Ocean Classroom – Message on a (plastic) bottle

The United World Challenge

An important aspect of the United World Challenge is to bring awareness to just how much plastic litters our oceans. In his milestones, Tez takes note of plastic debris he collects. Unfortunately, such large debris are only part of the story. In our programming exercise today, we will use publicly available data to compare the abundance of microplastics in the Eastern Pacific Garbage Patch between 1972-1987 and 1999-2010[1]. Dr. Miriam Goldstein generated the data from net samples collected during research expeditions led by the California Current Ecosystem Long Term Ecological Research (CCE-LTER)[2]. Today's exercise was suggested by microplastic expert, Dr. Jenni Brandon, Senior Scientist, Applied Ocean Sciences.

The Science

Plastic debris that we can see in the ocean are called **macro**plastics. More formally, macroplastics are defined as plastics larger than 5 mm, whereas smaller plastics are called **micro**plastics. Macroplastics are not only an eyesore: abandoned fishing gear, for instance, presents a risk of entanglement for fish and marine mammals. Sessile organisms, such as mussels, can also settle on floating plastic and be carried great distances. Because plastic takes longer to degrade and sink than natural debris such as wood and algae, any invasive species that hitch a ride is likely to disperse further and cause additional problems. In milestone 27, Tez collected a great example of plastic debris with hitchhikers (shown above)!

Microplastics, such as the beads in facial cleaners, can enter the ocean directly through wastewater discharge, or they can be formed when large plastics break down. Microplastics have been found in sediments, in marine organisms ranging from small plankton to larger animals, and even in human tissue. Scientists are still shedding light on the consequences of rampant microplastics, but these small particles are thought to be vehicles for pollutants, and to modify how fast organic matter sinks to the bottom of the ocean. Trapping of organic matter at depth is one of the mechanisms that removes carbon dioxide from the atmosphere. This means that by slowing down the sinking of organic matter, small plastic particles can slow down the removal of a greenhouse gas from our atmosphere, and therefore affect our climate.

Programming Activity

For this activity, we will use the R programming language and RStudio, which are both free and open source. First, you will need to install R and RStudio[3]. You will then be ready to begin.

---

[1] The data was published in Goldstein M.C., M. Rosenberg, and L. Cheng. 2012. Increased oceanic microplastic debris enhances oviposition in an endemic pelagic insect. Biol. Lett. 8: 817–820. doi: 10.1098/rsbl.2012.0298 and can be obtained at https://oceaninformatics.ucsd.edu/datazoo/catalogs/ccelter/datasets/213.

[2] To learn more about the CCE-LTER, including current research and outreach opportunities, visit https://cce.lternet.edu/.

[3] Instructions to install RStudio can be found at https://rstudio.com/products/rstudio/download/

Before You Start

- ☐ Download all files for this activity and put them in the same folder:
  - The programming script:
    - *OceanClassroom_Microplastics.R*
  - The data files:
    - *Microplastics.csv*
- ☐ Make a copy of the .R file in case the script stops working after some lines of code are modified.
- ☐ Double-click on the .R file called *OceanClassroom_Microplastics.R*
- ☐ When RStudio opens, find your working windows:
  - The OceanClassroom_Microplastics.R panel contains your script.
    - This is where you will make modifications.
    - In the top right corner of this panel, locate the **Run** button (runs the line where your cursor is located) and the **Source** button (runs the entire script).
    - Notice that some line starts with #. These lines will be ignored when you run your script. We say that the lines are commented.
  - The **Console** can be used to test some lines of code before adding them to the script. In this exercise, we will use the **Console** to explore our data.
  - The **Plots** panel will display the plot you create.

Getting Started

- ☐ Make sure your working directory is the one with all your files. At the very top, click: Session > Set Working Directory > Choose Directory...
- ☐ Click **Source** to run your script or type `source("OceanClassroom_Microplastics.R")` in the **Console**.
- ☐ Note that the script installs R libraries called ***rnaturalearth*** and ***rnaturalearthdata***. These libraries will be used to make our map, similar to how we used ***oce*** and ***ocedata*** in the Ocean Classroom activity on ocean depth. You will see that a plot also appears, and that it is automatically saved in your folder under the name *OceanClassroom_MicroplasticsPlot.png*
- ☐ Currently, the plot shows a region of the Pacific Ocean between Hawaii and North America, where the Eastern Pacific Garbage Patch is located. A magenta asterisk shows you approximately where Tez was when he collected the water container shown on page 1.
- ☐ Today, instead of immediately adding the data to our map, we will spend some time exploring our script and our data. First, **line 33** shows you that the function `read.table()` is used to load data from *Microplastics.csv* into a variable called `plastics`. The file extension *.csv* tells you that the data are separated by commas, which we specify in the `read.table()` function using `sep = ','`:

```
plastics <- read.table('Microplastics.csv', sep = ',', header = TRUE)
```

- ☐ To see the data contained in the variable `plastics`, simply type `plastics` in the **Console**. You should see 235 rows of data! You can also find the dimensions of `plastics` by typing `dim(plastics)` in the **Console**: the first number tells you the number of rows, and the second number tells you the number of columns. Remember that you can type `?dim` in the **Console** if you want more information on the function.

