanism for generating daily meteorological timeseries across multiple points that are spatially autocorrelated. Concerning the former point, while GWGEN simulates daily precipitation, temperature, cloud cover, and windspeed, it does not currently handle other variables that might be important in land surface modeling, such as humidity or wind direction. On the

latter point, the lack of explicit simulation of spatial autocorrelation may make GWGEN unsuitable for certain applications, e.g., regional high-resolution hydrological modeling in small catchments (< ca. 2500 km^2), where having the capability to simulate flood and other extremes is important. This is because the the weather generator could, e.g., simulate rainfall on different days in different parts of the catchment, where in reality storm events would be highly autocorrelated in space and controlled by mesoscale meteorological conditions.

5.5 Discussion and Outlook

GWGEN successfully downscales monthly to daily meteorology, for any point on the globe, in any climate, in any season, and in any time in recent earth history and into the near future (e.g., next century). It extends the original Richardson-type weather generators to simulate wind speed along with precipitation, temperature, and cloud cover. The model requires only monthly values of the meteorological variables to be downscaled, and does not rely on any other spatial information, e.g., whether or not the location is in the tropics.

In general, the results of our downscaled meteorology are excellent, with all simulated variables showing both very high correlation and limited bias when compared to observations. We improved the simulation of daily precipitation amount by replacing the Gamma distribution used in the original Richardson-type weather generators with a hybrid Gamma-GP distribution, which results in the improved simulation of heavy precipitation events. The GP distribution is based upon a globally fixed shape and location parameter, which may be an oversimplification, but is still ten times more accurate than traditional methods that used Gamma alone. Our extensive sensitivity analysis to determine the best coefficients for the shape and location parameters of the GP distribution suggest that further improvements might come through correlating the GP parameters to geographic region and/or seasonality (Maraun et al., 2009; Rust et al., 2009) or by introducing a dynamical location parameter (Frigessi et al., 2002). Finally, we introduced a step to correct for systematic bias in the downscaling of temperature and wind speed.

Despite the limitations noted above, GWGEN will be useful in a wide range of applications, from global vegetation and crop modeling, to large-scale hydrologic analyses, to understanding animal behavior, to forecasting of fire, insect outbreaks, and other ecosystem disturbances. GWGEN may even be envisaged as a potential replacement for very large and cumbersome gridded datasets of high-temporal resolution meteorology such as CRUNCEP (Viovy and Ciais, 2016), especially for models that use meteorological forcing at a daily timestep. The weather generator is particularly suited for the incorporation into models that run on a spatial grid, for example, GWGEN can readily be incorporated into existing DGVMs such as LPJ-LMfire (Pfeiffer et al., 2013) or LPJ-ML (Bondeau et al., 2007) that already rely on a weather generator to provide daily meteorology for certain processes.

While GWGEN does not handle spatial autocorrelation, in most DGVMs there is no lateral connection between gridcells, and therefore an explicit representation of spatial autocorrelation in the driving daily meteorological data would have no effect on the model output. We further note that if the monthly data used to drive the model are spatially autocorrelated — this would be the case when using gridded climatology for example — then the result of the weather generator will also preserve this autocorrelation, at least when the model results are analyzed on monthly or longer timescales.

The limitations present in this version of GWGEN could be addressed in future versions. Methods for simultaneous multisite weather generation exist (D. S. Wilks, 1998, 1999b,c) and could be adapted to GWGEN. However, even simpler methods to approximate spatial autocorrelation could be possible. Running GWGEN with gridded monthly meteorology — this is the primary application we foresee for the current version of the model — means that the input variables are already highly correlated in space, i.e., the monthly climate in one gridcell generally closely resembles neighboring cells, outside of complex terrain containing sharp, monotonic climate gradients, e.g., rain shadows. Thus, one simple way of achieving a measure of spatial autocorrelation in GWGEN would be to impose a spatial autocorrelation field on the sequence of random numbers used to impose stochastic noise in the downscaling functions. If the random number sequence is similar between gridcells, then, e.g., rain is likely to fall on the same day, given that the transition probabilities will likely also be similar. Over moderate distances, e.g., <50's of km, it might even be sufficient to use the same random seed across all gridcells in a neighborhood. This would have the effect of producing strongly autocorrelated daily meteorology in space, with the only variations being imposed by the underlying input monthly climatology.

Furthermore, it would be straightforward to include additional meteorological variables in the model framework, handling, e.g., humidity in the same way that temperatures, cloud cover, and wind speed are disaggregated. Other variables, such as pressure and wind direction, might be more difficult using the basic GWGEN structure because of the importance of autocorrelation, particularly at high spatial resolution, and might benefit from a different approach towards weather generation. Finally, GWGEN only downscales meteorology from monthly to daily values; for models that require an even shorter timestep, e.g., 6-hourly, some extension of the model functionality would be required. For certain variables, e.g., temperatures, sub-daily downscaling could be easily implemented (Cesaraccio et al., 2001), for other variables, such as precipitation, a large literature on downscaling methods exists (e.g. Bennett et al., 2016), and global datasets of hourly meteorology for model calibration are available (e.g., the Integrated Surface Database, Smith et al., 2011).

5.6 Conclusions

Compiling a global database of daily precipitation, temperature, cloud cover, and wind speed measurements, we explored the relationship between daily meteorology and monthly summaries first described in the context of weather downscaling by Shu Geng and Auburn, 1987. Our analysis of more than 50 million individual records showed that daily-to-monthly relationships are relatively stable in space and time, and constant across a very wide range of stations from all latitudes and climate zones. With the resulting relationships, we parameterized a WGEN/SIMMETEO-type weather generator, with the intention of creating a generic scheme that could be applied anywhere over the earth's land surface for the past, present, and (near) future.

5.7 Code availability

GWGEN, is open source software, and the code, utility programs for parameterization, evaluation and manipulating the raw weather station data, and complete documentation are available at (Sommer and Jed O. Kaplan, 2017). The original

weather station database can be made available upon request to the authors or downloaded from Hahn and Warren, 1999 and Menne et al., 2012b. The weather generator module is programmed in FORTRAN, the parameterization, evaluation and other supplementary tools are written in Python mainly using the numerical python libraries numpy and scipy (E. Jones et al., 2001), statsmodels (Seabold and Perktold, 2010), as well as matplotlib (Hunter, 2007) and psyplot (Sommer, 2017) for the visualization. Detailed installation instructions can be found in the user manual: https://arve-research.github.io/gwgen/.

5.A Supplementary material

5.A.1 Sensitivity analysis

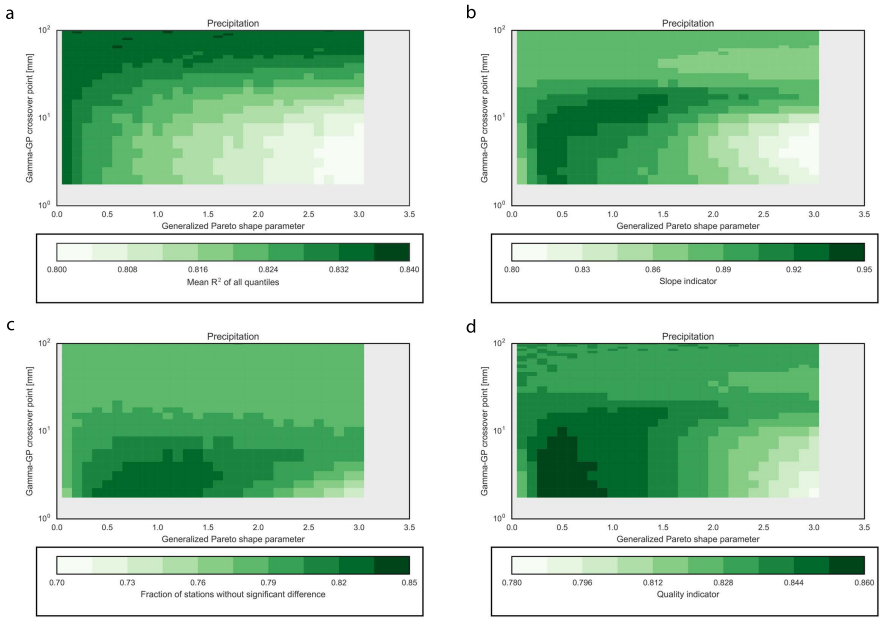


FIGURE 5.15: Results of the sensitivity analysis for the (a) correlation coefficient R^2, (b) deviation from a slope of unity, (c) the fraction of significant different station years, (d) the mean of (a) - (c). For the plots in (a) and (b) we used the means of the 25th, 50th, 75th, 90th, 95th and 99th percentiles. In general, 1 (dark green) is best, 0 (white) is worst. The dark red fields indicate experiments that failed because of a too low threshold and too high GP shape parameter. Note the logarithmic scale on the y-axis.

Author contributions. JOK conceived the model and analyses, wrote the prototype code and performed preliminary analyses, PS developed and documented the final version of the code (including parameterization and evaluation), performed all of the final analyses, and created the graphical output. Both authors contributed to the writing of the manuscript

Acknowledgements. This work was supported by the European Research Council (COEVOLVE, 313797) and the Swiss National Science Foundation (ACACIA, CR10I2_146314). We thank Shawn Koppenhoefer for assistance compiling and querying the weather databases and Alexis Berne and Grégoire Mariéthoz for helpful suggestions on the analyses. We are grateful to NOAA NCDC and the University of Washington for providing free of charge the GHCN-Daily and EECRA databases, respectively.

References

Bennett, James C., David E. Robertson, Phillip G.D. Ward, H.A. Prasantha Hapuarachchi, and Q.J. Wang (2016). "Calibrating hourly rainfall-runoff models with daily forcings for streamflow forecasting applications in meso-scale catchments". In: *Environmental Modelling & Software* 76, pp. 20–36. ISSN: 1364-8152. DOI: http://dx.

doi.org/10.1016/j.envsoft.2015.11.006. URL: http://www.sciencedirect.com/science/article/pii/S1364815215300979.

