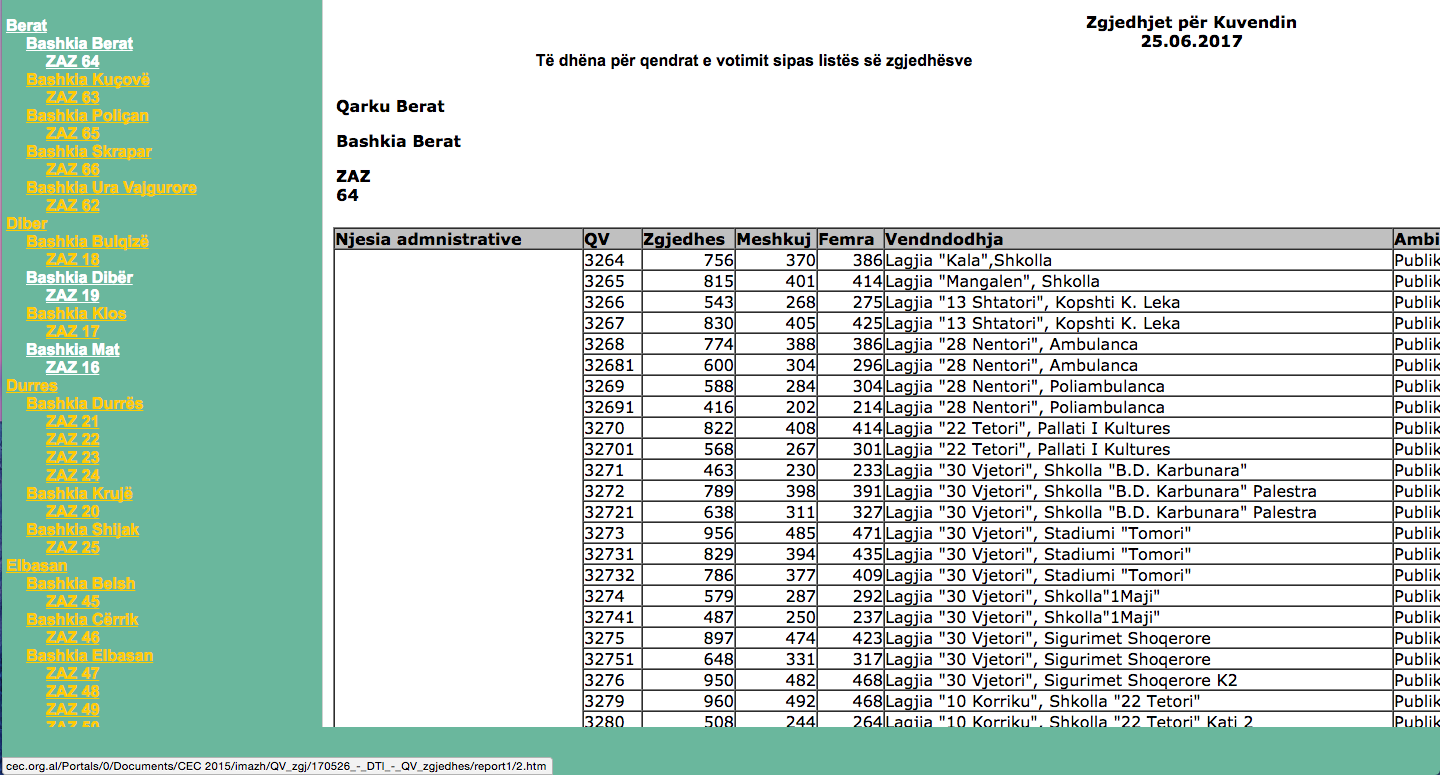
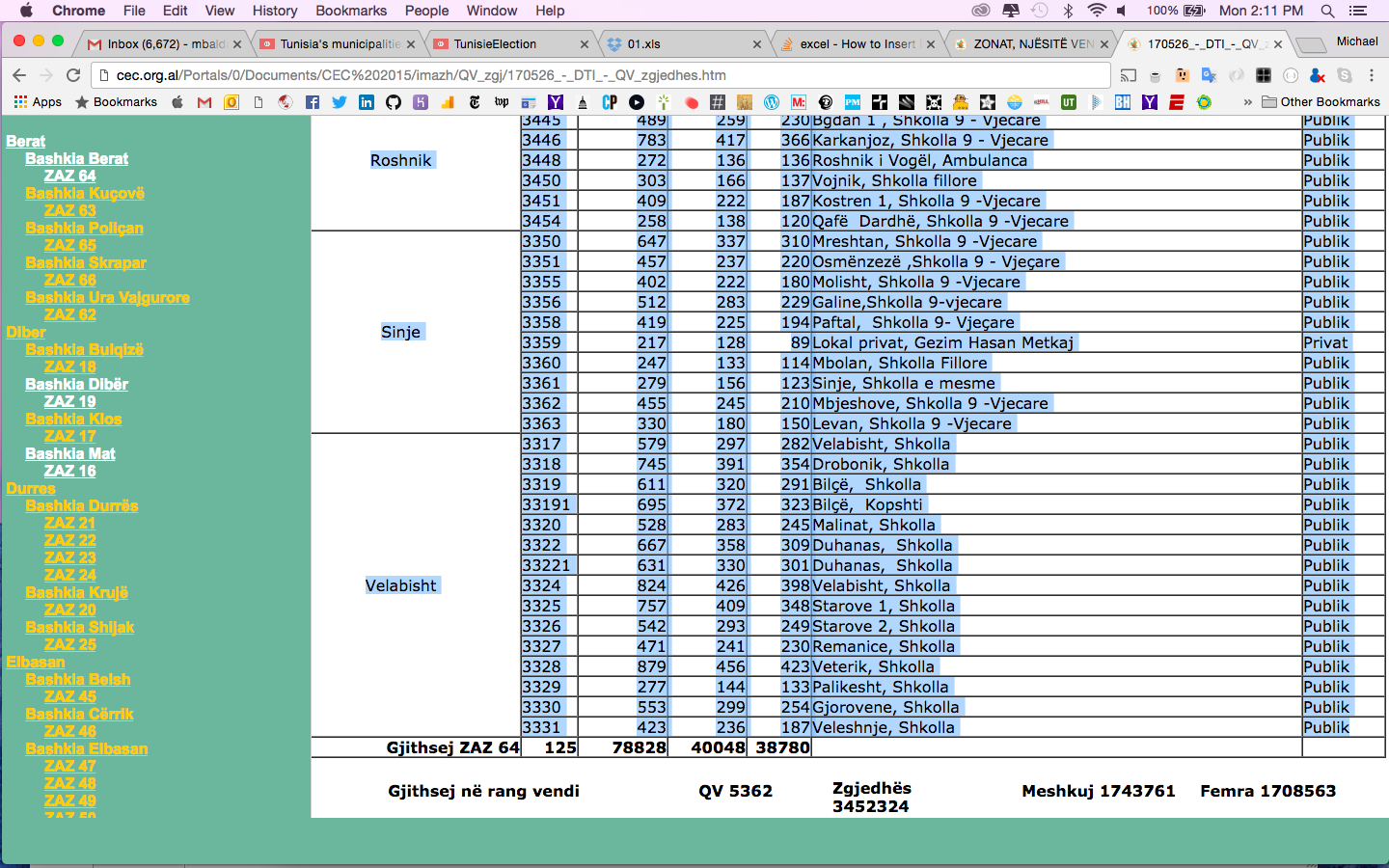
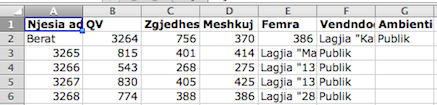
**1) Scrape KQZ website to put polling station data in Excel spreadsheet**

For the 25 June 2017 elections, KQZ published “Të dhëna për qendrat e votimit sipas listës së zgjedhësve” in HTML table format at the following URL: <http://cec.org.al/Portals/0/Documents/CEC%202015/imazh/QV_zgj/170526_-_DTI_-_QV_zgjedhes.htm>. Disaggregated by ZAZ, the published data table included the following: Qarku, Bashkia, ZAZ, Njesia administrative, QV, Zgjedhes, Meshkuj, Femra, Vendndodhja, and Ambienti:

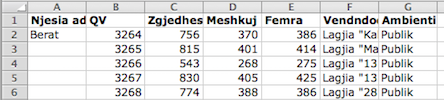
To be able to use this data, we need to take it from the HTML page and put into a spreadsheet format (XLS, at first). The process of taking data from a HTML page and putting it into a spreadsheet is called “scraping”. There are many ways to “scrape” a web page, but one of the least technical ways to do so is to simply copy, paste and clean. This can be quite time consuming, but for small-to-medium size datasets it is often faster to copy and paste than to write and run a “scrape code” that automates the process.



