Empirical Climate Model Web Application

CP499: Team Software Project

Final Report

Abe Binder, Clara Richter, Sunny Chiu

Table of Contents

1 Executive Summary

An *empirical climate model* (ECM) takes a statistics-based approach in modelling the global Earth temperature as dependent on various forcings, such as the levels of greenhouse gases and solar radiation. Our web application visualizes such a model. Specifically, through an interactive interface, the application enables users to directly explore how four forcings can account for the variation and change in Earth’s mean global temperature record. The application is intended to be used in educational settings. For instance, it will be part of an online curriculum developed by the Biological Sciences Curriculum Study (BSCS). It is live at https://ecm.coloradocollege.edu/.

2 Introduction

2.1 Background

Scientists use climate models to understand the relationships between climate forcings and responses in the system that makes up Earth's climate. An *empirical climate model* (ECM) is one type of model that, as opposed to many others used in modern research that utilize physical laws, is constructed entirely based on data.

In 2010, Dr. Judith Lean from the Naval Research Laboratory devised an empirical climate model. Lean used the *global mean surface temperature record* to construct a time-series regression model that describes how temperature anomaly (T_a) depends on four forcings. The four forcings in questions were El Nino southern oscillation (ENSO, or E), volcanic aerosols (V), solar radiation (S), and an aggregate measure of various anthropogenic forcings[1] (A). Mathematically, the model was defined as follows, where w_E is the weight of the ENSO factor, *E(t)* the measure of the ENSO factor at time *t*, and similarly for the other three factors:

$$Ta\,(t) = b_0 \ + c_E E(t) + c_V\, V(t) \ + c_S S(t) + c_A A(t)$$

Incorporating *time lags*, Lean's model was a slightly modified version of the one above. Time lags refers to the phenomenon that, for some forcings, there is a certain delay before the full effect on temperature is reached within the system -- a temporal shift between forcing and measured response. For example, according to Dr. Steve Getty, "it takes some amount of time for sulfur-bearing volcanic aerosols to react chemically to produce reflective sulfates and then disperse in the stratosphere". Lean's model, based on these observations, was defined as follows:

[1] These include greenhouse gas levels as well as ozone levels in the upper atmosphere.

$$Ta\ (t) = b_0 + \Delta T_{ENSO}\ (t) + \Delta T_{VOLC}\ (t)\ + \Delta T_{SOL}\ (t)\ + \Delta T_{ANTH}\ (t)$$

where

$$\Delta T_{ENSO}\ (t)\ = c_{E1}E(t-11)\ + c_{E2}E(t-5)\ + c_{E3}E(t-3)$$

$$\Delta T_{VOLC}\ (t)\ = c_{V1}V(t-10)\ + c_{V2}V(t-3)$$

$$\Delta T_{SOL}\ (t)\ = c_{S1}S(t-1)$$

$$\Delta T_{ANTH}\ (t) = c_{A1}A(t)$$

The significance of this model is that, although substantially less complex than most other modern climate models, analysis shows that it is quite accurate in predicting global Earth temperature. In particular, Lean found that the best-fit model could account for 83% of the variance in the global temperature record. In other words, despite its relative simplicity, the model is an indication something profound about the Earth’s climate system: four key forcings alone can to a large degree explain the short-term variance as well as long-term change in the global mean temperature record.

The overall objective with this project was to develop an application that visualizes this model and communicates its insight in an engaging way to learners of science. A second goal was to update Lean’s initial model to include recent data, that is, data that was recorded after her publication from 2010.

2.2 System Boundaries

It was decided upon that the application was to be web based, that is, be accessible via the Internet and via the URL “ecm.coloradocollege.edu”. The heart of the application would be a single page, displaying two graphs as well as controls for modifying these graphs.