Bhatt, Samir, Peter W. Gething, Oliver J. Brady, Jane P. Messina, Andrew W. Farlow, Catherine L. Moyes, John M. Drake, John S. Brownstein, Anne G. Hoen, Osman Sankoh, Monica F. Myers, Dylan B. George, Thomas Jaenisch, G. R. William Wint, Cameron P. Simmons, Thomas W. Scott, Jeremy J. Farrar, and Simon I. Hay (2013). "The global distribution and burden of dengue". In: *Nature* 496.7446, pp. 504–507. DOI: 10.1038/nature12060. URL: http://dx.doi.org/10.1038/nature12060.

Bondeau, Alberte, Pascalle C. Smith, SÖNke Zaehle, Sibyll Schaphoff, Wolfgang Lucht, Wolfgang Cramer, Dieter Gerten, Hermann Lotze-Campen, Christoph MÜLler, Markus Reichstein, and Benjamin Smith (2007). "Modelling the role of agriculture for the 20th century global terrestrial carbon balance". In: *Global Change Biol.* 13.3, pp. 679–706. ISSN: 1354-1013 1365-2486. DOI: 10.1111/j.1365-2486.2006.01305.x.

Cesaraccio, C., D. Spano, P. Duce, and R. L. Snyder (2001). "An improved model for determining degree-day values from daily temperature data". In: *Int. J. Biometeorol.* 45.4, pp. 161–9. ISSN: 0020-7128 (Print) 0020-7128 (Linking). DOI: 10.1007/s004840100104. URL: http://www.ncbi.nlm.nih.gov/pubmed/11769315.

Dai, Aiguo (2006). "Precipitation Characteristics in Eighteen Coupled Climate Models". In: *J. Climate* 19.18, pp. 4605–4630. DOI: 10.1175/jcli3884.1. URL: http://dx.doi.org/10.1175/JCLI3884.1.

Elith, Jane, Catherine H. Graham, Robert P. Anderson, Miroslav Dudík, Simon Ferrier, Antoine Guisan, Robert J. Hijmans, Falk Huettmann, John R. Leathwick, Anthony Lehmann, Jin Li, Lucia G. Lohmann, Bette A. Loiselle, Glenn Manion, Craig Moritz, Miguel Nakamura, Yoshinori Nakazawa, Jacob McC. M. Overton, A. Townsend Peterson, Steven J. Phillips, Karen Richardson, Ricardo Scachetti-Pereira, Robert E. Schapire, Jorge Soberón, Stephen Williams, Mary S. Wisz, and Niklaus E. Zimmermann (2006). "Novel methods improve prediction of species' distributions from occurrence data". In: *Ecography* 29.2, pp. 129–151. ISSN: 1600-0587. DOI: 10.1111/j.2006.0906-7590.04596.x. URL: http://dx.doi.org/10.1111/j.2006.0906-7590.04596.x.

Friend, A. D. (1998). "Parameterisation of a global daily weather generator for terrestrial ecosystem modelling". In: *Ecol. Modell.* 109.2, pp. 121–140. ISSN: 0304-3800. DOI: Doi10.1016/S0304-3800(98)00036-2.

Frigessi, Arnoldo, Ola Haug, and Håvard Rue (2002). "A Dynamic Mixture Model for Unsupervised Tail Estimation without Threshold Selection". In: *Extremes* 5.3, pp. 219–235. ISSN: 1572-915X. DOI: 10.1023/A:1024072610684. URL: http://dx.doi.org/10.1023/A:1024072610684.

Furrer, Eva M. and Richard W. Katz (2008). "Improving the simulation of extreme precipitation events by stochastic weather generators". In: *Water Resour. Res.* 44.12, n/a–n/a. ISSN: 00431397. DOI: 10.1029/2008wr007316.

Geng, S., F. W. T. P. Devries, and I. Supit (1986). "A Simple Method for Generating Daily Rainfall Data". In: *Agric. For. Meteorol.* 36.4, pp. 363–376. ISSN: 0168-1923. DOI: 10.1016/0168-1923(86)90014-6.

Geng, Shu and J. S. Auburn (1987). "Weather simulation models based on summaries of long-term data". In: *Weather and Rice: Proceedings of the international workshop on the Impact of Weather Parameters on Growth and Yield of Rice, 7-10 Apr 1986*. Ed. by International Rice Research Institute. Los Baños, Philippines: International Rice Research Institute, pp. 237–254.

Gerten, Dieter, Sibyll Schaphoff, Uwe Haberlandt, Wolfgang Lucht, and Stephen Sitch (2004). "Terrestrial vegetation and water balance—hydrological evaluation

of a dynamic global vegetation model". In: *J. Hydrol.* 286.1-4, pp. 249–270. ISSN: 00221694. DOI: 10.1016/j.jhydrol.2003.09.029.

Gordon, H. A. (1981). "Errors in Computer Packages. Least Squares Regression Through the Origin". In: *Journal of the Royal Statistical Society. Series D (The Statistician)* 30.1, pp. 23–29. ISSN: 00390526, 14679884. URL: http://www.jstor.org/stable/2987701.

Guenther, A., C. N. Hewitt, D. Erickson, R. Fall, C. Geron, T. Graedel, P. Harley, L. Klinger, M. Lerdau, W. A. Mckay, T. Pierce, B. Scholes, R. Steinbrecher, R. Tallamraju, J. Taylor, and P. Zimmerman (1995). "A Global-Model of Natural Volatile Organic-Compound Emissions". In: *Journal of Geophysical Research-Atmospheres* 100.D5, pp. 8873–8892. ISSN: 2169-897x. DOI: Doi10.1029/94jd02950.

Hahn, C.J. and S.G. Warren (1999). "Extended Edited Synoptic Cloud Reports from Ships and Land Stations Over the Globe, 1952-1996 (with Ship data updated through 2008)". In: DOI: 10.3334/CDIAC/cli.ndp026c. URL: http://dx.doi.org/10.3334/CDIAC/cli.ndp026c.

Harris, I., P. D. Jones, T. J. Osborn, and D. H. Lister (2014). "Updated high-resolution grids of monthly climatic observations - the CRU TS3.10 Dataset". In: *Int. J. Climatol.* 34.3, pp. 623–642. ISSN: 08998418. DOI: 10.1002/joc.3711.

Haxeltine, A. and I. C. Prentice (1996). "BIOME3: An equilibrium terrestrial biosphere model based on ecophysiological constraints, resource availability, and competition among plant functional types". In: *Global Biogeochem. Cycles* 10.4, pp. 693–709. ISSN: 0886-6236. DOI: Doi10.1029/96gb02344.

Haxeltine, Alex, I. Colin Prentice, and Ian David Creswell (1996). "A coupled carbon and water flux model to predict vegetation structure". In: *J. Veg. Sci.* 7.5, pp. 651–666. ISSN: 1654-1103. DOI: 10.2307/3236377. URL: http://dx.doi.org/10.2307/3236377.

Hijmans, Robert J., Susan E. Cameron, Juan L. Parra, Peter G. Jones, and Andy Jarvis (2005). "Very high resolution interpolated climate surfaces for global land areas". In: *Int. J. Climatol.* 25.15, pp. 1965–1978. ISSN: 0899-8418 1097-0088. DOI: 10.1002/joc.1276.

Hunter, J. D. (2007). "Matplotlib: A 2D Graphics Environment". In: *Computing in Science Engineering* 9.3, pp. 90–95. ISSN: 1521-9615. DOI: 10.1109/MCSE.2007.55.

Jones, Eric, Travis Oliphant, Pearu Peterson, et al. (2001). *SciPy: Open source scientific tools for Python*. [Online; accessed 2017-02-18]. URL: http://www.scipy.org/.

Kaplan, J. O., N. H. Bigelow, I. C. Prentice, S. P. Harrison, P. J. Bartlein, T. R. Christensen, W. Cramer, N. V. Matveyeva, A. D. McGuire, D. F. Murray, V. Y. Razzhivin, B. Smith, D. A. Walker, P. M. Anderson, A. A. Andreev, L. B. Brubaker, M. E. Edwards, and A. V. Lozhkin (2003). "Climate change and Arctic ecosystems: 2. Modeling, paleodata-model comparisons, and future projections". In: *Journal of Geophysical Research-Atmospheres* 108.D19. ISSN: 2169-897x. DOI: Artn817110.1029/2002jd002559.

Krinner, G., Nicolas Viovy, Nathalie de Noblet-Ducoudré, Jérôme Ogée, Jan Polcher, Pierre Friedlingstein, Philippe Ciais, Stephen Sitch, and I. Colin Prentice (2005). "A dynamic global vegetation model for studies of the coupled atmosphere-biosphere system". In: *Global Biogeochem. Cycles* 19.1, n/a–n/a. ISSN: 08866236. DOI: 10.1029/2003gb002199.

Kucharik, Christopher J., Jonathan A. Foley, Christine Delire, Veronica A. Fisher, Michael T. Coe, John D. Lenters, Christine Young-Molling, Navin Ramankutty, John M. Norman, and Stith T. Gower (2000). "Testing the performance of a dynamic global ecosystem model: Water balance, carbon balance, and vegetation

structure". In: *Global Biogeochem. Cycles* 14.3, pp. 795–825. ISSN: 1944-9224. DOI: 10.1029/1999GB001138. URL: http://dx.doi.org/10.1029/1999GB001138.

Lafon, Thomas, Simon Dadson, Gwen Buys, and Christel Prudhomme (2012). "Bias correction of daily precipitation simulated by a regional climate model: a comparison of methods". In: *Int. J. Climatol.* 33.6, pp. 1367–1381. DOI: 10.1002/joc.3518. URL: http://dx.doi.org/10.1002/joc.3518.

