Component Specification

- **Software components**: (High level description of the software components such as: data manager, which provides a simplified interface to your data and provides application specific features (e.g., querying data subsets); and visualization manager, which displays data frames as a plot. Describe at least 3 components specifying: what it does, inputs it requires, and outputs it provides)

    - Basic Python data manipulation tools
        - Pandas: tool for creating dataframes which work with indexed data which can be of many formats
            - Input: CSV to read in
            - Formats and combines data sets so they're consistent for comparison
            - Output: Data-frame and CSV with merged climate opinion, transport habits and census datasets.
        - Numpy: tool for working with numerical arrays.
            - Used mostly for the regression calculations
            - Input: Pandas dataframes
            - Output numpy arrays for use with SciPy
    - Statistical tools (the output will combine out)
        - Scipy: The main functionality of SciPy library is built upon NumPy and its arrays thus make substantial use of NumPy.
            - Input: variables to correlate
            - Functions to use: Stats.linregress and Stats.pearsonr to get the pearson correlation, and determination coefficients as well as the t-test for significance.
            - Output: array containing different coefficients.
        - Seaborn: Seaborn is based on Matplotlib.
            - Inputs: variables to plot.
            - Seaborn will provide the visualization of statistical outputs such as, scatter plots, boxplots between two or more variables.
            - Output: figures and the array obtained from Scipy will be printed on top as text.

    - Visualization manager (with interactive user interface including drop down boxes)
        - Inputs:
            - numpy arrays of regression, correlation, significance values
            - User input of variables to be investigated in drop down box
        - Outputs:
            - Maps showing the relationships between chosen variables on a state by state basis
            - Ideally some way to download static maps as pdf/eps/png/etc.

    For the data visualization manager we are considering the following options:

        - Altair https://altair-viz.github.io/

            - Interactive data visualization tool. Simple declarative syntax and easy to export to a web format. The tool is relatively new so we may encounter bugs.
        - Bokeh https://bokeh.pydata.org/en/latest/
        - Seaborn (see above)
        - Plotly https://plot.ly/python/
            - Possible issues with limited number of calls to API available in free version
            - Has tools and worked examples for maps, and interactive drop-down boxes
        - Folium https://pypi.org/project/folium/
            - Uses leaflet.js, but without the need for dealing with javascript
            - Works in browser easily

- **Interactions to accomplish use cases**: (Describe how the above software components interact to accomplish at least one of your use cases)
    - Use case: Reveal most impactful relationships between climate opinion and demography
        - Data formatted and combined in Pandas
        - This data is is passed to numpy for use with SciPy to calculate regression/correlation coefficients
        - SciPy passes out numpy arrays which are then used with:
            - Seaborn to generate box plots of data
            - Pandas probably? To generate tabulated output values which can be saved.
    - Use case: Reveal relationship between a specific climate opinion and demographic variable (choose by the user)
        - Data formatted and combined in Pandas
        - This data is is passed to numpy for use with SciPy to calculate regression/correlation coefficients
        - SciPy passes out numpy arrays which are then used with:
            - Interactive data visualization tool, which generates maps for investigating the spatial patterns in the relationships between given variables

- **Preliminary plan**: (A list of tasks in priority order)
    - Clean datasets and prepare for merge (done?)
    - Combine the datasets (for now we will merge only Census data and Climate Opinions Survey data) (done?)
    - Calculate correlation/regression/significance of all possible relationships (done?)
    - Plot this and find most significant/interesting
    - Create maps of specific relationships
    - Find way to allow users to select specific relationships and output/plot results