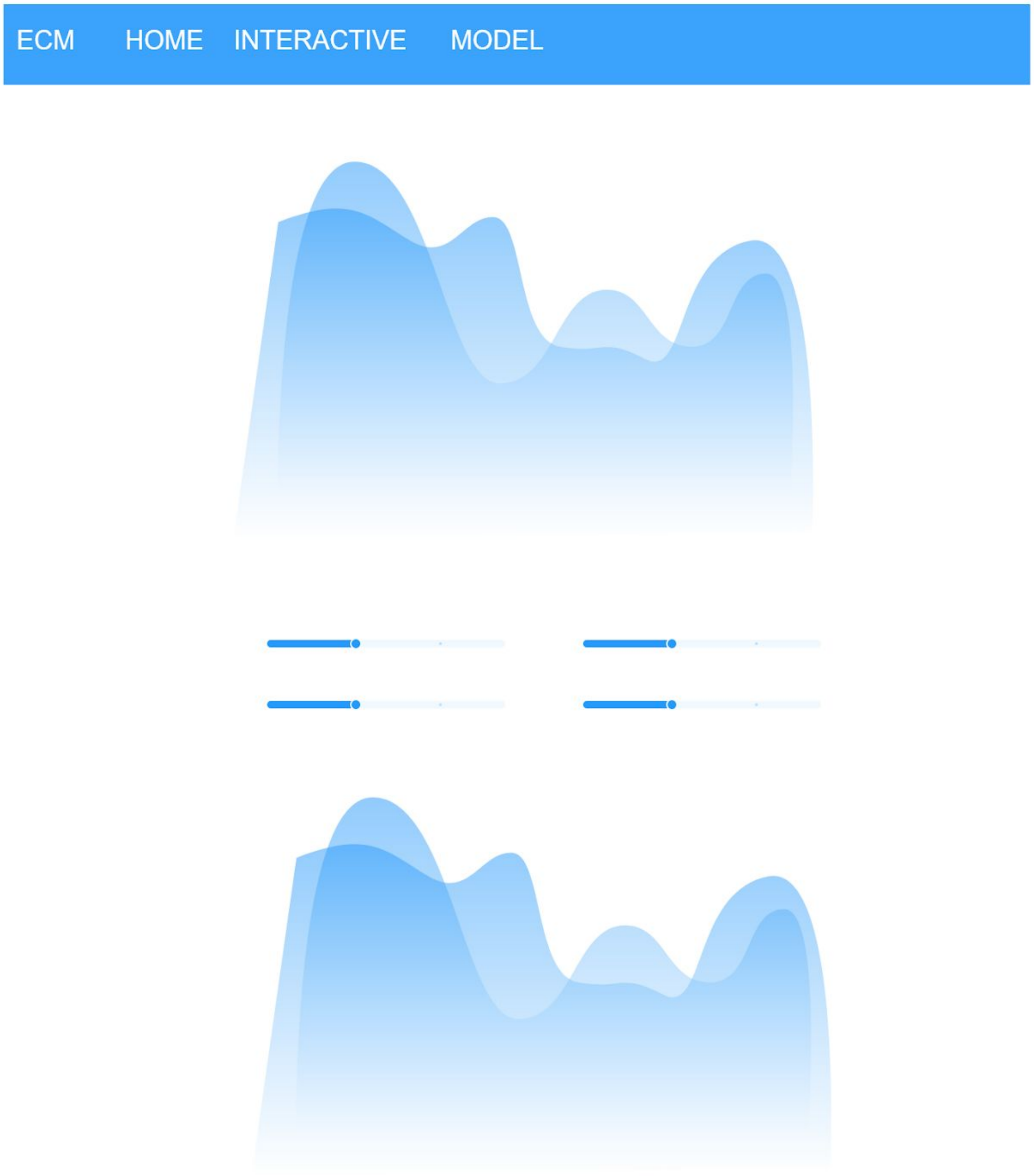
