UNIVERSITY OF CALIFORNIA
Santa Barbara

Renewable Energy Siting Predictors Observed from National Data for Wind and Solar

A Capstone Project submitted in partial satisfaction of the requirements for the degree of
Master of Environmental Data Science
for the
Bren School of Environmental Science & Management

By

Paloma Cartwright
Joe DeCesaro
Daniel Kerstan
Desik Somasundaram

Committee in charge:
Allison Horst
Ranjit Deshmukh
Grace Wu

June 2022

Renewable Energy Siting Predictors Observed from National Data for Wind and Solar

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.

_____

Paloma Cartwright Date

_____

Joseph DeCesaro Date

_____

Daniel Kerstan Date

_____

Desik Somasundaram Date

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:

_____

Dr. Ranjit Deshmukh Date

_____

Dr. Allison Horst Date

Abstract

Climate change is a pressing, global problem. Energy production is one of the major sources of carbon dioxide emissions and the decarbonization of energy is one of the proven pathways to addressing climate change. The predominant strategy for decarbonization of energy sources in the US is to generate electricity using renewable energy like wind and solar. However, one major issue energy developers face is having limited knowledge of the most important factors for determining successful siting of utility-scale onshore wind and solar photovoltaics. A better understanding of these influential factors will save time and resources for the deployment of renewable energy projects. This project aimed to fill this knowledge gap by identifying the important factors for determining successful siting of renewable energy. First data was gathered on the potential determining factors of siting. Next, the relationships between renewable energy siting and important factors were analyzed. Finally, projection maps were generated for the contiguous US that identified a location's favorability for the siting of renewable energy. The models used for this project include lasso regression, logistic regression, random forest, maxent, and geographically weighted regression.

Our models found that the overwhelmingly most important factor for determining wind siting success is capacity factor. We also observed that other factors are not that important compared to wind capacity factor. Our models found that there is not one single factor that is overwhelmingly the most important for utility-scale solar siting favorability. However, the two factors that were consistently the most important are renewable portfolio standard and population density. Future research should continue and refine the work in this project to aid in the rapid decarbonization of energy sources.

Executive Summary

The decarbonization of energy sources to mitigate climate change is one of the greatest challenges facing the world today. There is significant pressure to achieve the increasingly prevalent, highly ambitious clean energy goals. Even where physical conditions are well suited for renewable energy, there is often significant variation in their potential and their use (Pierce et al., 2021). Limited knowledge exists of the determining factors for successful siting of wind and solar energy projects. Without understanding the relationships between renewable energy development and local or regional siting criteria, clean energy policies could result in unintended consequences or end up being less effective (Wu et al., 2020). Dr. Grace Wu piloted a study predicting energy siting locations in the western United States. The objective of this capstone project is to expand on this previous work and determine the most influential factors for successful siting of utility-scale onshore wind and solar photovoltaics for the contiguous United States (US) through three main objectives:

1. Gather and prepare data for each of the factors to be included in the models.
2. Identify and analyze the relationship of renewable energy siting with these factors.
3. Generate projection maps of siting favorability for utility-scale wind and solar power plants using statistical techniques and machine learning algorithms.

To achieve these objectives the team used four processing steps: data gathering and management, creation of a complete raster stack, machine learning analysis, and siting projection map generation.

Data is stored on the Bren School of Environmental Science & Management server, named Taylor, and code is documented on the *energysiting* GitHub repository. Upon completion of this project, the data was transferred to the Client for their storage and continued analysis. The team primarily used R for writing and producing code with some Python used in the data gathering process. To complete some calculations QGIS was used. This only occurred when the processing time in R and Python was not efficient for the task. Any areas where QGIS was used are fully documented with details on the decision to use the software and what functions were performed. Data, code, and analysis were tested thoroughly to ensure that full functionality and reproducibility is achieved.

Data was gathered from various sources to determine their effect on solar and wind siting. Some of the variables include the slope of the land, distance to transmission lines, and capacity factor for the given energy technology. These data were in 7 file types, from 15 sources, and constituted over 60 GB of storage. Once gathered, the data was converted from its file type into a raster. All data were converted to have the same projection, extent, and resolution. Once converted the variables were saved into *.tif* files. The variables were then read into a single raster stack for the analysis.

Among the data was the location data for utility-scale wind and solar. The threshold for utility-scale wind energy was 10 MW and 5 MW for solar. For wind and solar, projects were filtered to only include those from 2017 forward to best align with the time period of the existing dataset for the variables of the analysis. The data for the locations of utility-scale projects is known as presence data. Presence data was made into a raster with the same projection, extent, and resolution as the factors to run the analysis.

To make models for the analysis, pseudo-absence points were generated. Pseudo-absence points are points where solar and wind are currently not located. These points were generated on a one-to-one scale based on the presence data. The area of each project present was calculated and a pseudo-absence polygon was created of the same area. The Client provided a dataset for both wind and solar called “site

suitability". Site suitability was distinct for each energy technology and provided locations of where wind and solar projects could possibly be sited. Based on the site suitability data, pseudo-absences were randomly distributed within these areas. The data was masked so that a pseudo-absence could not be placed where presence data occurred. Pseudo-absence data was made into a raster with the same projection, extent, and resolution as the variables to run the analysis.

Once presence and pseudo-absence data were finalized, zonal statistics were calculated for each project. For the purposes of this study, the zonal statistics are the average of each factor's value within the boundaries of the project. Using these averages we were able to analyze the effect of a variable on the project in our models.

Five different modeling techniques were used for the analysis of factors on renewable energy siting. These include logistic regression, random forest, Maxent, Lasso regression, and geographically weighted regression (GWR). GWR was used to understand how a factor's effect changed across space. All other model types were used to determine the most important siting factors and to generate projection maps of siting suitability. The team chose to use different machine learning methods to analyze which would be the most accurate and be able to compare results between models.

To make the models the R package "caret" was used. This package allows the use of multiple machine learning algorithms using the same package platform. The package is also used to conduct k-fold cross-validation and to assess the accuracy of the models made for this analysis.

For utility-scale wind energy, the most important contributing factor to predicting siting suitability is the wind capacity factor. This factor was chosen as the most important factor in all models. As a result, areas like the midwest were deemed the most favorable for wind energy. When making projection maps for the contiguous US, a range of models was considered. Using the receiving operating characteristic (ROC) performance metric, random forest was determined to be the highest-performing while maxent was determined to be the lowest-performing.

