Constructing Visualization Tools and Training Resources to Assess Climate Impacts on the Channel Islands National Marine Sanctuary

Technical Documentation

by

Olivia Holt
Diana Navarro
Patricia Park

Committee in Charge:

Dr. Samantha Stevenson-Michener
Dr. Carmen Galaz García

May 2024

Client:

Dr. Samantha Stevenson-Michener

Bren School of Environmental Science & Management

Development of a Web Application and Training Toolkit for Climate Impacts on the Channel Islands National Marine Sanctuary

As developers of this Capstone Project documentation, we archive this documentation on the Bren School's website such that the results of our research are available for all to read. Our signatures on the document signify our joint responsibility to fulfill the archiving standards set by the Bren School of Environmental Science & Management.

____

Olivia Holt

____

Diana Navarro

____

Patty Park

The Bren School of Environmental Science & Management produces professionals with unrivaled training in environmental science and management who will devote their unique skills to the diagnosis, assessment, mitigation, prevention, and remedy of the environmental problems of today and the future. A guiding principle of the School is that the analysis of environmental problems requires quantitative training in more than one discipline and an awareness of the physical, biological, social, political, and economic consequences that arise from scientific or technological decisions.

The Capstone Project is required of all students in the Master of Environmental Data Science (MEDS) Program. The project is a six-month-long activity in which small groups of students contribute to data science practices, products, or analyses that address a challenge or need related to a specific environmental issue. This MEDS Capstone Project Technical Documentation is authored by MEDS students and has been reviewed and approved by:

____

Samantha Stevenson-Michener

____

Carmen Galaz García

____

Date

Table of contents

1. Abstract

The Channel Islands Marine Sanctuary (CINMS) comprises 1,470 square miles surrounding the Northern Channel Islands: Anacapa, Santa Cruz, Santa Rosa, San Miguel, and Santa Barbara, protecting various species and habitats. However, these sensitive habitats are highly susceptible to climate-driven 'shock' events which are associated with extreme values of temperature, pH, or ocean nutrient levels. A particularly devastating example was seen in 2014-16, when extreme temperatures and changes in nutrient conditions off the California coast led to large-scale die-offs of marine organisms. Global climate models are the best tool available to predict how these shocks may respond to climate change. To better understand the drivers and statistics of climate-driven ecosystem shocks, a 'large ensemble' of simulations run with multiple climate models will be used. The objective of this project is to develop an educational toolkit and Python-based web application to visualize ecologically significant climate variables near the CINMS. The web application will be used by researchers from the University of California, Santa Barbara (UCSB) to analyze climate model output, and by CINMS staff to develop new indicators of shocks to marine ecosystems. The educational toolkit will be part of the ongoing development of the UCSB Bren School's Climate DataLab training environment.

2. Executive Summary

In order to address the increased vulnerability from the climate-induced 'shock' events it is important to understand the factors that drive environmental impacts around the Channel Islands.

A collaborative project between the University of California, Santa Barbara (UCSB) and CINMS, funded by the National Oceanic and Atmospheric Association (NOAA), aims to develop climate-based indicators to enhance understanding and management of the CINMS. This MEDS capstone project focuses on creating a set of instructional code materials and an interactive web application to support CINMS staff and UCSB researchers.

The approach begins with utilizing cloud computing resources to access climate model output, to facilitate reproducibility and avoid large dataset downloads. After data exploration, relevant variables are selected to create visualizations including maps, time series, and vertical profiles tailored to the CINMS area. The visualizations are presented through a public Github repository with tutorials on data access, plotted variables, and additional exploration tips. Additionally, an interactive web application was created by integrating code from the educational toolkit. The web application results will inform development of climate indicators by CINMS staff and be integrated into the Climate DataLab educational training environment developed and maintained by the client.

3. Problem Statement

The Channel Islands houses a wealth of marine and land biodiversity that holds many plants and animals species, in which, about 145 of these species can be found only at the Channel Islands. Because of its unique locations, most of these species can only survive on the island with special ecological conditions that these species have adapted to. Unfortunately, external factors, such as climate change, caused by human activities, are changing the environment that these species thrive in. Understanding the physical mechanisms driving impacts in the Channel Islands is critical to management and conservation.

A newly NOAA-funded collaboration between UCSB and CINMS is using information from large ensembles of climate model simulations to determine the range of possible future climate-driven 'shocks' to marine ecosystems. Listed on the NOAA's website contains a webpage on the science needs assessment, including information relevant to the Channel Islands. Listed on this website are data and analysis needs such as "predicted impacts to sanctuary resources over short and long term" (NOAA, 2024) and "Biotic and abiotic tipping points relevant to biological communities" related to the Channel Islands region (NOAA, 2024). Currently, the Channel Islands National Marine Sanctuary staff do not have easy access to these climate outputs that would help them visualize these potential impacts, nor do they have a way to visualize the effect of climate change on specific ecological variables that influence the overall health of the Channel Islands.

In order to project future climate change impacts, researchers need to work with these climate models to achieve these visualizations. However, there is a lack of information on how to work with this type of data, focusing on the Channel islands. Few comprehensive textbooks or tutorials have been published for users interested in working with climate models to get started. The goal of this MEDS capstone project is to support the CINMS staff in the development of new, quantitative climate-based indicators, bringing together place-based knowledge with ecological insights, as well as physical climate information. This will be the first of its kind creating climate based observations for the Channel Islands sanctuary.

4. Specific Objectives

The goal of this capstone project is to help facilitate the process of understanding potential risks towards the Channel Islands with tutorials and visualizations using climate data. The tutorials will make creating visualizations for marine ecosystem indicators more accessible to those not familiar with working in python or with large ensemble data. This goal can be broken down into two main objectives:

Preparing easy to follow educational Python tutorials for UCSB researchers and other interested users, based on data from a large ensemble of Community Earth System Model version 1 (CESM1.0) simulations.

