Lab: Producing Data Maps in Datawrapper

Version 2, March 22, 2021, 6:40pm

Log in to Datawrapper at datawrapper.de. Click on the "login" link at the upper right.
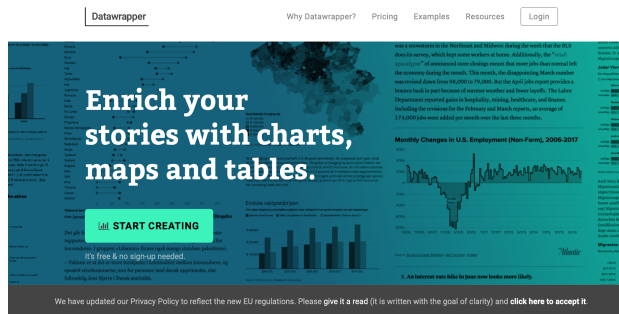
Upon logging in again, you'll see a grid of projects you've done before.
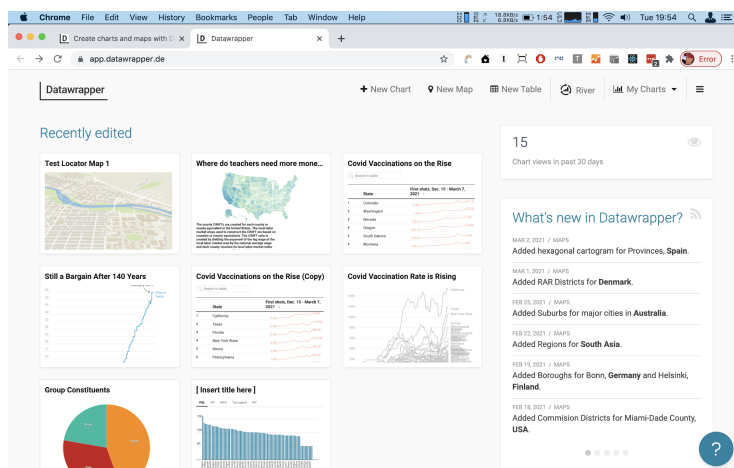
Click "New Map" at the top center of the page.

You'll find a page with three choices.
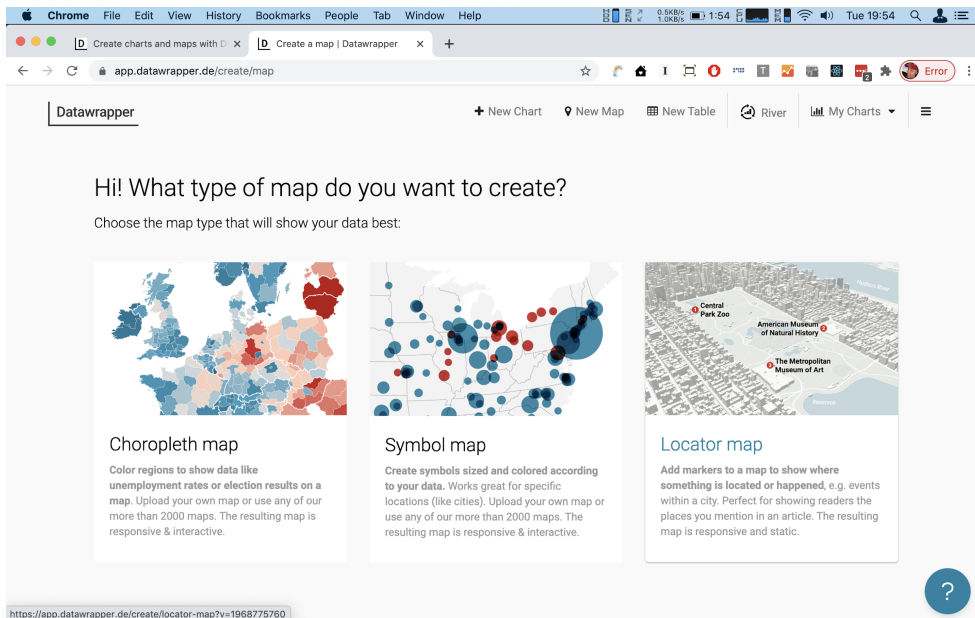
I. Symbol Map: U.S. Power Plants

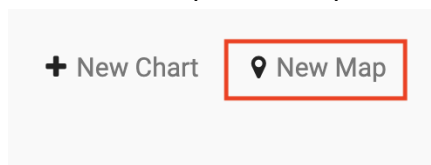This will be a good test of what Datawrapper calls a "symbol map," and which is also known as a "proportional symbol map" or "bubble map" – although only the latter when the symbol in question is a circle! Symbols can be anything – **circles**, squares, the **"spikes"** we saw today, Christmas trees… **words themselves**, even.. but they are best when they are abstract symbols. The simpler the better.
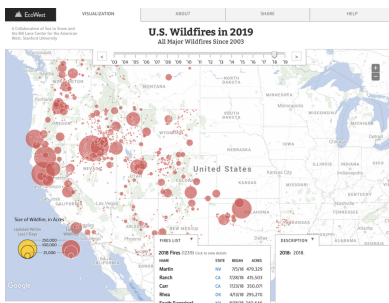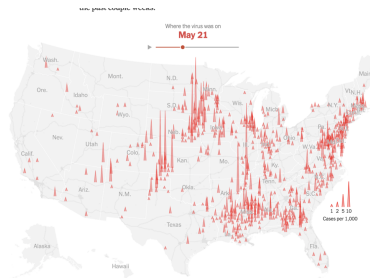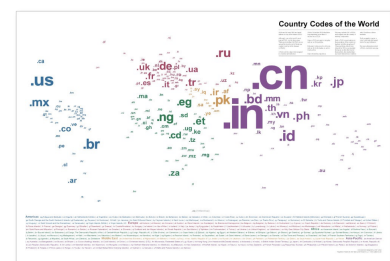
EcoWest   *NYTimes*   *Communication Arts*

Our goal here is to **produce a map of the United States showing the location and power source of all of the electrical generation plants** tallied by the Energy Information Administration.

To do that, we'll need to get the data from the EIA, convert it to a format we can upload to Datawrapper, and also simplify the data table itself so we run a lower chance of *breaking Datawrapper.* First, let's get the data.

1. Grab the data from EIA.gov

The EIA has been rolling out a new geodata portal built on ArcGIS online, but let's not go there – it's biased even more towards people already working in GIS. Instead, load up their map of electrical infrastructure in the US. It's a bit of a bear.

You can navigate their data sets by clicking on the "layers/legend" button at the upper right, but the main thing we're interested in is that link at the lower left entitled "layer information and map data."

This takes us to a page – let's call it "rich," not cluttered. There's a ton of data here. You'd think their only purpose in life was to… give us information about energy. I guess it is!

Note that all of these, in the "map data" column, all link to Shapefiles (the ArcGIS "shp" file format). If you're a GIS hound, that's exciting news, but for everybody else it's a bit technical. Especially for data sets that are really just tables of lat and long points on a map.

Let's continue well down the page to … Power Plants.

County						
Power Plants	Operable electric generating plants in the United States by energy source. This includes all plants that are operating, on standby, or short- or long-term out of service with a combined nameplate capacity of 1 MW or more.	EIA-860, Annual Electric Generator Report, EIA-860M, Monthly Update to the Annual Electric Generator Report and EIA-923, Power Plant Operations Report.	April 2020	7/10/2020	Shapefile	D F G S U
Railroads	Class 1 freight and other railroads in the United States. Class 1 includes the following freight railroad companies: Burlington Northern	Derived from: Homeland Infrastructure Foundation – Level Data (HIFLD)*	May 2018	10/8/2019	-	D G S U

By the way, those colorful icons at the right are category tags. (D: "Energy Disruptions," F: "Flood Vulnerability", G: "Gulf of Mexico" – I don't think that system is long for the world. Nice try, though, EIA!)

Click the link for "Shapefile," which should result in a file called **PowerPlants_US_EIA.zip** downloading to your computer. Your browser may automatically unzip it, but actually just keep it the way it is if you can.

We are now done with the EIA.

