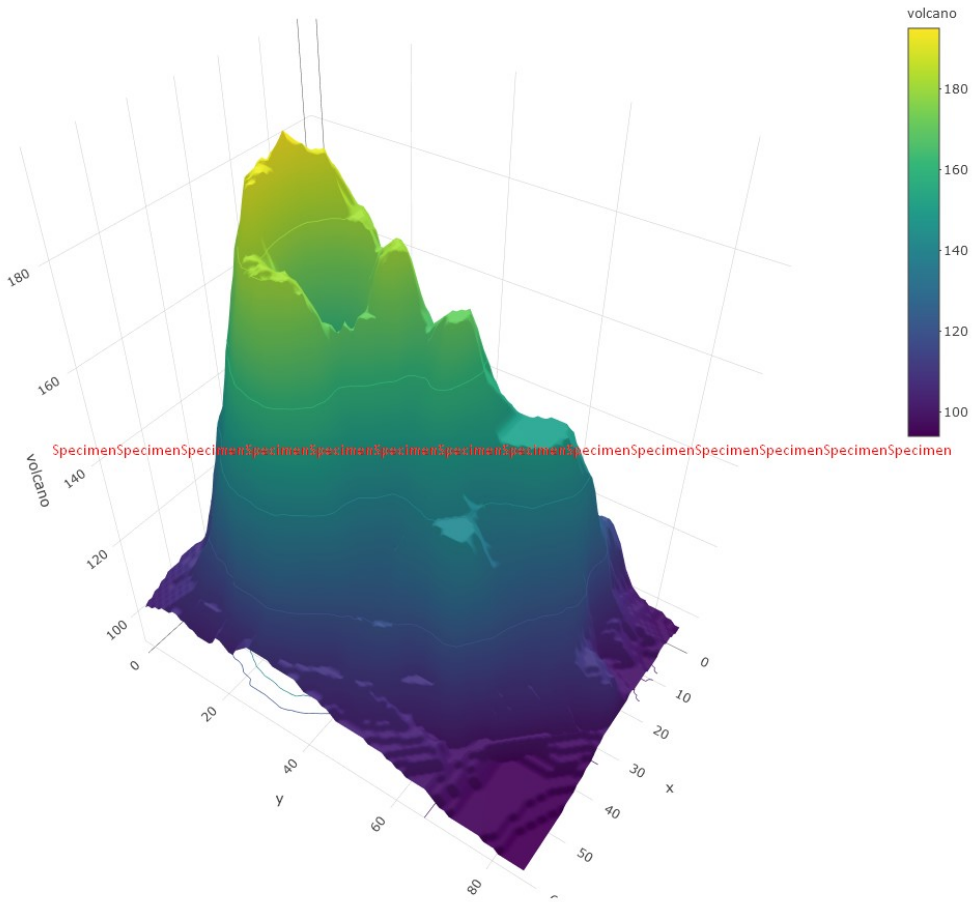
W5 Exercise: 3D Mapping

OBJECTIVE. Use the third dimension for mapping. The goal is to create an interactive 3D scene using geodata. First, you will work a tutorial using provided curated datasets, then you'll venture out to find your own data to visualize in 3D. Total: 10+5pts.

In this exercise, you will use R and Plotly to prepare a 3D visualization. Such a map uses the third dimension (besides the two planar dimensions x and y, or lat and lon) to encode a geographic attribute. An example of such a map is given below for Maungawhau, a volcano in Mount Eden, a suburb of Auckland, NZ.



Step 1: Create a 3D visualization of Maunghawau. Open RStudio, navigate to *File → New File → R Script* to create a new RScript, which you should immediately save to your *GEOG_5963_6960_Exercise_1* directory (which you need to create beforehand). Copy and paste the following line of code into the script:

```
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
```