Creating a web application to host the data visualizations created from the Python tutorials for Channel Islands National Marine Sanctuary staff to view the water parameters of interest.

5. Summary of Solution Design

5.1 Description of data

A large ensemble is a set of simulations run with a given climate model which only varies the initial conditions. In this project the large ensemble climate model data used is CESM1.0 which is able to give historical, present and future readings of the outcome variables (Kay, J. E., et al., 2015).

In order to retrieve the data from a large ensemble, Pangeo's framework for accessing data was used. Pangeo is an open source community project that utilizes geoscience packages to promote the use of *Big Data.* Big data refers to large ensembles of climate simulations using models like CESM1.0, as well as other similar models. More information on how Pangeo was utilized to access the data will be provided in section 8.3.1.

5.2 Data Access

Access to the CESM1.0 dataset involves several steps to retrieving specific experiments, temporal averaging frequencies, and variables. A main issue that concerned the team in the beginning of this project was the method of data access. Given the large amount of data, downloading files locally was not a viable option, but if the user uses a high computing system (HPC), NetCDF files are available for download through the Climate Data Gateway hosted by NCAR. We did not pursue this option for ease of reproducibility between novice researchers and climate scientists. By referencing past tutorials in Pangeo Gallery, data was sourced from Amazon Web Services(AWS) using the intake package. Once the url was retrieved using the intake package, the team loaded the data catalog into a dictionary of xarray datasets. There are multiple options for choosing which data to download, and the team used 'search' argument to select frequency, component, experiment, and variable. These options will be discussed in detail in section 8.3 User Documentation.

As each simulation from the CESM1 large ensemble is very large (ranging from about 20 GB each) and the ensemble contains 40 simulations in total, the team decided that choosing the maximum, minimum, and mean of these ensembles would be the best practice for our visualizations. At the request of the client, the team settled on two distinct experiments: 20C runs (1920 - 2000), and RCP 8.5 high-emissions simulation of future climate change runs (2000 - 2100). The selection of variables depends on their availability in the catalog and the level of interest to the Channel Islands' sanctuary staff. The variables that meet both criteria are dissolved oxygen, temperature, and salinity. These are a few of the parameters that are of primary concern to the sanctuary staff as they can indicate potential ecological issues stemming from climate change and rising global temperatures.

Section 8.3 User Documentation will lightly touch on this as well, but the team would like to acknowledge that Pangeo tutorials are referenced to source the data from Amazon web services.

5.3 Educational Toolkit

A portion of the deliverables for the project involves creating three educational Python tutorials that help the user interact with the CESM database. These tutorials produce three different types of visualizations: time series, maps, and vertical profiles. They demonstrate how to access the data, understand the structure, and create a visual output of the time frame and environmental variable the user is interested in. Each notebook follows a template structure created by the team for consistency and understanding. Because these notebooks aim to serve as educational resources, they emphasize the procedural steps leading to the final output, particularly the visualizations.

These notebooks focus on the Channel Islands Marine sanctuary, so the geographic coordinates are narrowed down to the region surrounding and this includes Southern California. Visual representations enable the audience to properly understand the spatial distribution of climate factors.

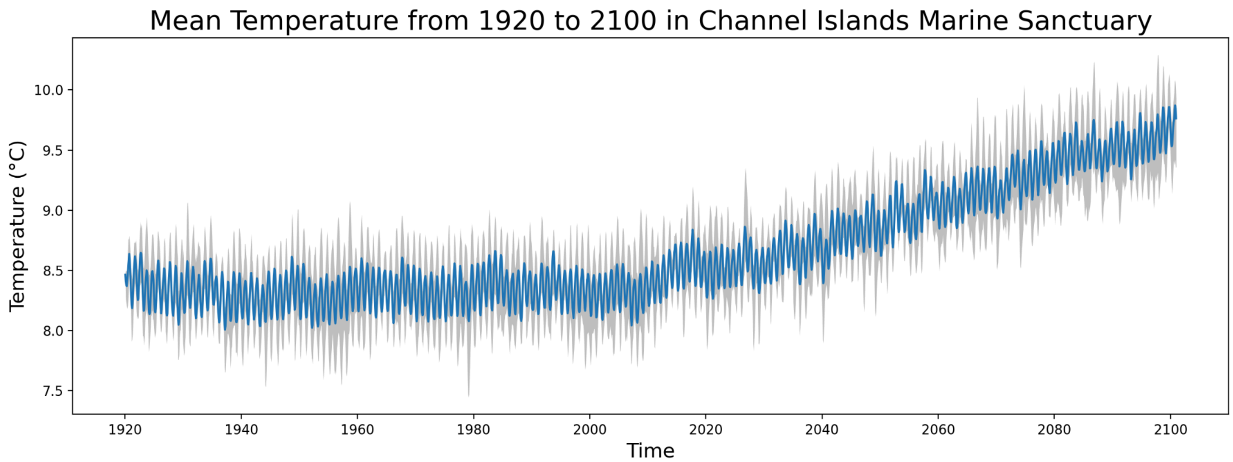


Figure 1. Time Series of Temperature of Channel Islands Marine Sanctuary

A section of the final deliverables includes a [GitHub repository]() that hosts the toolkit. This repository includes a detailed tutorial on data access with Pangeo, variable visualization, and data exploration tips to understand the dataset. The repository is contained in the [Channelislanders GitHub organization]() and is publicly available to anyone interested in the research. The README file of the GitHub repository discusses the set up of the environment and package versions required to run the code. This repository contains all elements and necessary documentation within the notebooks to complete the toolkit and to allow for reproducibility and manipulation of each element in the notebook.

5.3.1 Data Access

These notebooks demonstrate data accessibility through Amazon Web Services. This section of the notebooks will include the url, retrieval using the intake package, and dictionary search of the data catalog. It is important to highlight that while we have a distinct section dedicated to data exploration, we encourage the user to constantly review the data they work with after subsetting or wrangling any part of the data. These notebooks will demonstrate how the team achieved this understanding.