2. Use Mapshaper to convert the data to a spreadsheet

Meet Mapshaper
Instead of cracking open QGIS, ArcGIS or any other desktop GIS program, let's point our browsers to the handy website Mapshaper (mapshaper.org), created by Matthew Bloch, a cartographer and graphics editor at The New York Times. The NYT graphics desk has, at various points, spawned a number of tools like Mr. Data Converter, Mr. People, AI2HTML, and the briefly notorious "Configurator" (that was mine).
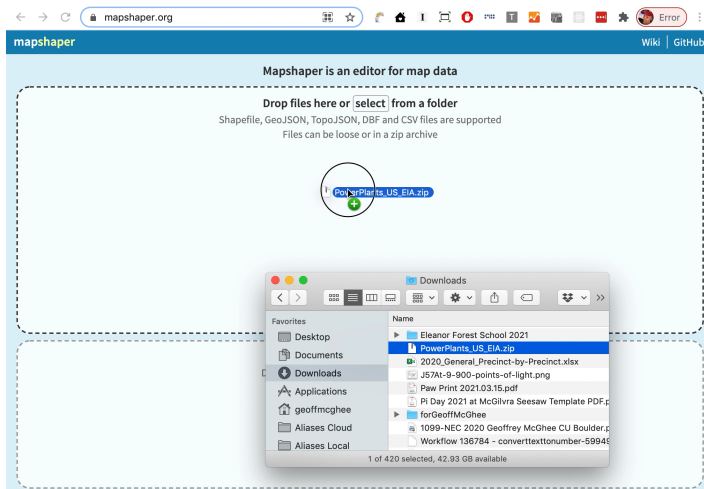
- Open mapshaper.org.
- Drag and drop the file 'PowerPlants_US_EIA.zip" into the Mapshaper window where it says "drop files here."
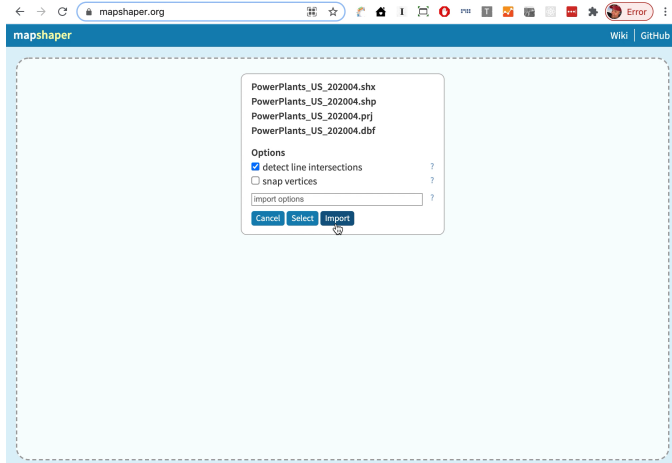
Import confirmation screen
- This will show you the contents of the zip archive, which is the shp file and its dependencies.
- **Click "import."**
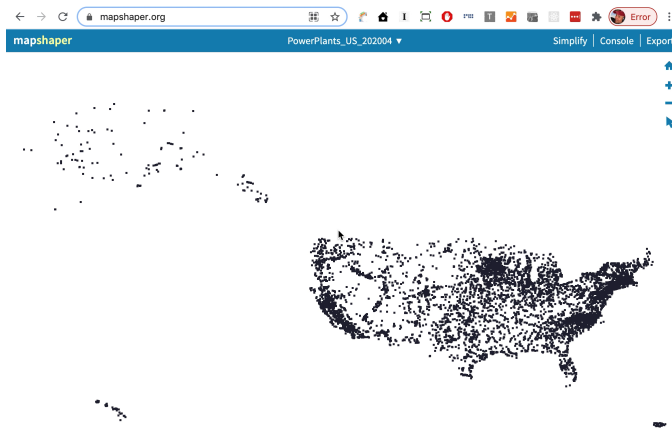
Geodata preview
As I mentioned, this file only contains lat and lon points with tabular data attached to them. It doesn't really have to be a shp file at all – I mean that it doesn't have any complex geographic shapes.
Click the "pointer" icon at the right side of the screen.
Hover over the points – you'll be able to view the data associated with each point.
Now **click on "PowerPlants_US_202004"** in the blue header area.
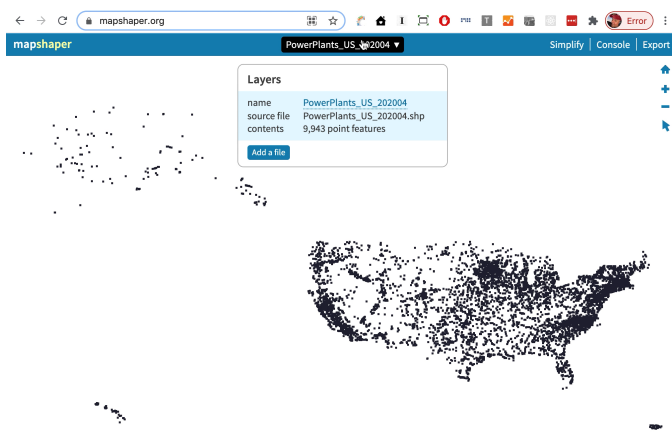
Layer information
This gives you details on the layers in this shp file - of which there is only one, as is typical. How many features does it contain?
Ah, 9,943 points. Each one a power plant. If this was a particularly complex file, we might want to edit it or simplify it – just you wait – but for this one, all we have to do is export it in a format that Datawrapper can read.
Click "export" at upper right.
Export menu
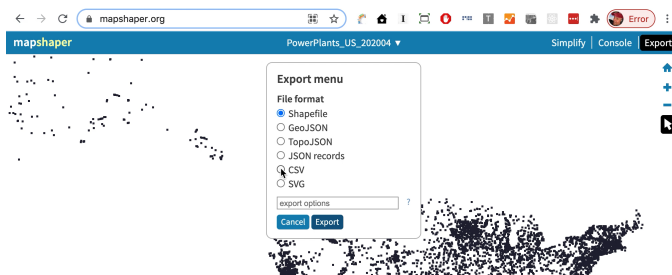
The default export format is SHP, but we don't want that! We want something that Datawrapper can read.

What can Datawrapper read? It depends.

1. **If you want MAP OUTLINES** made of lines or polygons – like states, counties, election districts, census tracts, etc – not the data itself, choose "GeoJSON" or "TopoJSON." This would be the case if you wanted administrative areas that didn't exist in Datawrapper's library, like "Burger King Regional Sales Areas" or "Places in America More than 50 Miles from a Library." With these, you could get Datawrapper to plot data into those boundaries, even if DW has never heard of them.

2. If you want **tabular data to place into Datawrapper's built-in map outlines**, of which it has over 1,000, then choose "CSV."

In this case, we want CSV.
Choose "CSV" and click "export."

Exported file: PowerPlants_US_202004.csv
The resulting file will look something like this, except that the full number of columns is not visible in this view.

In fact, it has far too many columns for comfort. This is making the file much bigger than it needs to be.

Let's trim down the columns in Excel.

If we didn't trim down the columns, this might fail to load in Datawrapper or, as happened to me, there are too many columns to select the Latitude and Longitude columns in Datawrapper. Literally their UI broke.

Here's the complete list of column names. Open the file in Excel (or import it into Google Docs if you don't have Excel) and **delete the bolded columns to reduce the file size**.
A lot of work? Maybe… but easier than learning GIS for this one assignment! : D

Plant_Code	PrimSource	**Solar_MW**
Plant_Name	Install_MW	**Wind_MW**
Utility_Na	Total_MW	**Geo_MW**
Utility_ID	**Coal_MW**	**Bat_MW**
sector_nam	**NG_MW**	**Other_MW**
City	**Crude_MW**	source_des
County	**Bio_MW**	tech_desc
StateName	**Hydro_MW**	**Source**
Zip	**HydroPS_MW**	Period
Street_Add	**Nuclear_MW**	Latitude
		Longitude
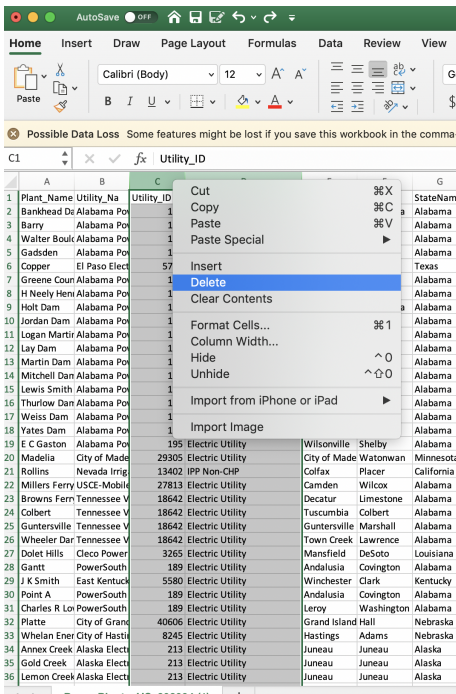
To delete a column, **right-click on the column header letter (like "A") and select "Delete" from the pop-up menu**. To delete multiple adjacent columns, click and drag across to select from left to right, then right-click and choose "delete."