Leemans, Rik and Wolfgang P Cramer (1991). "The IIASA database for mean monthly values of temperature, precipitation, and cloudiness on a global terrestrial grid". In: *International Institute for Applied Systems Analysis, Laxenburg, Austria.*

Lieth, Helmut (1975). "Modeling the Primary Productivity of the World". In: *Primary Productivity of the Biosphere*. Ed. by Helmut Lieth and Robert H. Whittaker. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 237–263. ISBN: 978-3-642-80913-2. DOI: 10.1007/978-3-642-80913-2_12. URL: http://dx.doi.org/10.1007/978-3-642-80913-2_12.

Maraun, D., H. W. Rust, and T. J. Osborn (2009). "The annual cycle of heavy precipitation across the United Kingdom: a model based on extreme value statistics". In: *Int. J. Climatol.* 29.12, pp. 1731–1744. DOI: 10.1002/joc.1811. URL: http://dx.doi.org/10.1002/joc.1811.

Matalas, N. C. (1967). "Mathematical assessment of synthetic hydrology". In: *Water Resour. Res.* 3.4, pp. 937–945. ISSN: 1944-7973. DOI: 10.1029/WR003i004p00937. URL: http://dx.doi.org/10.1029/WR003i004p00937.

Menne, Matthew J., Imke Durre, Bryant Korzeniewski, Shelley McNeill, Kristy Thomas, Xungang Yin, Steven Anthony, Ron Ray, Russell S. Vose, Byron E. Gleason, and Tamara G. Houston (2012a). "Global Historical Climatology Network - Daily (GHCN-Daily), Version 3.22". In: DOI: 10.7289/V5D21VHZ. URL: http://dx.doi.org/10.7289/V5D21VHZ.

Menne, Matthew J., Imke Durre, Russell S. Vose, Byron E. Gleason, and Tamara G. Houston (2012b). "An Overview of the Global Historical Climatology Network-Daily Database". In: *J. Atmos. Oceanic Technol.* 29.7, pp. 897–910. DOI: 10.1175/jtech-d-11-00103.1. URL: http://dx.doi.org/10.1175/JTECH-D-11-00103.1.

Mitchell, Timothy D. and Philip D. Jones (2005). "An improved method of constructing a database of monthly climate observations and associated high-resolution grids". In: *Int. J. Climatol.* 25.6, pp. 693–712. ISSN: 0899-8418 1097-0088. DOI: 10.1002/joc.1181.

New, M., M. Hulme, and P. Jones (1999). "Representing twentieth-century space-time climate variability. Part I: Development of a 1961-90 mean monthly terrestrial climatology". In: *J. Climate* 12.3, pp. 829–856. ISSN: 0894-8755. DOI: Doi10.1175/1520-0442(1999)012<0829:Rtcstc>2.0.Co;2.

— (2000). "Representing twentieth-century space-time climate variability. Part II: Development of 1901-96 monthly grids of terrestrial surface climate". In: *J. Climate* 13.13, pp. 2217–2238. ISSN: 0894-8755. DOI: Doi10.1175/1520-0442(2000)013<2217:Rtcstc>2.0.Co;2.

New, M., D. Lister, M. Hulme, and I. Makin (2002). "A high-resolution data set of surface climate over global land areas". In: *Climate Research* 21.1, pp. 1–25. ISSN: 0936-577x. DOI: DOI10.3354/cr021001.

Neykov, N. M., P. N. Neytchev, and W. Zucchini (2014). "Stochastic daily precipitation model with a heavy-tailed component". In: *Natural Hazards and Earth System Science* 14.9, pp. 2321–2335. ISSN: 1684-9981. DOI: 10.5194/nhess-14-2321-2014.

Parlange, Marc B. and Richard W. Katz (2000). "An Extended Version of the Richardson Model for Simulating Daily Weather Variables". In: *J. Appl. Meteorol.* 39.5,

pp. 610–622. DOI: 10.1175/1520-0450-39.5.610. URL: http://dx.doi.org/10.1175/1520-0450-39.5.610.

Pfeiffer, M., A. Spessa, and J. O. Kaplan (2013). "A model for global biomass burning in preindustrial time: LPJ-LMfire (v1.0)". In: *Geosci. Model Dev.* 6.3, pp. 643–685. ISSN: 1991-959x. DOI: 10.5194/gmd-6-643-2013. URL: http://www.geosci-model-dev.net/6/643/2013/gmd-6-643-2013.pdf.

Prentice, I. C., W. Cramer, S. P. Harrison, R. Leemans, R. A. Monserud, and A. M. Solomon (1992). "A Global Biome Model Based on Plant Physiology and Dominance, Soil Properties and Climate". In: *J. Biogeogr.* 19.2, pp. 117–134. ISSN: 0305-0270. DOI: Doi10.2307/2845499.

Prentice, I.C. (1989). *Developing a Global Vegetation Dynamics Model: Results of an IIASA Summer Workshop*. IIASA Research Report. IIASA, Laxenburg, Austria. URL: http://pure.iiasa.ac.at/3223/.

Richardson, C. W. (1981). "Stochastic simulation of daily precipitation, temperature, and solar radiation". In: *Water Resour. Res.* 17.1, pp. 182–190. ISSN: 00431397. DOI: 10.1029/WR017i001p00182.

Rust, H. W., D. Maraun, and T. J. Osborn (2009). "Modelling seasonality in extreme precipitation". In: *The European Physical Journal Special Topics* 174.1, pp. 99–111. DOI: 10.1140/epjst/e2009-01093-7. URL: http://dx.doi.org/10.1140/epjst/e2009-01093-7.

Rymes, M.D. and D.R. Myers (2001). "Mean preserving algorithm for smoothly interpolating averaged data". In: *Sol. Energy* 71.4, pp. 225–231. DOI: 10.1016/s0038-092x(01)00052-4. URL: http://dx.doi.org/10.1016/S0038-092X(01)00052-4.

Seabold, Skipper and Josef Perktold (2010). *Statsmodels: Econometric and statistical modeling with python*.

Sitch, S., B. Smith, I. C. Prentice, A. Arneth, A. Bondeau, W. Cramer, J. O. Kaplan, S. Levis, W. Lucht, M. T. Sykes, K. Thonicke, and S. Venevsky (2003). "Evaluation of ecosystem dynamics, plant geography and terrestrial carbon cycling in the LPJ dynamic global vegetation model". In: *Global Change Biol.* 9.2, pp. 161–185. ISSN: 1354-1013. DOI: 10.1046/j.1365-2486.2003.00569.x. URL: http://onlinelibrary.wiley.com/doi/10.1046/j.1365-2486.2003.00569.x/abstract.

Smith, Adam, Neal Lott, and Russ Vose (2011). "The Integrated Surface Database: Recent Developments and Partnerships". In: *Bull. Amer. Meteor. Soc.* 92.6, pp. 704–708. DOI: 10.1175/2011BAMS3015.1. eprint: https://doi.org/10.1175/2011BAMS3015.1. URL: https://doi.org/10.1175/2011BAMS3015.1.

Sommer, Philipp S. (2017). "The psyplot interactive visualization framework". In: *The Journal of Open Source Software* 2.16. DOI: 10.21105/joss.00363. URL: https://doi.org/10.21105/joss.00363.

Sommer, Philipp S. and Jed O. Kaplan (2017). "GWGEN v1.0.2: A global weather generator for daily data". In: DOI: 10.5281/zenodo.889213. URL: https://github.com/ARVE-Research/gwgen.

Stephens, Graeme L., Tristan L'Ecuyer, Richard Forbes, Andrew Gettelmen, Jean-Christophe Golaz, Alejandro Bodas-Salcedo, Kentaroh Suzuki, Philip Gabriel, and John Haynes (2010). "Dreary state of precipitation in global models". In: *Journal of Geophysical Research: Atmospheres* 115.D24, n/a–n/a. ISSN: 2156-2202. DOI: 10.1029/2010JD014532. URL: http://dx.doi.org/10.1029/2010JD014532.

Sun, Ying, Susan Solomon, Aiguo Dai, and Robert W. Portmann (2006). "How Often Does It Rain?" In: *J. Climate* 19.6, pp. 916–934. DOI: 10.1175/jcli3672.1. URL: http://dx.doi.org/10.1175/JCLI3672.1.

Viovy, N. and P. Ciais (2016). Online Database. URL: http://dods.extra.cea.fr/data/p529viov/cruncep.

Walter, H and H Lieth (1967). "Climate diagram world atlas". In: *VEB Gustav Fischer Verlag Jena, Jena*.

Wei, Y., S. Liu, D. N. Huntzinger, A. M. Michalak, N. Viovy, W. M. Post, C. R. Schwalm, K. Schaefer, A. R. Jacobson, C. Lu, H. Tian, D. M. Ricciuto, R. B. Cook, J. Mao, and X. Shi (2014). "The North American Carbon Program Multi-scale Synthesis and Terrestrial Model Intercomparison Project – Part 2: Environmental driver data". In: *Geosci. Model Dev.* 7.6, pp. 2875–2893. ISSN: 1991-9603. DOI: 10.5194/gmd-7-2875-2014. URL: http://www.geosci-model-dev.net/7/2875/2014/.

Wilks, D. S. (1998). "Multisite generalization of a daily stochastic precipitation generation model". In: *J. Hydrol.* 210.1-4, pp. 178–191. ISSN: 0022-1694. DOI: 10.1016/S0022-1694(98)00186-3.

— (1999a). "Interannual variability and extreme-value characteristics of several stochastic daily precipitation models". In: *Agric. For. Meteorol.* 93.3, pp. 153–169. ISSN: 0168-1923. DOI: 10.1016/S0168-1923(98)00125-7.

— (1999b). "Multisite downscaling of daily precipitation with a stochastic weather generator". In: *Climate Research* 11.2, pp. 125–136. ISSN: 0936-577x. DOI: 10.3354/cr011125.