For utility-scale solar energy, the most important factors contributing to site suitability are a state's renewable portfolio standard and the population density of the area. However, these factors were not overwhelmingly important like the capacity factor for wind energy. After these, multiple models gave high importance to regionality factors. Location favorability was more evenly distributed across the country for solar compared to wind. The largest concentrations of favorable areas for solar siting are along the west and east coasts. The least favorable areas are found in the Wyoming/Montana region.

Future work should expand on the processes established in this project. For example, more variables could be added to the analysis such as the electricity price of the area or if the state's renewable portfolio standard has a solar-specific carveout. The time range of included projects could be expanded to include other years to get a larger dataset of utility-scale projects. In the future, projects that are below utility-scale can be included to understand if the siting factors change with various scales of energy production. Also, this work could be applied to other regions with new data as the workflow should remain consistent. The code used for this project is all available on the *energysiting* GitHub repository. With the code from this project, future analysis can be conducted swiftly as reproducibility was an important consideration throughout this work.

Table of Contents

Renewable Energy Siting Predictors Observed from National Data for Wind and Solar **1**

Abstract **3**

Executive Summary **4**

Table of Contents **6**

Problem Statement **7**

Specific Objectives **7**

Summary of Solution Design **7**

Design and Implementation Plan (DIP) 7

Technical Documentation 7

Data and Metadata 8

Industry Survey 8

Software and Tools 8

Repository 8

Products and Deliverables **9**

Summary of Testing **10**

Testing Data 10

Testing Algorithms & Analyses 10

User Documentation **10**

Documentation 11

Pre-Processing 11

Analysis 16

Dashboard 20

Archive Access **23**

References **24**

Appendix **26**

Problem Statement

Climate change is a growing threat and one of the ways we can mitigate the impacts is through rapid decarbonization of energy sources (Wu et al., 2020). Renewable energy provides a technologically proven pathway to reach decarbonization goals. Studies have shown the need for both wind and solar photovoltaics (PV) to meet decarbonization goals (Williams et al., 2021). Building out these power generation sources can be conflicting with other societal goals such as agricultural production and biodiversity protection. To build sufficient power plants, significant resources are spent by utilities and developers to find suitable areas. However, limited knowledge exists pertaining to the determining factors for successful siting of utility-scale onshore wind and solar PV energy projects. Identifying the most influential determinants of renewable energy siting would inform better electricity system planning, better use of limited resources in the siting of future power plants, and allow for more rapid decarbonization of energy sources. Further, the expansion of renewable energy and the electrification of other areas of society will require a larger capacity than the current electrical infrastructure. Electrical infrastructure can take significantly longer to build compared to renewable energy power plants. A better understanding of successful renewable energy siting can help streamline the planning of necessary electrical infrastructure aiding in the planning of a proactive buildout.

Specific Objectives

The objective of this capstone project is to expand on the pilot study by Dr. Grace Wu (the Client) and determine the most influential factors for successful siting of utility-scale onshore wind and solar photovoltaics for the contiguous United States (US) through three main objectives:

1. Gather and prepare data for each of the factors to be included in the models.
2. Identify and analyze the relationship of renewable energy siting with these factors.
3. Generate projection maps of siting favorability for utility-scale wind and solar power plants using statistical techniques and machine learning algorithms.

Summary of Solution Design

The general process used to carry out this study is summarized below and detailed in the User Documentation. All processing during the project involved technical reporting and documentation. Coworking was facilitated through GitHub and documents on the *energysiting* organization in the project repositories. The analysis methods are described in detail in the User Documentation section of this document.

Design and Implementation Plan (DIP)
The DIP is a Bren deliverable and was finalized and submitted at the end of the Winter quarter to be archived by the Bren School. Faculty Review Presentations were on March 2nd, 2022 where students received feedback from Bren Faculty on the DIP. Any necessary changes from this presentation were considered and implemented in the DIP prior to submission.

Technical Documentation
The technical documentation, this document, provides information on the project's objectives, solution design, and products. It also details the testing of the product and user documentation for future use.

Data and Metadata
Literature provided by the Client was reviewed by students to inform the analysis. Datasets were procured based on the pilot study, literature review, and recommendations from the Client and Faculty Advisor. Students, with the guidance of the Faculty Advisor and Client, developed a survey that was sent to industry professionals to gather industry knowledge.

Data and metadata were downloaded from various sources and stored on the Taylor server at Bren. Data were cleaned and formatted in RStudio and QGIS. A table of the data sources, the variables they represent, and the file structure is provided in Table II of the Appendix. A *metadata.md* file was created which describes the datasets used in the project as well as the links to the preprocessing scripts for each environmental variable. This allows the user to immediately view the process taken to create the raster layer from the raw data. Each of these variables is visualized in the "Variables" tab on the energysiting dashboard.

Industry Survey
A survey was prepared by the team to gain insights from wind and solar developers about the most important factors to successful siting based on their firsthand experience. The survey was approved by the UCSB Human Subjects Committee on 02/18/2022. The survey included questions such as "Please score the following factors on a scale of 1 to 5 based on their importance when choosing locations for wind or solar project development." It was distributed to industry professionals however, low response rates did not allow for this portion of the project to advance further. The contents of the industry survey can be managed from Qualtrics Survey Link. The Client has been given access to the Qualtrics survey as a collaborator and necessary additional material to proceed with this portion of the project in the future if desired.

Software and Tools
To complete this project, RStudio, Anaconda Navigator with JupyterLab, QGIS and GitHub were used for the organization of coding materials. The *energysiting* GitHub organization holds any coding materials used during the project and a GitHub project board organized weekly tasks for students. ZenHub tracked tasks for each student. Zoom was used to facilitate weekly group meetings with the Client and Faculty Advisors if it was not in person. Slack was used for updates to the client and faculty advisor as well as for communication within the team. Google Drive organized shared materials and other Google Suite tools facilitated co-working on projects. Zotero was used to keep track of anything referenced in this project. The Bren School's server, Taylor, contained all of the data used on the project until it was transferred to the client. RStudio and Jupyter on the Taylor server ensured the environment for the code is consistent across all machines. All software and tools being used during the project are open access and can be downloaded for free.