When you've deleted all or most of the above columns (remember, each deletion removes almost 10,000 cells from the data!), **save the CSV file.**

3. Import the CSV into Datawrapper

Now that we've got our data, let's resume where we were at the "New Map" screen.
Choose "symbol map."

It's okay to be a bit thrown by this first screen. "I thought we were doing dots!"
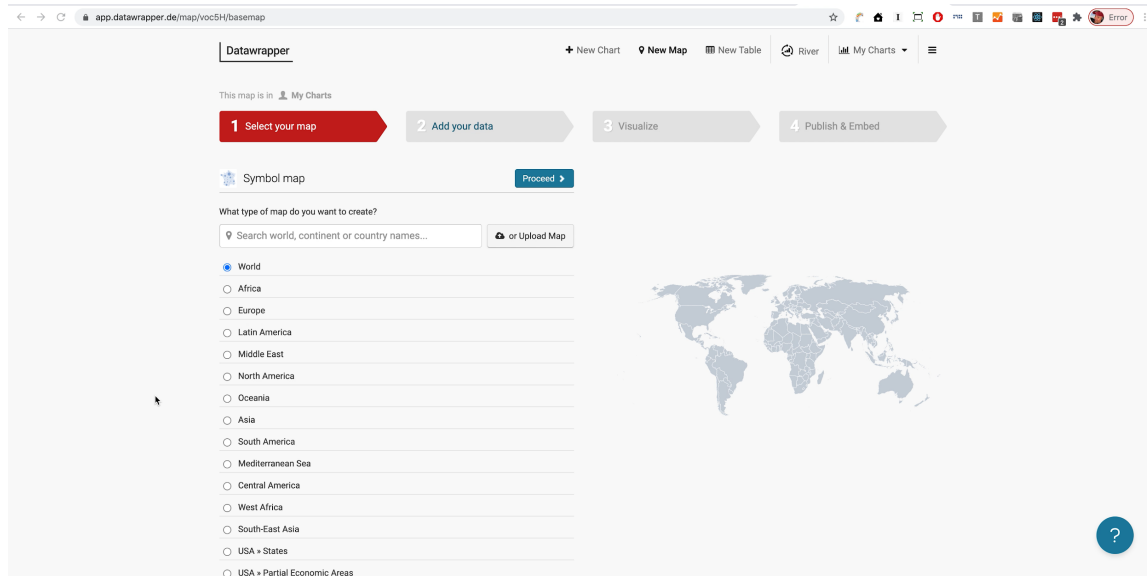
This screen is asking two things:
1. **What geographic scope** are we talking about here? Whole world? US? Montana? and
2. **What boundaries** do you want Datawrapper to show on the map? State lines? Counties?

Since our map is going to consist of dots indicating power plants in the United States, the answer to 1) is **the US, including Alaska and Hawaii** and 2) hmm, I'd say **"state boundaries,"** since those will do the best at situating the plants in context.
Then **click "proceed."**
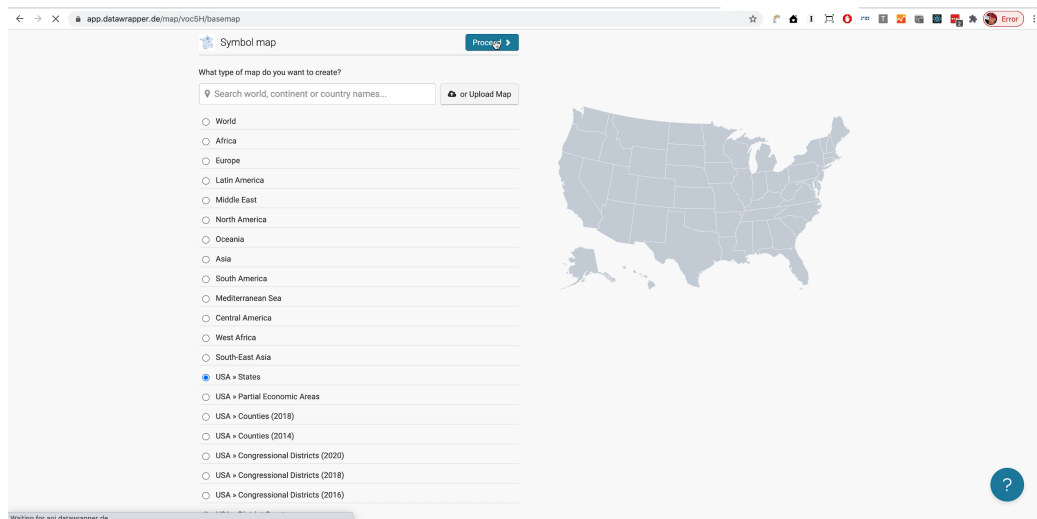
Add data to the map

Datawrapper's prompt is "add the first symbols to the map," which you can certainly do one at a time. You can type in an address, and Datawrapper will add it as a point on the map.
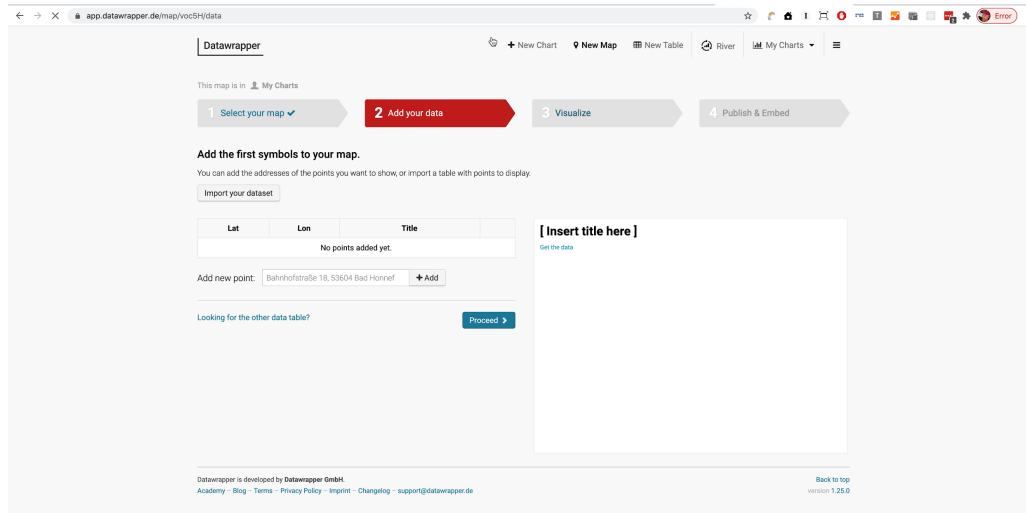
But we have 9,943 reasons not to do that. Instead, **click "import your dataset."**
The first screen gives you a tabular layout with two columns, into which you can paste data. But we have more columns than that, so click again, this time on **"Alternatively, click here to upload a csv-file."** (cute little German-ism there with the hyphen)
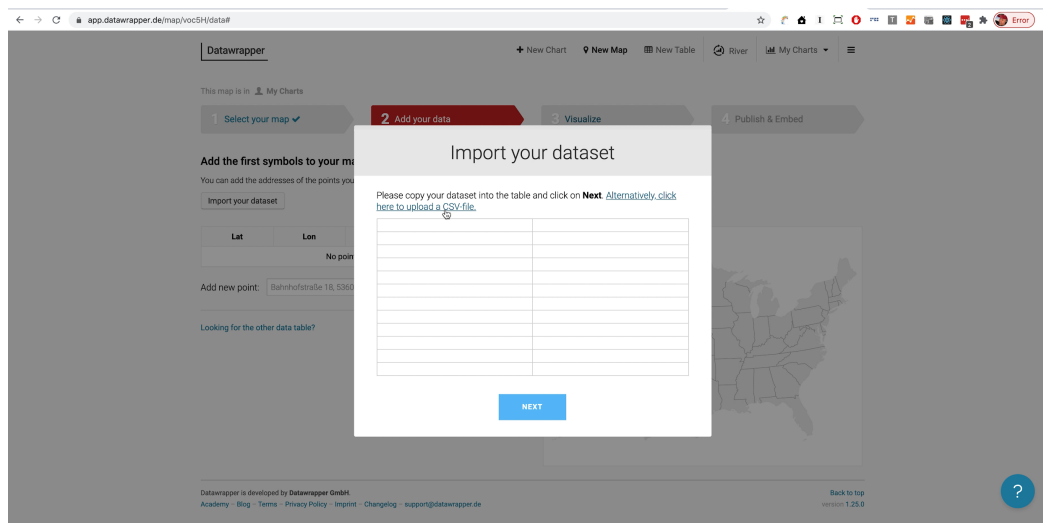
This is where it gets hairy.