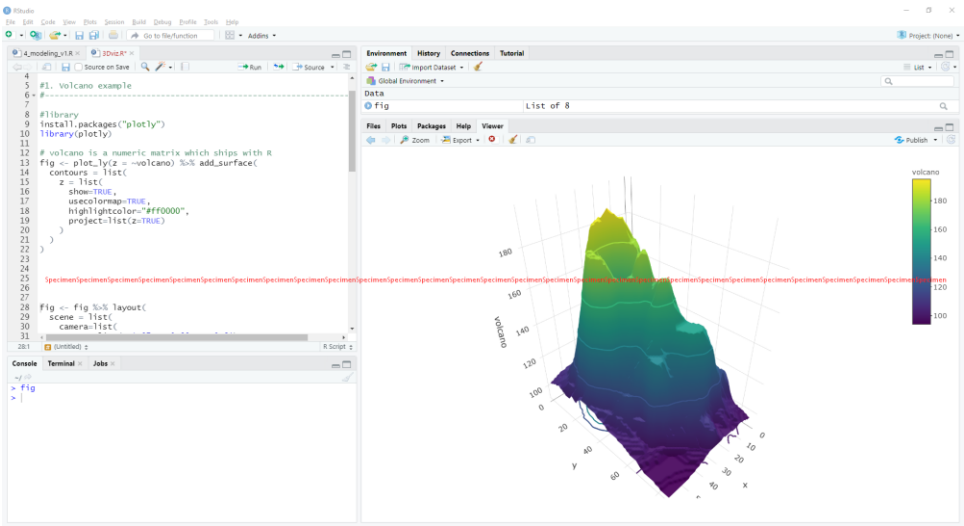
This line will set the working directory to the directory your script is stored in (your *GEOG_5963_6960_Exercise_1* directory).

Then, you start writing code to create a 3D plot using the volcano data set, which ships with R. Copy and paste the following into your script:

```
install.packages("plotly")
library(plotly)

fig <- plot_ly(z = ~volcano) %>% add_surface(
 contours = list(
  z = list(
   show=TRUE,
   usecolormap=TRUE,
   highlightcolor="#ff0000",
   project=list(z=TRUE)
  )
 )
)
```

The first two lines of code above install and load the Plotly library. Plotly is an open source graphing library available for R, Python and JavaScript. Go check out the documentation here. The second block of code creates the 3D visualization using the volcano dataset. Type `volcano` into your console, hit enter and you will see that the dataset is merely a 2D matrix of dimensions 87 * 61, which stores elevation values. Yes, this simply is a digital elevation model (DEM)! Run the code, then type `fig` into your console. The 3D visualization appears in the RStudio Viewer (it's interactive, try it!):



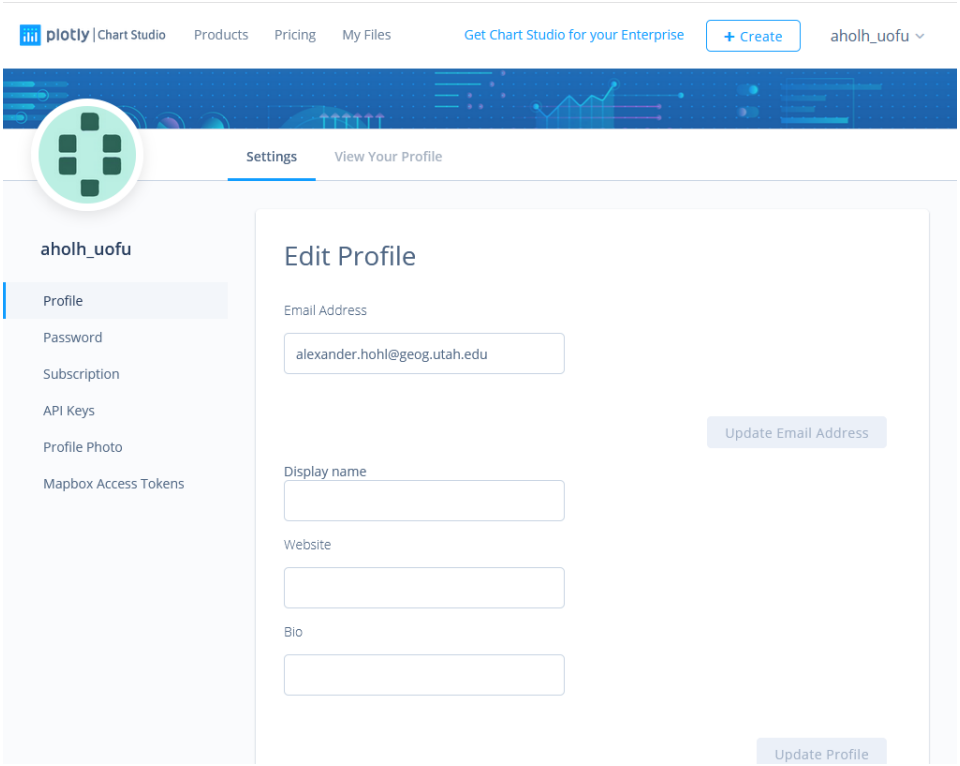
By adding the following lines of code, you can change the default viewpoint position:

```
fig <- fig %>% layout(
 scene = list(
  camera=list(
   eye = list(x=1.87, y=0.88, z=-0.64)
  )
 )
)
fig
```

In your Viewer window of RStudio, navigate to *Export → Save as Web Page*. This allows you to export your 3D visualization as a HTML file, so that you can conveniently access it through your browser.