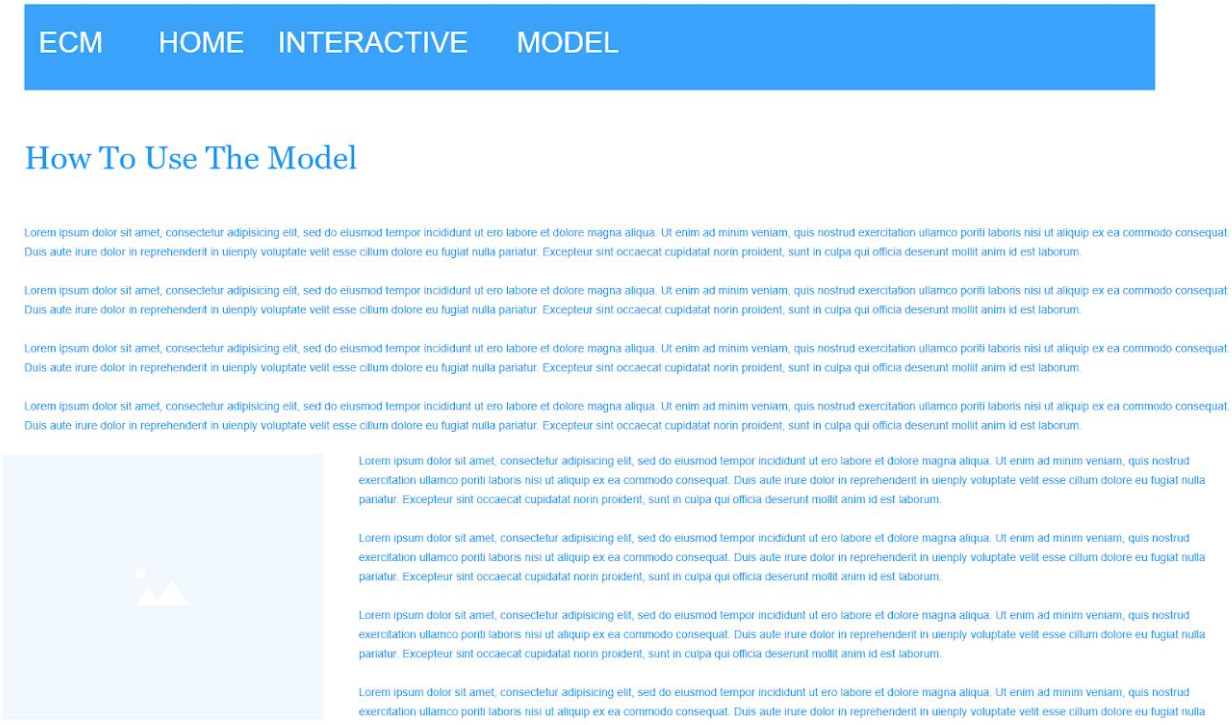