5.3.4 Data Exploration

Exploring data before wrangling is a critical step in comprehending the dataset and involves multiple stages of observing each output. The toolkit is designed to address any challenges that may arise during data manipulation.

In the toolkit, defining xarray datasets corresponding to the experiment is one of the initial steps of exploration. Each xarray is a compilation of multiple netCDF files, and within this xarray are lists of dimensions, coordinates, data variables, indexes, and attributes. The 'attributes' section is particularly useful and displays all information that allows the team to navigate through the data. This section can be interpreted as the metadata for the particular file extracted from the xarray.

5.3.3 Data Wrangling

Manipulating the data is the next process displayed in the notebooks. This varies with each notebook. The general structure of the mapping notebook for this section consist of:

1. Merging of the grid data that contains all latitude and longitude coordinates.
2. Regridding the data through `xesmf` for smoother indexing of coordinates.

The vertical profile and time series notebooks contain a slightly different structure for this section since there is no need to use the properly gridded data. The general structure for the time series notebook consists of:

1. Defining xarray datasets corresponding to the two experiments for each variable
2. Taking the mean over the member_id dimension
3. Merging the 20C and the RCP8.5 datasets together
4. Subset the dataset further to surface level data (z_t = 0)

Finally, the general structure for the vertical profile notebook consists of:

1. Defining xarray datasets corresponding to the two experiments for each variable
2. Subsetting the data to a specified latitude and longitude
3. Saving the data to an .nc file to use for plotting

The team explored all options to enhance user experience. Deciding the best approach to explore the original data was challenging.

5.3.4 Plotting

This section of the toolkits will produce the visualizations. Each notebook demonstrates how to properly plot the data and explains what the user is actually seeing.

Mapping

This section demonstrates how to calculate the mean across the time and member_id dimension. After this, the user will be able to subset the area they wish to focus on and finally plot using matplotlib package.

This area of the toolkit varies slightly depending on the variable used. In this section, the team focused on sea surface temperature. This variable only contains one depth level, so finding the mean of time and all members was only needed for this example. This will produce a map subsetting Channel Islands and surrounding waters, this data subsetted the area slightly larger than the marine sanctuary.

Time Series

Here, an example code is shown to the user, letting them know how to change various parts of the code to view other variable outputs. Included in this section is telling the user how to download their subsetted data as a .nc file, and how to read it back in using the package xarray.

Another code example is shown on how to create a graph that shows the max, mean, and min as seen in figure 1. Users are given additional links of how they an customize their plot from the matplotlib website.

Vertical Profile

This section demonstrates how to loop through each member_id to calculate the mean of temperature over the entire time frame. Then, the data is converted to an xarray DataArray and the overall mean of temperature is calculated from all member_id's. After this, the user will be able to plot using the matplotlib package.

5.4 Web Application

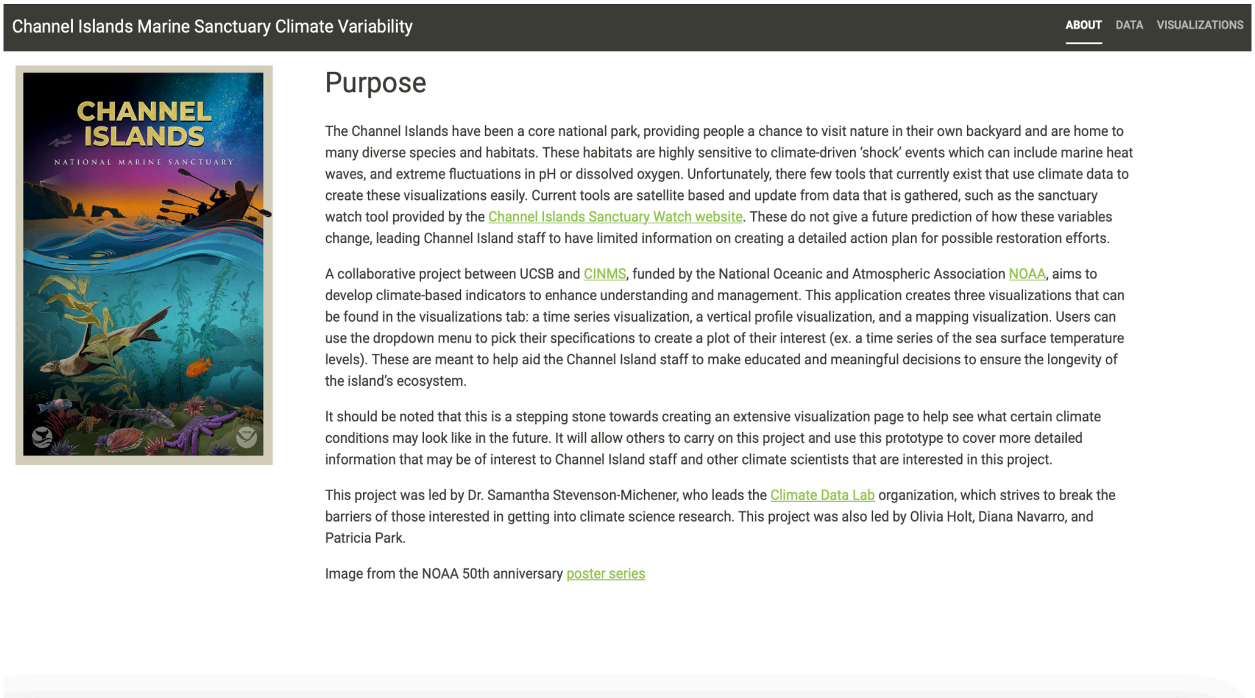
To create the web application that will show the three types of visualizations from the toolkit, the team used Shiny for Python. This is a Python framework designed for building web applications, offering multiple packages that facilitate the creation of dashboards. The code developed in the tutorials will serve as the foundation for the plots in the web application. Through this application, users can visualize various parameters within multiple experiments around the Channel Islands and Southern California. This dashboard will also discuss information regarding the data used and the connections between using Big Data and its applications to sanctuary management.