Repository
Students maintained all coding processes in a GitHub repository which will be submitted to the Client upon completion of the Capstone project. This repository is being organized into three main folders;

1. *docs,* which has documentation information for the project, like metadata and user access information,
2. *preprocessing,* containing all of the scripts used in the preprocessing of the variables included in the project, and
3. *analysis*, which contains the scripts used in the analysis and completion of the project.

There are two other folders that are not critical to the analysis: data_downloads and archive.

1. *data_downloads*, contains any code that was used to access or download data used in the analysis. Few datasets required any downloading code.
2. *archive*, contains old code that has since been updated for the analysis or has been dropped for other means of completion.

Products and Deliverables

This section provides a summary of all products and deliverables during the completion of the project.

Table I. Products and Deliverables

Deliverable	**Date Due**	**Delivered to School/Client**	**Description**
Draft Design and Implementation Plan	Feb 10, 2022	School	Draft documentation
Re-Run Pilot Study	Feb 25, 2022	Client	R script
Metadata Documentation	Mar 1, 2022	School	.md file
Design and Implementation Plan Faculty Review Presentation	Mar 2, 2022	School	Presentation and Panel Discussion
Design and Implementation Plan	Mar 11, 2022	School	Completed Document
Industry Survey Analysis (Stretch)	Apr 15, 2022	Client	Qualtrics survey, R script
Revise Machine Learning Algorithm	Apr 15, 2022	Client	R script
Regression Analysis	Apr 25, 2022	Client	R script
Draft Siting Projection Maps	Apr 29, 2022	Client	R script, images
Draft Technical Document and Project Repository	Apr 29, 2022	School	Draft documentation, GitHub repository
Final Siting Projection Maps	May 26, 2022	Client	R script, images

Table I. Products and Deliverables			
Deliverable	**Date Due**	**Delivered to School/Client**	**Description**
Capstone Final Presentations	May 26, 2022	School	Presentation
Revised Technical Documentation and Project Repository	Jun 3, 2022	School	Final Document, GitHub repository, data archive information

Summary of Testing

Testing Code
The scripts used in the project were checked for reproducibility across different operating systems (OS). This includes functions that check for the presence of packages and install them if they are not present on the user's machine. At each step of data curation and combination, students performed unit testing on intermediate outputs to ensure the data was combined in correct formats allowing for proper functionality. Code review was performed by the team members to ensure the access to code and documentation is preserved after the project.

Testing Data
Reviews will be conducted as data is downloaded and used to validate that data aligns with the provided metadata and can guide the analysis as is intended. Data types and structures were tested throughout the code to validate that calculations are being performed on the correct data types. Students ensured there was enough presence/absence data being used in the regression to avoid errors due to row-wise deletion. Any changes made to the datasets to handle gaps will be added to the metadata documentation. There is a built-in test telling users how many observations will be used in a regression to ensure that users are aware of list-wise deletion possibilities.

Testing Algorithms & Analyses
Regressions will be checked to ensure that the respective assumptions are satisfied. We will also test for multicollinearity through examination of the correlation coefficients. We will conduct sanity checks on our machine learning model output to ensure that siting predictions are feasible. For example, predicted locations should not include bodies of water or other excluded areas. The performance of the machine learning models will be tested using the metrics of the Receiver Operator Characteristic Curve and the Area Under the Curve (ROC/AUC). K-fold cross validation was conducted for all the machine learning models using the resampling functionality in the caret package. The current analysis uses 10 folds which is generally considered to be sufficient to reduce significant bias.

User Documentation

To successfully complete the project, students organized the *energysiting-main* GitHub repo into three main folders; docs, pre-processing, and analysis. The docs folder contains documentation portions of the project that are more informative rather than coding materials. The pre-processing folder contains the

code scripts used to process the data downloaded from its original formats into usable information for the purposes of this project. The analysis folder contains the code used to conduct the various forms of analysis with the data. All three of these folders and their contents are expanded upon in detail below. Figure 1 shows the order of the workflow from raw data to analysis in a flowchart.