Figure 1: Mockup of Interactive Graph Page With Slider Controls

The first graph visualizes the climate models predictions of global mean temperature alongside “gold standard” readings of global mean temperature, data averaged from hundreds of weather stations around the world. The second graph is the four forcings of the model graphed on top of each other.

In addition to the “interactive” page, the web application would include a set of static pages with

a) information about how to use the web application, and b) context about the climate model.



How To Use The Model

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

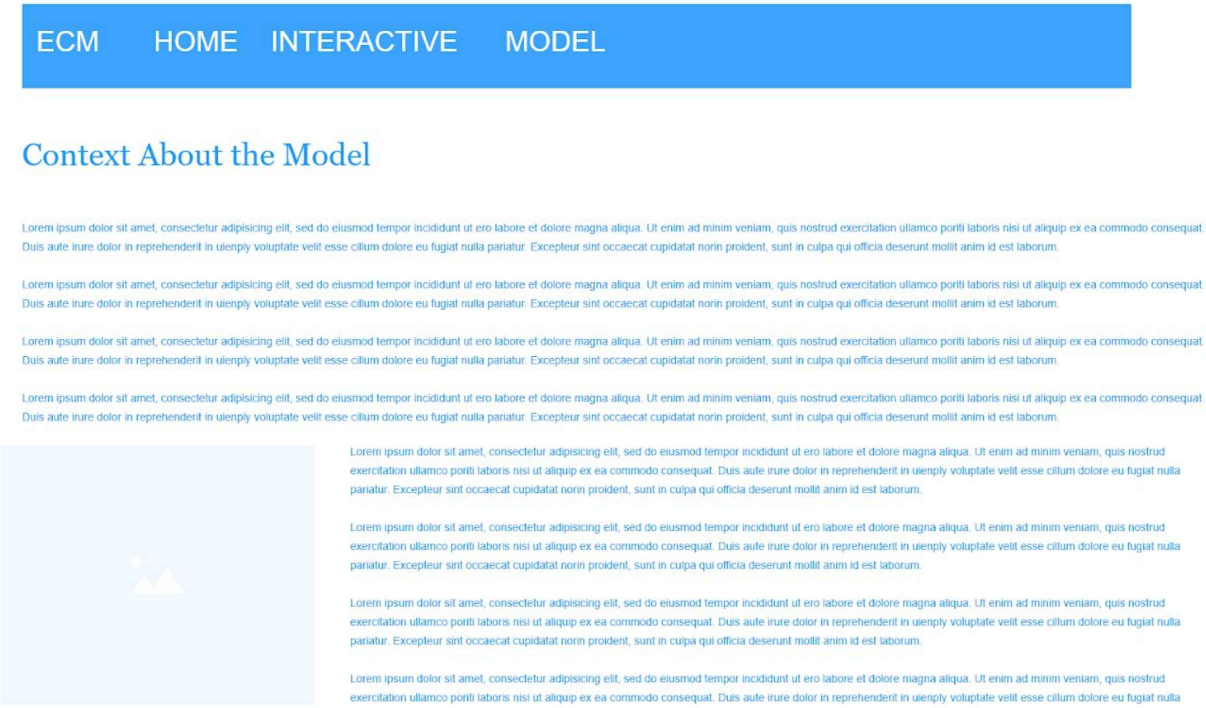
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla

Figure 2: Mockup of page detailing how to use model



Context About the Model

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat norin proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut ero labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco poriti laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in uienply voluptate velit esse cillum dolore eu fugiat nulla

Figure 3: Mockup of page with context about the model

2.3 The Client

The client of the project was Dr. Steve Getty, educator of science and earth scientist. Getty's intention was for the application to be used in various educational contexts. First and foremost, Getty himself will be using the application in introductory science classes that he occasionally teaches at Colorado College. Moreover, the application will be used in an online curriculum called *Carbon Connections* developed by the Biological Sciences Curriculum Study. As of early April, the application can be accessed via the curriculum website (see http://carbonconnections.bscs.org/interactives/).

While the application is most effectively used in a structured classroom setting with an instructor who provides guidance, the application is designed to also be used by someone who is not necessarily in an educational context. It was with this use case in mind that we decided to include a number of additional pages, besides the model visualization page, where contextual information about the model, as well as a description of how to use the web application is given.

3 User Stories

The following “user stories” describes the use cases of the application. All features required for the completion of above user stories were successfully implemented, except those of user story 3.10; the application cannot be accessed without an Internet connection.

3.1 Accessing the application from carbonconnections.org

Student would like to access the application from the links on “carbonconnections.org”, by which they accessed the old version of the application. To accomplish this, student goes to “carbonconnections.org” and follows the curriculum. When it is time to use the climate model application, they click on the given link and a new tab opens that directs to “ecm.coloradocollege.edu/interactive”.

3.2 Investigating the accuracy of the empirical climate model

Student would like to know how the (best-fit) model predictions of temperature anomaly compares to the actual observations. To accomplish this, student goes to the “Interactive” page. They look at the line graph at the top of the page, and how the lines (of actual observations and best-fit predictions respectively) correlate. They also look at the R^2 value of the progress bar to the right of the graph.

3.3 Investigating the role of each forcing

Student would like to know how the four individual component forcings of the empirical climate model individually affect Earth’s average surface temperature. To accomplish this, student goes to the “Interactive” page. They scroll down the page to get to the bottom graph, which includes one line for each of the four forcings. They inspect the line of each forcing component.

3.4 Investigating the significance of the anthropogenic forcing

Student would like to know how human influences relates to, and whether it can explain, the long-term increase of Earth’s global average temperature. To accomplish this, student goes to the “Interactive” page. Student slides the bar that adjusts the weight of the human forcing, and inspect the how the graph has changed.