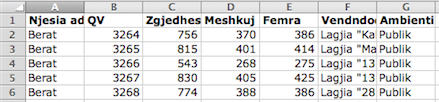
For each ZAZ, using your scrollpad or mouse, highlight the table data from the HTML page starting from the Njesia administrative until the bottom (don’t include the last row of the table “Gjithsej ZAZ #”).

Copy the data and paste it into a blank Excel spreadsheet. You will find that it will not paste in a clean format. Specifically, it will only paste the njesia administrative in the first cell in which it appears:

Now you have to engage in a tedious process to format the spreadsheet: insert an empty a column before column A (a new A column) and delete “empty cells” in the new A column to shift njesia administrative values to the left:

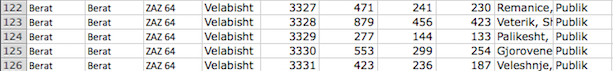


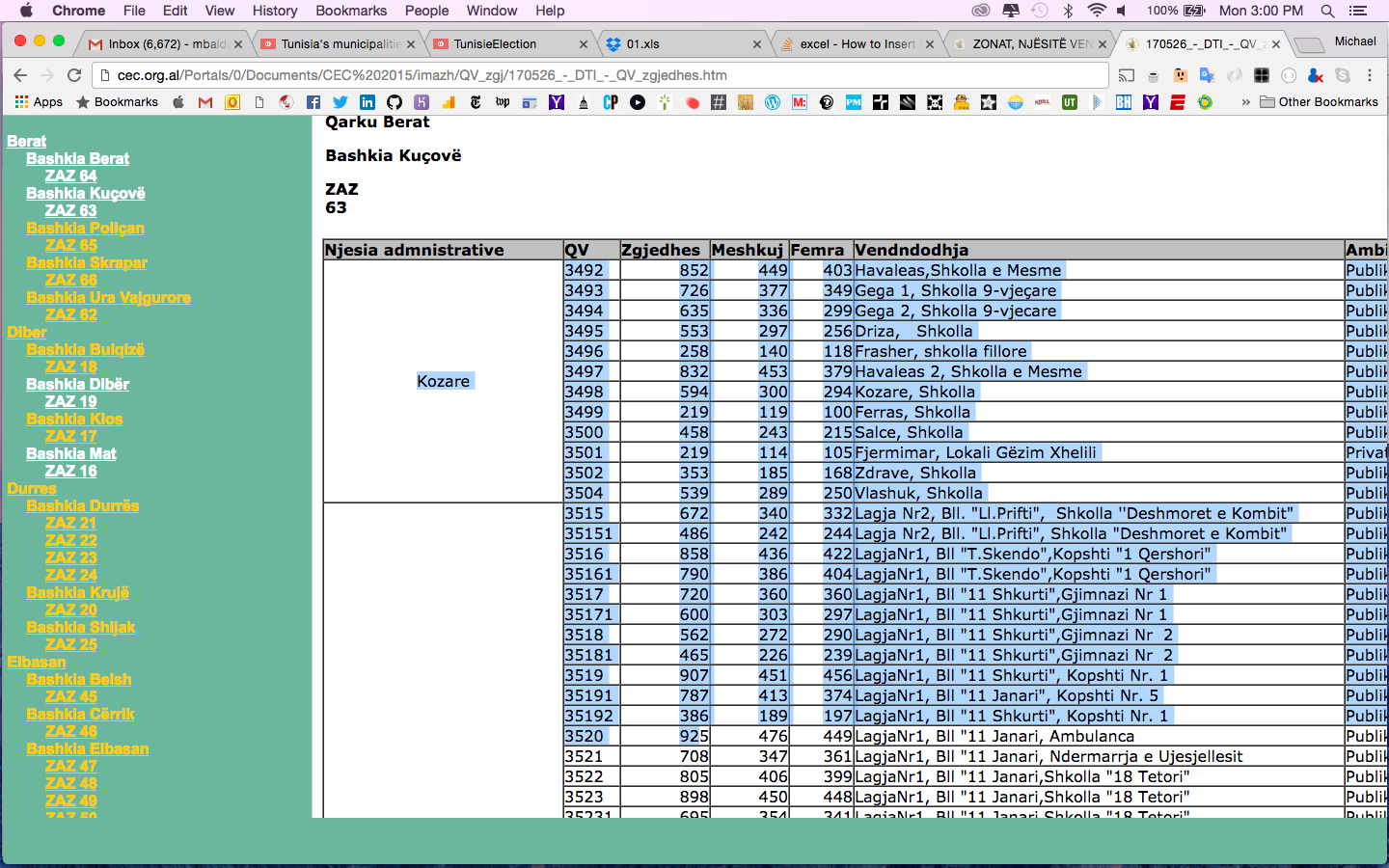
For each njesia administrative value, drag the bottom right corner of the cell to fill the empty cells with the value. The result should be that each QV has a corresponding njesia administrative value:



You also want to capture the qarku, bashkia, and ZAZ values for the polling stations, which are located at the top of the HTML page. You’ll need to insert three more empty columns and label them qarku, bashkia, and zaz respectively.

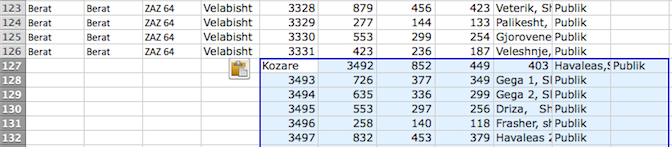
Enter the qarku, bashkia and zaz names in the first empty cells (A2, B2 and C2). For the qark and bashkia values, you can drag and fill until you fill all the rows. For the zaz value, in cell C3, you need to enter the formula: =C2. If you drag and fill, the number of the ZAZ will increment. From cell C3, drag and fill until you get to the last row.



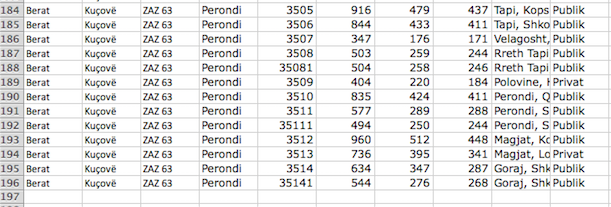


For the next ZAZ, using your scrollpad or mouse, highlight the table data from the HTML page. This time, don’t include the table head row which contains the column names and don’t include last row of the table “Gjithsej ZAZ #” either.

Click on the first empty cell in column E below the last row of data in the spreadsheet. Then click paste.



You’ll need to engage in tedious data formatting again (and again) until you’ve captured all ZAZ data in the spreadsheet.



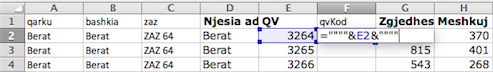
Once you’ve captured all the ZAZ data in the spreadsheet, you need to create a new column of values that contains a “text” value for the QV. The reason: many QV values contain leading zeroes (e.g. 0923). Later, when you convert an Excel spreadsheet to a CSV file, leading zeroes will be automatically removed. To maintain the integrity of data that contains leading zeroes, you need to wrap the values inside quotation marks. This preserves a numerical value as a text value. Computer jargon lesson: in CSV files, a text value is referred to as “string” while a numerical values is referred to as “real”. Real values cannot contain leading zeroes and, thus, must be converted to strings (inside quotation marks).

There are many ways to wrap real values in quotation marks, but the easiest is:

1) Insert a blank column next to the column with values to wrap in quotation marks (column F)

2) Give the column a name (qvKod)

3) In the first empty cell in the new column, enter the following formula: =“”””&E2&””””” (note: use four quotation marks on each side of the &)



In F2, you should now see the value from E2 in quotation marks (e.g. “3264”)

4) Click on the bottom right corner of F2 and drag it to the last row

5) When you’re done, “copy” the entire column and “paste special” as “values”

Software applications that read CSV files have difficulty parsing column names that contain spaces. To make the Excel spreadsheet “machine-readable”, replace the column names with “camelCase” names (e.g. Njesia administrative should change to “njesiaAdministrative”, QV to “qv”, Femra to “femra”).