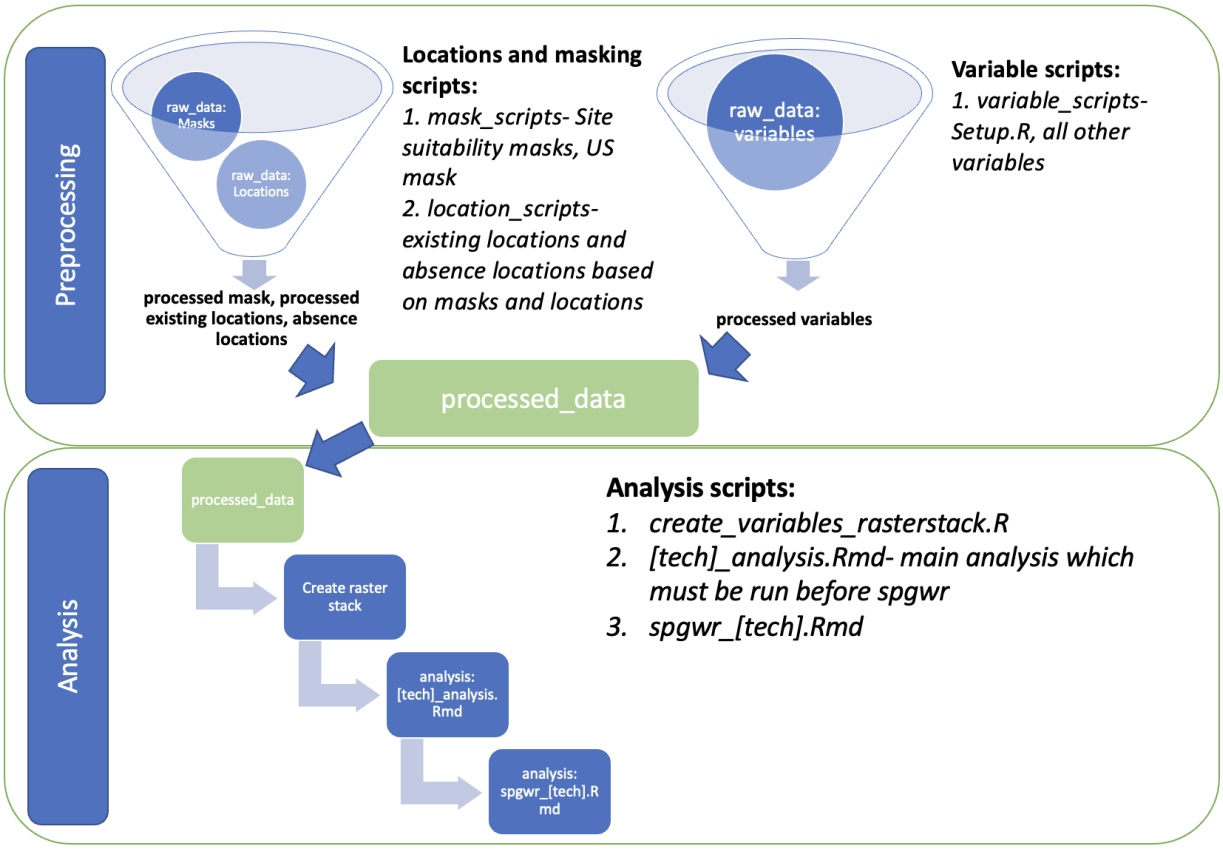

Figure 1: Flowchart of the workflow from raw data to analysis

There are two other folders that are not critical to the analysis in the *energysiting-main* GitHub repo: data_downloads and archive. *data_downloads*, contains any code that was used to access or download data used in the analysis. Few datasets required any downloading code. *archive*, contains old code that has since been updated for the analysis or has been dropped for other means of completion.

The other repositories in the *energysiting* GitHub organization are the *pilot* and the *energysiting-dashboard* repos. The *pilot* repo only contains the code provided by the Client for the purposes of the pilot study and will not be expanded upon further in this document. The *energysiting-dashboard* repo contains code to make the flexdashboard Github page and details on this code can be found below.

Documentation

This information is contained in the *docs* folder in the repository. This folder contains *metadata.md* which is the file containing all of the information on the sources, collection, and preprocessing of the environmental variables used in the project. There is also a document in this folder called

technical_doc.Rmd which is this document. It also contains the references.bib file which has citation information to be used in the technical_doc.Rmd file. The docs folder also contains *zero_to_postGIS.md* which contains instructions on how to set up a PostGIS database of Open Street Maps data as used for the roads dataset in this study.

Pre-Processing

The *preprocessing* folder on GitHub is divided into three sub-folders; *variable-scripts*, *location-scripts and mask_scripts.* These will be described in detail below.

- Variable Scripts (*variable_scripts*): detailed information about the variables represented in the processes described below, please refer to the *metadata.md* file.
  - *setup.R*
    - This script establishes all of the common information in the preprocessing scripts in the *energysiting-main* repository. This script is sourced using the *here* package in all other scripts so users must ensure they are working in an R Project to ensure there are no file path issues when trying to run the remainder of the preprocessing scripts. When users git clone the repository, the R Project should be generated automatically.
    - The base projection is set to geodetic database code "EPSG:5070", which is the coordinate reference system (CRS) used in our project. This will be the projection throughout the analysis. The base shapefile of the US was downloaded from the National Weather Service website and filtered to remove territories and states not in our study area. This was used to create the extent box for the area of interest and rasterized.
  - *acquisition.R*
    - This file sources the *setup.R* script first. Then it reads in the land acquisition geoTIFF file, projects it to the correct CRS, and then masks it to our area of interest. The file is then saved and outputted to the *processed_data* folder on Taylor.
  - *env_exclusions.R*
    - This file sources the *setup.R* script and then reads in the environmental exclusions geoTIFF file, projects it to the correct CRS, and masks it to the area of interest. The file is then saved and outputted to the *processed_data* folder on Taylor.
    - This file also contains a note about preprocessing which had to be done in ArcGIS due to the format the data was in when it was provided to the team.
  - *pop_density.R*
    - This file sources the *setup.R* script and then reads in the population density ADF file. The file is wrapped so that it can be stored correctly due to the file type. It is then reprojected to the CRS of the base raster and cropped to the area of interest. The file is then saved and output to the *processed_data* folder on Taylor.
  - *regions.R*

        - This file sources the *setup.R* script and makes a list of the states found in each region. The aoi_state object from the *setup.R* script is then used to make columns for each region where the column has a value of 1 if the state is in that region and a value of 0 if it is not. The region object is vectorized then rasterized separately by each region. The region objects are masked to the area of interest and then saved separately and output to the *processed_data* folder on Taylor.
    - *roads_qgis_*
        - There are two files that begin with the same extension due to the steps taken to process this data.
            - *roads_qgis_input.R*
                - The geopackage containing data for all roads in the US was read into R and projected into EPSG 5070. The data was then converted to a spatial vector and rasterized before it was masked to the area of interest.
                - The file was then saved and outputted to the *qgis_inputs* folder on Taylor.
            - Because of the superior computing power of QGIS (seconds compared to days in R), this output file was passed to QGIS to calculate the euclidean distance to a road for each cell. When the raster was exported from QGIS, caution was taken to ensure the extent matched the area of interest. This file was saved on the *qgis_ouputs* folder on Taylor.
            - *roads_qgis_output.R*
                - This outputted geoTIFF file from QGIS was then read into R, rasterized, masked, and outputted to the *processed_data* folder on Taylor.
    - *rps.R*
        - The csv file containing the renewable portfolio standard (RPS) for each state in the continental US was read into R and joined with the *aoi_state* table which was created in the *setup.R* file. This was done to add geometries to each of the states in the RPS list. This was then vectorized and then rasterized before masking for our area of interest. During rasterizing, the *fun* argument was set to *max* to ensure the larger RPS target was chosen in the case where one grid cell extends over more than one state boundary. The file is then saved and outputted to the *processed_data* folder on Taylor.
    - *slope.R*
        - The tiff file with global elevation data was read in and reprojected to EPSG 5070. The data was then masked to include only the area of interest. This file was then outputted to the *intermediate_files* folder on Taylor so that it can be read in as a stars raster. This was done so that slope could be computed using the *slope* function from the *starsExtra* package. The file is then saved and outputted to the *processed_data* folder on Taylor.
    - *solar_capacity.R*

