[SQUEAKING]

[RUSTLING]

[CLICKING]

HELENA VALLICROSA: All right, so we are back to the code. And now we're going to start applying the concepts that we've been talking to on R. So the first thing is to read vectorial maps. How do we read vectorial maps? Well here I included a web page where you can download vectorial maps. They are basically physical vectors of the global Earth, so you can find the coastal barriers.

The packages we're going to use are these three. I have already installed them. But feel free to install them, and then lock them into the session. There are several ways to read vectorial maps. I just include three of them here. There's this one. My personal preference is the second one. There's this third one that I just put them in common.

So in the first one this, which is using this function right here, the projection needs to be specified previously. So here is what I am doing. I'm just saying that I'm using the projection longlat, which is longitude latitude, WGS84. So I have to put this between commas, and using this particular function here. So this is going to be the projection.

And when I'm reading the map, what I have to do is to specify where this map is located in your computer, and then specify what projection do you want it to be. So you'll have that map here. It's giving me a warning, just information. But everything worked fine. And my personal preference is this one, where you don't even have to specify the projection.

So it's going to be read by the package itself. You only have to specify the route from your computer. You use this. It gives you some information-- what it has read, what it includes. And then I'm just going to plot, just type plot, and the map you want to plot, and you're going to see how this map looks like.

You see this is the boundaries of the ground. And then we move on to raster maps. This is pretty simple. The king or the best personal preference, but the best package to use raster is the package raster. So you can use it to manage everything you need from rasters. I'm just going to load it into the session. And it's as easy just to put raster here and the route of your map.

I'm going to chart a temperature map that I've taken from WorldClairm. And it's in 10 minutes resolution. We're just going to see how it looks like. And this is how by default the map is created. Also, if you feel like magnifying your map, you can just click on zoom here. And this is going to pop out in an outside window. So you will be able to see it bigger.

And then there is a particular way of displaying maps which is using NC format, which is when you are working on time series maps. Just because the maps are pretty heavy data, this is a particular way to store time series in a pretty efficient way. But that makes it a little tricky to work with. So I'm just going to show you how to do that. I'm going to clean my environment to not being distracting.

These are the packages you're going to need to go through this code. It's going to load them all. And this is a map that I have obtained based on this reference here. If you click on that link, you would be able to download this information from the internet. It's a nitrogen deposition map that goes from, if I remember correctly, from 1901 to 2020 or something like that. So every time to time there is a map. So that's why it's a time series. So you have to picture, let's say a book. And each page of the book is a map. So you have them all together. And they are all the same size, the same resolution. But you have a repetition in time of the same map.

OK, first of all, is to open the file. So nc_open where the map is. You take the map. And if that would be a raster, you would be able to plot it right away. So that's what I try here. No worries. It's not going to work, but it's just to show you that it's a different way of organizing maps. So how do we do it, to extract it and to convert it into a map we can see, we can visualize, we can understand?

The first thing we need to do is to see what are the names of the dimension of the map. Just click names, the map, and dimensions. And this is going to return the three dimensions that we need to know which is the longitude, the latitude, and the time. But sometimes, the author's name, this variable is differently. So here is longlat, but it could be longitude latitude, or time could be years, or things like that. So that's why we need to make sure how this is named.

So we're going to extract the longitude. Here is where we get the map structure. We have longitude. We variable get the longitude. OK, and we store it in a vector. And here we want to extract the latitude. We repeat the same, but with latitude instead. And here we are extracting the information related to the time.

And now we see that the latitude is cut in 360 different columns or rows, and the longitude likewise, but 720. And then we want to get the actual information. So what's the content of the cells? So we do this like that.

So here is the content. Something else we need to take into account is the NA values. So when creating a map, there is a value that you use to name your NA.

Sometimes it's NA right away. But some other people use numbers like 9,999, or just a big number. And here what we are doing is to explore how the authors have named this NA value. So this one I want to see. I'm curious. This is the big number they named as missing value. So to avoid this number to mess with our results, I recommend you to change it to NA. So that's what I'm doing here. I'm replacing every number that looks like this, to an NA. Don't worry that you are not going to knock out important data. It's just mostly going to be NAs. We run this chunk of code.