— (1999c). "Simultaneous stochastic simulation of daily precipitation, temperature and solar radiation at multiple sites in complex terrain". In: *Agric. For. Meteorol.* 96.1-3, pp. 85–101. ISSN: 0168-1923. DOI: 10.1016/S0168-1923(99)00037-4.

Wilks, D. S. and R. L. Wilby (1999). "The weather generation game: a review of stochastic weather models". In: *Prog. Phys. Geog.* 23.3, pp. 329–357. ISSN: 0309-1333. DOI: 10.1177/030913339902300302.

Wilks, Daniel S. (2010). "Use of stochastic weathergenerators for precipitation downscaling". In: *Wiley Interdiscip. Rev. Clim. Change* 1.6, pp. 898–907. ISSN: 17577780. DOI: 10.1002/wcc.85.

Woodward, F. Ian, Thomas M. Smith, and William R. Emanuel (1995). "A global land primary productivity and phytogeography model". In: *Global Biogeochem. Cycles* 9.4, pp. 471–490. ISSN: 1944-9224. DOI: 10.1029/95GB02432. URL: http://dx.doi.org/10.1029/95GB02432.

Woolhiser, D. A. and G. G. S. Pegram (1979). "Maximum Likelihood Estimation of Fourier Coefficients to Describe Seasonal-Variations of Parameters in Stochastic Daily Precipitation Models". In: *J. Appl. Meteorol.* 18.1, pp. 34–42. ISSN: 0894-8763. DOI: Doi10.1175/1520-0450(1979)018<0034:Mleofc>2.0.Co;2.

Woolhiser, D. A. and J. Roldan (1982). "Stochastic Daily Precipitation Models: 2. A Comparison of Distributions of Amounts". In: *Water Resour. Res.* 18.5, pp. 1461–1468. ISSN: 0043-1397. DOI: DOI10.1029/WR018i005p01461.

Woolhiser, D. A. and José Roldán (1986). "Seasonal and Regional Variability of Parameters for Stochastic Daily Precipitation Models: South Dakota, U.S.A". In: *Water Resour. Res.* 22.6, pp. 965–978. ISSN: 00431397. DOI: 10.1029/WR022i006p00965.

Chapter 6

pyleogrid

An Ensemble method for Gridding Paleo Proxy Climates

6.1 Introduction

- Why gridding
  - Data-Model-Intercomparisons)
  - Easier to handle
  - Energy balance, compatibility with models, grid cell (Area) averages
  - Stability of observation network through time
  - Not just spatial grid, also regular timestep (problems with pseudo gridding, including time 'windows' or 'slices'
  - Understanding past climates, different forcings – independent of models (difficult from point-cloud)
  - Filling the gaps
  - Spatial scales (Samarthein chironomids etc)
- Importance of uncertainties
  - Necessary for data comparisons, interpretation skill of the data
  - Proxy climate reconstruction uncertainties are higher (inverse modelling, not always properly defined (MAT)) than for instrumental data
  - Age uncertainties can be high (centennial to multi-millennial)
- Why Tps (can also use something else, but has been used before)
  - Extrapolation to the gaps (Previous work by mauri et al and davis)
  - Pseudo-gridding (marcott, Marsicek 2018, Margo, 2009, Bartlein et al ?2013) has holes
  - Data assimiliation (pages2k? Need to look the paper up again) – depends on model
  - Bayesian data assimiliation (Weitzel 2019) – depends on model
  - Does not require interpolation of time (Marsicek 2018)
- Other
- Introducing constraints; climate, training set size, spatial coverage etc

6.2 Data

The ensemble based gridding method is adapted to paleo-climates. In this study, we describe the method using a large set of western Eurasian fossil pollen assemblages that have been transformed to summer (June, July and August) (JJA) temperatures. We focus on pollen data because it is the spatially most widely available proxy during the Holocene, but it is important to mention that the reconstruction method is agnostic to the climate proxy, because it does not explicitly use the pollen assemblages but rather alters the standard climate reconstruction method under the aspect of it's methodological uncertainties. As such, the following sections describe the fossil and modern pollen database for this use case (section 6.2.1) and the associated uncertainties of the temperature reconstruction method (section 6.2.2) and the dating of the fossil pollen samples (section 6.2.3).

6.2.1 Pollen database

The source data for this study is a subset of the latest development version of the POLNET database, a northern hemispheric, sub-tropical collection of pollen assemblages (Basil A. S. Davis and Kaplan, 2017; Sommer et al., 2019). The purpose of this database is to generate the source for large-scale climate reconstruction during the Holocene (past 12'000 years) that can be used for model-data comparisons. The subset that we use in this study to describe and develop the gridding method contains fossil and modern pollen assemblages of western Eurasia, a region that has already been under investigation in the previous study by Mauri et al., 2015.

The database contains raw pollen counts from various publicly available and private data sources, in total 1350 datasets with 80500 fossil samples. The majority of the fossil pollen data (see figure 6.1) comes from the EPD (94%), other publicly available databases , and PANGAEA. The rest has been obtained either through private communications from the author, the private database that has also been used by ibid., and 2 sites have been digitized.

38: Add reference. Binney, Cao et al., 2019, ACER database

39: Add reference. probably not possible...

The modern calibration dataset (XXXX samples, see figure 6.2) is mainly based on the version 2 of the EMPD (B. A. S. Davis et al., 2013).

Describe origins of modern calibration data, link back to chapter 4

6.2.2 Site-based holocene temperature estimates

A standard approach for site-based climate reconstruction from fossil pollen assemblages is the so-called modern analogue technique (MAT) (also called *k*-nearest neighbors). This technique estimates the climate of the fossil sample as the (weighted) climate average of the most similar modern samples (i.e. the closest modern analogues). It has the major advantage that it requires little parameterization efforts and can be applied over a large spatial area that covers many different climate regimes (Mauri et al., 2015). We apply this method but vary it in our probabilistic setup, such that it better represents the spatial domain of the modern analogues.

For this purpose, we follow the standard approach and assign a JJA temperature to each modern calibration sample (figure 6.2), taken from the corresponding grid cell in the WorldClim dataset, version 2 at 30 seconds (Fick and Hijmans, 2017).

In the next step every pollen assemblage is transformed from raw counts to percentages, based on the total sum of terrestrial pollen counts in the sample. In order to measure the similarity between a (transformed) fossil pollen assemblages $\{f_i\}$ and modern pollen assemblage $\{m_i\}$ with use squared-chord distance from the R package *rioja* (Juggins, 2017), defined as

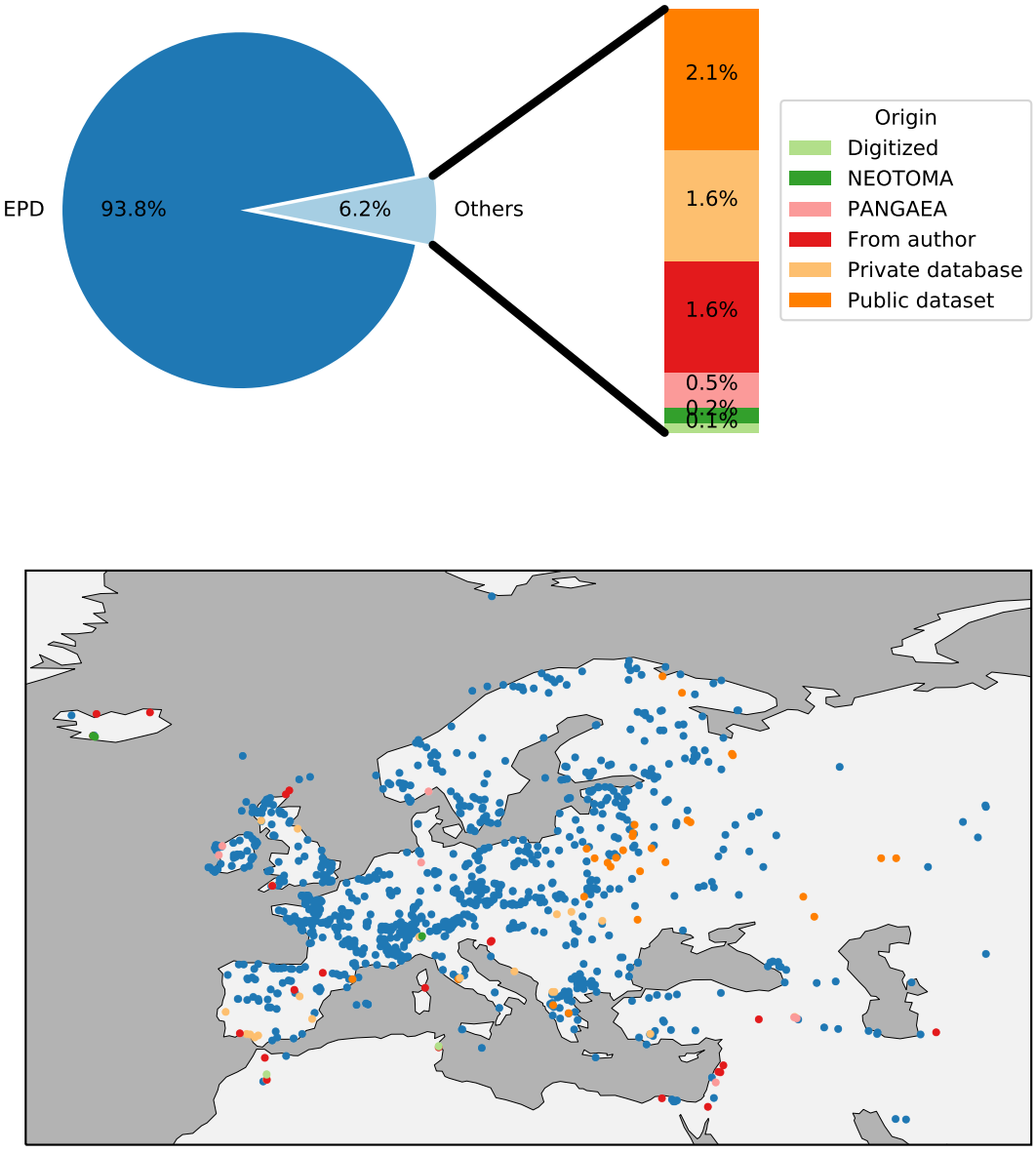


FIGURE 6.1: Data origins for the fossil POLNET database in western Eurasia (1351 in total). The majority comes from the EPD and other public datasets, and some sites where obtained through direct communication from the authors and private databases.