        - The input tiff file for solar capacity was read into R, and reprojected into EPSG:5070 before being masked to the area of interest. The file is then saved and outputted to the *processed_data* folder on Taylor.
    - *substations110_qgis_*
        - Similar to the roads data, there are two files that begin with the same extension due to the steps taken to process this data.
            - *substations110_qgis_input.R*
                - The shapefile containing data for all substations in the US was read into R, filtered to only include substations with a minimum voltage of 110V which was the chosen threshold for our project, and projected into EPSG 5070. All of the extra information was then removed, just keeping the geometry information, and all rows with empty geometries were removed. The data was then vectorized and rasterized before it was masked to the area of interest.
                - The file was then saved and outputted to the *qgis_inputs* folder on Taylor.
            - Because of the superior computing power of QGIS (seconds compared to days in R), this output file was passed to QGIS to calculate the euclidean distance to a substation for each cell. When the raster was exported from QGIS, caution was taken to ensure the extent matched the area of interest. This file was saved on the *qgis_ouputs* folder on Taylor.
            - *substations110_qgis_output.R*
                - This outputted geoTIFF file was then read into R, rasterized, masked, and outputted to the *processed_data* folder on Taylor.
    - *svi_*
        - There are four files in the folder that begin with the svi_ extension. This is because each of these scripts are processing a different category from the social vulnerability index dataset. Each of these files followed the same steps for processing.
        - The shapefiles were read in and all potential columns of interest were extracted. The –999 values in the data were replaced with NA. The data was then reprojected to EPSG:5070 before the geometries were bootstrapped with the dataset. The indicator of interest was selected and vectorized then rasterized. The files were then saved and outputted to the *qgis_inputs* folder on Taylor.
            - *svi_overall.R*
                - Indicator of interest: “SPL_THEMES”
            - *svi_pci.R*
                - Indicator of interest: “E_PCI”
            - *svi_socioeconomic.R*
                - Indicator of interest: “SPL_THEME1”

- *sci_unemp.R*
    - Indicator of interest: “EP_UNEMP”
- *transmission110_qgis_*
    - Similar to the roads data, there are two files that begin with the same extension due to the steps taken to process this data.
        - *transmission110_qgis_input.R*
            - The shapefile containing data for all transmission lines in the US was read into R, filtered to only include transmission lines with a minimum voltage of 110V which was the chosen threshold for our project, and projected into EPSG 5070. All of the extra information was then removed, just keeping the geometry information. The data was then vectorized and rasterized before it was masked to the area of interest.
            - The file was then saved and outputted to the *qgis_inputs* folder on Taylor.
        - Because of the superior computing power of QGIS (seconds compared to days in R), this output file was passed to QGIS to calculate the euclidean distance to a substation for each cell. When the raster was exported from QGIS, caution was taken to ensure the extent matched the area of interest. This file was saved on the *qgis_ouputs* folder on Taylor.
        - *transmission110_qgis_output.R*
            - This outputted geoTIFF file was then read into R, rasterized, masked, and outputted to the *processed_data* folder on Taylor.
- *wind_capacity.R*
    - The input tiff file for wind capacity was read into R, and reprojected into EPSG:5070 before being masked to the area of interest. The file is then saved and outputted to the *processed_data* folder on Taylor.

➢ Location Scripts (*location_scripts)*

- *solar_hull_location.R*
    - This file sourced the *setup.R* script and then read in the solar location geojson file from the *Kruitwagen* study. This file contained global predicted solar locations. The geojson file was read into R using the *sf* package, transformed to EPSG:5070, and cropped to the area of interest. Locations of projects are then filtered for capacities greater than 5 MW as established in conversations with the Client. The location geometries were then buffered by 500 meters under the advice of the Client for the mask made to be included along with site suitability data. The locations points used for analysis were not buffered as discussed with the Client. The *unique_id*, *capacity_mw*, and *geometry* columns were selected from the data to make it streamlined. The location's data was also filtered for projects installed in 2017 or later. Using the *sf* package functions *st_convex_hull* and *st_union*, the locations were combined and then a hull was drawn around

each individual project id. For the location points an *area_m2* column was added to the data frame so that it could be used later to generate area-equivalent pseudo-absence points. With the hulls made for each project, the locations were then rasterized using the *terra* package and output to the *processed_data* folder on Taylor.

- *wind_hull_location.R*
  - This file sourced the *setup.R* script and then read in the wind location csv from the US Wind Turbine Database (USWTDB) with the added project IDs by the Berkeley Lab. The csv was then made into an *sf* object by identifying the longitude and latitude coordinates and setting the CRS as EPSG:4326. The metadata for this dataset did not contain a CRS so the team discussed with the Client and determined the CRS was EPSG:4326 based on industry standards. The locations were then reprojected to EPSG:5070 and cropped to our area of interest. The location data was then cleaned and filtered for those with project capacities greater than 10 MW as discussed with the Client. Location points were buffered by 2000 meters as discussed with the Client for the mask made to be included along with site suitability data. The locations points used for analysis were buffered by 500 meters as discussed with the Client. The individual turbines were grouped by project ID (*p_id*), their mean capacity calculated, and cleaning further. The location's data was also filtered for projects installed in 2017 or later. Using the *sf* package functions *st_convex_hull* and *st_union* the locations are unioned and then a hull was drawn around each individual project id. For the location points an *area_m2* column was added to the data frame so that it could be used later to generate area-equivalent pseudo-absence points. With the hulls made for each project, the locations were then rasterized using the *terra* package and output to the *processed_data* folder on Taylor.
- solar_absence_location.R
  - This file sourced the setup.R script and read in three main inputs: the buffered mask of existing solar locations, solar site suitability mask, and solar location data with the radii of absence locations to be generated. The inputs are used to create the region of interest to sample the absence points from. A custom random points function is used to generate absence points of equivalent area for each existing location using the randomPoints function in the *dismo* package. One important note is that the n for the randomPoints is set to be 4 times n to get enough points as per the solution specified here: dismo::randomPoints generating fewer points than requested in R - Geographic Information Systems Stack Exchange. The absence points are saved as points, polygons and raster.
- wind_absence_location.R
  - This file sourced the setup.R script and read in three main inputs: the buffered mask of existing wind locations, wind site suitability mask and wind location data with the radii of absence locations to be generated. The inputs are used to create the region of interest to sample the absence points from. A custom

random points function is used to generate absence points of equivalent area for each existing location using the randomPoints function in the *dismo* package. One important note is that the n for the randomPoints is set to be 4 times n to get enough points as per the solution specified here: dismo::randomPoints generating fewer points than requested in R - Geographic Information Systems Stack Exchange. The absence points are saved as points, polygons and raster.

