

Interactive Visualization of Cancer Data

AUTHOR: Joseph Ramaswami, IT Analyst/Programmer

SUMMARY

The North Carolina Central Cancer Registry (NCCCR) publishes data on cancer incidence and mortality on its website every year. Historically, the data were published as tables in PDF files. The NCCCR now has interactive data visualization tools on its website that allow the public to interact with plots of cancer incidence and mortality in North Carolina.

CHALLENGE

Many software solutions allow for interactive data visualizations. Some require additional infrastructure like database or application servers. Others require subscription to cloud services. The NCCCR wanted to add interactive data visualization to the NCCCR website without any additional infrastructure or expense.

SOLUTION

Two important characteristics of the cancer data allowed the NCCCR to develop a custom solution to produce interactive data visualizations. First, the amount of data is relatively small. There are just under 40,000 observations. These observations are split across multiple web pages, so the highest number of observations for a single page is just over 3,500. With moderate compression, the data can be sent to the client web browser. Second, the number of possible plots and interactions is limited. The data are distributed across different pages by incidence or mortality, and by category such as sex or race. The web visitor can control three attributes of the data to be plotted: year, cancer site, and a category variable such as male or female.

The open-source programming language Python was used to script the process of generating static HTML and JavaScript files. A script reads and cleans data from Microsoft® Excel workbooks. Another script uses a templating library to generate the static files. Data are embedded directly into the JavaScript files as multidimensional arrays to minimize file size. The data are sent to the visitor's web browser along with the HTML web page. The open-source JavaScript library Plotly was used to plot the data in the visitor's web browser. Some interaction behavior comes built into Plotly, but custom JavaScript code allows the visitor to use web controls to change the plots. This solution produces static HTML and JavaScript files that can be published to the NCCCR's web server without any additional infrastructure or expense.

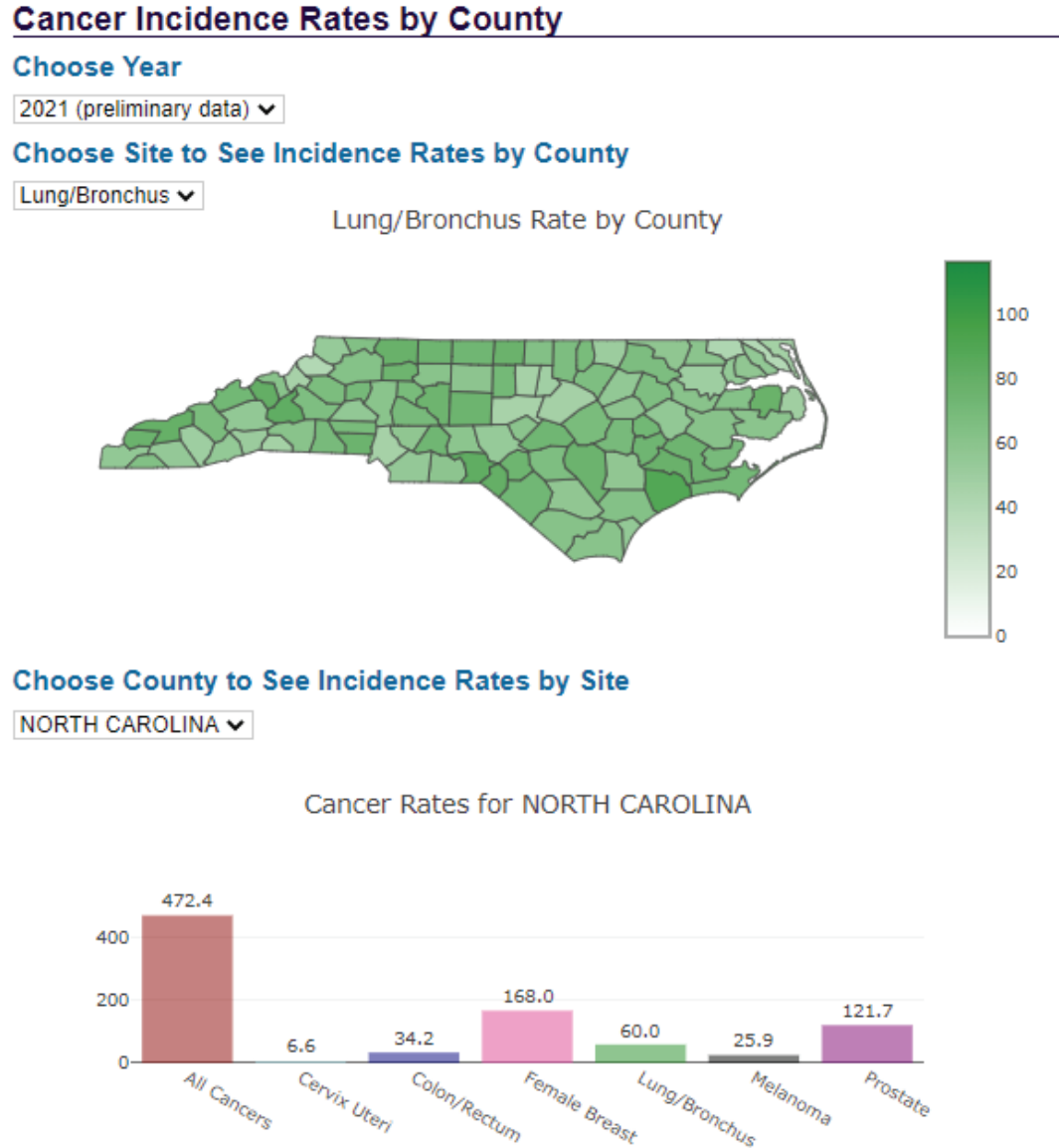
RESULTS

The NCCCR now has data visualization tools available without the additional expense and administrative overhead of extra infrastructure or subscription services. An example of a plot is shown in Figure 1.

CONCLUDING REMARKS

Using Python scripts to perform all of the data ingestion and generation tasks makes the entire process repeatable. When data are updated or added, the Python scripts can be run to generate a new set of static HTML and JavaScript files ready to publish.

Figure 1.



U.S. Department of Health and Human Services
Centers for Disease Control and Prevention