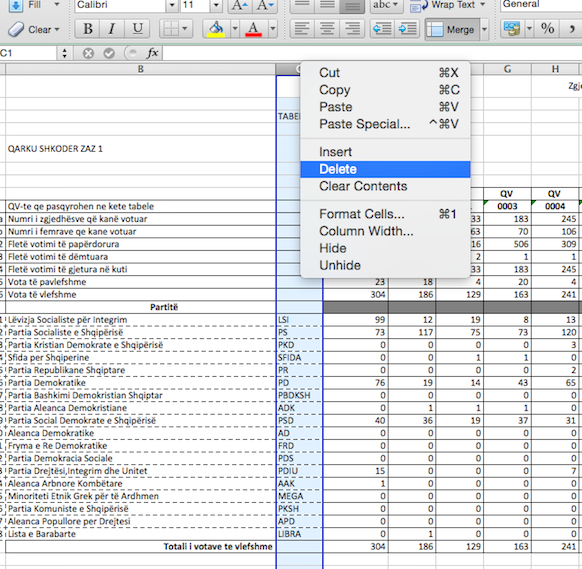
Macintosh HD:Users:mbaldassaro:Desktop:screenshots:Screen Shot 2017-08-07 at 4.24.11 PM.png

At this point, you could technically convert this Excel spreadsheet to a CSV file and you would have a full list of polling station data in a machine-readable format – the critical first step to making data open!

But we’re not done yet: leave the Excel spreadsheet it in XLS format and next we’ll add results data.

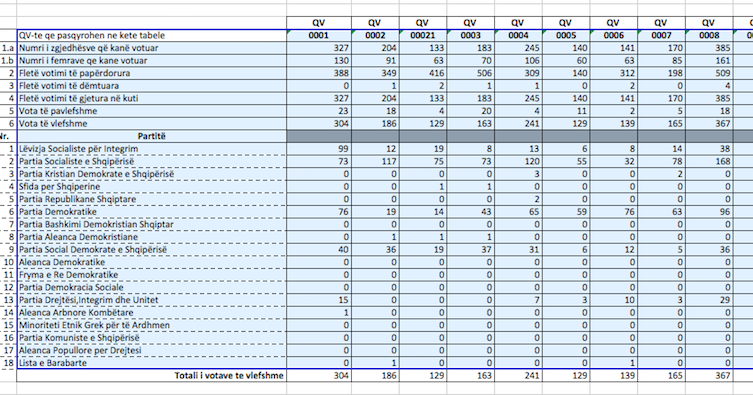
**2) Add results data to the Excel spreadsheet containing polling station data**

Download the 2017 ZAZ results sheets and put them files in a folder (you always want to keep source data whenever possible). It will be helpful in case you need to check or reference these documents at a later stage.



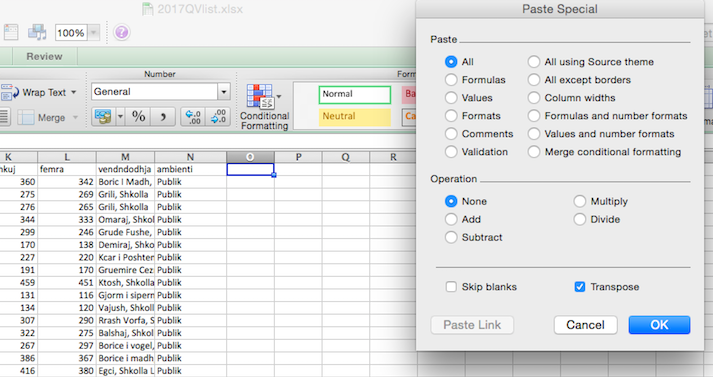
Open the 01.xls file containing all the polling station-level results data for ZAZ 1. In this first file, delete column C containing abbreviations of the political parties.

Once you have deleted column C, select the data from B9:CC35, starting with the cell containing the text “QV-te qe pasqyrohen ne kete tabele” and ending with the cell containing the vote results for the “Lista e Barabarte” at QV 0076. Do not include the data in the last row containing the text “Totali i votave te vlefshme” or the data in the last column labeled “Totali”

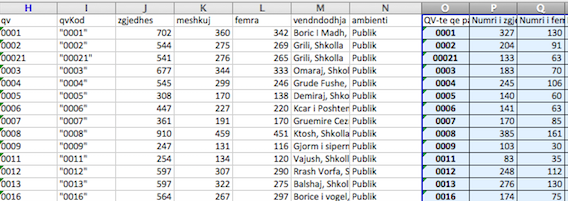


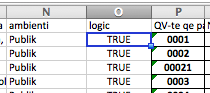
Copy the selected data and, in the spreadsheet containing the polling station data you scraped earlier, click on the empty cell immediately to the right of the word “ambienti”. From the empty cell, select “Paste Special” and the Paste Special dialogue box will appear on the screen.

In the paste special dialogue box, check the box for “Transpose” and click “OK”

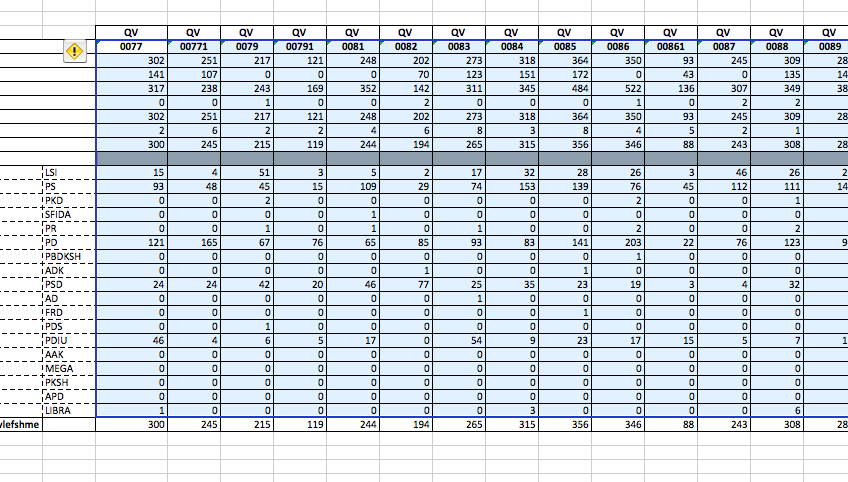


The “Transpose” function will change the orientation of the data, rearranging data contained in columns to rows. The data contained in column C of 01.XLS will now fill top row and data contained in subsequent columns will fill the rows beneath.



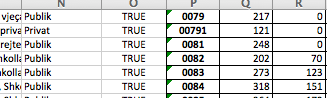
The QV numbers should automatically match up. If you find it useful, you can insert a column to add a logic test and return a TRUE response if the QV codes match. The generic formula for a logic test is: =IF(A2=B2, TRUE)

Open the 02.xls file containing all the polling station-level results data for ZAZ 2. This time, do not include the column containing the description data for each row. Only select and copy the data from D9:DK35, starting with the cell containing the text “0077” and ending with the cell containing the vote results for the “Lista e Barabarte” at QV 0347.



Click on the empty cell in the column labeled “QV-te qe pasqyrohen ne kete tabele” and in the row with polling station data for QV 0077. Follow the Paste Special > Transpose process you followed earlier to fill the rows with the data.

*Note: it is helpful to use the “Freeze Panes” function to freeze the top row and keep the column names visible while you are pasting data. To freeze the top row the column names, select the top row and go to Window > Freeze Panes.*



The logic test column can be used to verify you’re pasting correctly.

Repeat this tedious process until you have copied and pasted all the data from all 90 ZAZ files. The Excel spreadsheet should ultimately contain all polling station data scraped from the HTML page and results data from the ZAZ files.

After you have finished copying and pasting all data from the ZAZ files, you must:

\*Delete the logic test column – this column has no practical value

\*Delete the empty Partitë column – this column has no value at all

\*Change the column labels to camelCase format – for parties, I suggest using the abbreviations from column C of the ZAZ Excel spreadsheets

\*Optional: you can delete the redundant column containing QV numbers not enclosed in quotation marks - leaving it in the file neither helps nor hurts.

*Note: if you used the Freeze Panes option to freeze the top row with the column headers, you can unfreeze it now.*

Now you have a full dataset that includes all polling station voter and results data.