Oh, yeah. Sorry about that. Now, it should go. It's taking a while because there's a lot of data. Oh yeah, here we have it. So now all the NAs would be replaced into NA value. And now we no longer need to have opened them up. So we can close it and it doesn't occupy space in our session. Just close it because we already have the information we need.

And how are we going to transform this information we have in different chunks into an organized map that we can actually visualize and work with? First of all, here I'm going to run an example. I'm just going to use the package raster right here.

I need to include this t here, which means transpose, because otherwise if I don't put transpose, the image would look flipped. Then this is the object. These are the different dimensions-- longitude, latitude, and this number 1 is going to be the first time frame. So I'm going to just get the first map that we have.

Then it asks us for the minimum of the x-axis. I'm going to put the minimum of the longitude, the maximum of the longitude, and just basically setting up the format of this raster map. You have to specify as well the projection. So this is going to be the projection that it's the same that we've used before in other maps. And we just run this chunk of code.

And we can plot our results. This is how the map looks like. This is the nitrogen deposition in, I believe in 1901. And it includes also the sea. So that's why it doesn't look like the Earth that we're used to. And then how can we store this information in our computer? How can we write this raster?

So it's just as easy as write raster, put the name of the file we want to store, file name, and your route as follows.

So now we are moving to the resolution. And you might have paid attention when downloading the data from the WorldClaim that it allows you to download the maps in different resolutions. But in order to name the different resolutions, WorldClaim, it uses time. So why is that?

Well actually, you could face three different ways to approach resolution. The first of all is the surface using the surface square. One example would is 1 kilometer square. That refers to the map when we have it in a representation. In a gridded representation, we would have each side of the grid calculated by the surface that it's occupying. So just take this that's 1 kilometer. And that's 1 kilometer. The problem of this approach is that it's not always accurate. Because the part of the grid that's closer to the equator is smaller than the one that it goes to the pole.

So it stretches when it goes to the edges. And so that makes that this 1 kilometer is not consistent throughout the whole map. And to overcome this problem, it came with the angle or the time approach, which treats the Earth as a round surface. This is not perfect because, the round is not 100% perfect, but it's better than using the surface. So what they do is to consider on one side the angle that this opening is.

So just let's take this square right here. And let's just plot it on top of this. So that the angle that covers this part of the surface, that would be displayed in the angle, or also the time that the Earth needs to rotate and cover this amount of surface. So for you to have a reference, for example, 1 kilometer square resolution would be the same as 30 seconds in time.

So going back to the code. Here I downloaded two different resolutions for the temperature maps on WorldClaim. One is 10 minutes. And the other one is 30 seconds. The 10 minutes is OK to download. The 30 seconds, it might take a while. If your internet connection is slow, you can just believe what I'm saying. But if you want to try it out, obviously feel free.

Just to remind you how the map looks like, it's going to look like this. Let's just check the resolution here on both maps. Here we have this number here. But what does it mean? Here we have this other number here. So when we consult the resolution in our maps, it's going to be shown in degrees, so using the angle approach that I previously mentioned.

So how can we change resolution in a map? We can use the factor aggregate. So if we have a number that's bigger than 1, we are going to combine cells. So if we have in this case, I'm combining in a factor of 4, meaning that four cells horizontal and four cells vertical are going to be combined in one.

There are different ways to calculate the resultant cell. So it could be using the mode, or it could be using the mean, depending on what kind of data you have in your map. But you could also increase your resolution by using a number that's lower than 1. So let's if you put, let's say 0.5, you are going to divide a cell in 4. Because we're working in two dimensions.

I mean this is technically correct, but conceptually it doesn't really make a lot of sense. Because you are creating fake data. It's not that you are actually including the resolution of how good you are approaching reality. You are just creating more data for no more accuracy.

So in this case, I'm going to aggregate the map that we have in a factor of 4. And I'm just going to consult how the resolution has changed. So initially I aggregated this one. This was the initial resolution. And once the resolution was changed, I see the degrees has gone bigger because the surface is bigger. So the time or the degree to cover that surface is bigger.

So if you plot that, you might be able to see that the quality of the image is a little worse.