Datawrapper wants to know which column has latitude values to read. Obviously, in this case it's "Latitude."

Since Latitude and Longitude are the last two columns, you have to click carefully to make sure the **second-to-last box is highlighted**:
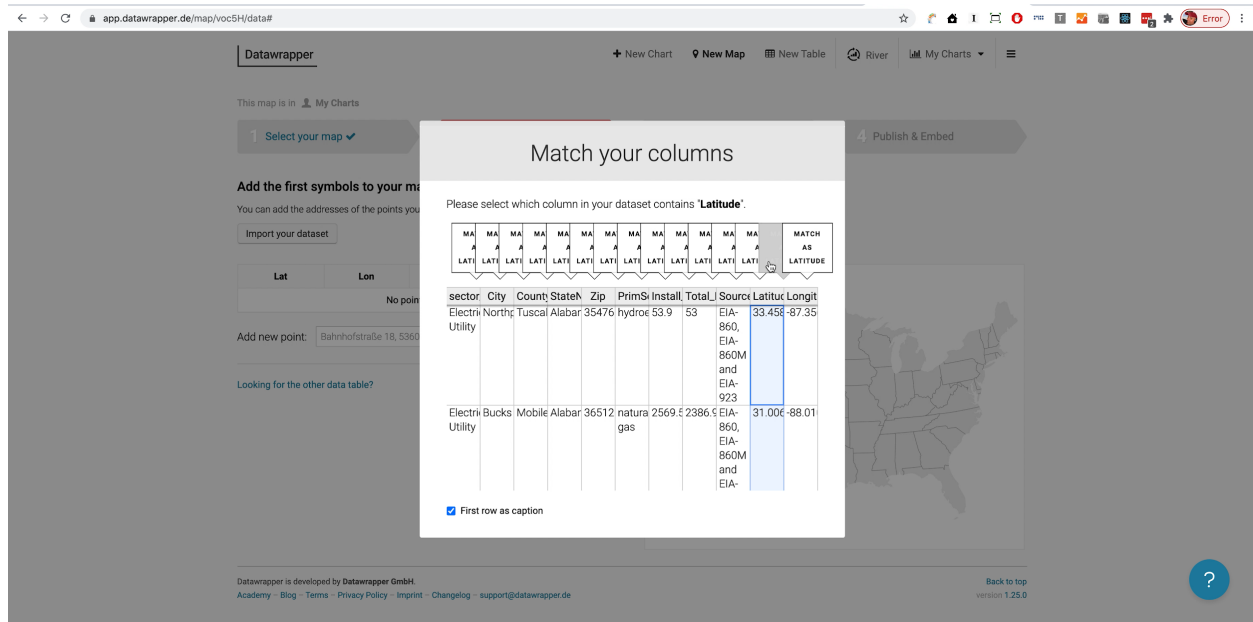
Then put your mouse pointer in the white area on the left or right and scroll down. This is so you scroll the popup window, not the 9,943 rows of data. If you scroll correctly, you'll see a button marked "NEXT". Click it and repeat the process for "Longitude."
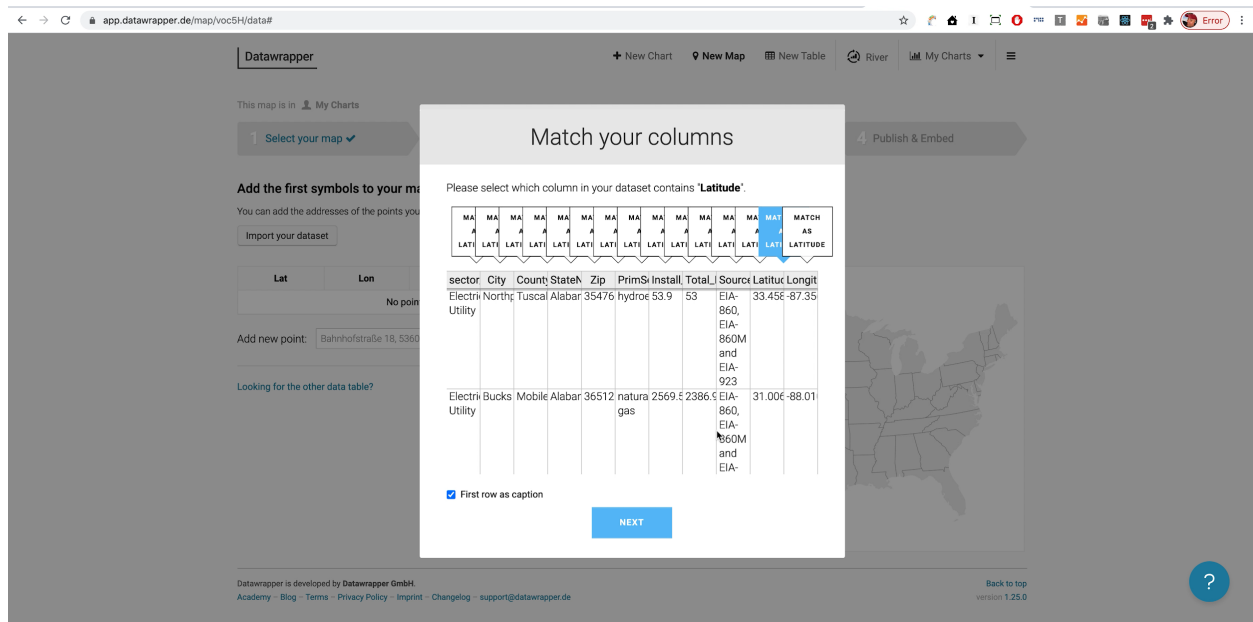
Hopefully, the next NEXT will result in "Congratulations, you've successfully imported your dataset. Click on the button below to see your map."

Don't give up
There might now be a considerable delay while your browser – and Datawrapper – chew on the data. If Chrome or Firefox ask you if you want to cancel the page because it's unresponsive, just tell them to %$#% right @#$@$. Eventually, you should see this.

When you get to this screen, click "Proceed," which will take us to the Visualize screen, where we can start to turn this into an infographic:

Add colors and size to the map symbology

Leave the symbol shape as "circle," unless you're feeling adventurous.

For "Size by," **choose "Total_MW."** The circles (or whatever) should now take on different sizes.
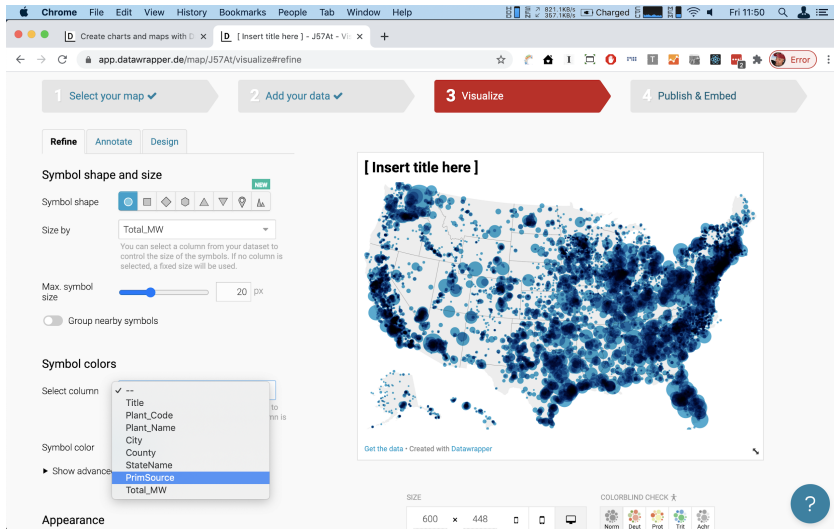
Change the colors
For "symbol colors," choose the field "PrimSource" – for primary source of energy.

You can also select a color palette, but we'll actually want to enter custom values.\

Add a Legend
Click "show color legend" and give it the caption "Energy Source"

Make Map Zoomable
Under "appearance," activate "make map zoomable".
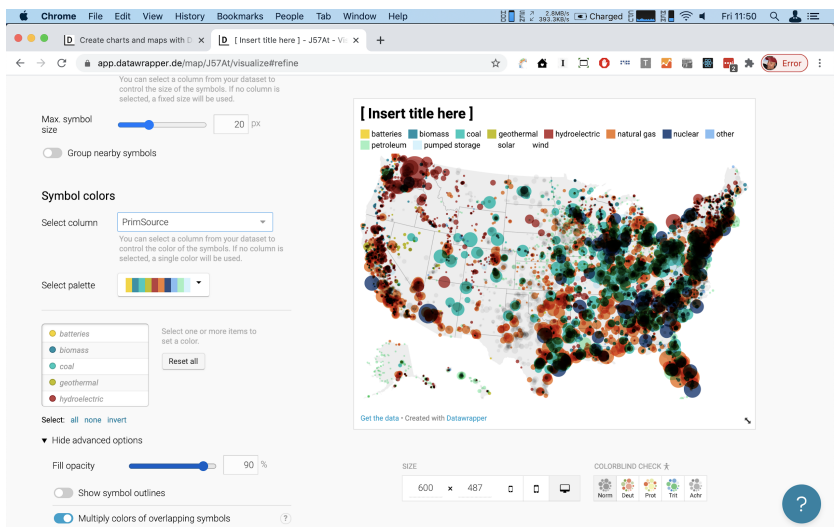
Let's refine those colors. Colors mean something here.