**3) Adding key columns to be able to join CSV and shapefile data**

To be able to visualize your data on a map, you need to join your data to a file that contains geocoded data called a shapefile. Geocoded data can be stored in many file formats, but the most common are CSV, JSON / GeoJSON (Geospatial JavaScript Object Notation), and SHP (Shapefile).

A brief note on geospatial data and these three formats: There are three common kinds of geospatial data: point, line, and polygon.

* “Point” refers to a single latitude and longitude point that correlates to a specific landmark or address (e.g. a polling station).
* “Line” refers to multiple latitude and longitude points that form a road or waterway (e.g. road connections from CDO’s office to a polling station)
* “Polygon” refers to a cluster of latitude and longitude points that form a district, state, or country (e.g. bashkia Tiranë)

Point data is straightforward and can be added to a CSV file using latitude and longitude columns. A CSV file with geocoded point data can be as simple as:

place latitude longitude

KVV 41.317644 19.813512

Partia Demokratike 41.325544 19.823499

Partia Socialiste 41.327060 19.816882

To prove that simple point data in a CSV file renders a map, try the following:

1. Put the above data in an Excel spreadsheet and save it as a CSV file
2. Go to <https://www.google.com/maps/d/>
3. Click the “+ create a new map” button at the top
4. Click “Import” in the dialogue box at the top left corner of the screen
5. Upload (or drag) the CSV file you just created
6. When prompted to “Choose columns to position your placemarks”, make sure “latitude” and “longitude” are checked and click “Continue”
7. When prompted to “Choose a column to title your markers”, make sure “place” is checked and click “Finish”

You should get a map with three point icons for KVV, Partia Demokratike and Partia Socialiste.

Bravo! You can change base maps, add styles, add more layers and even share the map with your friends!

Line and Polygon data are more complex and, while they can be contained in CSV files, they are more commonly found in JSON / GeoJSON or SHP files.

SHP files are old-fashioned and tricky to work with (the subject of a separate tutorial at some point), so we’ll focus on how data is stored in modern JSON / GeoJSON files. Note: to be able to open CSV files, you’ll need a text editor like Atom (<https://atom.io/>) or Sublime Text (<https://www.sublimetext.com/>).

*Note: going forward, I will refer only to GeoJSON, a variation of JSON that uses the same structure and format. You’ll see files saved with a suffix of .json or .geojson - the “geo” prefix is used to indicate the file contains geospatial data.*

A GeoJSON file is structured as follows:

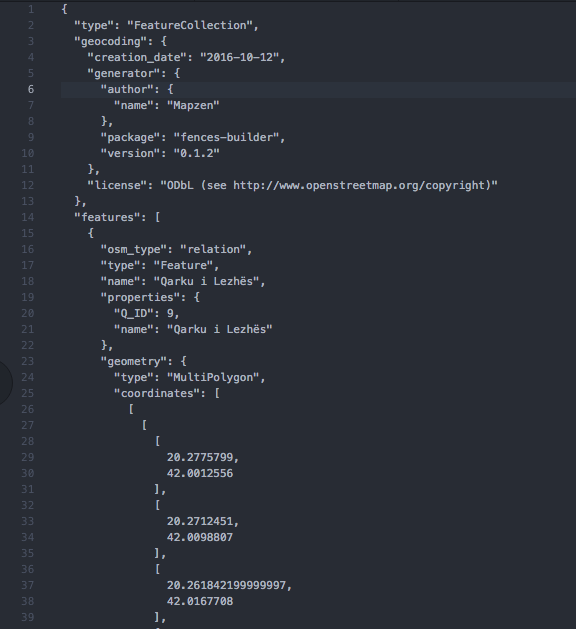
* At the beginning of the file is some basic metadata that explains the “type” of data contained in the file, usually a “Feature” (a single point, line, or polygon) or “Feature Collection” (multiple points, lines or polygons)
* Each Feature in a GeoJSON file contains two key components:
  + Properties: elements of the Feature which, at a minimum, includes the “name” of the feature itself (e.g. bashkia Tiranë)
  + Geometry: latitude and longitude coordinates for each feature that creates a point, line or polygon. For polygons, the geometry is a large number of points that connect to form location boundaries.

Below is a snapshot of the first 30+ lines of a GeoJSON file containing qark data (this GeoJSON file is available in the AlbaniaElectionData GitHub repository). You can see at the beginning of the file is metadata that states the type is “FeatureCollection”, which indicates that it contains multiple points, lines or polygons (in this case, polygons). There is some other data explaining the date, author, and license, which you don’t need to be concerned with at this stage.

Beneath the metadata, you find “features” which includes two important elements: “properties” and “geometry”. Here “properties” has only two attributes:

* “Q\_ID”: 9,
* “name”: “Qarku I Lezhës”

You can also see “geometry” has multiple latitude and longitude points. These points go on and on for hundreds of lines of code. Collectively these individual longitude and latitude points constitute the boundaries of Qarku I Lezhës.



What we care about in this GeoJSON file is the “properties”, which is where polling station data we want to visualize will eventually be joined. It is *technically* possible to manually add all the data that you want to visualize by simply adding a whole bunch of attributes to the “properties” section. IF you wanted to add population data for Lezhë, you could manually add a new property attribute:

“Q\_ID”: 9,

“name”: “Qarku I Lezhës”,

“population”: 135613

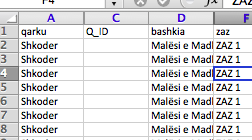
Rather than manually adding all the data we want to visualize, we’re going to join the data automatically from the CSV file with the GeoJSON file. To be able to join the data and shapefile together, we need a shared property attribute in both files. This shared property used to join data and shapefile is called a “key” value.

As you might have guessed, this GeoJSON file has 12 features – one feature for each of the 12 qarks. For each feature, there is a “properties” element that has a “Q\_ID” value. For “Qarku I Lezhës”, the “Q\_ID” value is 9. If you were to scroll through all the features, you would find the following “Q\_ID” values:

|  |  |
| --- | --- |
| name | Q\_ID |
| Berat | 1 |
| Diber | 2 |
| Durres | 3 |
| Elbasan | 4 |
| Fier | 5 |
| Gjirokaster | 6 |
| Korce | 7 |
| Kukes | 8 |
| Lezhe | 9 |
| Shkoder | 10 |
| Tirane | 11 |
| Vlore | 12 |

To be able to data from our CSV file data with the qark shapefile, we need to insert a column with the label “Q\_ID”. In the CSV file, the values for “Q\_ID” in the cells in that column should correspond to the “Q\_ID” values in the GeoJSON file.

In the CSV file, insert a column next to the qarku column with the label Q\_ID



In the empty cells in the Q\_ID column, enter the corresponding Q\_ID value found in the GeoJSON file, and drag and fill until you reach the end of the qark (in the case above, enter 10 for all the Q\_ID values that correspond to qarku Shkoder).



Repeat this process until all the corresponding Q\_ID values in the CSV file match the Q\_ID values for qarks in the GeoJSON file.

In addition to the GeoJSON file containing qark feature data, we also have GeoJSON files containing the 61 bashkia and 373 njesia administrative features. In the bashkia GeoJSON file, there is a properties element called “BAS\_ID”. The 61 BAS\_ID “properties” element values of the bashkia GeoJSON file are below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| bashkia | BAS\_ID | bashkia | BAS\_ID | bashkia | BAS\_ID |
| Belsh | 41 | Klos | 24 | Përmet | 66 |
| Berat | 11 | Kolonjë | 72 | Pogradec | 75 |
| Bulqizë | 21 | Konispol | 124 | Poliçan | 15 |
| Cërrik | 42 | Korçë | 73 | Prrenjas | 46 |
| Delvinë | 121 | Krujë | 33 | Pukë | 104 |
| Devoll | 71 | Kuçovë | 13 | Pustec | 77 |
| Dibër | 22 | Kukës | 83 | Rrogozhinë | 115 |
| Divjake | 51 | Kurbin | 91 | Rroskovec | 57 |
| Dropull | 61 | Lezhë | 92 | Sarandë | 125 |
| Durrës | 31 | Libohove | 64 | Selenicë | 126 |
| Elbasan | 44 | Librazhd | 48 | Shijak | 36 |
| Fier | 52 | Lushnje | 54 | Shkodër | 105 |
| Finiq | 122 | Malësi e Madhe | 102 | Skrapar | 17 |
| Fushë - Arrëz | 101 | Maliq | 74 | Tepelenë | 68 |
| Gjirokastër | 62 | Mallakaster | 55 | Tiranë | 111 |
| Gramsh | 43 | Mat | 25 | Tropojë | 88 |
| Has | 81 | Memaliaj | 65 | Ura Vajgurore | 19 |
| Himarë | 123 | Mirditë | 94 | Vau i Dejës | 108 |
| Kamëz | 112 | Patos | 56 | Vlorë | 128 |
| Kavajë | 113 | Peqin | 45 | Vorë | 119 |
| Kelcyre | 63 |  |  |  |  |

Next to bashkia, insert a column with the label “BAS\_ID” and, in the empty cells in the BAS\_ID column, enter the corresponding BAS\_ID value found in the GeoJSON file, and drag and fill until you reach the end of the bashkia.