3.5 Adjustment of lags

Student would like to know how time lags affect the accuracy of the model. To accomplish this, student goes to the “Interactive” page. Student scrolls to the bottom of the page, where the “Time Lags” tab is located. Student opens up tab content, enters different values for each lag component, and hits the “Update” button. They inspect how the new best-fit compares with actual observations, and what the new best-fit R^2 value is.

3.6 Investigating future temperature anomalies

Student would like to, using the model, see predictions of temperature anomalies of the future. To accomplish this, student goes to the “Future Predictions” page and views the graphs and texts which describes an extrapolation technique of data into the future, and how the model predicts future global Earth temperature given this projected data.

3.7 Learning about the context of the model

Student would like to understand more about the forcings and what they represent. To accomplish this, student goes to the “Responses, Forcings” page, and reads the contextual paragraphs displayed.

3.8 Using a screen reader to access application content

Student would like to navigate the site, and learn the site content, using a screen reader. To accomplish this, student goes onto the web app on a desktop or laptop and uses the screen reader to read the information out loud. The student can navigate through the website through the keyboard or mouse. When the focus of the screen reader is on an appropriate website section, the screen reader reads the provided label or description.

3.9 Mobile usage

Student would like to use the app on their mobile device. Student opens the app on mobile, and the all of the application functionalities can be accessed in an intuitive and easy way.

3.10 Offline usage

Educator wants to use the application for a class, but the Internet connection at the school is unreliable. Student load the page while at home. At school they can reload the page, and still access and use all of its functionality, despite unreliable Internet connection.

4 Architecture

The software architecture of the application is based on Vue.js, an open-source JavaScript framework. Frameworks such as Vue (React, Angular being alternatives) comes with **a lot** of different functionalities that is conveniently bundled up for the developer. One important feature of Vue.js is the *single-page application* aspect. In traditional websites, pages are stored as separate resources on the server. Whenever a client goes to a page, a request to the server has to be made. By contrast, a single-page application--from the client-server perspective--consists of a single-page. From the perspective of the user of the website, however, there are multiple pages. But these are only *virtual* pages; JavaScript, executing all client-side, is responsible for making it appear as separate pages when in fact everything on the website is in one single resource.

The *Empirical Climate Model Web Application* uses this architecture, which has a number of benefits, most importantly from a performance perspective. If a traditional architecture was to be used, then the user would experience more of a delay (load time in making the request) when toggling between the pages. With a single-page application architecture, the webpage is only loaded once.

Additional Node.js libraries are used for extra functionality such as multiple linear regression, graphics, and other user interface features. The source code is organized according to the (virtual) pages; each page is a separate module in /src/views. The modules responsible for model computations are located in the src/model directory. The page module Interactive.vue, has different visual components located in separate modules in src/components. These components interact with the model modules in order to retrieve predictions and statistics used for visualizations.

It should be noted that the model visualization page is the only one that use JavaScript and has “interesting” content from a development point of view; the rest merely contains of static HTML code.

4.1 External Packages

- ***Fontawesome-icons*** was used for the arrow icons on the expandible tab headers, used to indicate whether tab is open or closed.
- ***Bootstrap-vue*** was the frontend framework used for this project.
- ***math.js*** was used for various statistics-related computations and is utilized in the file “src/model/model-util.js”
- ***ml-regression-multivariate-linear*** (part of the larger JavaScript modelling library ***ml.js***) was used to perform the regression, in other words, train the model to the data.
- ***Papaparse*** was used to parse the .csv file containing the data into a JavaScript two-dimensional array.
- ***Vue-router***; the Vue third party standard, was used to accomplish routing.
- ***Babel*** Vue.js uses babel to transpile JavaScript code into various version compatible with different browsers.
- ***Webpack*** is also used by Vue.js to bundle the modularized JavaScript code into a single file.
- ***Mocha*** was the testing framework used for unit testing. ***Chai*** was the testing library used to access a richer set of assertion types.