Under "select palette," click on each one of the color key items to select it, and paste the hexadecimal color into the "color" value that appears when you click the color chip.
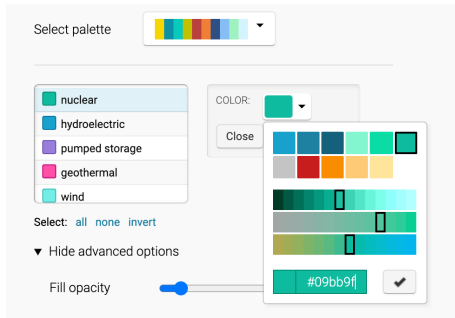
Nuclear: #09bb9f
Hydroelectric: #18a1cd
Pumped Storage: #987ddb
Geothermal: #ff4fa7
Wind: #73ede7
Batteries: #15607a
Solar: #c1f531

Biomass: #fa8c00
Coal: #989898
Natural gas: #aa957f
Petroleum: #5e5e5e
Other: #ac9752

When you're done, the map should look something like this:
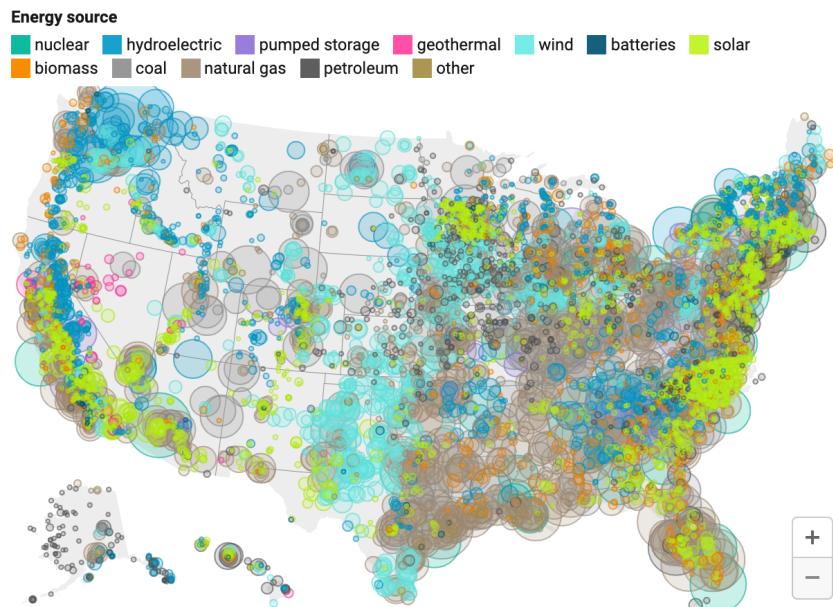
Damn. Pretty noisy! But we can improve it with interactivity.

Under "Legend," **click "show advanced options."**

Check "enable highlighting on hover" – this means that when a user puts their mouse pointer over the category, the dots associated with that category highlight, and the others disappear. Pretty neat, right?
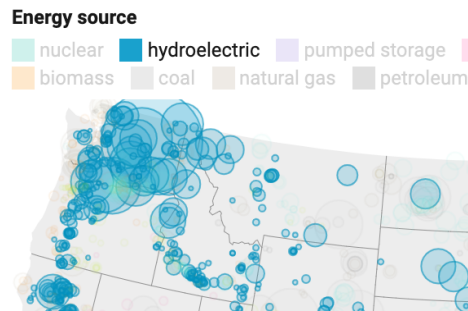
Click "Proceed" to get to the annotation phase. *If the screen goes blank, don't freak. Just scroll up!*

For **title**, enter: "9,900 Points of Light"
For **description**: "The Energy Information Administration regularly updates the national tally of electricity generation plants around the U.S. Since 2018, TK coal plants have shut down and TK wind, solar and battery plants have opened."

Data source: U.S. Energy Information Agency
Link to data source: https://www.eia.gov/maps/layer_info-m.php

Check Tooltips, then "customize tooltips":
Tooltip title: {{ plant_name }}, {{ city }}, {{ statename }}
Tooltip body: "Energy source: {{ primsource }}
Capacity in MW: {{ total_mw }}"
This will give you:
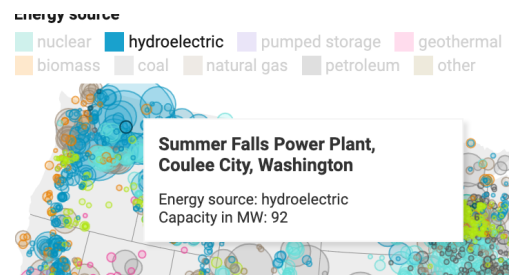
Then **publish!**

Optional: Make a series of bar charts to show share of energy source

Since we had a CSV, I decided to load it into a spreadsheet as well and do some summary analysis.

While it might not make sense to roll up the power plants by state, since – except Texas! – most states share and sell energy with other states – we can roll them up by the type of energy source they use (primary energy source, I should say).

If you open the Google Sheet "EIA Power Plants 202004" and go to the third sheet, entitled, "Summary Table w Category" you can **copy the populated cells to the clipboard**.

Go to Datawrapper and select "new chart."

For "upload data," paste the contents of "Summary Table w Category."
Inspect the data. If it looks like this, you're good.

Click proceed.

For chart type, choose "bar chart" then click proceed again.
For labels, select the column "energy source."
Check "show values."
For "appearance" under "bar values" check "customize"
Put in the same color chips you did for the map.

Also check:

"Bar background"

Sorting and Grouping: check "Reverse Order," "Group bars by Column ("Type")," "Show Group Labels," and "Show Bar Labels."
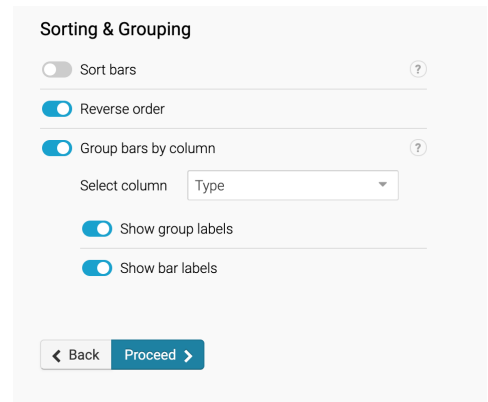
Proceed to "annotate"

Title: "Fossil Sources Still Predominate"

Desc: "U.S. electrical generation capacity by energy source, April 2020"

Source: U.S. Energy Information Agency

Etc.

You should wind up with something like this:

Publish.

If you were putting these graphics into Wordpress, for example, you could embed them both next to each other or in sequence in an… infographic story.

II. Choropleth (Value by Color) Map: Montana Election Results

Now we'll turn our attention back to the first option in Datawrapper's map creation menu: the **"choropleth"** map, which I will agree is somewhat unfortunately named. We could call it a **value-by-color** map, but *not* a heatmap! That's something else, sometimes not even a map.

As I mentioned in the lecture Monday, choropleth maps use:

Administrative boundaries – "human" boundaries, the somewhat arbitrary lines we have drawn on maps to manage, claim and subdivide space. "Canada, you get everything above 49º north, except for those nice islands in the Northwest etc…" We generally don't use choropleth maps for natural features, although they come in handy to represent water basin boundaries (water is only gonna flow one way, so those boundaries are real).

"Normalized" data that represents *rates* and *ratios* rather than raw numbers. There's no point in saying that California or Texas have more unemployed people than other states. They have more of practically everything than most other states. So we compare the unemployment **rate**, right? We can normalize data by dividing it by the square mileage – say, for population density – or by the total population, for per-capita statistics like Covid-19 cases per 100,000 people. Again, for **raw "count" data, you're better off using a Symbol/Bubble** or Dot map – or just a chart.

Hi! What type of map do you want to create?