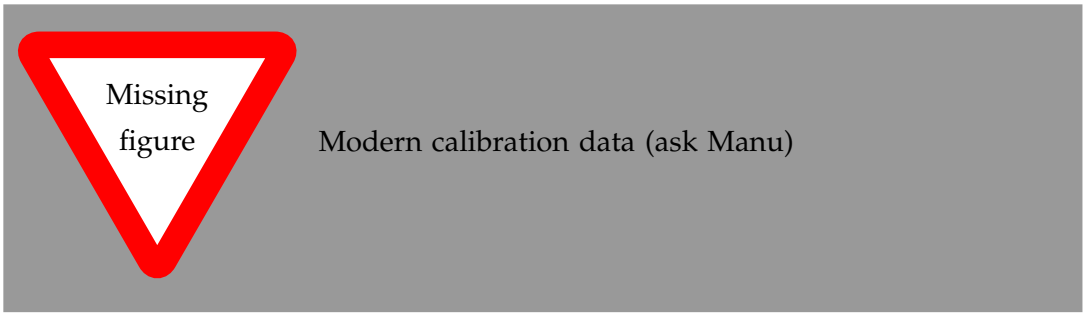


FIGURE 6.2: Modern calibration database

$$d = \sum_i \left(\sqrt{f_i} - \sqrt{m_i} \right)^2$$

This is done for every modern and fossil sample in the database. The standard, non-probabilistic setup would now compute the climate of the fossil sample as the mean climate of the k closest analogues (e.g. $k = 6$), eventually weighted by their corresponding distance d. There are many variations of this technique (see for example Birks et al., 2010, including various measures of similarity, choices about k, the maximum allowed distance d between modern and fossil assemblage, subsampling of the calibration dataset to avoid spatial autocorrelation, and by grouping pollen taxa into so-called plant-functional types (PFTs) (B. A. S. Davis et al., 2003; Mauri et al., 2015, e.g.). They all, however, have in common that the categorical, multimodal distribution of the climate of the modern analogues is oversimplified into a unimodal distribution represented by the mean of the analogue climates. Therefore, in our ensemble approach, we explicitly do not take the mean but sample the climate of the analogues directly. This is further discussed in the methods section 6.3.2 and 6.4.1.

40: Add reference.

41: Add reference.

42: Add reference.

43: Add reference. Guiot and de Vernal, 2011; Telford and Birks, 2009, 2005

6.2.3 Age uncertainties

In addition to the methodological uncertainties of the climate reconstruction method (previous section 6.2.2), we provide a framework to handle dating uncertainties. During the gridding step (see next section 6.3.3), every sample is weighted by the age difference to the target reconstruction age. The previous studies by B. A. S. Davis et al., 2003 and Mauri et al., 2015 do not take this uncertainty, that can be as high as multiple centuries, into account although they influence the gridded temperature reconstruction.

The reason is a systematic problem of pollen samples that we overcome here with the recent developments in the pollen community. In palynology, each sample in a sediment core is is dated using a so-called age-depth model, a function that maps each depth of the sediment core to an age. This function is based on a few chronological control points where the age has been determined instrumentally (for lake sediments in the Northern Hemisphere, these are commonly radiocarbon (^{14}C dates) and interpolates/extrapolates to the depths of the sample locations. Various methodologies exist to define these age-depth models, ranging from simple linear interpolation methods (Bennett, 1994) to the more recently developed bayesian techniques of the Bchron (Haslett and Parnell, 2008) and BACON (Blaauw and Christen, 2011) models.

The early approaches have been proven to provide unreliable uncertainty estimates (R. J. Telford et al., 2004) and there has been no standardized way to report the uncertainties, if they are reported at all. For this reason we (and previous studies) cannot rely on the age uncertainties reported in the pollen database. An alternative approach is to recalculate the chronology for every dataset in the database (see Goring, 2019, for instance), but this also requires parameterization for reliable uncertainties and goes beyond the scope of this study.

Instead, we follow an approach that is based on two aspects: age uncertainties are higher for older samples, and samples that are farther away from the radiocarbon dates (i.e. chronological control points). Additionally, samples behave differently if the sample is surrounded by two chronological points (i.e. the sampe age is interpolated) or not (sample age is extrapolated). These relationships are illustrated in figure **??**, based on all datasets (ca. 30'000 samples) from the Neotoma paleoecology database (Williams et al., 2018) that have age-depth models estimated with BACON, a model that has been proven to provide more reliable age uncertainty estimates (Trachsel and Richard J Telford, 2016). These uncertainties in Neotoma are reported as two sigma confidence intervals of lower and upper sample age bounds, but for the sake of implementation (section 6.3.1 assumes a normal distribution), we use the one sigma uncertainty of the maximum of the two deviations. The grayscale density plots in the background shows the high dispersal of the data and the number of samples decreases strongly with higher distance to the control point or older samples (red lines). Nonetheless, the mean of the data (blue lines) reveals the increasing nature of both relationships, as mentioned before.

Figure **??** also shows two models that have been fitted to the data. The first one is a standard simple univariate linear model $y = a + b \cdot x$ (orange line). This model already simulates the increasing trend of both variables although it does not capture the non-linear relationship between age and age-uncertainty. A possible reasons for this non-linearity might be the time-dependency of the radiocarbon calibration curve and it's associated errors. This gives the motivation to use a constrained linear Generalized Additive Model (GAM), a smooth semi-parametric model of the form

$$\mathbb{E}[y|X] = \beta_0 + f_1(X_1)$$

in the univariate case, or

$$\mathbb{E}[y|X] = \beta_0 + f_1(X_1) + f_2(X_2)$$

in the bivariate case. The feature functions $f_{1,2}$ are based on penalized B splines with a constraint for monotonic increasing, $\mathbb{E}[y|X]$ is based on a normal distribution and has been fitted with the *pyGAM* software package (Servén et al., 2018). This model enables to better simulate the non-linear features as can be seen with the green lines in figure 6.3.

These results approve the initial hypotheses and justify the choice of a bivariate Generalized Additive Model (GAM) for predicting age uncertainties based on the distance to the chronological control point, and the age of the sample. The two models, together with a bivariate simple linear regression model, and again for interpolated and extrapolated samples, are shown in the central column of figure 6.4. Both model classes (simple linear and GAM) are able to reproduce the general shape of the observed data, although the GAM better resolves the non-linear relationship between the three variables.

The final uncertainties, predicted for the data set presented in the previous section 6.2.1, are shown in the supplementary figure 6.7.