The repository, located within the Channelislanders organization, contains several essential files for running the interactive Python web application. The `myapp` folder contains:

- **app.py**: This file serves as the main script for the web application, utilizing specific syntax to define the application layout, interactions, and sessions. Inputs are designed to be user interactive, allowing users to select these options:
    - Climate variables: temperature, dissolved oxygen, and salinity
    - Statistics: mean, maximum, and minimum value across the member_id dimension
    - Experiments: historical simulations and RCP 8.5 simulation
        - Note: The code within ‘app.py’ handles these inputs and triggers appropriate data processing and visualization.
- **plots.py**: This file houses functions responsible for generating interactive plots and visualizations using the library, Matplotlib.
- **requirements.txt**: This text file specifies all the required Python packages and libraries needed to run the application.
- **shared.py**: This module manages the data source and selection of specific subsets required for the application. It defines variables and functions that can be imported into ‘app.py.’

In this setup, 'app.py' acts as the main driver for the web application, defining the interface and behavior, while 'plots.py' handles the visualization logic. The 'requirements.txt' ensures that the necessary packages are installed, and 'shared.py' abstracts data management tasks, making the application maintainable for Dr. Stevenson-Michener.

5.4.1 About



Channel Islands Marine Sanctuary Climate Variability

ABOUT DATA VISUALIZATIONS

CHANNEL ISLANDS
NATIONAL MARINE SANCTUARY

Purpose

The Channel Islands have been a core national park, providing people a chance to visit nature in their own backyard and are home to many diverse species and habitats. These habitats are highly sensitive to climate-driven 'shock' events which can include marine heat waves, and extreme fluctuations in pH or dissolved oxygen. Unfortunately, there few tools that currently exist that use climate data to create these visualizations easily. Current tools are satellite based and update from data that is gathered, such as the sanctuary watch tool provided by the Channel Islands Sanctuary Watch website. These do not give a future prediction of how these variables change, leading Channel Island staff to have limited information on creating a detailed action plan for possible restoration efforts.

A collaborative project between UCSB and CINMS, funded by the National Oceanic and Atmospheric Association NOAA, aims to develop climate-based indicators to enhance understanding and management. This application creates three visualizations that can be found in the visualizations tab: a time series visualization, a vertical profile visualization, and a mapping visualization. Users can use the dropdown menu to pick their specifications to create a plot of their interest (ex. a time series of the sea surface temperature levels). These are meant to help aid the Channel Island staff to make educated and meaningful decisions to ensure the longevity of the island's ecosystem.

It should be noted that this is a stepping stone towards creating an extensive visualization page to help see what certain climate conditions may look like in the future. It will allow others to carry on this project and use this prototype to cover more detailed information that may be of interest to Channel Island staff and other climate scientists that are interested in this project.

This project was led by Dr. Samantha Stevenson-Michener, who leads the Climate Data Lab organization, which strives to break the barriers of those interested in getting into climate science research. This project was also led by Olivia Holt, Diana Navarro, and Patricia Park.

Image from the NOAA 50th anniversary poster series

Figure 2: About Page of Web Application

The About page displays detailed information regarding the objective of the capstone and the connection to the larger work through Climate Data Lab. This section emphasizes the importance of the project in acting as a stepping stone towards use of large ensemble climate models.

5.4.2 Data

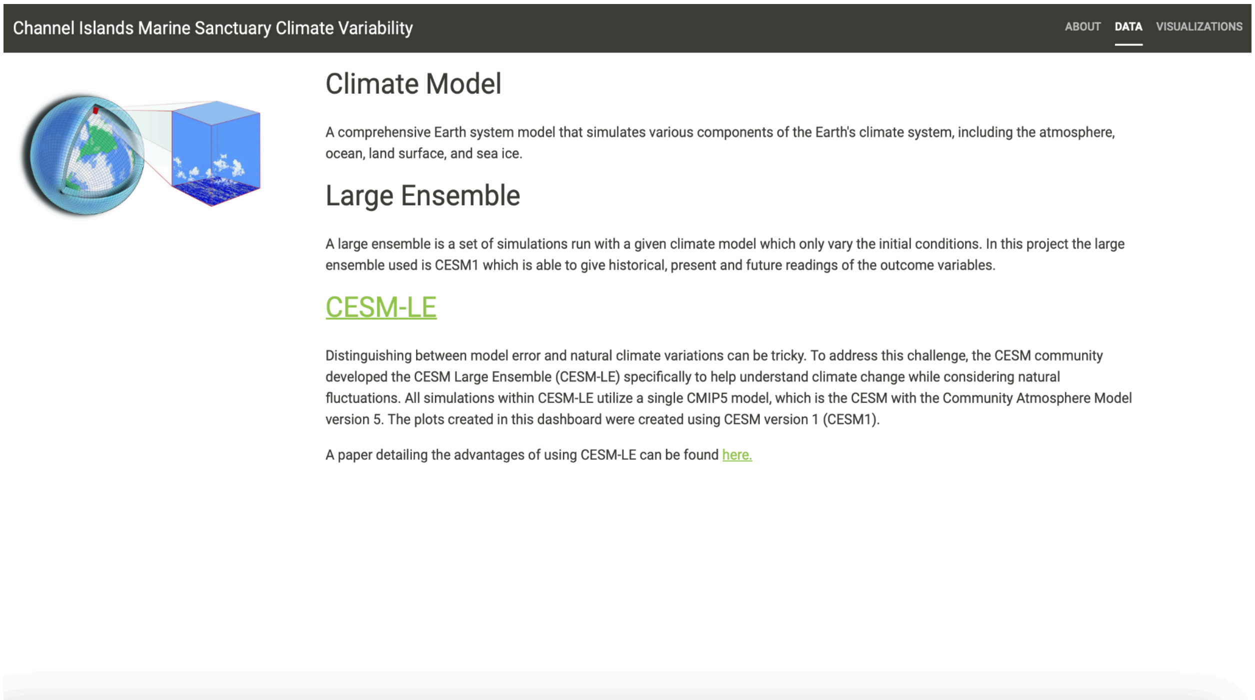


Figure 3: Data Page of Web Application

The data page in the web application displays information regarding CESM1.0. The primary audience will not be familiar with large ensemble climate models, so the team decided to add a section dedicated to this. Making the connection between the visuals and data is imperative towards promoting use of these large data sets.