Repeat this process until all the corresponding BAS\_ID values in the CSV file match the BAS\_ID values for bashkia in the GeoJSON file.

Finally, for the 373 njesia administrative features, in the njesia administrative GeoJSON file, there is a properties element called “COM\_ID”. The 373 COM\_ID “properties” element values of the njesia administrative GeoJSON file can be found in an Excel spreadsheet named “geokey.xlsx” enclosed with this tutorial.

Next to njesiaAdministrative, insert a column with the label “COM\_ID” and, in the empty cells in the COM\_ID column, enter the corresponding COM\_ID value found in the GeoJSON file, and drag and fill until you reach the end of the njesiaAdministrative. Repeat this process until all the corresponding COM\_ID values in the CSV file match the COM\_ID values in the GeoJSON file.



As a last step, you need to do a “find and replace” the following letters: ç with c and ë with e. Unfortunately, CSV files do not read these Albanian alphabet characters and automatically replace them with punctuation marks, symbols, or spaces. This creates a massive headache when trying to work with the data later.

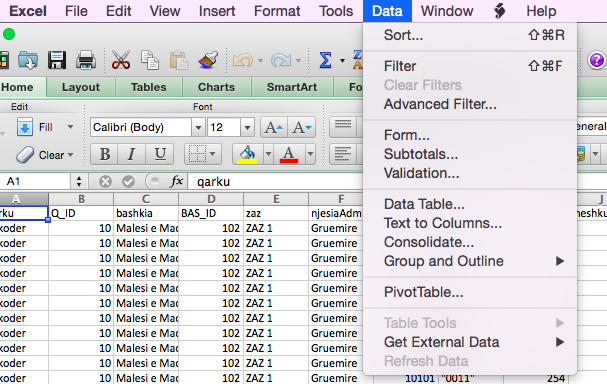
At this juncture, you can save the file as a CSV file, you have a full machine-readable dataset of polling station registered voters and results data as well as “key” values to join CSV data with shapefile data.

*Note: it’s always good to keep the original XLS file and make a duplicate to save as a CSV file just in case the CSV file somehow gets lost or corrupted.*

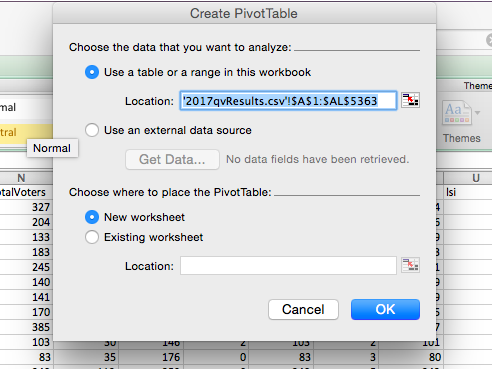
Now we’re ready prepare our datasets for merger using Pivot Tables!

**4) Processing and analyzing datasets using pivot tables**

Now that you have a full CSV file with polling station data and “key” values, you can begin processing the data you want to visualize. The most efficient way to process data is with a pivot table. A pivot table is a data processing tool that allows you to summarize spreadsheet data in a quick and efficient manner.

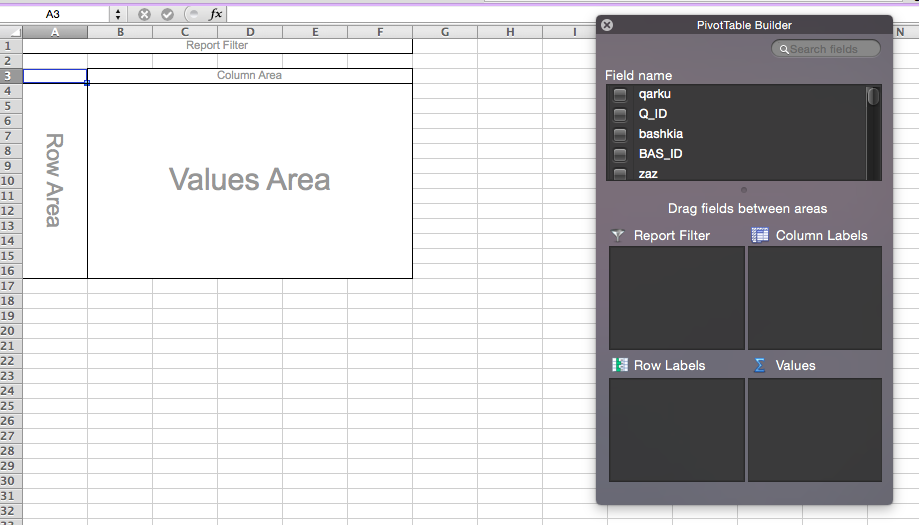


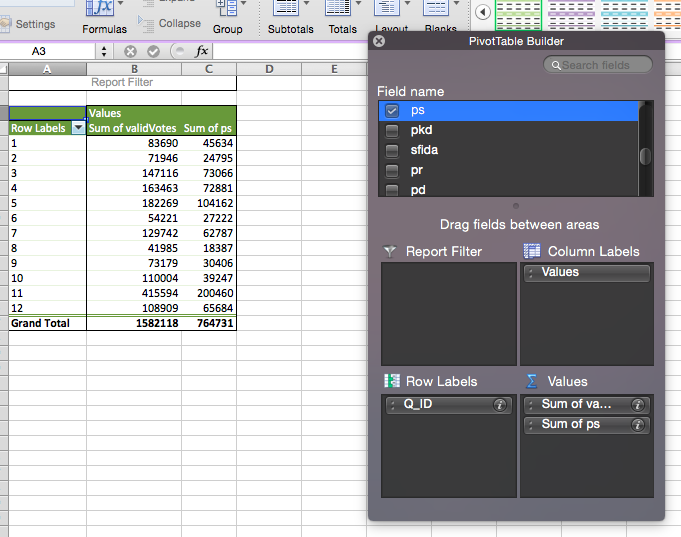
Let’s start by using a pivot table to summarize the total number of valid votes and the number of votes received by the Socialist Party in each qark. With the CSV file open, in the menu bar, click on Data and select PivotTable:

A dialogue box should appear with two boxes checked.

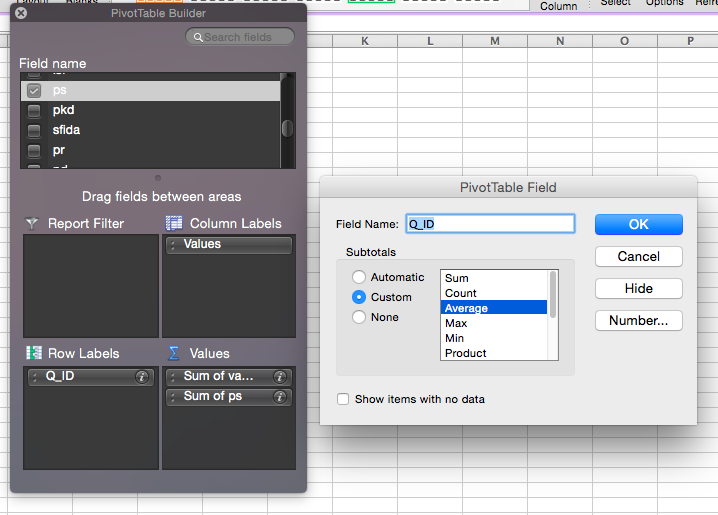
The first is “Use a table or range in this workbook”. The range should include all the data in the spreadsheet and follow this format: ‘filename.csv’!$ A$1:$Z$9999.