5 Validation

The test suite below is performed in, and expected to be passed for, the browsers listed below. The requirement of a bullet point whose text is striked through was not met. In the footnote of the bullet point follows a comment or explanation.

- Google Chrome
- Mozilla Firefox
- Safari

- ~~Internet Explorer~~[2]
- Microsoft Edge

5.1 General

1. Go to *carbonconnections.org.* Click on the link to get to the climate model application. Assert that the website directed to is *ecm.coloradocollege.edu*.
2. Go to *ecm.coloradocollege.edu*. Assert that no error has been thrown.
3. Go to *ecm.coloradocollege.edu*. Assert that all visual components appear in the interface:
    a. Heading
    b. Tabs
        i. Home
        ii. Model
        iii. Info
        iv. Documentation
    c. Graph 1
        i. Two lines appear, one for predictions and one for actual observations (the third, the best-fit line, is hidden under predictions)
    d. Progress bar
    e. Slide bars
    f. Graph 2
        i. Four lines appear, one for each forcing
    g. “Advanced input” tab
4. Assert that the slider bar values are at 1.
5. Assert that, when clicking “Advanced Input” tab, the tab expands and the following visual components appear:
    a. Duration inputs
    b. Lag inputs
    c. “Update” button, “Reset” button
6. When zooming, all the components shrink but remain fully visible
7. ~~Go to ecm.coloradocollege.edu. Make sure there is Internet connection, such that the page loads as expected. Turn off connection. Reload page. Assert that page loads as though Internet connection was had.~~[3]

[2] As of now, the application is not running in Internet Explorer. For a reason the authors cannot explain, Vue fails to transpile the source code into a correct version of JavaScript and so a syntax error is thrown when the model visualization page of the application is opened.

[3] This feature was never implemented. However, Vue.js does provide functionality for offline usage (developers call applications with this functionality “progressive web apps”), so it shouldn’t be impossible to achieve,

5.2 Pages

1. Go to *ecm.coloradocollege.edu*. Assert that “home” page is displayed.
2. For each page <page>, for each of the other pages <other page>, from <page>, assert that <other page> is displayed

5.3 Slide bars

1. Set all slide bar values to 0.
    a. For graph 1, assert that the “predictions” line is straight and parallel to the X-axis.
    b. For graph 2, assert that there is a single line, straight and parallel to the X-axis, and intersecting Y-axis at 0.
    c. For progress bar, assert that the “fit indicator” is at 0.
2. **For each slide bar**, have it be set to 1 and the other slide bars set to 0.
    a. For graph 1 AND graph 2, assert that, when given value of slide bar is adjusted to 2, the y-values along the “predictions” line (graph 1) and forcing line (graph 2) both increase by a factor of 2.
    b. For graph 2, assert that, besides the forcing line in question, there is a single line at y = 0 (representing the factors whose amplitudes are set to 0).
    c. For the progress bar, assert that the “fit indicator” value is greater than 0, but below best-fit R^2 horizontal line.
3. With all slide bars set to 1, when the anthropogenic forcing slide bar is increased to 2, assert that, the “predictions” line tilts downward, such that the y-values of it are underestimated compared to the actual observations.
4. **For each slide bar**, have it be set to 1 and the other slide bars set to 0. Increase its value to 1.1. Assert that the “fit indicator” value is less than the best-fit R^2 value. Decrease its value to 0.9. Again, assert that the “fit indicator” value is less than the best-fit R^2 value.
5. **For each slide bar**, assert that, when either of the following types of user interactivity occurs, the graphs are updated:
    a. User ***drags*** the bar with finger or cursor
    b. User ***presses with finger*** or ***clicks*** somewhere along the bar
6. Assert that, when in a zoomed in mode, sliding bars can be adjusted equally well.

5.4 Duration Inputs

Go to “model” tab. Open up “Advanced inputs” tab.