- Mask Scripts (mask_scripts)
  - site_suitability_solar.R
    - This file sourced the setup.R script and rasterized the site suitability layer for solar before reprojecting it into EPSG:5070. The script ensures that the domain of the data is within the US base raster. Finally, the site suitability mask raster is saved for further analysis.
  - site_suitability_wind.R
    - This file sourced the setup.R script and rasterized the site suitability layer for wind before reprojecting it into EPSG:5070. The script ensures that the domain of the data is within the US base raster. Finally, the site suitability mask raster is saved for further analysis.
  - US_base_raster.R
    - This script writes the US base raster and shapefile using the terra package.
  - us_map_script.R
    - This script writes US shapefiles with state information.

Analysis

The *analysis* folder contains a single folder called *complete_stack* that contains all R and Rmarkdown files used to complete the factor importance analysis and generate projection maps of siting favorability. To complete this part of the analysis logistic regression, lasso regression, random forest, and Maxent methods were used. Also, this section contains the documentation for the geographically weighted regression (GWR). The GWR was used to understand how the factors affecting siting favorability changes over the contiguous US. Data was split into training and testing datasets to ensure bias is reduced in the model. Cross-validation was also used to assess model performance and compare models to identify the "best" model. Projection maps and analysis plots were saved to the project *dashboard* repo.

- *create_variables_rasterstack.R*
  - This file takes the raster layers produced in the pre-processing section above and combines them into a complete raster stack to run the analysis and generate maps. This file sources the *setup.R* script to get file paths and use necessary packages. Two lists of file paths are made for the raster stacks, one for wind and one for solar energy. A list of names for each variable in the raster stack was then generated to be fed into a function.
  - The function *create_raster_stack* is made to feed in three variables: tech, cov.names, and cov.filePaths. This function uses the factor names and file paths that were generated previously to make the raster stack based on the "tech". The raster stack is generated by first making a list with all the factor tifs and then using the *stack* function. Finally, the generated raster stack is saved to a designated file path with the appropriate technology.

This function is run for solar and wind technologies separately. The *tictoc* package is used to see how long the function takes to complete this last step.

- Line 84 and beyond in this document is scratch code that was explored to try various methods to make the raster stack. This includes a function called *addto_raster_stack* which tries to add variables to an already made raster stack. This function was able to run but it was slower than just remaking a raster stack from scratch as the *create_raster_stack* does. There is also a function called *check_raster_stack* that ensures the names of the variables used in the raster stack match the rasters in the stack. This function is used in other areas of the analysis and does not need to be saved.
- Lines 134 and beyond contain commented-out code that was originally taken from the pilot study script. This is not used for the purposes of our analysis.

➢ *wind_analysis.Rmd*

- The analysis is split into five main sections: Setup, Generate Covariate Data Inputs: Read In Raster Stack, Zonal Stats Calculations, Run Simple Logistic Regression and Caret: ML Methods.
  - Setup
    - The analysis Rmd sourced the *setup.R* script and loaded additional libraries. Some basic inputs such as seed and whether Regions are to be included in the models are required in the first part of the setup. The saved location data for presence and absence locations are read in various formats as required for the analysis. The region of interest, based on site suitability mask and presence locations buffered mask, from which the absence locations were sampled is also read in during this section.
  - Generate Covariate Data Inputs: Read In Raster Stack
    - This section reads in the raster stack and ensures that the variables are as expected through a few quick checks.
  - Zonal Stats Calculations
    - This section conducts zonal stats by variable to get a value associated with each of the presence and absence locations to be saved in a dataframe and stored in a csv for future use.
  - Run Simple Logistic Regression
    - The data is first prepared for the glm model before some exploratory analysis based on term plots and correlation plots.
    - This section builds a generalized linear regression model using the glm packages and shows the resulting coefficients in stargazer table format.
    - The model is then used to generate a prediction raster for the entire United States.
  - Caret: ML Methods
    - The caret package allows for the streamlined process of model building for various machine learning techniques. The variables to be included in the model are specified to be consistent across all the methods.

Additionally, the split between training and test datasets along with the k-fold cross validation parameters are also set to be common amongst glm, lasso, maxent and random forest methods.

- The sections for glm, lasso, maxent and random forest methods model building are very similar except for model-specific parameters that can be modified. All models generate a variable importance plot along with a prediction map for the entire US.
- All four models are then evaluated based on the receiving operating characteristic (ROC) performance metric. The performance results across the various folds within each method are then visualized through a series of plots.

➢ *solar _analysis.Rmd*

- The analysis is split into five main sections: Setup, Generate Covariate Data Inputs: Read In Raster Stack, Zonal Stats Calculations, Run Simple Logistic Regression and Caret: ML Methods.
  - Setup
    - The analysis Rmd sourced the *setup.R* script and loaded additional libraries: "dismo", "sp", "rgdal", "fasterize", "ggplot2", "dplyr", "gdalUtils", "maptools", "rgeos", "stargazer", "randomForest", "ranger", "gdistance", "tmap", "vip", "GWmodel", "RColorBrewer", "ModelMap", "gstat", "corrplot" , "caret", "elasticnet", "caretSDM"
    - Some basic inputs such as seed and whether Regions are to be included in the models are required in the first part of the setup.
    - The saved location data for presence and absence locations are read in various formats as required for the analysis.
    - The region of interest, based on site suitability mask and presence locations buffered mask, from which the absence locations were sampled is also read in during this section.
  - Generate Covariate Data Inputs: Read In Raster Stack
    - This section reads in the raster stack and ensures that the variables are as expected through a few quick checks.
  - Zonal Stats Calculations
    - This section conducts zonal stats by variable to get a value associated with each of the presence and absence locations to be saved in a dataframe and stored in a csv for future use.
  - Run Simple Logistic Regression
    - The data is first prepared for the glm model before some exploratory analysis based on term plots and correlation plots.
    - This section builds a generalized linear regression model using the glm packages and shows the resulting coefficients in stargazer table format.
    - The model is then used to generate a prediction raster for the entire United States.