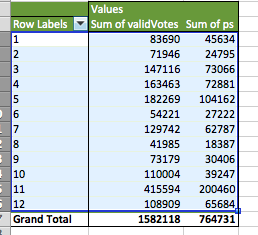
The second is “New worksheet”, which will open a new tab in the current file for your pivot table. With both options checked, click “OK”

A new worksheet tab should open with an empty pivot table and a “Pivot Table Builder” dialogue box. To build your pivot table of valid votes and PS votes by qark, under field name, tick three boxes: “Q\_ID”, “validVotes”, and “ps” (note: I used names for totalVotes and validVotes in English. Your names may be different).

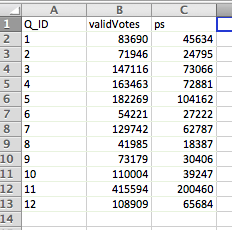
By default, your pivot table should take the value for the first box you checked and place it under “Row Labels”. Subsequent values will be placed under “Values”. The first value I checked was Q\_ID, and the second and third values were “validVotes” and “ps”. You should now have a pivot table with 12 rows – a Q\_ID that corresponds to a qark, the sum of valid votes in each qark, and ps votes received in each qark.

Note: it is important to check and make sure the values in the Pivot Table are reflected correctly. Obviously, the values for validVotes and ps should be “Sum” values. For Q\_ID (or BAS\_ID or COM\_ID), the value should be “Average”. Why? If Tiranë has 1000 polling stations, then there are 1000 Q\_ID values of 11 in the spreadsheet. The “average” value of all these Q\_ID values will be 11, but the “count” would be 1000 and the “sum” value would be 11000. To make sure the values are reflected correctly, click on the “?” next to the Q\_ID property under Row Labels and a PivotTable Field dialogue box will pop up. IF the Q\_ID value is incorrect, i.e. it is a “count” or “sum” value, change it to “Average” and click “OK”.





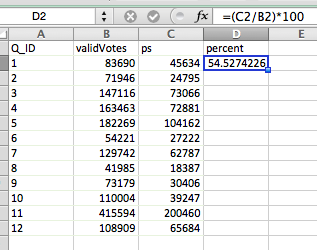
Now that you have your pivot table, select and copy the 12 rows with the Q\_ID, validVotes and ps data. Don’t copy the column labels or Grand Total data – you will not need this data. Open a brand new Workbook (**not** a new tab in the existing sheet) and, with your cursor in the empty A2 cell, paste the data you copied from the pivot table. Once you have copied and pasted this data, you should delete the Pivot Table tab in the CSV file.



Once you have pasted the copied data in the new Workbook, add the column labels in the top row:

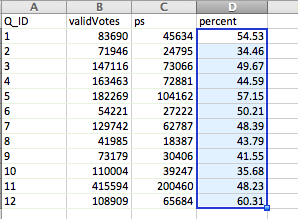
Q\_ID, validVotes, and ps.

In the blank column to the right of the ps column, add a column label: percent



In the first empty cell under the percent column, enter the formula to calculate the percent of the valid vote received by ps in each qark:

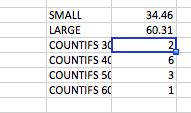
=(C2/B2)\*100

Drag and fill the cells below so that you get the vote percentages for ps in all 12 qarks. Once you have finished, copy the entire column and paste special as “values”

Once you have pasted the formula calculations as values, reduce the number after the decimal point to just two integers (using the .00 -> .0 number formatting tool).

Now you can save the Workbook as a CSV file with a user-friendly name like “psQark2017.csv”Note: many Excel formulas can help you quickly and easily analyze data. Three are particularly useful when doing basic data analysis to will help you think through your plan for data visualization: SMALL, LARGE and COUNTIF.

SMALL will give you the smallest value in an array. If you wanted to know the smallest vote percentage value received by ps in all the qarks, you would use:



=SMALL(D2:D13, 1)

In this case, D2:D13 is the array of values you are considering and the 1 is the value: first – or smallest value. If you were looking for the third smallest value, you would write: =SMALL(D2:D13, 3)

LARGE will give you the largest value in an array. If you wanted to know the largest vote percentage value received by ps in all the qarks, you would use:

=LARGE(D2:D13, 1)

Similar to SMALL, D2:D13 is the array of values and 1 is the largest value.

COUNTIF will get a count of the numbers that meet specified criteria in your array. For example, if you wanted to know how many values are greater than 40, you would enter: =COUNTIF(D2:D13, “>40)

Here we are using the COUNTIFS function, which allows you to set multiple criteria. For example, if you wanted to know how many values are greater than 40 AND less than 50, you would enter:

=COUNTIFS(D2:D13, “>40”, D2:D13, “<50”)

This is a particularly useful function that will help you prepare your visualizations, specifically map visualizations. It is helpful to know the smallest and lowest values as well as the number of values within specified ranges so you can determine how many different color gradients you will need.

While these formulas are not so useful for small datasets, e.g. 12 qarks, they become very useful with larger datasets, e.g. 373 njesia administrative.

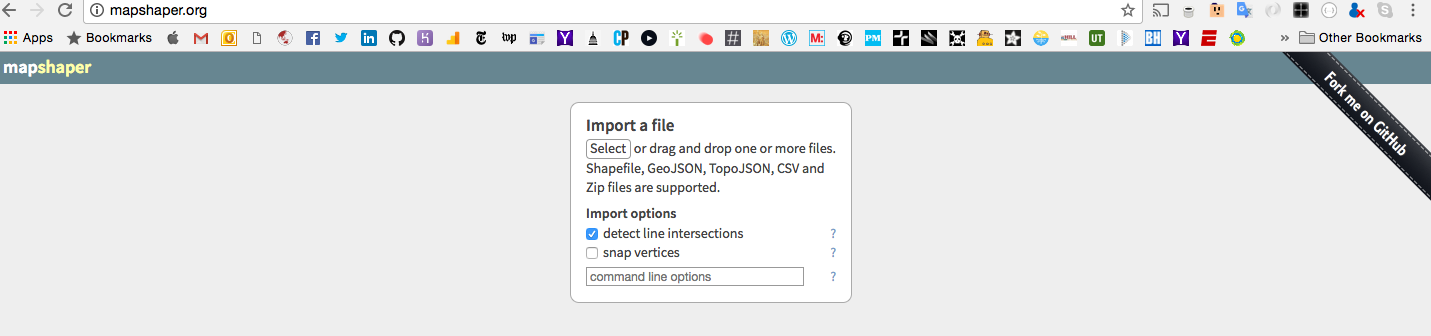
Finally, be sure not to leave these formulas and outputs in your CSV files! These formulas are helpful, but make your dataset “dirty”. Dirty data is difficult to work with in general, but specifically when you are merging CSV and GeoJSON files.

5) Joining CSV files and GeoJSON files with MapShaper

Now that you have a processed CSV dataset with “key” values that correspond to values in a GeoJSON shapefile, you can join the CSV and GeoJSON files together. The simplest way to join CSV data and a GeoJSON file is with a free online tool called Mapshaper: <http://mapshaper.org/>

Mapshaper enables you to upload a GeoJSON file and a CSV file and, with a couple lines of code, join the data together. You can export the new GeoJSON file with the data contained in the CSV file and use it to make map visualizations.

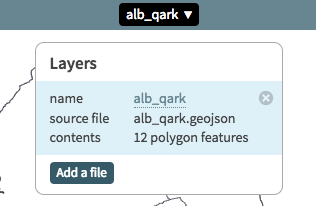
Step-by-step instructions:

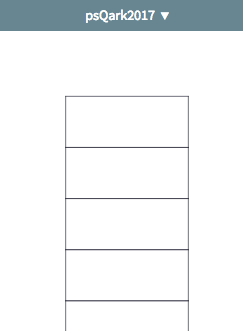


* Go to <http://mapshaper.org/> and upload the alb\_qark.geojson file. An outline of the qark map should be automatically rendered and the name of the file (“alb\_qark”) should appear at the header at the top of the screen.



* Click on the downward-pointing arrow next to “alb\_qark”. In the dialogue box, there should be a button labeled “Add a file” at the bottom.



* Click on the “Add a file” button and upload the CSV file you just created containing the number of votes and vote percentage the Socialist Party received by qark and click the “Import” button (you should see a bunch of empty cell blocks on the screen). *Note: to return to the view the map, click on the downward-pointing arrow next to “alb\_qark” and, when the dialogue box pops up, click on the “alb\_qark” layer*
* In the top right corner of the screen, click on “Console” and a console window should open with a “$” displayed where you can enter commands.