Choose the map type that will show your data best:
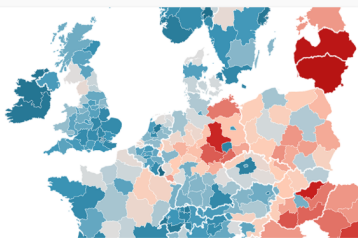
Choropleth map

Color regions to show data like unemployment rates or election results on a map. Upload your own map or use any of our more than 2000 maps. The resulting map is responsive & interactive.
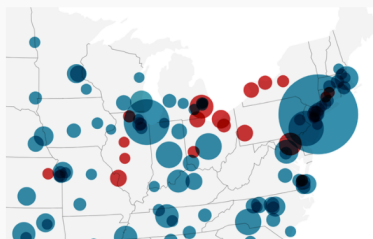
Symbol map

Create symbols sized and colored according to your data. Works great for specific locations (like cities). Upload your own map or use any of our more than 2000 maps. The resulting map is responsive & interactive.

Locator map

Add markers to a map to show where something is located or happened, e.g. events within a city. Perfect for showing readers the places you mention in an article. The resulting map is responsive and static.

I mentioned administrative boundaries. Those can be precious commodities if you make a lot of maps, and a data journalist who does GIS will generally keep a stash of them on their computer. But that's less and less necessary now that services like Datawrapper and Flourish are here, and building out their library of boundaries.

Datawrapper has things like US counties, congressional districts, etc, up to and including European Parliament areas, city districts of Kabul, Afghanistan, Australian postal codes, etc. So if the boundaries are already in Datawrapper, all you have to do is load up a spreadsheet with values to populate those areas. It will even color them for "no data" if you have some missing values.

1. Get the Data

"An Extremely Detailed Map of the US Election"

In what is becoming a bit of a tradition, The New York Times has rolled out precinct-level results from the 2020 election for many – though not all – US states. They have an interactive map that lets you zoom in and hover over every election precinct, and see the vote totals and the margin of victory.

This took tremendous time and effort on their part, but they've kindly shared the fruits of their labor on the code-sharing platform Github. The data file is enormous, and I don't recommend downloading it, because it requires a GIS application to read and it will make your computer cough and smoke to render it, as it did mine.

However, it was worth it, because I was able to trim out just the Montana precincts, which are available to download from my website in two formats: GeoJSON and CSV. You will need them both to complete this lab.

Data files you need to download:

mt-precincts-with-results.json

GeoJSON file with all 662 election precinct boundaries in Montana. These vary greatly in size, based on population and are smaller in the center cities. This file includes the election results – but Datawrapper won't read the values. It will only use the map boundaries.

Size: 8.4mb (too large for Datawrapper)

Snippet:

mt-precincts-with-results.csv

CSV file with total votes, democratic and republican votes, vote margin in percentage, and votes-per-square-kilometer calculation. This is the file Datawrapper will read to plot data on the map boundaries.

Size: 22kb

Snippet:

GEOID	votes_dem	votes_rep	votes_total	votes_per_sqkm	pct_dem_lead
30001-20	7	88	97	0.1	-83.5
30001-03	227	625	869	0.7	-45.8
30001-21	44	120	165	0.1	-46.1
30001-04	34	181	221	0.1	-66.5
30001-17	34	96	137	0.2	-45.3

2. Use Mapshaper to simplify the map data

As I mentioned above, the GeoJSON file is too big for Datawrapper to process. Map files can be no larger than 2mb, and this one clocks in at 8.4. Much too big. Also, while we want this to be a detailed map, we don't need absolute perfection – especially since nobody will see the street grid in this basemap.

Process the data file
Let's go back to Mapshaper.org, which also can accept GeoJSON files as an upload format, in addition to the zipped SHP file that we gave it last time..
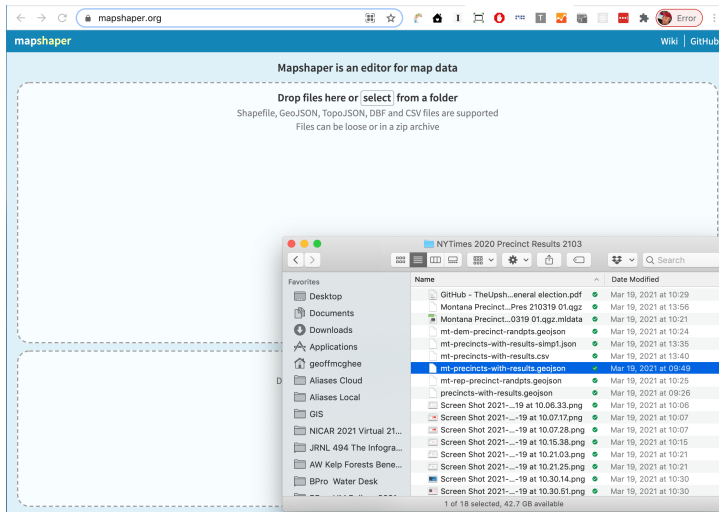
Drag the file named **mt-precincts-with-results.json** into the upload window.

It should confirm the upload and ask you if you want to import it. **Click "import."**
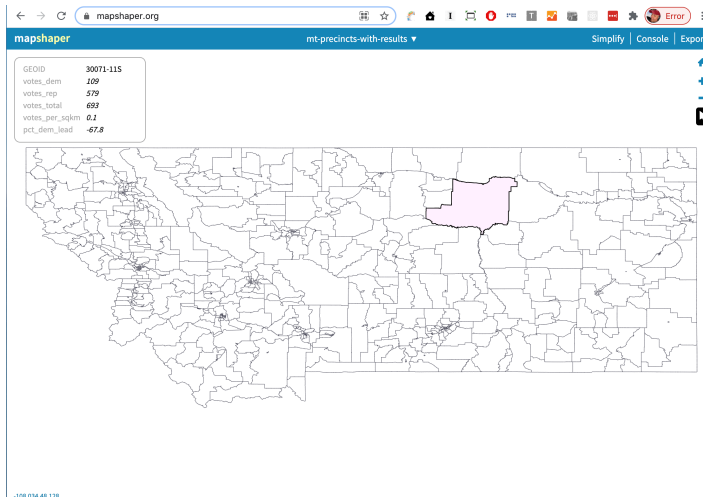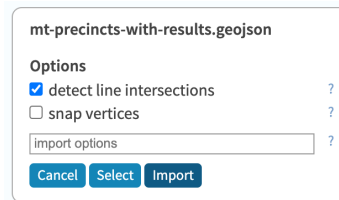
Take a look at the map. As with the point data with power plants, Mapshaper will give you a look at the data.
Click the pointer icon at right, which enables you to inspect the shapes on mouseover. You should see information from each of the six columns in the tabular data.

Note that Montana looks weird. This is "unprojected" map data, which is a kind of unpolished representation.
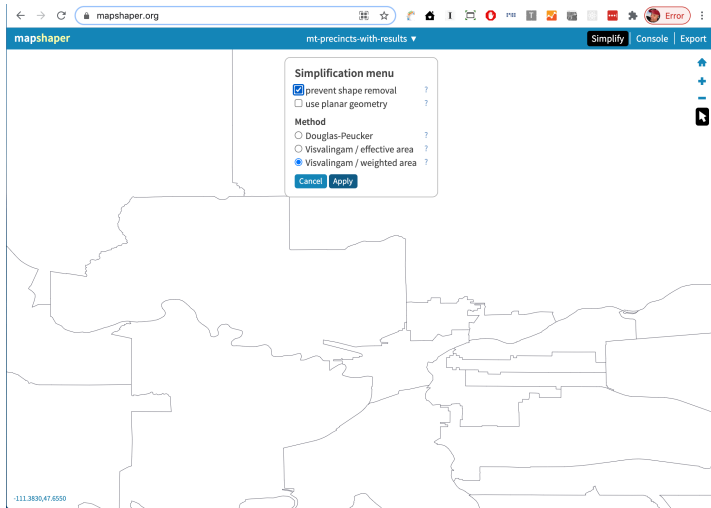
Zoom in close to get a good view of an area with lots of shapes – like a center city. We're going to start simplifying the geometry, and we want to have a close-up look to see how much the simplification degrades the image. You don't want to go too far.

Then **click the "simplify" link** on the upper right.
Make sure **"prevent shape removal" is checked**.