Step 2: View your visualization in Plotly Chart Studio. Plotly Chart Studio is a browser-based tool that offers additional functionality, such as annotations, dashboarding, and advanced sharing capabilities. While some functionality is free, other functions are hidden behind a paywall. However, with an example as small as our visualization of Maunghawau, we can easily illustrate the capabilities of Plotly Chart Studio within the free tier.

Sign up for an account here and use your UMail address, which you'll have to verify. Once logged on to your account, your screen should look like this:

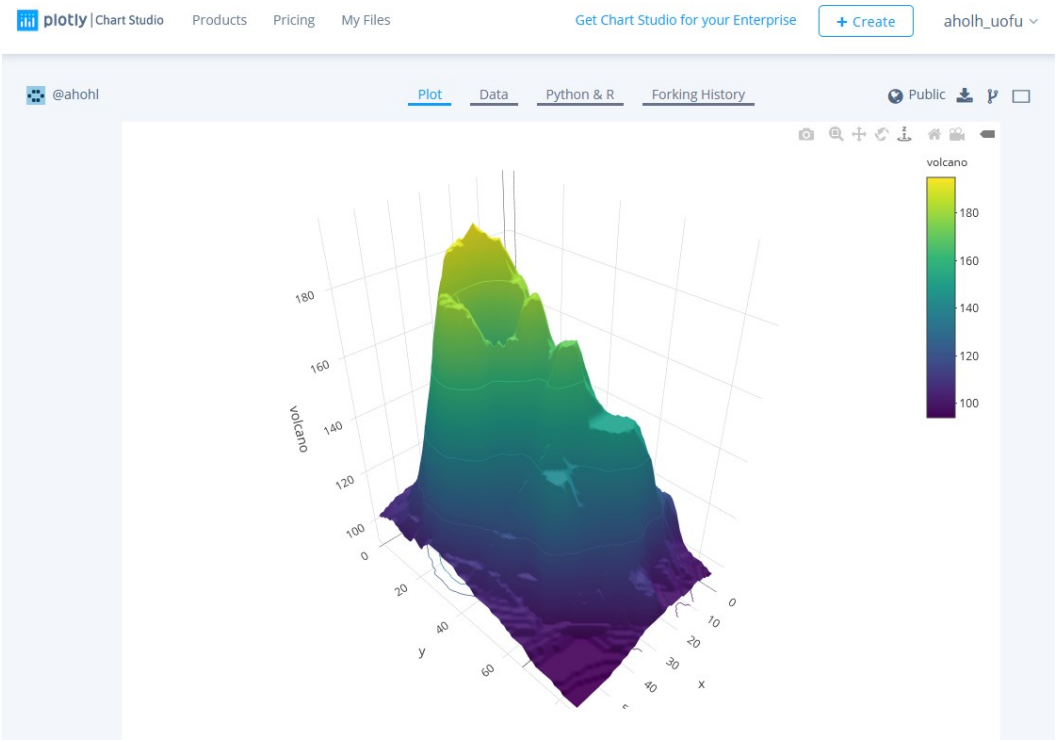


Next, you'll create your API key: Navigate to *API Keys*, click *Regenerate Key*, and copy and paste your username and key into a text editor to keep them handy.

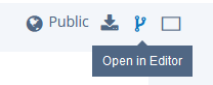
Paste the following code into your script:

```
Sys.setenv("plotly_username"="...")
Sys.setenv("plotly_api_key"="...")
chart_link = api_create(fig, filename="volcano")
chart_link
```