* In the console, enter the following command:

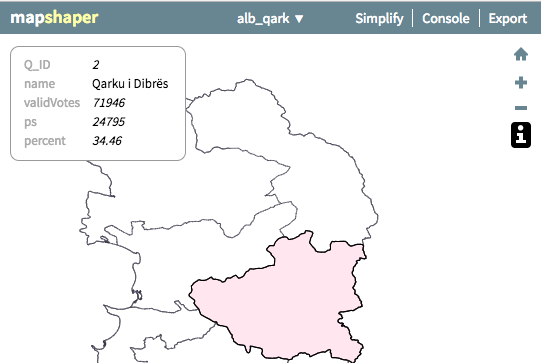
$ join [filename] keys=Q\_ID,Q\_ID

and press enter. If everything is ok, you should see a response:

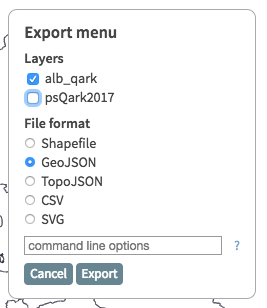
[join] Joined 12 data records

Macintosh HD:Users:mbaldassaro:Desktop:screenshots:Screen Shot 2017-08-14 at 12.01.08 PM.png

* To verify that the join worked correctly, click on the “i” icon on the right side of the screen and roll the cursor over a qark. You should see the data from the CSV file in the teaser that appears on the screen.



* Finally, in the top right corner of the screen, click “Export”. When the dialogue box pops us, under “Layers” make sure that only “alb\_qark” is checked and that the file format is “GeoJSON” then click “Export”

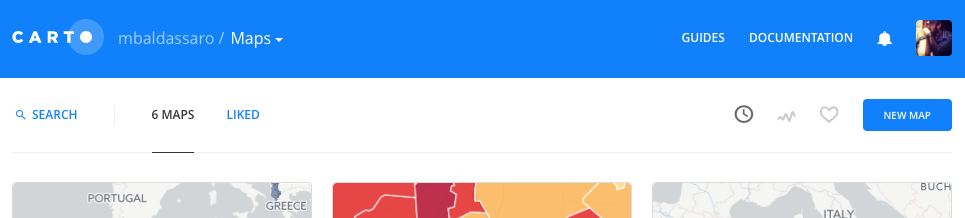


The GeoJSON file you just created will download to your computer. Make sure you rename the GeoJSON file with an appropriate name such as “psQark2017.json” or something similar.

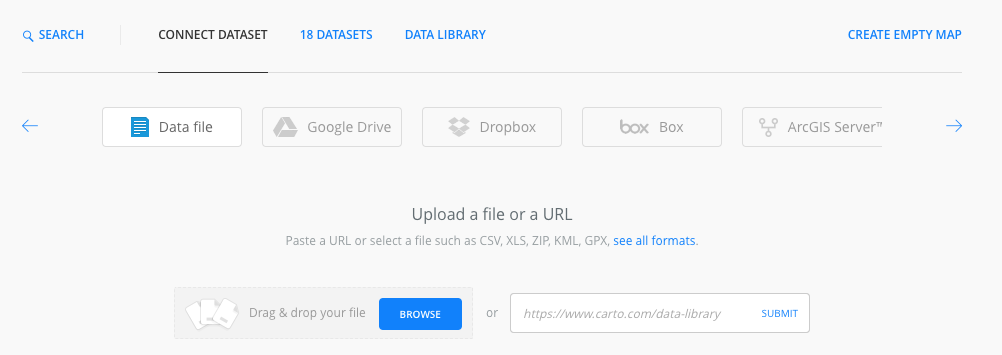
**6) Visualizing results data with Carto**

Now that you have a GeoJSON file with the data you want to visualize, you can create a map that visualizes your results data. The easiest way to create a map is to use a software-as-a-service platform like Carto: <https://carto.com/>. Carto is a web platform that enables you to upload a shapefile (CSV, GeoJSON, etc.) and create custom maps to visualize data. Carto allows you to create, host and share maps online for free (up to 250MB worth of storage space).

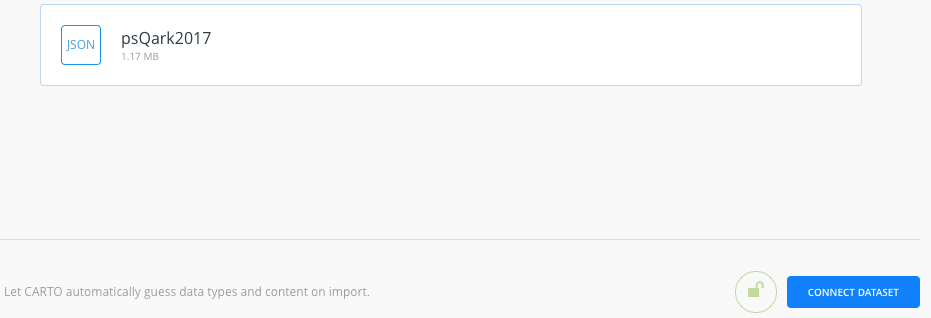
As a first step, go to <https://carto.com/> and sign up for a free account. Once you have created an account, you will be redirected to your dashboard. On your dashboard, click the blue “New Map” button in the top right corner of the screen.



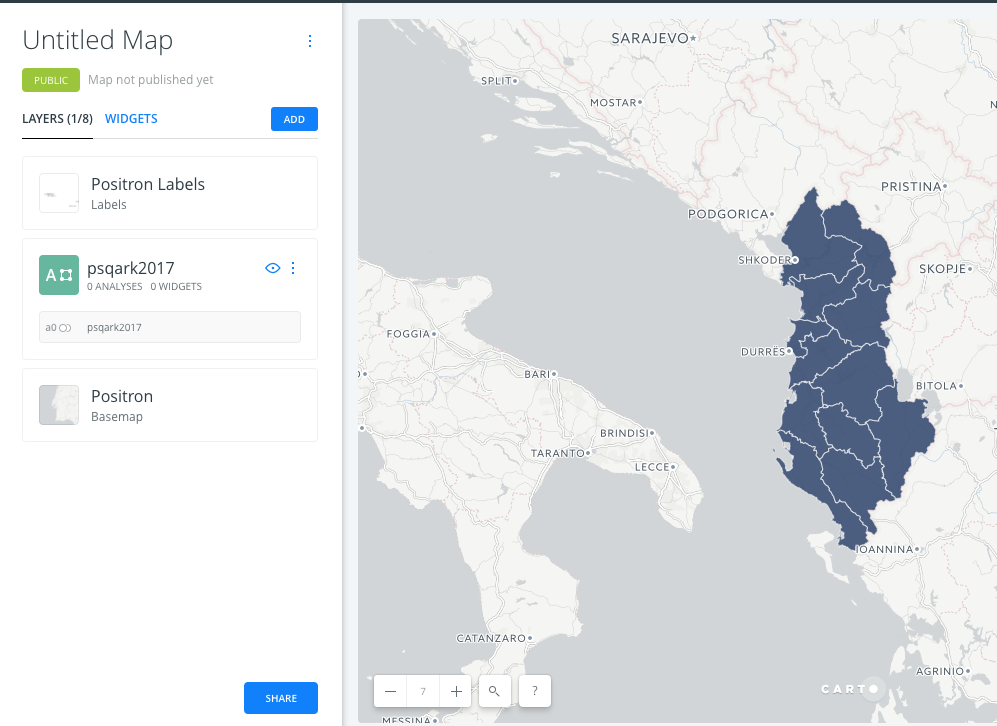
Choose “Connect Dataset”, click the “Browse” button and upload the GeoJSON file you created with Mapshaper.

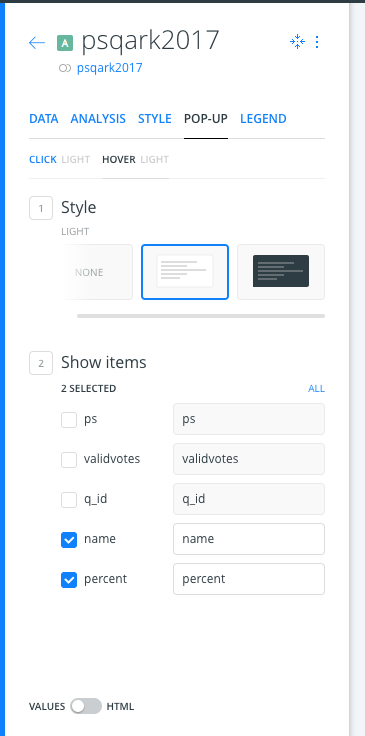
****

Once the file is uploaded, click the “Connect Dataset” button at the bottom.