5.4.3 Visualizations

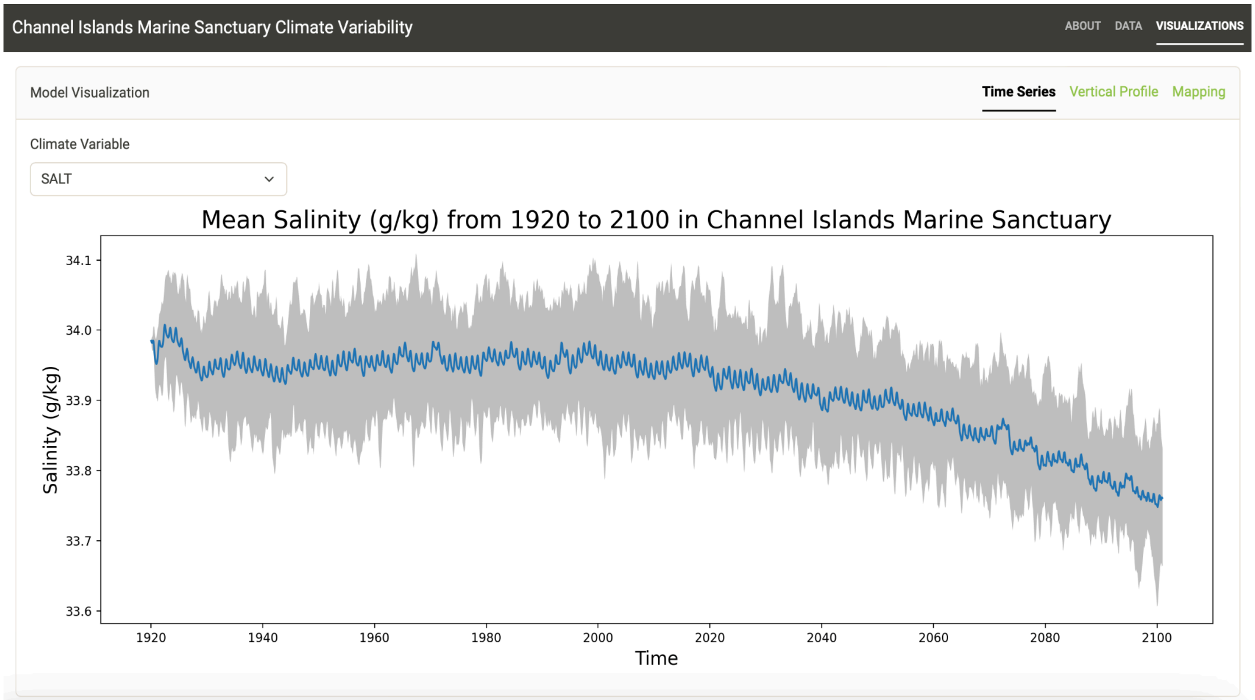


Figure 4: Visualizations Page of Web Application

The visualizations page will produce the output associated with the user inputs. This page will be divided into three separate sections:

1. Time series
2. Maps
3. Vertical profiles

Each of these sections allow the user to edit the input information, which will then result in the plots produced.

5.5 Issues

Issues that arose in the project resulted in a lack of high computing systems and the large data being handled. As a result of this, visualizations required a substantial amount of time to produce within regular computing systems. One way to mitigate this issue is to use Dask. Dask is an open-source Python library designed for parallel computing. It allows Python code to scale from multi-core local machines to large distributed clusters in the cloud. Dask offers a user-friendly interface by replicating the APIs of other libraries in the PyData ecosystem, such as Pandas, scikit-learn, and NumPy.

5.6 Summary

The solution design for this project takes a modular approach to accomplishing the deliverables. Each of these sections briefly cover the steps taken towards completing these deliverables, and provide further detail on the thought process towards the approach taken by the team.

6. Products and deliverables

Here, two main deliverables were completed for the client. These deliverables were highlighted from the client's proposal and completed to meet the School's guidelines for academic completion. The client has access to all deliverables created as mentioned in the proposal as ownership of the Github organization has been transferred to the client. A total of 3 notebooks were made in this project, which are a time series notebook, a vertical profile notebook, and a mapping notebook. All three notebooks can be found here located on the project's Github Repository. While this repository is currently public, the client has full authority to make it private if they choose to do so. The web application code can also be found on its respective Github Repository here. As with before, this repository is publicly accessible. The client will also receive a working web application that produces the wanted visualizations according to the specifications of the user.

7. Summary of testing

The testing plan is focused on implementation of the web application which can be found below:

7.1 Code

As with any educational-type notebook, the code written will only work with certain package versions and if the data contents have not been changed. When making these tutorials, the example notebooks featured on the Pangeo's website did not run properly, and the team had to make adjustments to the code in order to fully access the data. Our testing strategies included each team member running the other two notebooks from start to finish to see if the notebook ran completely through. Capstone staff also participated in this testing as well by git cloning the repository, opening the notebook, and running all cells in the notebook to completion. Minimal testing was done to see if minor changes to the code would still provide an output to what a user may be interested in viewing.