        - Caret: ML Methods
            - The caret package allows for the streamlined process of model building for various machine learning techniques. The variables to be included in the model are specified to be consistent across all the methods. Additionally, the split between training and test datasets along with the k-fold cross validation parameters are also set to be common amongst glm, lasso, maxent and random forest methods.
            - The sections for glm, lasso, maxent and random forest methods model building are very similar except for model-specific parameters that can be modified. All models generate a variable importance plot along with a prediction map for the entire US.
            - All four models are then evaluated based on the receiving operating characteristic (ROC) performance metric. The performance results across the various folds within each method are then visualized through a series of plots.

➢ *spgwr_wind.Rmd* and *spgwr_solar.Rmd*

- These files generate the geographically weighted regression for wind and solar factors. These two documents largely do the same function but just differ for using wind and solar data. These documents were split so that they could be run simultaneously on the server. The first part of the markdown document largely takes from the analysis markdown documents. This was done so that the GWR analysis can be run separately from all other analysis and be a standalone markdown.
- The *setup.R* script is sourced and the additional required libraries are loaded.
- The location data is then read in for both presence and absence points both as shape files and rasters. The raster variables stack is read in and subset for the desired factors in this analysis. The subset factors can be changed in this file for future analysis. An if statement is used to make sure that there are the correct amount of variables based on the list of variables made and desired for the analysis. Regional analysis is not included because of the nature of geographically weighted regression.
- Zonal stats are then read in for presence locations and pseudo-absence points. Pseudo-absence points are also referred to as background points (bg) as they were in the pilot study. Zonal stats are joined with the spatial points for existing and absence spatial data. To run the GWR all NA values are dropped.
- Pseudo-absence and presence data is then joined into one variable called *[tech]_regData*. This spatially joined data has a column called “treat” that is a 1 for presence points and 0 for pseudo-absences.
- The next portion of the markdown document follows the example for the “spgwr” package that is laid out in the https://rpubs.com/quarcs-lab/tutorial-gwr1, specifically, Chapter 9 - Geographically Weighted Regression.
- First the bandwidth for the GWR is calculated using the function *gwr.sel*. In this function the formula is the variables desired affecting “treat”, the data is the *[tech]_regData* object, and “adapt = TRUE” is used so that the bandwidth changes over regions. Next the

bandwidth that was just calculated is used to run the GWR using the function *gwr*. In the function, the formula is the same as in *gwr.sel* and adapt is set to the bandwidth that was just calculated. For the gwr used in this analysis "hatmatrix = TRUE" and "se.fit = TRUE" as was done in the example.

- Census polygon data was then loaded into the Rmd document and then the results from the GWR were applied to the census geometries via a spatial join. Maps were then made for each factor analyzed using *tm_fill*.
- As the GWR only used presence and absence points for the analysis it does not cover the entirety of the contiguous US so ordinary kriging was used to fill the gaps of data. For each technology a function was made for the kriging call *kriger_[tech]*. This function also saves a raster of the kriged variable. The last part of this Rmd runs the *kriger_[tech]* function for each of the desired variables using a for loop.

Dashboard

The documents described below can be found in the *energysiting-dashboard* repository of the *energysiting* GitHub organization. This repository contains all of the files used in the creation of the public facing dashboard found at https://energysiting.github.io/energysiting-dashboard/.

Description

- *index.html*
  - This file was generated from the index.Rmd file. This file and the associated R markdown file were named "index" so that it could be published using github pages.
- *index.Rmd*
  - This R markdown document contains the code used to generate the dashboard. When altered and knitted, changes are visible on the resulting html document. This R markdown has a specific yaml, and uses a package called *flexdashboard.* After being formatted this way, level 1 headers were then used to create the different tabs that appear at the very top of the dashboard. Rows within these tabs were created using level 2 headers. When paired with {.tabset}, these rows were capable of containing tab subsections. To display content in these tabs, or to display other subsections within rows, level 3 headers were used.
  - Setup chunk
    - The relevant packages were loaded first then a shapefile was read in and assigned the name "aoi" (area of interest). This shapefile was used to establish the background for static maps.
    - Functions:
      - *display*: used to display raster layers
      - *display_continuous*: similar to the display function but contains an extra argument to guarantee the legend of the plot is continuous
      - *display_vect*: used for plotting vector data
      - *display_mapview*: used to display interactive maps
      - *display_mapview_quality*: similar to display_mapview, but with extra arguments to adjust the quality of the map

            - *display_mapview_vect*: for displaying vector data interactively
            - *reso*: returns the resolution of a .tif file
            - *extent*: returns the extent of a .tif file
            - *pro*: returns the projection of a .tif file
            - *units*: returns the map units of a .tif file in meters
            - *type*: returns either "raster" or "vector" depending on the type of the file
            - *dt*: used to create a data table listing out the type, etent, projection, and resolution of a file
            - *display_table*: used to read a .csv file and create a data table using the DT package
        - The path to the geographically weighted regression results was then loaded and called "gwr_path"
            - *display_gwr*: used to display the geographically weighted regression plots.
- Overview
    - This section contains the code that was used to generate the first tab of the dashboard, titled "Overview". In this section, the summary, problem statement, specific objectives, testing, and references sections were written (copied from the technical documentation). This section also included the markdown code used to display pictures of the capstone group.
- Variables
    - This section contains the code used to generate the plots found under the "Variables" tab of the dashboard. The *display* function was used to display each plot, and the *dt* function was used to show plot characteristics including type, extent, projection, and resolution. A brief description of each variable was also included as text. When a new variable was added, the format was copied from the variable directly above, with the name of the file substituted for the new file.
- Locations
    - This section contains the code used to generate the plots found under the "Locations" tab on the dashboard. This tab was broken up into a wind row and a solar row. Within each row, plots were displayed, including a static plot for locations, an interactive plot for locations, a plot of the mask used to generate absence points, and a static plot of absence point locations. All of these plots were generated using the functions established in the setup chunk.
- Wind Analysis
    - This section contains the code used to display the graphs found under the "Wind Analysis" tab on the dashboard. To display the Logistic Regression Model Summary, the *includeHTML* function from the *htmltools* package was used because the table was saved as an .html file. The remaining graphs, excluding the gwr plots, were all saved as .png files, so the markdown code used to display

them on the dashboard was identical to the code used to display any other image. The geographically weighted regression plots were displayed using the function established in the setup chunk with the specific file name used as the argument for that function.