You will be redirected to the Carto editor where your GeoJSON file will be rendered as a map.





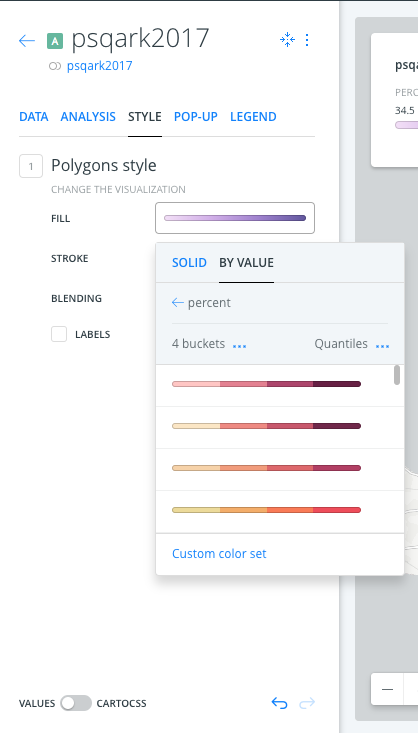
In the style menu, replace “Untitled Map” with an appropriate name (e.g. PS Vote Percentage by Qark 2017) and click on the layer name (here it is called “psqark2017”) to access the style editor.

In the style editor, click “POP-UP” and click on “HOVER”.

Choose the Style of the pop up box you would like to appear (or leave as is).

Check the boxes next to “name” and “percent”.

Now when you hover over, or click on, a qark, you should get the name of the qark and the percent of the vote that the Socialist Party received in the 2017 elections.

****

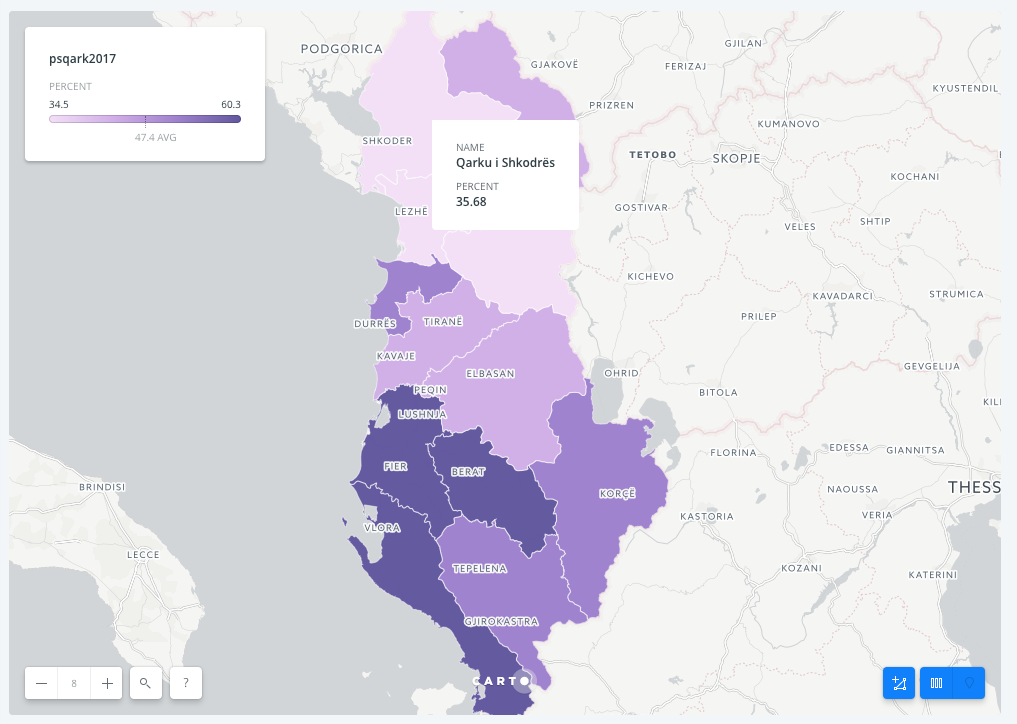
In the style menu, click “STYLE”. Select the FILL bar and a dialogue box should pop up.

In the dialogue box, select “BY VALUE”.

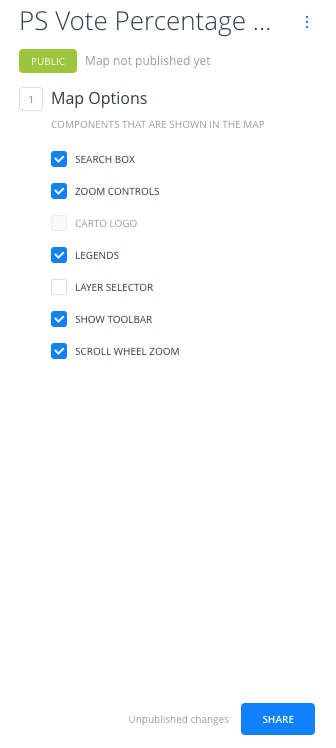
Click on “buckets” and select 4.

Scroll down click and select purple color gradient bar.

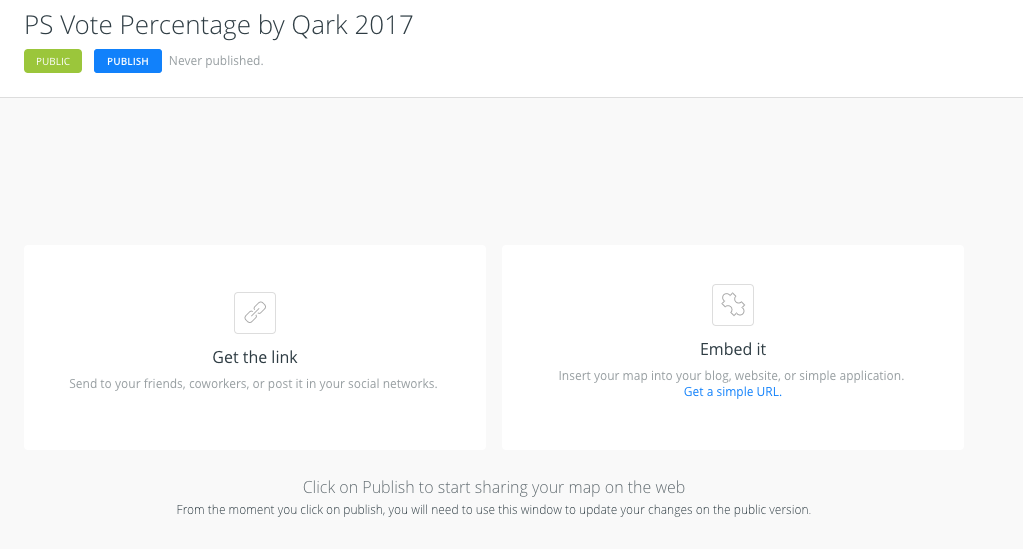
Once you have finished, you should have a map that looks like the one below:

****

**Macintosh HD:Users:mbaldassaro:Desktop:screenshots:Screen Shot 2017-08-14 at 6.00.35 PM.png**Once you have finished styling your map, click on the “map options” icon on the left side of the screen.

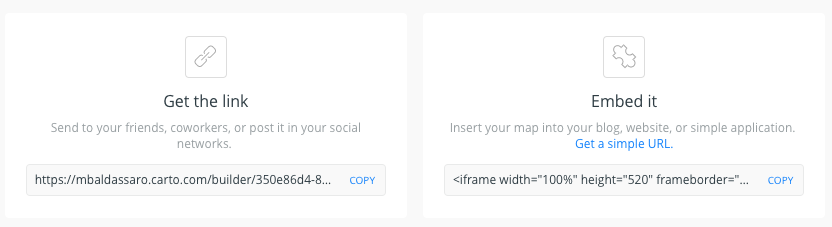
****

At the bottom of the map options menu, click on the blue “Share” button.

****

You will be redirected to a sharing options page. Click on the blue “Publish” button at the top of the page.

Once you have published, you will get a URL link that you can share as well as an embed link that can be used to embed your map in a website!

****