Instructions Module 14

**Background**

In this assignment, you will build an interactive dashboard to explore the Belly Button Biodiversity dataset Links to an external site., which catalogs the microbes that colonize human navels.

The dataset reveals that a small handful of microbial species (also called operational taxonomic units, or OTUs, in the study) were present in more than 70% of people, while the rest were relatively rare.

**Before You Begin**

1. Create a new repository for this project called belly-button-challenge. Do not add this Challenge to an existing repository.
2. Clone the new repository to your computer.
3. Inside your local git repository, copy the files from in the StarterCode folder contained within the Module 14 Challenge zip file. i.e. index.html, samples.json, and the static folder.

NOTE

You will not be required to access the samples.json file locally, but it is provided for reference.

1. Push the above changes to GitHub.
2. Deploy the new repository to GitHub Pages.

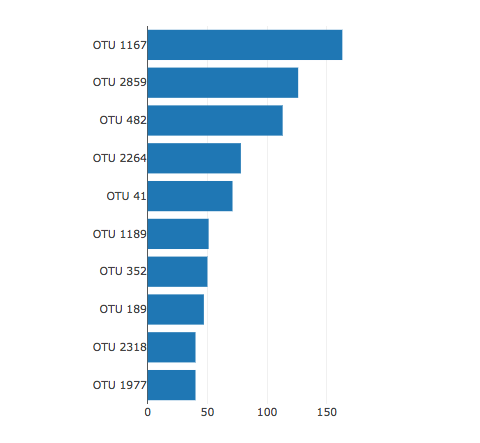
**Instructions**

**Complete the following steps:**

1. Use the D3 library to read in samples.json from the URL https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-classroom/v1.1/14-Interactive-Web-Visualizations/02-Homework/samples.json.
2. Create a horizontal bar chart with a dropdown menu to display the top 10 OTUs found in that individual.

* Use sample\_values as the values for the bar chart.
* Use otu\_ids as the labels for the bar chart.
* Use otu\_labels as the hovertext for the chart.

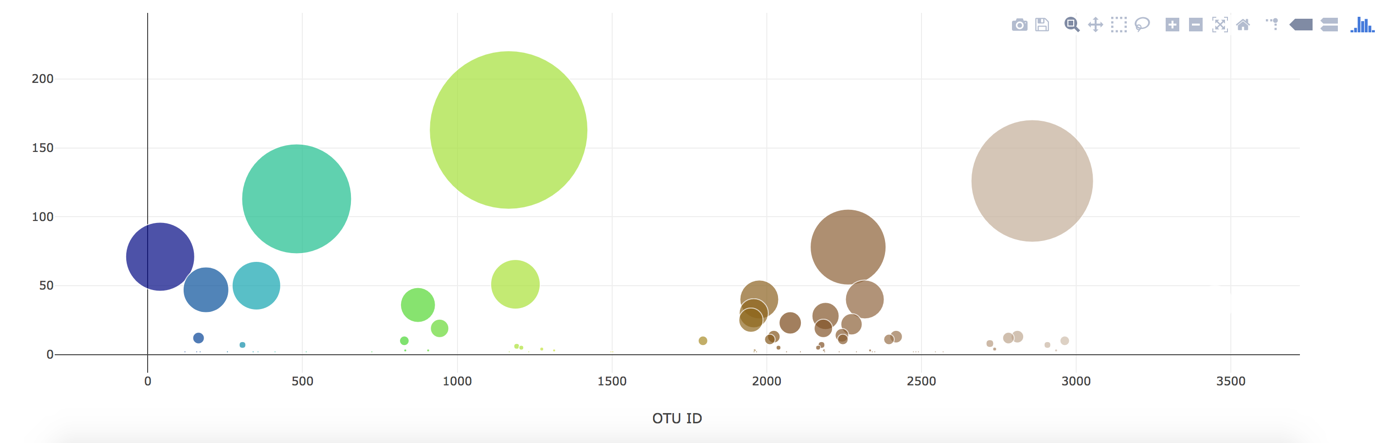
bar Chart



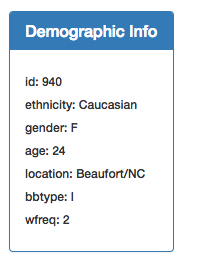
1. Create a bubble chart that displays each sample.

* Use otu\_ids for the x values.
* Use sample\_values for the y values.
* Use sample\_values for the marker size.
* Use otu\_ids for the marker colors.
* Use otu\_labels for the text values.