7.2 Product

One of the two major deliverables is creating a web application that can be viewed by Channel Islands staff for their use. For testing this web application, the team created a list of actions taken by the user and the expected results from that chosen action. As there were a very limited number of actions a user can choose to do on the web application, the team decided that it was not necessary to find and use a package to fulfill their testing needs. Many of the actions a user can choose to do are controlled by clicking on a tab or by clicking on a dropdown menu. The chosen information will then be checked with our expectations and if correct, no additional changes will need to be made.

8. User documentation

8.1 Overview and Purpose

The purpose of the educational toolkit is to provide step by step instructions for accessing and processing large ensemble climate data. Working with CESM1.0 is an arduous task, so this toolkit allows for users to understand how to interact with the dataset. For the purpose of our web application, the documentation illustrates how to generate various types of visualizations specifically focused on the data from the Channel Islands Marine Sanctuary. This toolkit is designed for users with a basic understanding of Python who are interested in exploring large ensemble climate data further.

The web application is intended for use and revision by Dr. Stevenson-Michener, specifically designed for the staff of CINMS. Its purpose is to serve as a catalyst to promote the utilization of large ensemble climate models in the realm of climate change.

8.2 Getting Started

In order to run through the educational toolkit, a virtual environment is created as a dedicated space for storing specific versions of packages and Python. There is an option to create the environment locally or through a server, and depending on the computing device, a server is recommended.

In order to set up the environment and correctly run through the toolkit, adhere to the following directions:

1. Open up the terminal on the server.
   *Note that the Channelislanders are using the Taylor server through Bren School.
2. Run the following terminal commands:

   `conda create -n **channelislanders** python=3.9 anaconda`

   `conda activate channelislanders`

   `python -m ipykernel install - - user -
   -name=channelislanders`

The name of the environment does not matter. For reference to the specific project, 'channelislanders' was used. The name is bolded in the instruction above.

3. After confirming the correct environment, proceed with installing the following packages:

```
`pip install xarray`
`pip install intake`
`pip install intake-esm`
`pip install requests`
`pip install aiohttp`
`pip install s3fs`
`pip install zarr`
`pip install gcsfs`
`pip install ipykernel`
`pip install cfgrib`
```

4. Restart the session
5. Choose "channelislanders" as the kernel
6. Run the toolkit.

If the toolkit is going to be ran locally then follow these instructions below:

1. Open the terminal locally.
2. In the terminal window type:

```
`conda create -n cmip6 -c conda-forge dask distributed
ipython netcdf4 xarray
intake-esm aiohttp`
```

This may take a few minutes depending on your computer.

3. In the same terminal, type:

   ```
   `conda activate cmip6`
   ```

4. After this, open up the notebook on the platform of choice. These notebooks were created through JupyterLab on the Taylor server, and also utilized in VisualStudio code.
5. If any of the packages are missing, an error message will appear to notify you which packages to download into the environment.
   a. There are multiple ways to download packages into the terminal, and the most common is

      ```
      `pip install package_name`
      ```

   b. Another way to download packages into the environment is using the

      ```
      `conda search -c conda-forge package_name`
      ```

      Conda search also searches for the package while conda-forge installs it. This same command can also be used in the terminal for the server if any other packages need to be installed.

8.3 Interacting with the Toolkit

The instructions for running and understanding are detailed in the notebooks located in the 'toolkit' repository. This documentation will provide an overview of the steps applicable to all repositories. Throughout this notebook, for loops were created to loop through the multiple climate datasets. This was done because loading in all the data through the computer created many issues and warnings of working with data that is too large. For more detailed information regarding the code, refer to the respective repository.

8.3.1 Pangeo Tutorial

This tutorial references the Pangeo gallery for CESM Large Ensemble Community Project (or LENS) on Amazon web services. Main packages used to source the data are xarray and dask which improves user experience by faster computing. The original data is roughly ~500 TB so accessing it through a regular computing system is impractical, using operations that involve 'lazy loading' that are shown through Pangeo inspired our tutorial setup and influenced packages used.

8.3.2 Understanding the Data

The notebook will begin with a short description of the data being used and its source. The data is loaded from Pangeo through Amazon Web Services, a separate file containing the grid data needs to be loaded in as well because it is not provided in the dataset.

A few things to take into consideration when running through the tutorial:

- ➢ The tutorial discusses accessing the ocean model, so if the user is trying to access a different component such as the atmosphere or land variables, this will need to be added/specified in the dictionary search.
- ➢ The loaded grid file is tailored for the ocean model, therefore the user must also provide and designate the file path. The atmosphere, land, and sea ice components contain different latitude and longitude information.
- ➢ We are accessing variables specifically for CINMS, so the user should also reference this list if interested in other climate variables.

8.3.4 Data Exploration

In this section of the tutorial, the user can explore the dataset and familiarize themselves with the data. This section dissects the xarray.Dataset for the user. This image displays sea surface temperature(SST) within experiment '20C.' The CESM1.0 data contains multiple `experiments` which are datasets that are associated with different time frames or emissions scenarios. This tutorial will explore three different experiments already discussed in Section 5.2 of the Solution Design but for clarification:

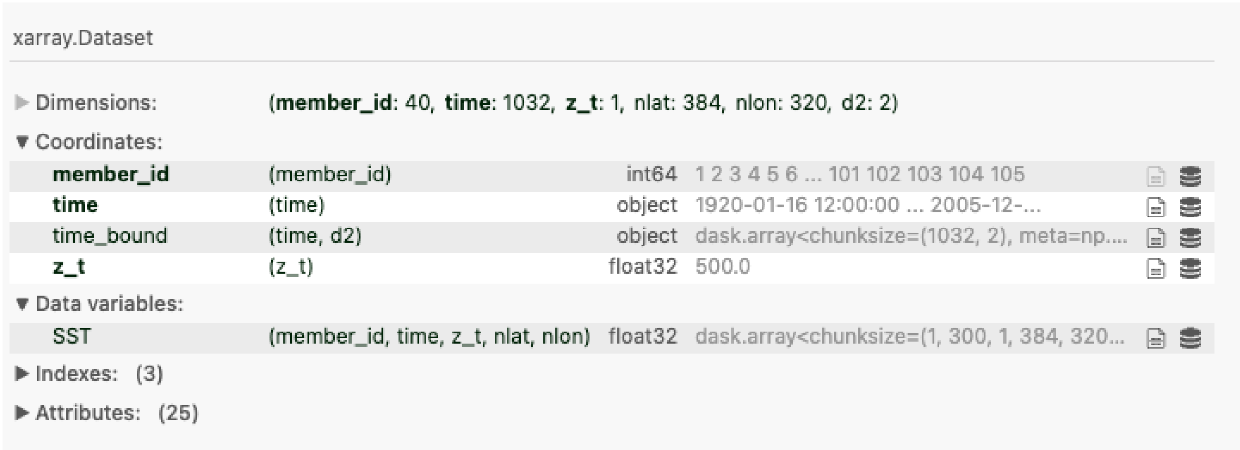


Figure 5: xarray Dataset summary.

- ➢ 20th Century (1920-2005)
- ➢ RCP 8.5 (2006-2100)

Along with choosing experiments, there is also an option for `frequency` of the data which allows the user to choose from annual, monthly, daily, etc. This can also be found on the official NCAR CESM website. Each climate variable associated with the experiment and frequency represents one dataset. The image displayed is the dataset associated with monthly averaged sea surface temperature (SST) for the 20th century simulations. Although dimensions will vary, it's important to understand what each represents:

- ➢ member_id : There are 40 different member_id's that represent different ensemble members used in the simulation. These members usually vary by initial conditions, parameter sensitivity, and various others.
- ➢ Time: This dimension will change depending on the frequency chosen and this particular example shows it contains 1,032 samples.
- ➢ z_t: this represents the depth of the sample.
- ➢ nlat: cell index referencing the latitude position to the corresponding sample
- ➢ nlon: cell index referencing the longitude position to the corresponding sample
- ➢ d2:2: temporal intervals providing information about the start and end times of each interval

As mentioned before, each data set will contain different dimensions but will most likely carry time, nlat, and nlon.

8.3.5 Data Wrangling
This section of the toolkit will vary depending on the notebook the user is working with.

Mapping

Within the mapping notebook, the user will need to merge the grid data with the climate variable data then regrid using the `xesmf` package.

In order to avoid unnecessary data into the data, the toolkit demonstrates how to subset the grid data by calling the 2 dimensions and 2 coordinates we are interested in:

- ➢ Dimensions: nlat and nlon
- ➢ Coordinates: TLAT and TLONG

After properly subsetting, the user will need to rename the coordinates 'TLAT' and 'TLONG' because the regridder package will not work unless it has specified 'lat' and 'lon' to work with. This is also done in the for loop so that the user does not have to code this multiple times.

The next part of this section creates a target grid. A target grid is a grid created that aligns with actual latitude and longitude coordinates. Notice that nlat and nlon coordinates are 384x320 but we need a 180x360 grid. This will be used in the `xe.Regridder` function. Within this function the user will specify the merged data set and target grid to align with. After this the user will be able to apply just `regridder` to the same merged data set.

This concludes the data wrangling section for Mapping.

Time Series

For the time series notebook, minimal coding is necessary in this section. Merging and regridding is not necessary because the user is still able to use nlat and nlon to locate the area that needs to be worked on.

- ➢ How to subset for a specific time frame
  - ○ This is done by using the subset syntax [ ]
- ➢ How to subset for a certain 'member_id' or how to find the mean of all the 'member_id'
- ➢ And how to slice for a specific time frame

Vertical Profile

Within the vertical profile notebook, there is minimal data wrangling necessary. The data wrangling involves subsetting the data to a specified location, and taking averages in order to condense the data to be plotted.

The data wrangling section of the vertical profile notebooks starts with subsetting the data to a specific latitude and longitude of interest.

- ➢ Dimensions: nlat and nlon

Next, a loop is created to take an average of the variable of interest for each member_id. Lastly, the data is converted to an xarray DataArray and an average of the variable of interest is taken from all member_id's.

8.3.6 Producing Visualizations

The final section of each notebook demonstrates the process of generating visualizations. Each notebook provides guidance on creating maps, vertical profiles, and time series plots for various climate variables to help users better understand and analyze the data.

Mapping

Producing visualizations for this section includes subsetting the area that the user wants to focus on, so in this section Channel Islands Marine Sanctuary (and surrounding waters) was included in the plots. This section of the mapping also involved showing the user how to calculate a statistic (mean, max, min) of any of the dimensions.

In the case of sea surface temperature:

- ➢ Find the mean of both time and member_id.
- ➢ Sea surface temperature has only 1 depth dimension (z_t) so finding the mean of this statistic was unnecessary.
    - ○ If the user wishes to plot the full depth, then finding the mean of all three dimensions will properly plot the climate variable.

In addition, this section also demonstrates how to plot a shape file over the original climate data. However, the user will need to ensure that the crs matches to properly align coordinates. In this case, importing the library, cartopy as ccrs, was extremely useful in projecting the shape file over the Channel Islands sanctuary. Since the regular latitude and longitude coordinates are used, if the user wished to utilize another shape file, they will be able to do so easily, as long as they subset the proper coordinates.

This was repeated with all climate variables, but the team would like to emphasize that since this data is large, it will take some time to run.

In this case, the team worked around this by subsetting the area and saving that to a netCDF file, and then finding the statistic across all dimensions. This was not an issue with sea surface temperature since it's smaller data array, but this should be noted in the toolkit.