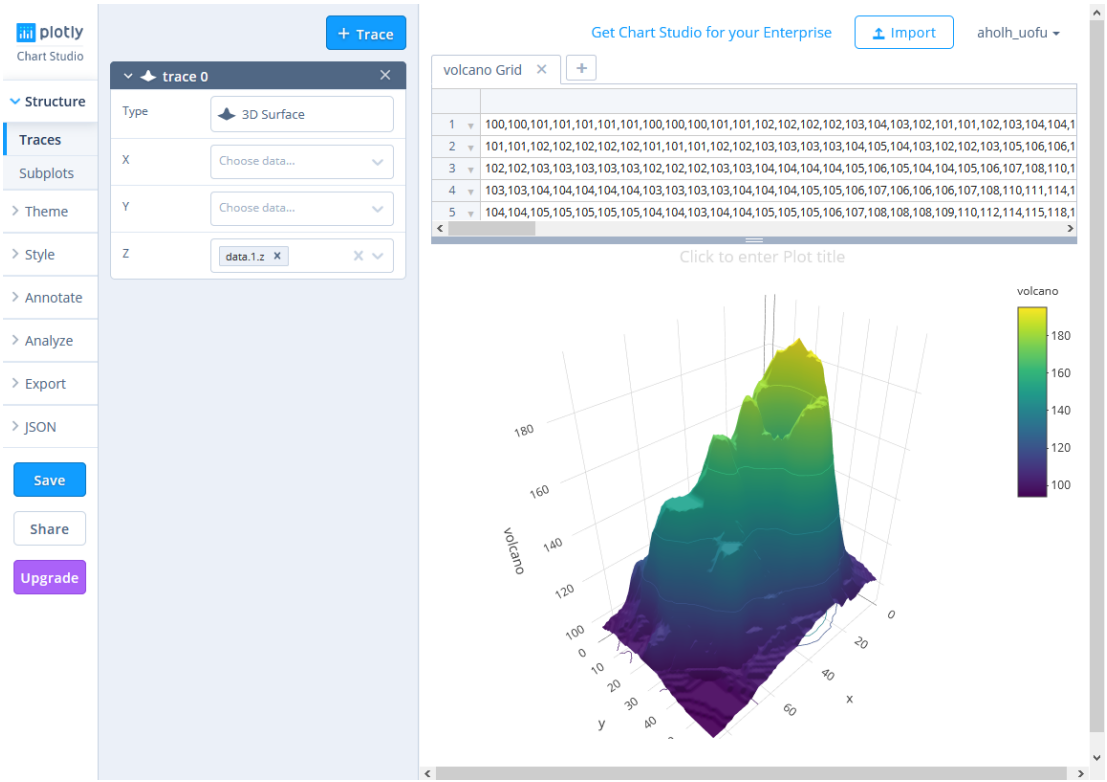
In the first and second lines, replace the three dots (...) with your username and API key. Running this block of code will open the Maunghawau 3D visualization in Plotly Chart Studio:



Explore this page, click on each of the tabs above (Plot, Data, Python&R, Forking History) and finally, click on the *Open in Editor* button in the top-right corner:



You now see Maunghawau in the Plotly Chart Studio Editor:



On the left side pane, click *Theme* → *Choose* and select the *Plotly Dark* theme. Save your plot by clicking the *Save* button. Then, click the *Share* button to obtain the URL of your plot. You may now tweet, facebook, linkedin, Instagram or tiktok about it.

Step 3: Grandeur Peak. There are additional steps involved if you want to take any raster dataset and create a 3D visualization of it using Plotly:

First, import the raster and rgdal packages:

```
library(raster)
library(rgdal)
```

Second, read the grandeur peak raster dataset, which you downloaded from CANVAS:

```
dem <- raster(x = "data/grandeur/w001001.adf")
```

Third, convert the raster object to a matrix:

```
dem_mat <- t(as.matrix(dem))
```