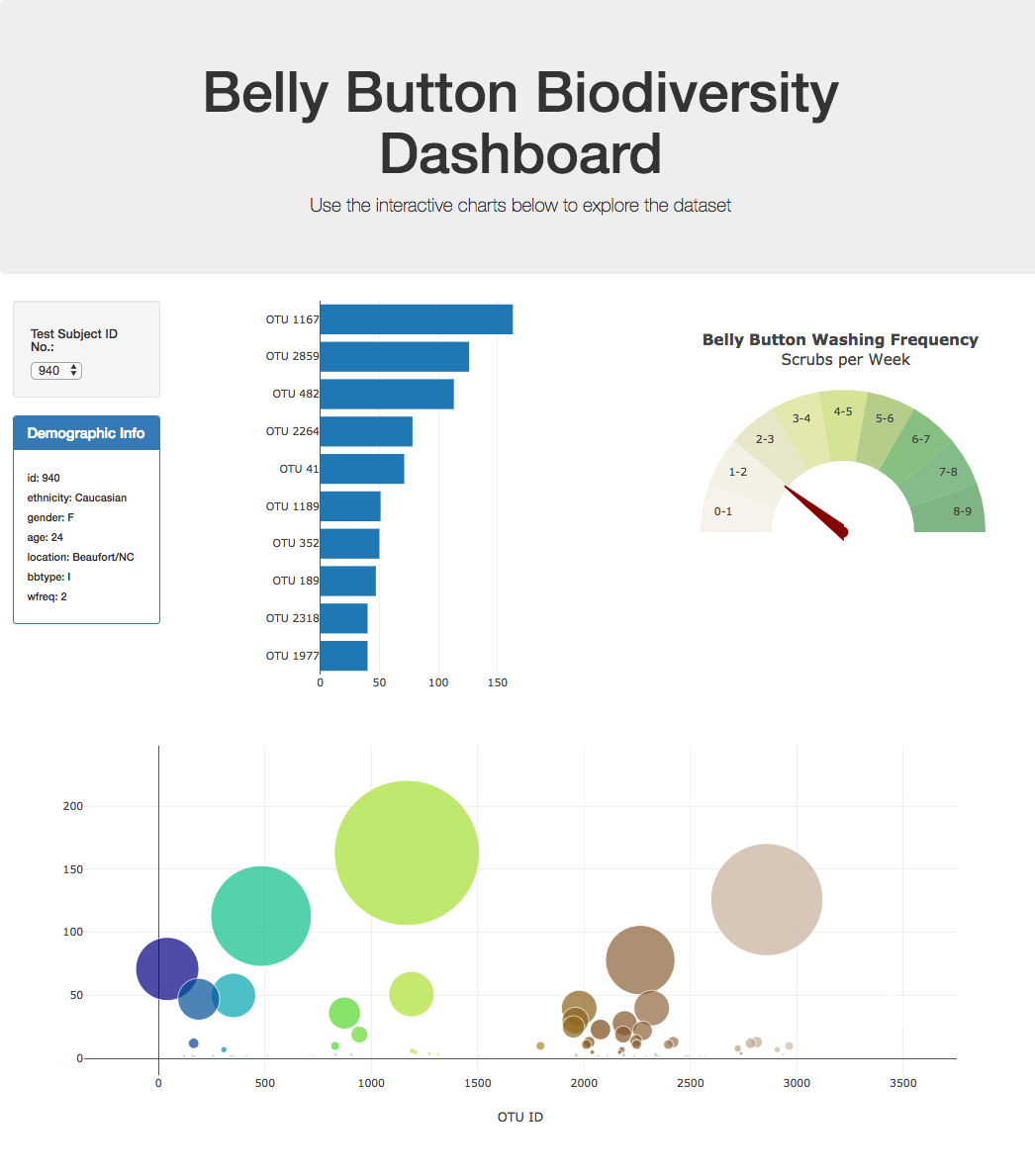
Bubble Chart



1. Display the sample metadata, i.e., an individual's demographic information.
2. Display each key-value pair from the metadata JSON object somewhere on the page.