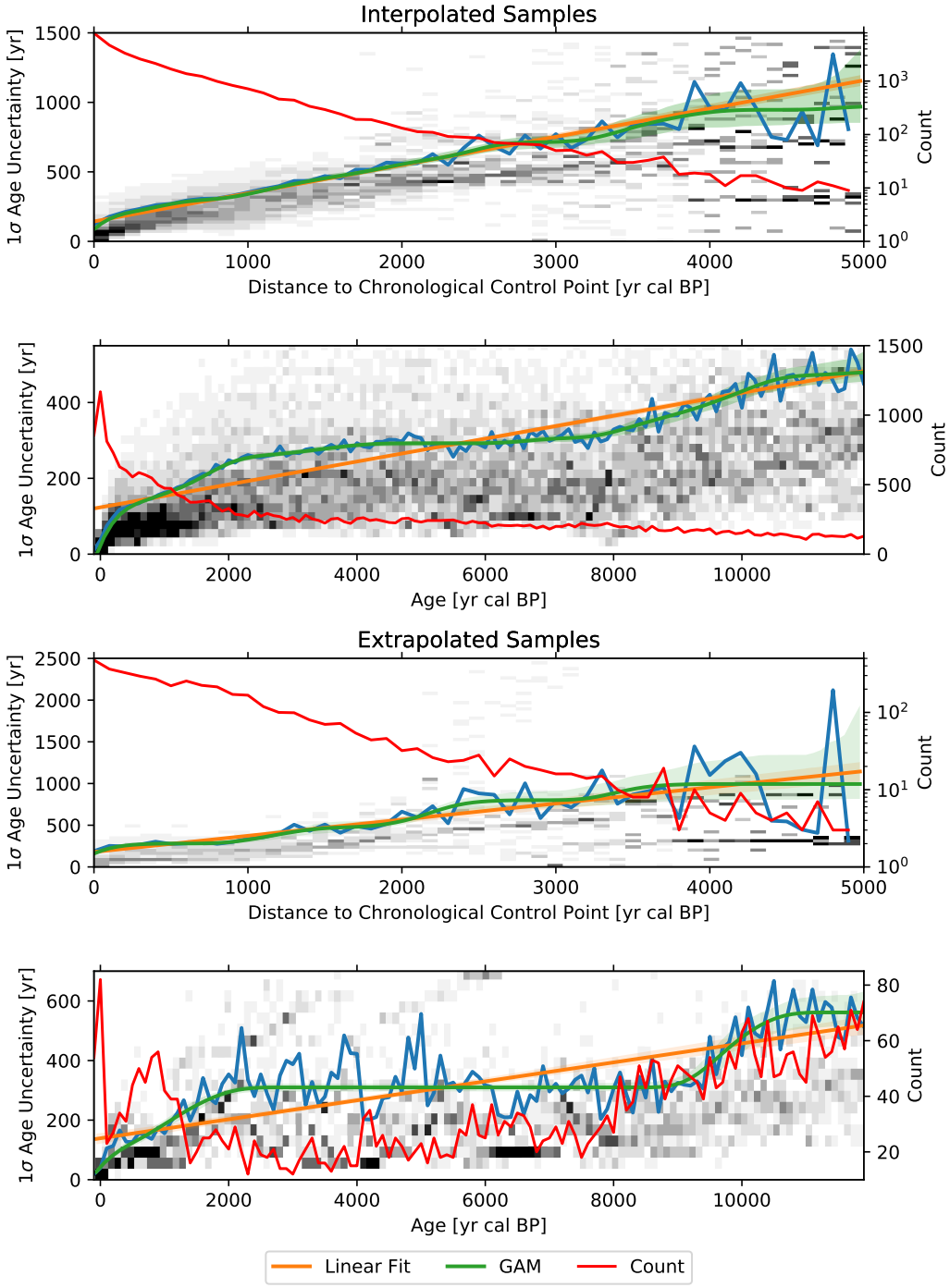


FIGURE 6.3: Univariate regression plots of (first and third) distance to chronological points, and (second and fourth) age to the one sigma dating uncertainty of the sample. The upper two plots contain only interpolated samples (i.e. samples that lie between two chronological control points), the lower extrapolated samples. Blue lines show the mean age uncertainty for the given distance (age). Orange and green lines show the linear and GAM fits of distance (age) to age uncertainty, and red lines show the number of samples for a given distance (age). The grayscale plot in the background shows a two-dimensional histogram (density plot) to illustrate the underlying data of the fits. For the purpose of a better visualization, each vertical bin of this histogram has been normalized to one. Means, counts and histogram are all based on 100 year bins in distance (age). The fits are estimated based on the unbinned data, the source data are all Neotoma datasets with BACON-based age-depth models. Note the logarithmic scale of the right count axis on the first and third plot.

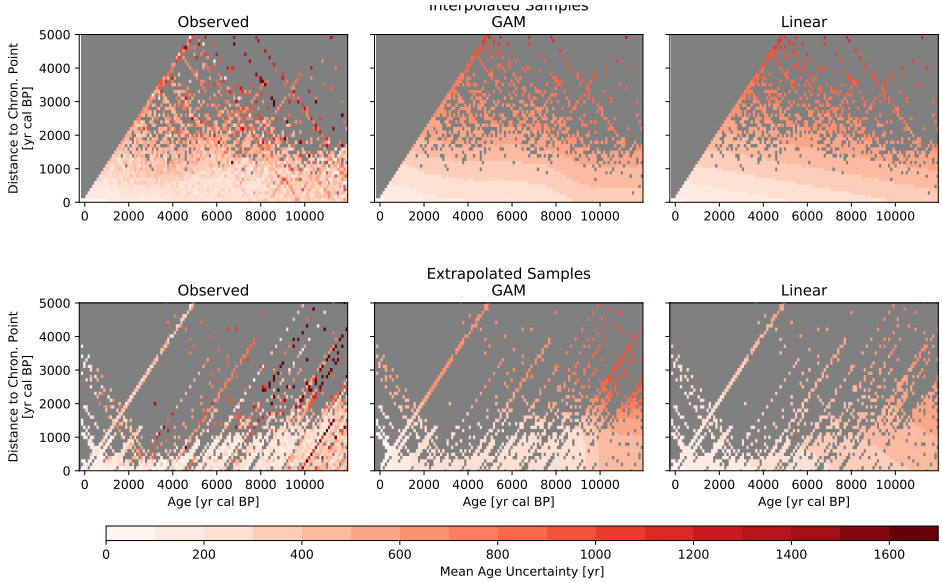


FIGURE 6.4: Bivariate models of age uncertainty. The top row shows interpolated samples (i.e. samples that lie between two chronological control points), the bottom row extrapolated samples. Plots in the left column show the observed data (samples of the Neotoma database with BACON-based age-depth models), central and right columns show the simulations of bivariate linear GAMs or bivariate linear regression models respectively. y-axes are the distance to the closest chronological control point, x-axes are the age of the sample, both binned into 100-years intervals. The color coding of each 100 by 100 years grid cell is based on the mean age uncertainty of all samples within this cell.

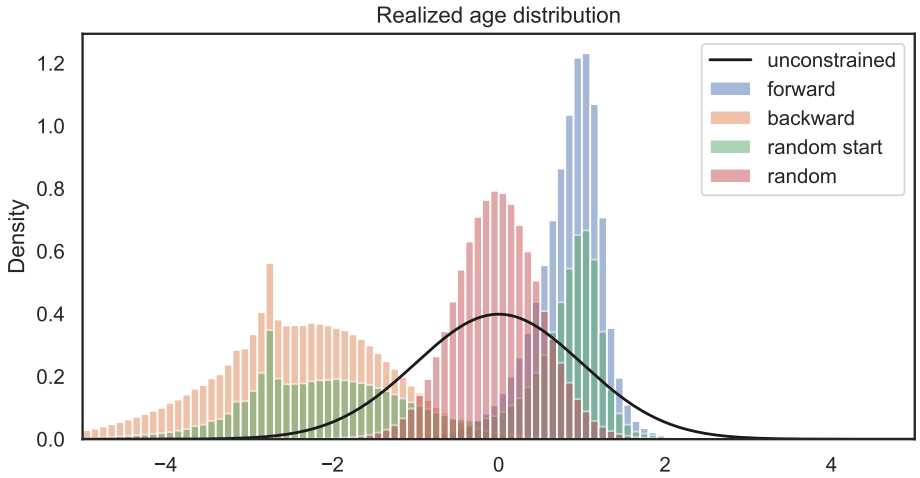


FIGURE 6.5: Histograms of standardized age sampling methods for the site in section 6.4.1 with and ensemble size of 10'000. Every sampled age has been centered at the reported age or the corresponding sample and scaled by its age uncertainty. The black line shows the unconstrained distribution (a standard normal with a standard deviation of 1), the other histograms show the realized distributions for each of the age sampling methods (section 6.3.1).

6.3 Method

With the intrinsic methodological uncertainties of climate and dating in mind, we present a new ensemble-based approach on gridding the reconstructions from the individual sites. Each ensemble member is generated with an randomized sample ages and climate, derived from the corresponding uncertainty measures (see previous sections 6.2.2 and 6.2.3), with additional constrains arising from the integrity of the individual dataset (sediment core). We explain these in more details in sections 6.3.1 and 6.3.2. The final gridding step for each ensemble member is based on a modified setup of Mauri et al., 2015, but can also be extended with other interpolation algorithms, as described in section 6.3.3). We implemented the method as the python package *pyleogrid* that efficiently scales to large datasets and ensemble sizes, and shortly describe it in section 6.3.4.

6.3.1 Constrained age sampling

Every dataset has an intrinsic monotonicity constrain that the sample deeper down the core has an older age. An inversion of this constrain is very rare and is usually visible in the stratigraphy of the core, such that affected samples are ruled-out before. As such, a classic unconstrained sampling of ages[1] using a normal distribution centered at reported sample age and a scale corresponding to the estimated age uncertainty (section 6.2.3) violates this constrain. Samples are inverted in such a case when their uncertainty intervals overlap and as such the individual ensemble member would not maintain the integrity of the individual core. We illustrate an example for such a core in section 6.4.1.

pyleogrid therefore implements different variants of this constrain.

[1]We call it the unconstrained distribution for convenience, but keeping in mind that every sampled age has to be older than -70 yr cal BP.

The intuitive approach

The most intuitive approach is to randomly draw a sample age and constrain the age of the neighboring sample with it. This can be done in a *forward* manner, such that every older sample has to be older than the previous younger sample, or in a backward manner, i.e. the younger sample has to be younger than the neighboring older sample. We will show in the paragraphs below that this method is not working, nevertheless we mention it here because of the intuitivity of the approach and because the reason for the failure is non-trivial.

As such, we demonstrate three different algorithms:

forward Starting with an unconstrained age distribution for the youngest sample in the core, every consecutive sample has to be older than the previous (i.e. the method works forward in age, but backward in time

backward Starting with an unconstrained age distribution for the oldest sample in the core, every consecutive sample has to be younger than the previous (i.e. the method works backward in age, but forward in time)

random start Starting with an unconstrained age distribution of a random sample in the core, we apply the *backward* algorithm for younger and *forward* algorithm for younger samples.

As such, *forward* and *backward* algorithms always start with an unconstrained age distribution of the youngest (oldest) sample for every ensemble member. Within the *random start* algorithm, every sample gets the chance to start with an unconstrained age distribution, because the starting point is random for every ensemble member. The constrained age distributions for the consecutive samples are implemented as truncated normal distributions.

The resulting age distributions from the three algorithms are shown in figure 6.5, together with another method, that is described later in this section. The figure shows the sampled age distributions by the various above-mentioned sampling methods for the site described in section 6.4.1. To make these age distributions comparable, we transformed them to a standard normal distribution (visualized as the unconstrained distribution in figure 6.5) prior to visualization, by subtracting the reported age and dividing by the estimated age uncertainty of the corresponding sample. It is obvious from this figure that all of the above-mentioned algorithms produce an artificial bias to the age distribution. The *forward* approach pushes the samples to the upper tail of the distribution, the *backward* approach pushes everything to the lower tail. The *random start* method produces a bimodal distribution with peaks at the upper and lower tail.

This is also shown with three exemplary samples from the site in the supplementary figure 6.8. The forward method works well for the young sample but pushes all older samples to the upper tail of their distribution, The backward method does the opposite and the random sort method creates a bimodal distribution for the sample in the center of the core, and backward behaves like the forward (backward) algorithm at the older (younger) part of the core.