1. In “start” text field, enter “1985”. In “end” text field enter “2005”. Assert that the “Update button is clickable”. Assert that, when clicked, the range of the X-axes are updated accordingly.
2. In “start”, enter “1978”. Assert that “invalid input” message is shown, and that “Update” button is unclickable.
3. Perform the same test with the “2018” as input.
4. Perform the same tests for “end” text field.
5. Perform the same test with “1990” as input for start, “1989”.
6. With “1990” for start and “1990” for end, assert that the graphs’ X-axes go from the beginning of 1990 to the end of 1990.
    a. Assert that the X-axis labels is 1990,1 and 1990,12

5.5 Lag Inputs

Go to “model” tab. Open up “Advanced inputs” tab.

1. Input the following in the “Lag features” textfield: E1: 0, E2: 3, E3: 5, V1: 0, V2: 3, S: 0, A: 0. Assert that the “Update button is clickable”. Assert that, when clicked, the “predictions” line of graph 1 deviates from that of “best fit”. Assert that the “fit indicator” is decreased.
2. For each of the lag inputs, type a negative number. Assert that “invalid message” is shown and that “Update” button is unclickable.
3. For each of the lag inputs, type a number that exceeds the number of available months in the past, assert that “invalid message” is shown and that “Update” button is unclickable.

5.6 Screen reader

1. Go to ecm.coloradocollege.edu. Navigate the menu bar with “tab” key. While doing so, assert that, that the title of each tab is read out loud.
2. For each tab content, use the tab to navigate the components. Assert that, for each component (excepting graphs), the component label is read.
    a. Assert that, clicking tab and enter will enable you to interact with different components

6 Development Process

6.1 How Github was Used

Git was our source control method and we used GitHub to host our source code. Our repository on Github was private, as per our clients request. One person created the Vue.js frame for the project. Once that existed, we split up user stories into different Github issues. When we chose stories to work on, we tried to choose them in a matter where each developer worked orthogonally to every other developer. That way, we avoided conflicting merges and repeated work. When someone was developing a feature, they would checkout a new branch for it, make their changes, and then merge their changes with the master branch and push.

6.2 How the Team Communicated

We met each day after our meeting with the professors to discuss what we would do for the day. Often, we worked in the same physical location so that when someone had an issue they wanted to discuss, they could reach out immediately. When we did not work together, we had a group chat IM open so that we could ask each other questions. We also met with our client three times a week to keep in him the loop and report our progress.

6.3 What tools were used

We used Git for source control, Excel as the gold standard to compare our JavaScript model computations against. We used the JetBrain’s IDE “Webstorm” for development. We adopted a SCRUM approach for organization.

6.4 Challenges encountered

The entire team had no prior experience with web development and faced a steep learning curve. A large majority of the time spent in the first week of development was learning and experimenting web technologies, such as Javascript, CSS, and different front-end framework. The time spent on debating about architecture (e.g. model on client or server side) and data/model security delayed the code production for this web application. Furthermore, the client relationships delayed the production by having at least three meetings and lectures a week to teach the science behind the model and discuss different user requirements for this application. There was no agenda for each meeting and, as a result, the meetings became unproductive and repetitive. A cause for these unproductive meetings is the client not knowing what he wanted or what was possible for the development team to accomplish. With the client's vague requirements and lack of understandability that this project is a software-based and not climate education focused, the client repeated lectures about the climate science and requested large alterations to the code base during the end of week three. Due to these interactions with the client, the team grew to have a tenuous and unpleasant relationship with the client.

In addition, there were aspect of teamwork that caused tension and unproductivity. Communication between the team members was sometimes too critical and not positive, which appeared condescending. At the end of the block, these issues were sorted with open and positive discussions. While the development of the code base could have been more productive if the team was not as modularized. For example, when a method had to be changed in another team members code section, we would have to wait for that team member to change it. With less modularization, the team could had worked independently on different sections of the web application and learn more about web development.

Other than interpersonal challenges, technical challenges existed after testing the application on different browsers and screen size. After doing these testing, the team discovered the certain libraries were not cross-platform responsive or functional. In order to resolve these problems, multiple libraries were incorporated and tested before the final product.