1. Update all the plots when a new sample is selected. Additionally, you are welcome to create any layout that you would like for your dashboard. An example dashboard is shown as follows:



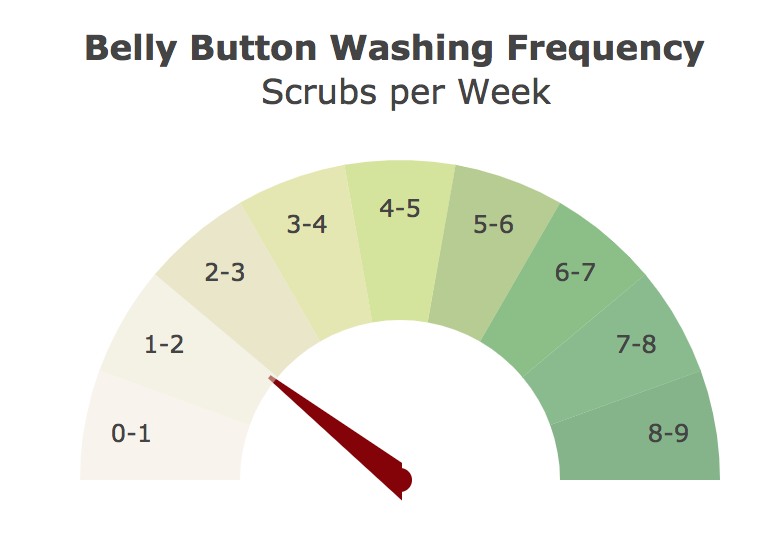
1. Deploy your app to a free static page hosting service, such as GitHub Pages. Submit the links to your deployment and your GitHub repo. Ensure that your repository has regular commits and a thorough README.md file

**Advanced Challenge Assignment (Optional with no extra points earning)**

The following task is advanced and therefore optional.

* Adapt the Gauge Chart from https://plot.ly/javascript/gauge-charts/ Links to an external site. to plot the weekly washing frequency of the individual.
* You will need to modify the example gauge code to account for values ranging from 0 through 9.
* Update the chart whenever a new sample is selected.

Weekly Washing Frequency Gauge



Hints

* Use console.log inside of your JavaScript code to see what your data looks like at each step.
* Refer to the Plotly.js documentation Links to an external site. when building the plots.

**Requirements**

**Bar Chart (30 points)**

* Chart initializes without error (10 points)
* Chart updates when a new sample is selected (5 points)
* Chart uses Top 10 sample values as values (5 points)
* Chart uses otu\_ids as the labels (5 points)
* Chart uses otu\_labels as the tooltip (5 points)

**Bubble Charts (40 points)**

* Chart initializes without error (10 points)
* Chart updates when a new sample is selected (5 points)
* Chart uses otu\_ids for the x values (5 points)
* Chart uses otu\_ids for marker colors (5 points)
* Chart uses sample\_values for the y values (5 points)
* Chart uses sample\_values for the marker size (5 points)
* Chart uses `otu\_labels for text values (5 points)

**Metadata and Deployment (30 points)**

* Metadata initializes without error (10 points)
* Metadata updates when a new sample is selected (10 points)
* App Successfully Deployed to Github Pages (10 points)

**Grading**

This assignment will be evaluated against the requirements and assigned a grade according to the following table:

Grade Points

A (+/-) 90+

B (+/-) 80–89

C (+/-) 70–79

D (+/-) 60–69

F (+/-) < 60

**Submission**

To submit your Challenge assignment, click Submit, and then provide the URL of your GitHub repository for grading.

NOTE

You are allowed to miss up to two Challenge assignments and still earn your certificate. If you complete all Challenge assignments, your lowest two grades will be dropped. If you wish to skip this assignment, click Next, and move on to the next Module.

Comments are disabled for graded submissions in BootCamp Spot. If you have questions about your feedback, please notify your instructional staff or your Student Success Manager. If you would like to resubmit your work for an additional review, you can use the Resubmit Assignment button to upload new links. You may resubmit up to three times for a total of four submissions.

IMPORTANT

It is your responsibility to include a note in the README section of your repo specifying code source and its location within your repo. This applies if you have worked with a peer on an assignment, used code in which you did not author or create sourced from a forum such as Stack Overflow, or you received code outside curriculum content from support staff such as an Instructor, TA, Tutor, or Learning Assistant. This will provide visibility to grading staff of your circumstance in order to avoid flagging your work as plagiarized.

If you are struggling with a Challenge or any aspect of the curriculum, please remember that there are student support services available for you:

Office hours facilitated by your TA(s)

Tutor sessions (sign up Links to an external site.)

Ask the class Slack channel/get peer support

AskBCS Learning Assistants

References