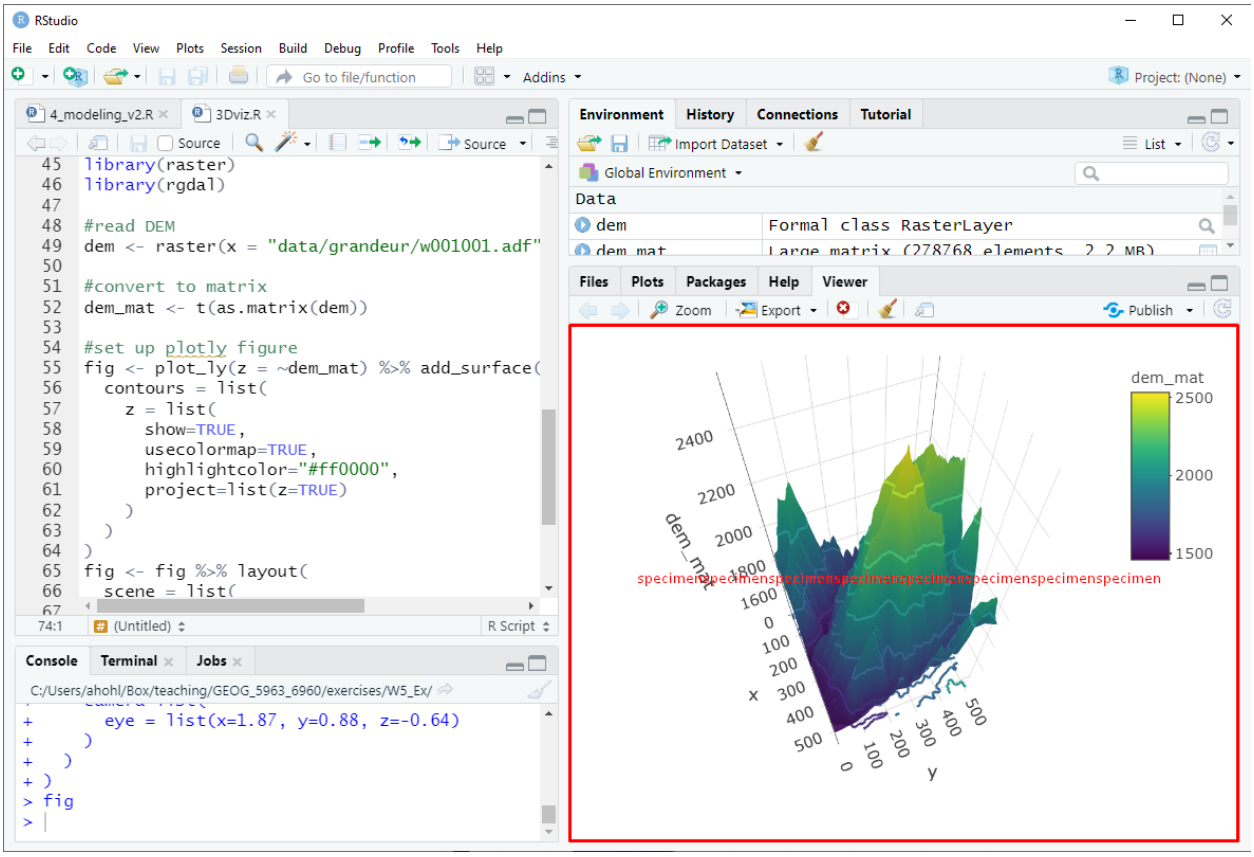
Finally, set up and create the plotly figure, similar the previous steps creating the volcano figure:

```
fig <- plot_ly(z = ~dem_mat) %>% add_surface(
  contours = list(
```

```
  z = list(
   show=TRUE,
   usecolormap=TRUE,
   highlightcolor="#ff0000",
   project=list(z=TRUE)
  )
 )
)
fig <- fig %>% layout(
 scene = list(
  camera=list(
   eye = list(x=1.87, y=0.88, z=-0.64)
  )
 )
)

fig
```

The plot shows up in the Viewer window of your RStudio:



In the Viewer window, click *Export → Save as Web Page* and save the visualization as a web page in a directory of your choice. You can now open, view and interact with the 3D visualization of Grandeur Peak in your browser. In case you are not aware of it: This is really cool!

Your Task: Get your own data!

To score the 40 points in this exercise, get your own raster dataset and create a 3D visualization with it, using plotly. Basically, replicate Step 3 above with your own data. Take any raster dataset of your choice (Maungawhau and Grandeur Peak excluded) and submit the HTML file that contains your visualization on CANVAS.

In this tutorial, we used raster data that represented elevation (DEM). However, be aware that you can use any raster dataset, even though I would refrain from multi-band rasters. Also, I recommend choosing a variable that varies continuously through space. Here are some ideas:

- Population density
- PM 2.5 concentration
- Travel time to health care facilities
- Temperature
- Bathymetry
- Crime density

Pro tip: It's easier if you use a small raster data set. As a reference, the Grandeur Peak raster has 524 columns and 532 rows.

Bonus: The visualizations produced in this tutorial all exaggerate the third dimension (the face of Grandeur Peak is steep, but certainly not as steep as the plot makes it seem). You get 5 bonus points if you find out how to "squeeze" the third dimension together to create a more realistic representation of topography. This is something I don't know how to do, and by finding out, you help me and future generations of students with 3D visualization. A potential way to success is to apply a scaling factor to your raster dataset, i.e. you multiply every cell by, let's say, 0.5.