- Wind Prediction
  - This section contains the code used to display static and interactive plots of the prediction results (which had been saved as .tif files). The functions used to display these plots were established in the setup chunk.
- Solar Analysis
  - This section was set up the same as the Wind Analysis section.
- Solar Prediction
  - This section was set up the same as the Wind Prediction section.

Archive Access

Data used by the students for the project is all open access. The open-access data is cited in the references of the final Technical Documentation as required under the Creative Commons licenses provided by the data sources. The data and coding materials used in this project was handed over to the Client upon completion of the project for their continued analysis. The data has been handed over to the Client through their personal Seagate hard drive to be uploaded to their cloud storage accounts.

References

Centers for Disease Control. (2021, August 27). *CDC/ATSDR Social Vulnerability index*. https://www.atsdr.cdc.gov/placeandhealth/svi/data_documentation_download.html

Earth Resources Observation And Science (EROS) Center. (2018). *Shuttle Radar Topography Mission (SRTM) 1 Arc-Second Global* [Tiff]. U.S. Geological Survey. https://doi.org/10.5066/F7PR7TFT

Geofabrik. (2018). *North America Open Street Map Data*. https://download.geofabrik.de/north-america.html

Hoen, B., Diffendorfer, J. E., Rand, J., Kramer, L. A., Garrity, C. P., Roper, A. D., & Hunt, H. (2022). *United States Wind Turbine Database* [Data set]. U.S. Geological Survey. https://doi.org/10.5066/F7TX3DN0

Kruitwagen, L., Story, K., Friedrich, J., Byers, L., Skillman, S., & Hepburn, C. (2021). *A global inventory of solar photovoltaic generating units—Dataset* [Data set]. Zenodo. https://doi.org/10.5281/zenodo.5005868

National Renewable Energy Laboratory. (2021, December). *National Solar Radiation Database*. https://nsrdb.nrel.gov/data-sets/api-instructions.html

National Renewable Energy Laboratory. (2022). *Wind Toolkit Data Downloads API | NREL: Developer Network*. https://developer.nrel.gov/docs/wind/wind-toolkit/

NC Clean Energy Technology Center. (2022). *Database of State Incentives for Renewables & Efficiency®*. DSIRE. https://www.dsireusa.org/

Nolte, C. (2020). *Data for: High-resolution land value maps reveal underestimation of conservation costs in the United States* (Version 4, p. 299063757 bytes) [Data set]. Dryad. https://doi.org/10.5061/DRYAD.NP5HQBZQ9

Pierce, J. C., Krause, R. M., Hofmeyer, S. L., & Johnson, B. J. (2021). Explanations for Wind Turbine Installations: Local and Global Environmental Concerns in the Central Corridor of the United

States? *Energies*, *14*. https://doi.org/10.3390/en14185830

Rose, A. N., McKee, J. J., Sims, K. M., Bright, E. A., Reith, A. E., & Urban, M. L. (2020). *LandScan 2019* (2019th ed.). Oak Ridge National Laboratory. https://landscan.ornl.gov/

The Nature Conservancy. (2021). *Power of Place Renewable Resource Areas and Environmental Exclusions*. https://tnc.maps.arcgis.com/apps/webappviewer/index.html?id=71b0605e44bf475ea55f6d369e668b2c

U.S. Department of Homeland Security. (2021a, September 30). *Electric Substations—Homeland Infrastructure Foundation-Level Data (HIFLD)*. https://hifld-geoplatform.opendata.arcgis.com/datasets/geoplatform::electric-substations/explore?location=14.046296,-9.857173,2.94

U.S. Department of Homeland Security. (2021b, December 23). *Electric Power Transmission Lines—Homeland Infrastructure Foundation-Level Data (HIFLD)*. https://hifld-geoplatform.opendata.arcgis.com/datasets/geoplatform::electric-power-transmission-lines/about

Williams, J. H., Jones, R. A., Haley, B., Kwok, G., Hargreaves, J., Farbes, J., & Torn, M. S. (2021). Carbon-Neutral Pathways for the United States. *AGU Advances*, *2*(1), e2020AV000284. https://doi.org/10.1029/2020AV000284

Wu, G. C., Leslie, E., Sawyerr, O., Cameron, D. R., Brand, E., Cohen, B., Allen, D., Ochoa, M., & Olson, A. (2020). Low-impact land use pathways to deep decarbonization of electricity. *Environmental Research Letters*, *15*(7), 074044. https://doi.org/10.1088/1748-9326/ab87d1

Appendix

Table II. Project Data Sources		
Variable	**Source**	**Data Description**
Land Acquisition	Christoph Nolte (Nolte, 2020)	.tif; 426.4MB
Environmental Exclusion	Provided by the Client from The Nature Conservancy (The Nature Conservancy, 2021)	.gdb; 58.8MB
Population Density	LandScan 2017 High-Resolution Global Population Data Set (Rose et al., 2020)	.tif; 3.9GB
Roads	Open Street Map through Geofabrik (Geofabrik, 2018)	.gpkg; 713.6MB
Renewable Portfolio Standard or Target	Database of State Incentives for Renewables & Efficiency (NC Clean Energy Technology Center, 2022)	CSV; 452B
Slope	USGS - Digital Elevation - Shuttle Radar Topography Mission (SRTM) Model (Earth Resources Observation And Science (EROS) Center, 2018)**)**	.GeoTIFF; 8.3GB
Solar Capacity	National Solar Radiation Database (National Renewable Energy Laboratory, 2021)	.csv; 4.7MB
Electric Substations	Homeland Infrastructure Foundation Level Data - Electric Substations (U.S. Department of Homeland Security, 2021a)	.shp; 55MB
Social Vulnerability Index	US Centers for Disease Control (Centers for Disease Control, 2021)	.shp; 273.2MB
Transmission Lines	Homeland Infrastructure Foundation Level Data - Electric Power Transmission Lines (U.S. Department of Homeland Security, 2021b)	.shp; 283.6MB
Wind Capacity Factor	National Renewable Energy Laboratory - WIND Toolkit (National Renewable Energy Laboratory, 2022)	.tiff; 4.3GB

Solar Unit Locations	Kruitwagen (Kruitwagen et al., 2021)	.geojson; 310.8MB
Wind Farm Locations	US Wind Turbine Database (Hoen et al., 2022)	CSV; 13.7MB