7 Summary

The Empirical Climate Model Web Application was requested by Steve Getty. Its purpose is to teach students about the factors that impact global mean temperature. The application is written in Javascript and HTML and uses Vue.js as JavaScript framework and Bootstrap as front-end framework. Students and teachers in an educational setting can access the application through a mobile or desktop browser. The web application is hosted on a colorado college virtual machine and is live at ecm.coloradocollege.edu

8 References

Lean, J., & Rind, D. (2008). How natural and anthropogenic influences alter global and regional surface temperatures: 1889 to 2006. Geophysical Research Letters, 35, L18701. https://doi.org/10.1029/2008GL034864

Lean, J., & Rind, D. (2009). How will Earth's surface temperature change in future decades? Geophysical Research Letters, 36, L15708. https://doi.org/10.1029/2009GL038932

Lean, J.L. (2018). Observation-based detection and attribution of 21st century climate change. WIREs Climate Change, 9:e511. https://doi.org/10.1002/wcc.511

[1] https://www.digitalocean.com/community/tutorials/how-to-install-nginx-on-centos-7

[2] https://www.digicert.com/csr-ssl-installation/nginx-openssl.htm

9 Appendix A: User Documentation

With the Empirical Climate Model Web Application, the user can:

1. Compare the observed record of Earth's mean monthly temperature with the best-fit model for the mean monthly temperature (Lean, 2018).
2. Use adjustable slide bars (or type value in box) to test the contributions of each forcing to the best-fit model. For example, a value of "0" indicates no contribution from the forcing. A value of "2" means twice the contribution in the adjustable model, compared with the best-fit model.
3. Compare contributions of each forcing in ˚C in the adjustable model for every month over nearly 4 decades (data through December, 2018).
4. Use the adjustable model to test a set of real climate claims gleaned from the media or elsewhere regarding factors affecting Earth's mean temperature. Testable claims
5. *Advanced*: Test ECM performance between forcings and mean global temperature response, Ta, when monthly lags in the temperature response are introduced.

6. *Advanced*: Test ECM performance between forcings and mean global temperature response, Ta, when the forcing signal is stretched or attenuated (i.e., a single month's forcing is distributed over several subsequent months).

10 Appendix B: Deployment Instructions

10.1 Setting up the Development Environment

For the development environment, you will need a version > 6 of Node.js. You will also need Git. You will also need access to the repository, since it is private. If you're a developer who wants to enhance or maintain the application, reach out to one of us, and we'll add you as a collaborator.

To set up development environment, clone the repository:

```
git clone https://github.com/clararichter/climate_model_app.git
```

The list of dependencies required for the project can be found in the package.json file at the root of the repository. To install the dependencies, run:

```
npm install
```

To compile the application, and to run locally, run:

```
npm run serve
```

If compilation was successful, the output will say so, and it will notify that application runs at http://localhost:8080/.

10.2 Installing The Product

If you wish to launch this application on a web server, you can find instructions to do so in this section.

10.2.1 Creating A New Server With The Application

10.2.1.1 Prerequisites

You will need a version of Node.js with a version > 6 as well as Git. You will need a web server; we used Nginx in deploying the application to a Colorado College server.

10.2.1.2 Procedure

Clone the repo on your server machine:

```
git clone https://github.com/clararichter/climate_model_app.git
```

Navigate inside repo and Install dependencies:

```
npm install
```

Compile assets:

```
npm build
```

A directory named “dist” should appear in the repo. Since we installed Nginx on a CentOS operating system, we used the this guide [1] to setup the server. Use the guide to serve the “dist” directory that was just created[4].

10.2.1.3 Optional (Use SSL)

Use this guide [2] to generate and use an SSL certificate with Nginx.

10.2.2 Run Tests

To run unit tests, run the following command:

```
npm run test:unit
```

[4] We soon realized that serving the distribution directory from the same directory where it was built is generally considered bad practice. This is not something we are intending to change, however.

End-to-end tests are not automated, but must be done manually. Refer to the document /tests/manual-e2e-testing.pdf to view a description of the tests.

10.2.3 Software structure

See https://github.com/clararichter/climate_model_app