- ☐ Since 235 rows is a lot, you can choose to display only the first 6 rows of the variable `plastics` by typing `head(plastics)` in the **Console**. You should see 4 columns called **Period**, **Longitude**, **Latitude**, and **Abundance**. In our case, these columns correspond to the time period when the samples were collected, the longitude and latitude at which they were collected, and the abundance of microplastics found in each sample (number per m^3).

- ☐ You can extract any column of `plastics` by typing `plastics$` followed by the column name in the Console. Try:
  `plastics$Longitude`

- ☐ To know how many rows a column contains, you can use `length(plastics$Longitude)`

- ☐ If you type `tail(plastics)`, you will see the last 6 rows. Note that our data has been grouped in two time periods: **1972-1987** and **1999-2010**. When using `head(plastics)`, you should see data from 1999-2010, and when using `tail(plastics)`, you should see data from 1972-1987.

- ☐ In this case, you probably believed me that there were only two different time periods. However, if you wanted to check, you could type `unique(plastics$Period)`. The function `unique()` displays repeated information only once, so you can easily tell there are only two unique values.

- ☐ You can also use RStudio to perform a number of regular mathematical operations on your data. For instance, you could try `abs(plastics$Longitude)` to get the absolute value of longitude, or you could add 3 to your latitude data by typing `plastics$Latitude + 3`. Although it does not make much sense to do this with latitude, you could multiply, divide, or take the square root of your data. The sky is the limit!

- ☐ To test if a sample was collected North of latitude 20, simply type `plastics$Latitude > 20`. Your **Console** displays TRUE or FALSE for each row.

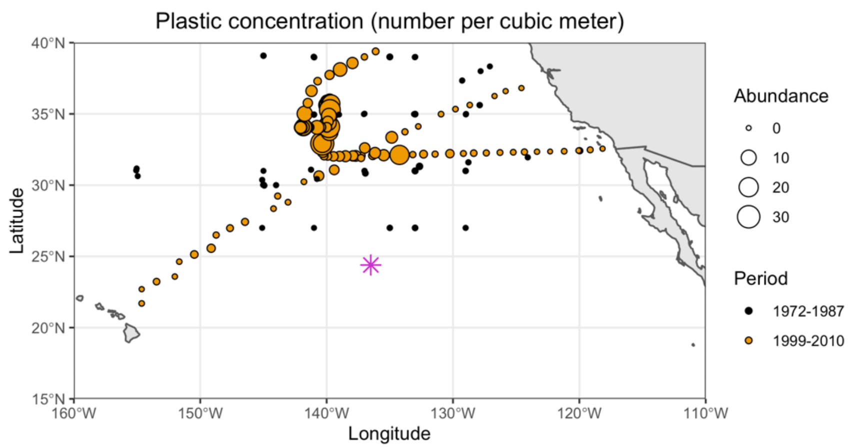
- ☐ Have you tried doing your math homework using RStudio? You can use the **Console** the same way you would use a calculator. Just remember that you need to use * to multiply. Try typing `x <- seq(2, 100, by = 2)`. This will create a vector with a sequence from 2 to 100 in jumps of 2. Type `x` to see what is stored in the variable. You can then perform operations on this vector such as `y <- x + 3`. Type `y` to see what is stored in the variable y. If you are working on an assignment, it is always good to copy any line that works for you into an R script so that you can save and repeat it.

- ☐ Now, let's come back to our map and add the microplastic data. **Delete** the `#` in front of **lines 39 to 41**. Notice that the function `geom_point()` extends over the three lines so they must all be commented or uncommented together. If not, the code will break.

```
#geom_point(data = plastics,
#    aes(x = Longitude, y = Latitude, size = Abundance, fill = Period),
#    colour = "black", pch = 21)+
```

☐ **Save** your changes and click **Source** to run your modified script.

☐ You should now see that circles were added to your map (below). The color of the circles shows you whether the samples were collected between 1972-1987, or between 1999-2010, while the size of the circles shows you the concentration of microplastics found in the samples. Would you agree that the circles from 1999-2010 are much bigger? Unfortunately, this means that the concentration of plastics in the Eastern Pacific increased from 1972-1987 to 1999-2010.



☐ To change the default colors of the circles, simply uncomment **line 42**. As you can see on the map above, I found that the circles for 1972-1987 were easier to see when they were black. Feel free to try with different colors by changing `'black'` and `'orange2'` to your favorite colors. You can find many color names at the following link: http://www.stat.columbia.edu/~tzheng/files/Rcolor.pdf

```
#scale_fill_manual(values=c('black', 'orange2'))+
```

☐ Now that you understand a bit more about coding, enjoy exploring! Some suggestions of what you can try:

- Use RStudio for your own calculations, or create a new *.csv* file with your own data by using an example from your math homework for instance.

- Create a map for a different location by changing `xlim` (longitude) and `ylim` (latitude) on **line 38**. Do you have any GPS data you could add to the map by modifying the *.csv* file to include your own data? If you change the names of the *.csv* file, remember that you will have to change the name of this file in your script too (**line 33**):
```
coord_sf(xlim = c(-160, -110), ylim = c(15, 40), expand = FALSE)+
```

- Change the name of the file that contains your plot or its size by modifying **line 51**:
```
ggsave('OceanClassroom_MicroplasticsPlot.png', width = 7, height = 4)
```