We explain this initially unexpected results with the overlapping age uncertainties in the core. The site that we describe here has 110 samples. As such, the probability that one sample draws a random age at the lower or upper tail of the distribution is very high. Now, most of the dating uncertainty intervals overlap and this forces all the consecutive samples to the tail of their age distributions. Another problem,

FIGURE 6.6: Realized age distribution for the entire dataset (section 6.2.1) with the *random* method (section 6.3.1). The individual sample distributions have been centered and scaled as in figure 6.5.

that is not shown here, arises from the differing sizes of the age uncertainties which highly depends on the distance to the chronological control point (see section 6.2.3). This can also lead to unsatisfiable requirements, if one sample is close to a control point (and as such has a lower age uncertainty) and the previous sample has been pushed far outside of the 95% confidence interval.

The random sorting approach

These strong biases of the intuitive approach led to another method, that we also show in red in figure 6.5 and supplementary figure 6.8, the *random* method. This method consists of two steps: in the first step we draw random age for each sample based on its unconstrained distribution[1]. In the second step, we order these random ages while maintaining the order of samples in each dataset. As such, we assign an age to each sample that is not necessarily drawn from its own distribution, but rather from the one of a neighboring sample. When samples overlap, this then truncates the tails of realized distribution and effectively decreases the reported age uncertainty, as can be seen in the figures 6.5, 6.8 and for the full dataset in figure 6.6. This approach violates the common approach that each sample has a unique confidence interval that it needs to explore and as such might introduce some hidden biases in the sampled distributions. Nevertheless, the algorithm is very fast and much closer to the desired joint distribution, than the previous *intuitive* approach.

The Gibbs sampling approach

The biases of the above-mentioned algorithms led to the development of a Markov chain Monte Carlo (MCMC) sampling algorithm. A standard MCMC approach, is to draw a set of random ages for all unconstrained sample distributions in a core at once, until this set of ages satisfies the monotonicity criterion. This is described with the following pseudo-code:

Algorithm 2 Classic MCMC approach

for *dataset* in *datasets* **do**
  Set $i = 0$
  Set $\vec{\mu}$ as vector reported ages in *dataset*
  Set $\vec{\sigma}$ as estimated age uncertainties
  Set $\vec{ages}$ to be of length $\vec{\mu}$
  while $i < 1$ **or not** $is\_monotonic(\vec{ages})$ **do**
    $\vec{ages} = \mathcal{N}(\vec{\mu}, \vec{\sigma}^2)$
    Set $i = i + 1$
  end while
end for

This standard approach however did not find a monotonic solution within ten million iterations for a high-resolution site such as it has been presented in the previous section. Therefore we decided to implement a Gibbs sampler, an algorithm that is commonly used in Bayesian inference.

Describe Gibbs sampler

6.3.2 Temperature sampling

- Sampling of analogue climates weighted by squared chord distance
- Can also use different methods (WAPLS, etc.)
- Random starting point
- Mask analogues that have a temperature distance more than 5°C??? From the previous/earlier sample – test this

As mentioned in section 6.2.2, our method alters the standard modern analogue technique (MAT) approach such that it does not use the weighted average, as it is commonly estimated, but

continue

6.3.3 Gridding

- 3D of climate (not anomaly)
- Distance in time through weighting
- Paleo-Elevation (ICE-6G)
- Can use any other method than tps

6.3.4 Implementation

6.4 Results

6.4.1 Site-based realized climate reconstruction: a use-case

Reconstruction of entity 11390

6.5 Discussion

- Maps of reconstruction
  - How high are the uncertainties
- Questions:
  - How does the uncertainty evolve with distance to the samples
    - At map boundaries
    - In gaps within the data
- How consistent is the interpolation uncertainty (from Tps) within the ensemble?

6.6 Conclusions

Supplementary material

6.A Estimated age uncertainties

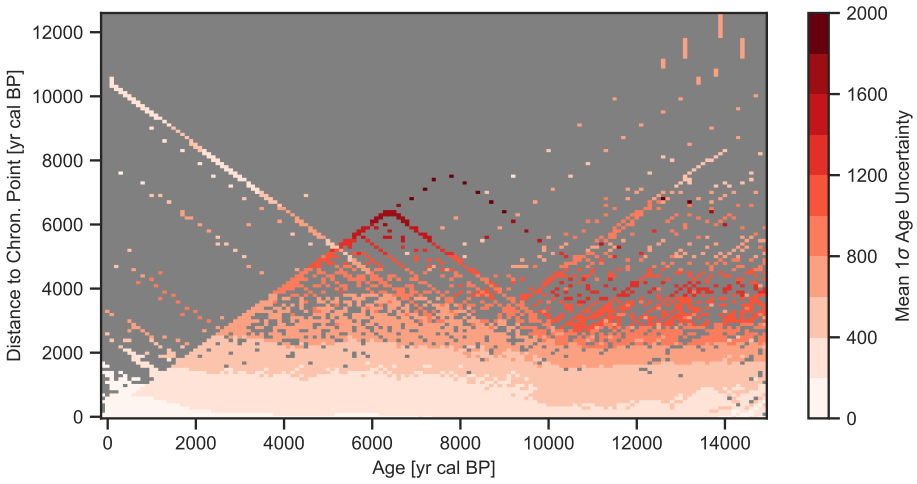


FIGURE 6.7: Estimated age uncertainties for the Eurasian dataset from section 6.2.1 with the same formatting as in figure 6.4.

6.B Example of generated age distributions

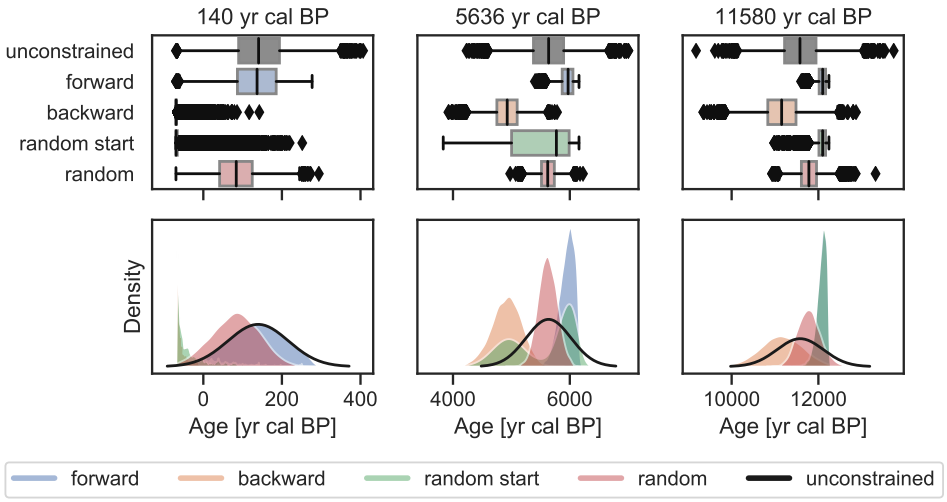


FIGURE 6.8: Example of three samples from the site in section 6.4.1 and their realized distributions. Sampling algorithms are explained in section 6.3.1. Top plots show the box plots of the realized distribution that are visualized with a kernel density estimate in the lower row.

References

Bennett, K.D. (1994). "Confidence intervals for age estimates and deposition times in late-Quaternary sediment sequences". In: *The Holocene* 4.4, pp. 337–348. DOI: 10.1177/095968369400400401. eprint: https://doi.org/10.1177/095968369400400401. URL: https://doi.org/10.1177/095968369400400401.

Birks, H. John B., Oliver Heiri, Heikki Seppä, and Anne E. Bjune (2010). "Strengths and Weaknesses of Quantitative Climate Reconstructions Based on Late-Quaternary Biological Proxies". In: *The Open Ecology Journal* 3.1, pp. 68–110. DOI: 10.2174/1874213001003020068.

Blaauw, Maarten and J. Andrés Christen (2011). "Flexible paleoclimate age-depth models using an autoregressive gamma process". In: *Bayesian Analysis* 6.3, pp. 457–474. ISSN: 1931-6690. DOI: 10.1214/11-ba618.

Davis, B. A. S., S. Brewer, A. C. Stevenson, and J. Guiot (2003). "The temperature of Europe during the Holocene reconstructed from pollen data". In: *Quat. Sci. Rev.* 22.15-17, pp. 1701–1716. ISSN: 02773791. DOI: 10.1016/s0277-3791(03)00173-2.

Davis, B. A. S., M. Zanon, P. Collins, A. Mauri, J. Bakker, D. Barboni, A. Barthelmes, C. Beaudouin, A. E. Bjune, E. Bozilova, R. H. W. Bradshaw, B. A. Brayshay, S. Brewer, E. Brugiapaglia, J. Bunting, S. E. Connor, J. L. de Beaulieu, K. Edwards, A. Ejarque, P. Fall, A. Florenzano, R. Fyfe, D. Galop, M. Giardini, T. Giesecke, M. J. Grant, J. Guiot, S. Jahns, V. Jankovska, S. Juggins, M. Kahrmann, M. Karpinska-Kolaczek, P. Kolaczek, N. Kuhl, P. Kunes, E. G. Lapteva, S. A. G. Leroy, M. Leydet, J. Guiot, S. Jahns, V. Jankovska, S. Juggins, M. Kahrmann, M. Karpinska-Kolaczek, P. Kolaczek, N. Kuehl, P. Kunes, E. G. Lapteva, S. A. G. Leroy, M. Leydet, J. A. L. Saez, A. Masi, I. Matthias, F. Mazier, V. Meltsov, A. M. Mercuri, Y. Miras, F. J. G. Mitchell, J. L. Morris, F. Naughton, A. B. Nielsen, E. Novenko, B.

Odgaard, E. Ortu, M. V. Overballe-Petersen, H. S. Pardoe, S. M. Peglar, I. A. Pidek, L. Sadori, H. Seppa, E. Severova, H. Shaw, J. Swieta-Musznicka, M. Theuerkauf, S. Tonkov, S. Veski, W. O. van der Knaap, J. F. N. van Leeuwen, J. Woodbridge, M. Zimny, and J. O. Kaplan (2013). "The European Modern Pollen Database (EMPD) project". In: *Vegetation History and Archaeobotany* 22.6, pp. 521–530. ISSN: 0939-6314. DOI: 10.1007/s00334-012-0388-5. URL: http://link.springer.com/article/10.1007/s00334-012-0388-5.