Drag the slider that appears to the right, while looking closely at the map linework. You can also go backward to restore detail.
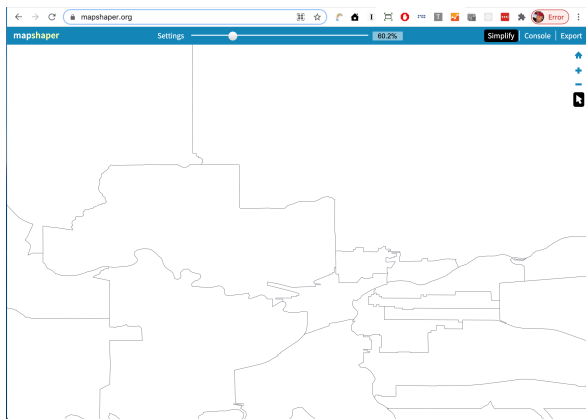
60% of original data retained
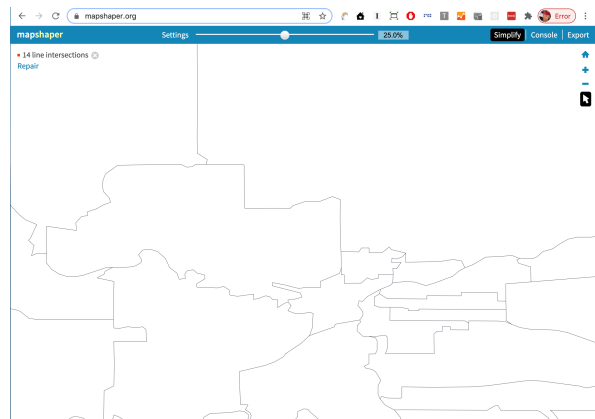
25% of original

I'd ballpark that 25% seems about right – it's a little crude really close up, but Datawrapper doesn't even let users zoom this far, so we should be fine. If it looks like a 1980s video game, you've gone too far.
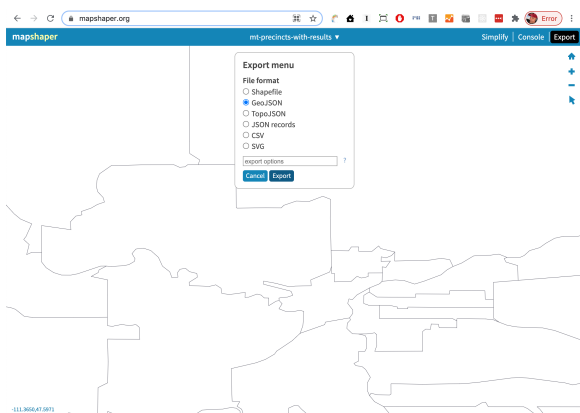
Now **click "export"** on the upper right hand corner, and **set the file format radio button to "GeoJSON."**

Click "export."
*By the way, I've already done it for you, but this is where you **can also export the tabular data from the map as a CSV***.

Locate the exported file in your downloads folder.

How big is it? Just a hair under 2mb (1,977 kilobytes or something?). Should work in Datawrapper now.

I suggest you rename it, to remind yourself that this is the smaller one. My personal convention is to add the level of simplification I've applied for future reference:
So "mt-precincts-with-results.json" becomes "mt-precincts-with-results-**simp25**.json,"
or whatever amount of simplification you applied.

Ok, we are done with Mapshaper. Unless the file is bigger than 2mb! Then you have to return to simplification and make it smaller still, then repeat the steps above.

3. Load the data into a Datawrapper map

Back to Datawrapper. **Click "New Map"** and **select "Choropleth map."** Datawrapper will ask what geography you want to show. I selected USA Counties (2018) down below, but honestly I don't think it matters because we are going to upload our own geography.
Click the "or Upload Map" button.

The next screen should look something like this. There's Montana, looking nicely projected here as well, contrary to Mapshaper.
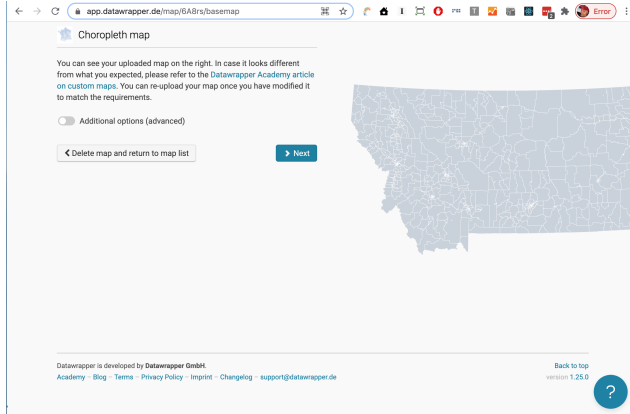
You can ignore the "additional options," which mostly have to do with map projections.

Click Next.
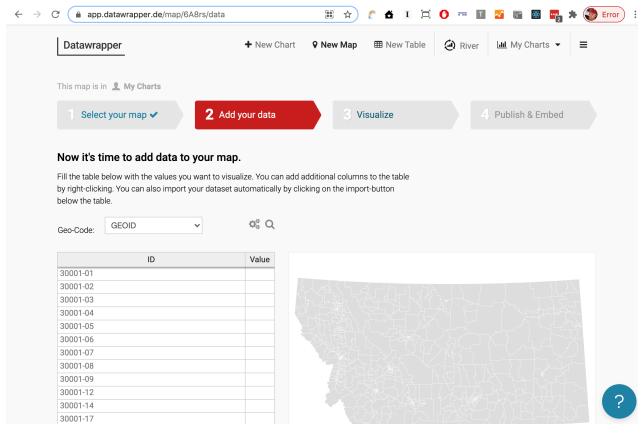
Add data
Now, we know that the map already has data in it, but Datawrapper isn't set up to use that data. Instead, it's modeled on the premise that you have two data sources:
1) **The geography** (GeoJSON file)
2) **The tabular data** (the CSV, or pasted data).

Click the "import" button down below.
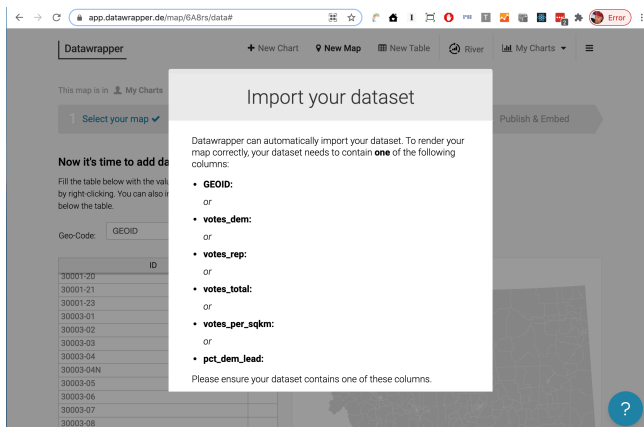
You don't have to do anything here – it's just Datawrapper telling you that your tabular data file has to have column names that match what's in the GeoJSON file –or **at least one matching column**.
This is called a "join," where you're taking two separate data files and stitching – or zipping – them together (I like the idea of zipping them together like the teeth of a zipper, but the "Zip means compress" concept muddies the issue).

Joining data
This is probably the most important thing we'll cover today. How you link, or "join" two separate data files together. Generally in GIS, you're joining geography, like individual state outlines, with tabular data (like population). You've got a GIS file with the outlines of all the states, including Colorado. And you've just gotten new tabular data from the Census that has updated population statistics, which you want to map.

The hypothetical GIS File has a column called "State_Name" with values like "Colorado." The CSV file has a column called "State", let's say, with values like "Colorado", as well as another column called "Pop_2020" with values like "2,793,330".

You want to get that value for Pop_2020 into the map so you can make a Symbol or Bubble Map (not choropleth, right? Because it's raw count data?). So you tell the GIS, let's say, to "join" the two files by connecting records where "State_Name" in one matches "State" in the other.

Or better yet, you have a numeric id or code, like a state postal code, to match up on because it's more rigorous (someone might say "New York State" in one place and "New York" in the other, and boom, no population for you because they won't match).

So here, we have a column called GEOID in both files, which is a unique identifier. That's how Datawrapper will recognize which data in the table gets displayed in the map. And it will use that matching ID to grab all five data columns and make them available to the map (Votes for Democrats, Votes for Republicans, Total Votes, Democratic Margin, Votes per Sq Mile).

Scroll down that warning screen to the **"Start Import" button. Click it.**

Click the link that says, **"Alternatively, click here to upload a CSV-file,"** and select the CSV file **"mt-precincts-with-results.csv."** Then click "Next."

This is a sort of unfortunate Datawrapper UI bug. They don't have enough room for all the columns, so the "Match" buttons tumble down to a second line. But make sure that you keep "Matched as GEOID" selected – don't click the lower one because that will select the wrong column.