Time Series

To create the time series visualization, the user must subset into a certain nlat and nlon in order for it to be made correctly. Users must also subset to a certain depth they are interested in to get the correct output of the graph. Users do have the choice to subset on which time frame they are interested in, however, a whole time frame overlooking the whole year is usually recommended for a time series plot. Once subsetted correctly, users will then be able to use the plot function by the package matplotlib to create their visualizations of interest.

Users can also input the maximum and minimum from the 'member_id' column by using the max() and min() functions and input it into the graph to get a visual representation of what the possible maximum and minimum values are from the dataset.

Vertical Profiles

To create vertical profile visualizations, the user must subset into a certain latitude and longitude in order for it to be made correctly. This subsetted data is downloaded into an '.nc' file which is then loaded in to pull for the plots. Once the data is loaded, users will then be able to use the plot function by the package matplotlib to create their visualizations of interest.

Users can use the mean, maximum, and minimum from the 'member_id' column by using the mea(), max(), and min() functions to get a visual representation of these values from the dataset. Other variables of interest can be selected instead of temperature, such as 'O2' (Dissolved Oxygen) and 'SALT' (Salinity).

To plot the vertical profiles it is important to note that the y-axis is inverted as the plot should look like a cross section of the ocean from the specified coordinates.

8.4 Using the Web Application

The web application primarily targets the CINMS staff as its audience and will serve as a baseline for further development.

The user is recommended to navigate through each tab within the application by the following order:

1. About
    a. In this section, users have the opportunity to delve deeper into the project's details and discover the fundamental purpose behind the creation of the web application. Here, they can explore comprehensive information about the project's objectives, goals, and the underlying motivation driving the development of the application. This tab serves as a valuable resource for users seeking a clearer understanding of the project's scope and significance within its domain.
2. Data
    a. Due to the infrequency of the data utilized in this application for sanctuary management, this section will elaborate on the details of leveraging large ensemble climate models. It will specifically describe CESM1 data and explore why these models are indispensable tools, offering insights into how they enhance our understanding of complex environmental dynamics and variability.
3. Visualizations
    a. After navigating through the other tabs, the user will be able to view the visualizations. These will be separated into three different panels that produce the plots.
        i. Time series contains two drop down menus where the user is able to select from different climate variables and statistic.
        ii. Vertical profiles and maps will show three drop down menus where the user will be able to select from climate variables, statistic, and experiments.

Due to the time constraints of the project, this web application will emphasize the importance of initial exploration of using these models.

8.5 Troubleshooting and Support

8.5.1 URL & Catalog Errors

If the user finds that they are running into issues accessing the data, then there are multiple options they can pursue, but first the team would also like to emphasize that CESM1.0 data remains consistent, so this toolkit should not run into any issues with the data itself. However, amazon web services could work entirely differently. If the issues stem from the url, then the user is recommended to check their resources page, to make sure the link has not changed.

Another error that may occur is climate variable data, amazon web services does not host all data displayed on NCAR website lists of variables. In order to check which variables are in each, the user will need to conduct a '.search' within the data catalog displayed as a dictionary. This will entirely depend on the variables being searched, but the NCAR list should display the exact name of how it should appear once loaded in, so if nothing appears then most likely AWS does not provide the data.

8.5.2 Package Errors

Since a virtual environment is created, there should be minimal errors in package versions, but the user should ensure that they are using an updated version of Python, in this case it is version 3.9.

8.5.3 Support

Following the completion of this project in late May 2024, the GitHub repository will be passed to Dr. Stevenson-Michener. Any future modifications will be managed by Dr. Stevenson or future collaborators.

9. Archive access

9.1 Data sharing & Access

Data for this project came from publicly available and open-source datasets through CESM1. The publicly accessible data is cited under references in the final Technical Documentation Plan.

9.2 Data Archival and Preservation

All code scripts were version controlled and archived using GitHub repositories. The study's data, metadata, and scripts were provided to the Client for their archives and ongoing research. Data used for the web application will be archived using Dryad, which is UCSB's institutional data repository. In addition to this, a metadata file and a README file will accompany the data preservation file. This data will be made publicly available and can be accessed through a DOI link.

10. References

Community earth system model. Large Ensemble Community Project (LENS) | Community Earth System Model. (n.d.). https://www.cesm.ucar.edu/community-projects/lens

Kay, J. E., Deser, C., Phillips, A., Mai, A., Hannay, C., Strand, G., Arblaster, J. M., Bates, S. C., Danabasoglu, G., Edwards, J., Holland, M., Kushner, P., Lamarque, J., Lawrence, D., Lindsay, K., Middleton, A., Munoz, E., Neale, R., Oleson, K., Polvani, L., & Vertenstein, M. (2015). The Community Earth System Model (CESM) Large Ensemble Project: A Community Resource for Studying Climate Change in the Presence of Internal Climate Variability. *Bulletin of the American Meteorological Society*, *96*(8), 1333-1349. https://doi.org/10.1175/BAMS-D-13-00255.1

Office of National Marine Sanctuaries. 2019. Channel Islands National Marine Sanctuary 2016
Condition Report. U.S. Department of Commerce, National Oceanic and Atmospheric Administration, Office of National Marine Sanctuaries, Silver Spring, MD. 482 pp.

Office of National Marine Sanctuaries. (2022). Science Needs Assessment of Climate Change. Retrieved from: https://nmssanctuaries.blob.core.windows.net/sanctuaries-prod/media/docs/2022-channel-islands-nms-climate.pdf

Sanford E, Sones JL, García-Reyes M, Goddard JHR, Largier JL. Widespread shifts in the coastal biota of northern California during the 2014–2016 marine heatwaves. Sci Rep. 2019 Mar 12;9(1):4216

Taylor, K. E., Stouffer, R. J., & Meehl, G. A. (2012). An Overview of CMIP5 and the Experiment Design. Bulletin of the American Meteorological Society, 93(4), 485-498. https://doi.org/10.1175/BAMS-D-11-00094.1