Davis, Basil A. S. and Jed O. Kaplan (2017). *HORNET Holocene Climate Reconstruction for the Northern Hemisphere Extra-tropics*. SNF-Research-Plan. last accessed Jan, 30th, 2018. URL: http://p3.snf.ch/project-169598#.

Fick, Stephen E. and Robert J. Hijmans (2017). "WorldClim 2: new 1-km spatial resolution climate surfaces for global land areas". In: *International Journal of Climatology* 37.12, pp. 4302–4315. DOI: 10.1002/joc.5086. eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/joc.5086. URL: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/joc.5086.

Goring, S.J. (2019). *Bulk Baconizing*. https://github.com/NeotomaDB/bulk-baconizing. DOI: 10.5281/zenodo.2545891.

Haslett, John and Andrew Parnell (2008). "A simple monotone process with application to radiocarbon-dated depth chronologies". In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 57.4, pp. 399–418. DOI: 10.1111/j.1467-9876.2008.00623.x. eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9876.2008.00623.x. URL: https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9876.2008.00623.x.

Juggins, Steve (2017). *rioja: Analysis of Quaternary Science Data*. R package version 0.9-21. URL: http://www.staff.ncl.ac.uk/stephen.juggins/.

Mauri, A., B. A. S. Davis, P. M. Collins, and J. O. Kaplan (2015). "The climate of Europe during the Holocene: a gridded pollen-based reconstruction and its multi-proxy evaluation". In: *Quat. Sci. Rev.* 112, pp. 109–127. ISSN: 0277-3791. DOI: 10.1016/j.quascirev.2015.01.013. URL: http://www.sciencedirect.com/science/article/pii/S0277379115000372.

Servén, Daniel, Charlie Brummitt, and Hassan Abedi (2018). *Dswah/Pygam: V0.8.0*. DOI: 10.5281/zenodo.1476122.

Sommer, Philipp S., Basil A. S. Davis, Manuel Chevalier, Jian Ni, and John Tipton (2019). "The HORNET project: applying 'big data' to reconstruct the climate of the Northern Hemisphere during the Holocene". In: *20th Congress of the International Union for Quaternary Research (INQUA)*. International Union for Quaternary Research. URL: https://app.oxfordabstracts.com/events/574/program-app/submission/94623.

Telford, R. J., E. Heegaard, and H. J. B. Birks (2004). "All age–depth models are wrong: but how badly?" In: *Quaternary Science Reviews* 23.1, pp. 1–5. ISSN: 0277-3791. DOI: 10.1016/j.quascirev.2003.11.003. URL: http://www.sciencedirect.com/science/article/pii/S0277379103003160.

Trachsel, Mathias and Richard J Telford (2016). "All age–depth models are wrong, but are getting better". In: *The Holocene* 27.6, pp. 860–869. DOI: 10.1177/0959683616675939. eprint: https://doi.org/10.1177/0959683616675939. URL: https://doi.org/10.1177/0959683616675939.

Williams, John W., Eric C. Grimm, Jessica L. Blois, Donald F. Charles, Edward B. Davis, Simon J. Goring, Russell W. Graham, Alison J. Smith, Michael Anderson, Joaquin Arroyo-Cabrales, Allan C. Ashworth, Julio L. Betancourt, Brian W. Bills, Robert K. Booth, Philip I. Buckland, B. Brandon Curry, Thomas Giesecke, Stephen T. Jackson, Claudio Latorre, Jonathan Nichols, Timshel Purdum, Robert

E. Roth, Michael Stryker, and Hikaru Takahara (2018). "The Neotoma Paleoecology Database, a multiproxy, international, community-curated data resource". In: *Quaternary Research* 89.1, pp. 156–177. DOI: 10.1017/qua.2017.105.

Chapter 7

Conclusions

- New tools that have been developed
- Quality standards of the tools
- Further development and potential usage

Appendices

Todo list

Appendix A

Computing climate-smart urban land use with the Integrated Urban Complexity model (IUCm 1.0)

Need to write chapter A

Appendix B

Publications and Conference contributions

B.0.1 Peer-reviewed

Cremades, R. and P. S. Sommer (2019). "Computing climate-smart urban land use with the Integrated Urban Complexity model (IUCm 1.0)". In: *Geoscientific Model Development* 12.1, pp. 525–539. DOI: 10.5194/gmd-12-525-2019. URL: https://www.geosci-model-dev.net/12/525/2019/.

Sommer, Philipp, Dilan Rech, Manuel Chevalier, and Basil Davis (2019). "straditize: Digitizing stratigraphic diagrams". In: *Journal of Open Source Software* 4.34, p. 1216. DOI: 10.21105/joss.01216. URL: https://doi.org/10.21105/joss.01216.

Weitzel, Nils, Sebastian Wagner, Jesper Sjolte, Marlene Klockmann, Oliver Bothe, Heather Andres, Lev Tarasov, Kira Rehfeld, Eduardo Zorita, Martin Widmann, Philipp S. Sommer, Gerd Schädler, Patrick Ludwig, Florian Kapp, Lukas Jonkers, Javier García-Pintado, Florian Fuhrmann, Andrew Dolman, Anne Dallmeyer, and Tim Brücher (2018). "Diving into the past – A paleo data-model comparison workshop on the Late Glacial and Holocene". In: *Bulletin of the American Meteorological Society*. DOI: 10.1175/bams-d-18-0169.1.

Sommer, Philipp S (2017). "The psyplot interactive visualization framework". In: *The Journal of Open Source Software* 2.16. DOI: 10.21105/joss.00363. URL: https://doi.org/10.21105/joss.00363.

Sommer, Philipp S. and Jed O. Kaplan (2017). "A globally calibrated scheme for generating daily meteorology from monthly statistics: Global-WGEN (GWGEN) v1.0". In: *Geosci. Model Dev.* 10.10, pp. 3771–3791. DOI: 10.5194/gmd-10-3771-2017.

B.0.2 Conference contributions

Sommer, P. S., B. A. S. Davis, and M. Chevalier (2019a). "Github and Open Research Data; an example using the Eurasian Modern Pollen Database". In: *EGU General Assembly Conference Abstracts*. Vol. 21. EGU General Assembly Conference Abstracts, p. 5669. URL: https://meetingorganizer.copernicus.org/EGU2019/EGU2019-5669.pdf.

Sommer, Philipp S., Basil A. S. Davis, Manuel Chevalier, Jian Ni, and John Tipton (2019b). "The HORNET project: applying 'big data' to reconstruct the climate of the Northern Hemisphere during the Holocene". In: *20th Congress of the International Union for Quaternary Research (INQUA)*. International Union for Quaternary Research. URL: https://app.oxfordabstracts.com/events/574/program-app/submission/94623.

Sommer, P. S. (2018). "Psyplot: Interactive data analysis and visualization with Python". In: *EGU General Assembly Conference Abstracts*. Vol. 20. EGU General Assembly Conference Abstracts. Provided by the SAO/NASA Astrophysics Data System, p. 4701. URL: http://adsabs.harvard.edu/abs/2018EGUGA..20.4701S.

Sommer, P. S., B. A. S. Davis, and M. Chevalier (2018a). "STRADITIZE: An open-source program for digitizing pollen diagrams and other types of stratigraphic data". In: *EGU General Assembly Conference Abstracts*. Vol. 20. EGU General Assembly Conference Abstracts. Provided by the SAO/NASA Astrophysics Data System, p. 4433. URL: http://adsabs.harvard.edu/abs/2018EGUGA..20.4433S.

Sommer, Philipp S., Manuel Chevalier, and Basil A. S. Davis (2018b). "STRADITIZE: An open-source program for digitizing pollen diagrams and other types of stratigraphic data". In: *AFQUA - The African Quaternary*. Nairobi (Kenya): AFQUA. URL: https://afquacongress.wixsite.com/afqua2018.

Sommer, P. and J. Kaplan (2017). "Quantitative Modeling of Human-Environment Interactions in Preindustrial Time". In: *PAGES OSM 2017, Abstract Book*, pp. 129–129.

Sommer, P. (2016). "Psyplot: Visualizing rectangular and triangular Climate Model Data with Python". In: *EGU General Assembly Conference Abstracts*. Vol. 18. EGU General Assembly Conference Abstracts. Provided by the SAO/NASA Astrophysics Data System, p. 18185. URL: http://adsabs.harvard.edu/abs/2016EGUGA..1818185S.

Sommer, P. and J. Kaplan (2016a). "Fundamental statistical relationships between monthly and daily meteorological variables: Temporal downscaling of weather based on a global observational dataset". In: *EGU General Assembly Conference Abstracts*. Vol. 18. EGU General Assembly Conference Abstracts. Provided by the SAO/NASA Astrophysics Data System, EPSC2016–18183. URL: http://adsabs.harvard.edu/abs/2016EGUGA..1818183S.

— (2016b). "Fundamental statistical relationships between monthly and daily meteorological variables: Temporal downscaling of weather based on a global observational dataset". In: *Workshop on Stochastic Weather Generators*. Vannes (France): University of Bretagne Sud. URL: https://www.lebesgue.fr/content/sem2016-climate-program.

Appendix C

New Software Tools - An Overview

This section mainly contains the latest version of the package, a short summary and an information table about where to find everything (Documentation, source code, etc.)

Need to write chapter C

C.1 Main packages

- psyplot
  - psy-simple
  - psy-maps
  - psy-reg
  - psyplot-gui
  - psy-strat
- straditize
- gwgen
- iucm
- EMPD
  - EMPD-admin
  - EMPD-viewer
  - EMPD-data
- POLNET
  - POLNET-viewer
  - POLNET-data

C.2 Other packages

- docrep
- sphinx-nbexamples
- model-organization
- funcargparse
- autodocsumm