Then **scroll down and click "Next."**

Import your dataset

In the next step, you will select the **column that should be visualized on your map**. The column needs to contain a number. Datawrapper will then automatically create a color scale for your values.
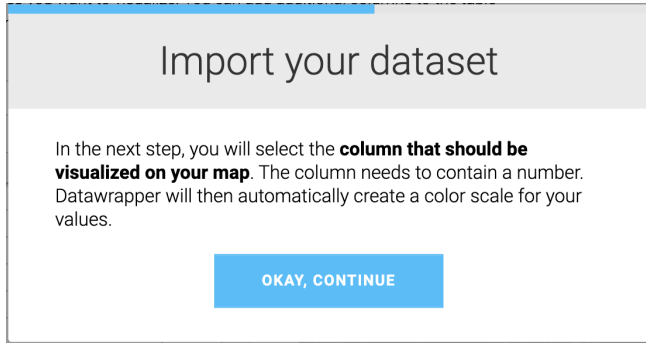
OKAY, CONTINUE

Now you'll get a message from Datawrapper saying you need to pick one column that will be the primary data point – the one that gets shown through the colors on the map.

Click "okay, continue."

This is a little confusing. What they're saying is, "which of these data points is the most important?" Because Datawrapper is going to color the map according to values in that column.

Match your columns

Please select which column in your dataset contains "**Values**".

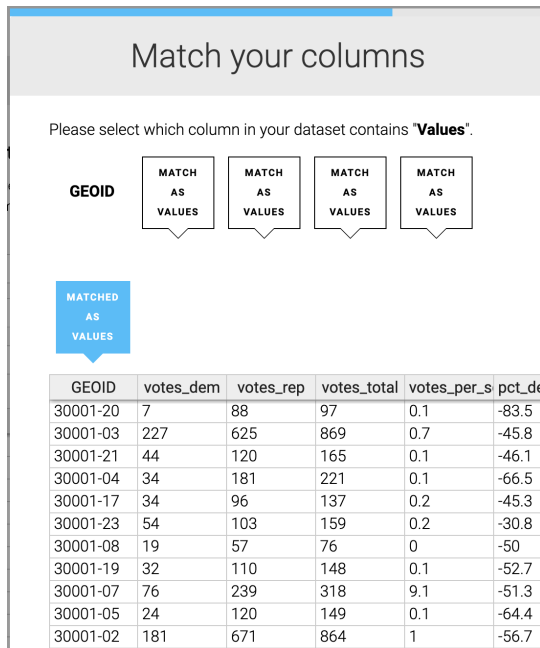| GEOID | MATCH AS VALUES | MATCH AS VALUES | MATCH AS VALUES | MATCH AS VALUES |

MATCHED AS VALUES

GEOID	votes_dem	votes_rep	votes_total	votes_per_s	pct_de
30001-20	7	88	97	0.1	-83.5
30001-03	227	625	869	0.7	-45.8
30001-21	44	120	165	0.1	-46.1
30001-04	34	181	221	0.1	-66.5
30001-17	34	96	137	0.2	-45.3
30001-23	54	103	159	0.2	-30.8
30001-08	19	57	76	0	-50
30001-19	32	110	148	0.1	-52.7
30001-07	76	239	318	9.1	-51.3
30001-05	24	120	149	0.1	-64.4
30001-02	181	671	864	1	-56.7

Since we want to show the margin between Democratic and Republican vote totals (in percentage), click the left-most button in the second row, which corresponds to the last column, **pct_dem (which is actually the margin). Confusing! I know! Just go with it. UI glitch.**

Then **scroll down and click "Next."**

If all went well, you should see a preview of the data that looks like this:

GEOID	pct_dem_
30111-56.3	-38.9
30111-56.2	-39
30111-56.1	-51

That it says "pct_dem" means that it grabbed the correct column. If it says something else, click "import your dataset" again to start over.

4. Adjust the Visualization Settings

Datawrapper has many, many sensible defaults built-in, but it's not picking well this time. This is America, and we need RED AND BLUE (and colors in between).
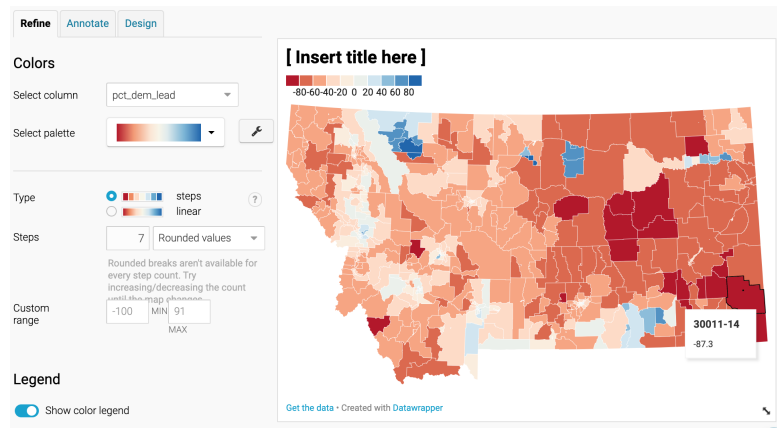
Also, we want what's called a **diverging color palette**, so that it has intensity in two directions, with a mushy middle. **Click "select palette."**
Mouse down to the ninth choice, **red-to-blue diverging**, and click it.
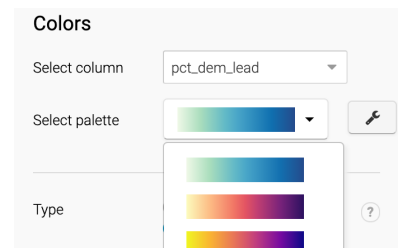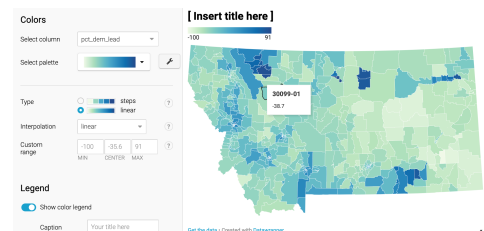
Change "type" to "steps" from "linear"
Set the steps to 7, and **change "linear" to "rounded values."**
You can leave "custom range" alone.

Check "show color legend," which is probably already activated.
For "Caption," enter "Vote Margin for Democratic vs Republican Candidate."
Leave "Labels" as "ruler"
Change "Format" to "0.0%" to show percentage with one decimal point of precision.
Click "Show advanced options" and:
Check "Enable highlighting on hover"
Size: 300px – this will give it more breathing room.

Appearance
Check "Make map zoomable." Yes, please, we've got detail!
Button position: **top right.**
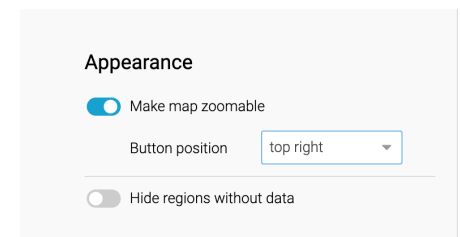
Click "Proceed." We're almost there.

On the Annotate Screen, enter the headline and description.

Title: **2020 Presidential Vote in Montana**
Description: **Precinct-level results, November 2020**
Data Source: Montana Secretary of State via NYTimes
Link to data source: https://sosmt.gov/elections/results/

Map Labels
Select column: **pct_dem_lead**
Number format: **0.0%**
Min zoom for labels (very important!) **Set this to 3** so it doesn't try to draw 662 data labels when you're looking at the whole state.
Check "prevent label overlapping" as well.

Tooltips

Scroll down to the lower part of the page. **Check "Show tooltips"** and then click "Customize tooltips."

For title, enter: "`Precinct {{ geoid }}`"
For body, "`Votes: <b>{{ votes_total }}</b><br>`
`Votes per sq mi.: <b>{{ votes_per_sqkm *.6 }}</b><br>`
`Trump/Pence: <b>{{ votes_rep }}</b><br>`
`Biden/Harris: <b>{{ votes_dem }}</b><br>`
`Margin for Democrats: <b>{{ pct_dem_lead }}</b>`"

This will insert live data values into the text that displays when you mouse over a precinct.

At this point, you're pretty much ready to publish.

Two things that are disappointing about the map as-is, though:
1) There's no city labels
2) The voting precincts are only identified by their unique identification code, which is lame
We can't fix 2), but we can do something about the city names.

Go back to the "Visualize" step and select "Annotate" from the smaller tabs.

Scroll down and find "Text annotations." Click the button labeled "+ Add Text Annotation."
Let this be a test of your Montana geography knowledge.

Click where Missoula is, or would be, and replace the placeholder text with "Missoula."
Change the type size to 12 px
And check "Text Outline" so it gets a little bit of extra separation from the map below.

Repeat as many times as you like with other place names.

Precinct-Level Presidential Election Results, 2020
Vote Margin for Democratic vs Republican Canddate
80% 60% 40% 20% 0% -20% -40% -60% -80%
Missoula