Computer-Assisted Language Comparison in Practice

Tutorials on Computational Approaches to the History and Diversity of Languages

How To Visualize Language Polygons with QGIS (How to Do X in Linguistics 15)

Frederic Blum
Max Planck Institute for Evolutionary Anthropology / University of Passau
Leipzig / Passau

This tutorial shows how to build beautiful maps using polygon data of language distributions and the geospatial software QGIS. For this purpose, I show in the first part how to extract a list of Glottocodes from CLDF datasets and how to use command-line tools to extract the polygon data from Glottography datasets. The second part focuses on the visualization of the extracted data with QGIS. In this part, I also show to include non-linguistic data like archaeological sites and how to prepare the map for printing. The tutorial is fully implemented with free and open software and intends to make map-making accessible for linguists and other researchers.

1 Introduction

A common representation of languages in quantitative studies is through the use of point coordinates in maps and statistical models. Since languages are distributed across areas, this critique certainly has a point. If some random mean of location is chosen for languages which are distributed across vast areas, two languages appear far further away from each other than they actually are. On the other hand, it is unclear how to model such distances, since only speakers at borders or overlapping areas are actually in contact, and the speakers in the mean locations or the far extremes are not. An example of such a discussion can be observed in the reviews and answers to the reviewers in (Guzmán Naranjo & Jäger 2024). Regarding the presentation of languages in maps, the answer to the critique is often “but we don’t have any other data”, especially in large quantitative studies with a worldwide sample.

Computer-Assisted Language Comparison in Practice
Volume 9, Number 1
URL: https://calc.hypotheses.org/9261 DOI: 10.15475/calcip.2026.1.4

Published on 04/05/2026

Alas, after the publications of multiple polygon datasets of the Glottography project (Ranacher et al. 2025, 2026), this is no longer true (if it ever was). The project published multiple datasets of existing publications like the “World Atlas of Languages” (Asher & Moseley 2018). In contrast to other collections of language distributions like Ethnologue, the Glottography data comes as open-access repositories and is not behind a paywall. In its current form, the project already covers more than 5,300 languages from 29 different publications and is likely to increase in coverage in the future. The presentation of the data using the Cross-Linguistic Data Formats (CLDF, Forkel et al. 2018) supports good integration with other existing datasets in CLDF like Lexibank (Blum et al. 2025) or Grambank (Skirgård et al. 2023). It is part of this tutorial to show how this integration can be leveraged through the command-line interface and SQLite queries.

The first question I asked myself as someone who has a map in almost all his articles was: “How can I make nice maps with such polygons”? While there are some online-tools which provide quick maps, I was not satisfied by this solution because of the lack of control, difficulties in reproduction, and unclear long-term reliability that these tools provide. For comparative purposes, I will briefly present one of those tools later on. While those online tools offer quick solutions and are easily accessible, QGIS offers a broad variety of customization for maps that provide complex representations of multiple data types. Especially when it comes to important publications, such maps can make the difference in how (re)viewers perceive the quality of the work and provide a more realistic representation of the distribution of languages across the world. This contribution thus seeks to provide a short tutorial on (a) how to extract the data for a set of languages, (b) to import the data to different visualization tools, and (c) how to make a beautiful map of language distributions using QGIS.

2 Software Requirements

We will be using QGIS (v4.0.1, Dawson et al. 2026), which is an open-source software for spatial visualization (https://qgis.org/download/). It is a very powerful tool, of which we will only make use of a small amount of its features. You can install the application from the link above on all common operating systems.

The tutorial requires a basic setup that involves a functional installation of Python 3 (>3.10) and GIT (https://git-scm.com/). With Linux and MacOS systems, this is straightforward and will not be covered here. For the usage with Windows, I recommend to have a look at the corresponding tutorial (Snee 2024). We also recommend the installation of SQLite (https://sqlite.org/).

For the initial data processing, we will be running Python from the command line. Here, we recommend using a virtual environment. On MacOS, you can set up one such environment with the following commands:

```
# Create an environment within the folder venv/
python3 -m venv venv/your-virtual-environment-1
# Activate the environment
source venv/your-virtual-environment-1/bin/activate
```

Now, we will be installing a couple of packages that we need to run the code using the Python package managing tool `pip`:

```
pip install pycldf
pip install csvkit
pip install cldfgeojson
```

We will also use multiple datasets that are curated on GitHub, which you can clone directly via `git` on the command-line:

```
git clone https://github.com/Glottography/asher2007world
git clone https://github.com/Glottography/queixalos2000linguas
git clone https://github.com/pano-takanan-history/blumpanotacana
```

All the code code and data that are used for this tutorial are curated on Codeberg, which you can also clone in the same way:

```
git clone https://codeberg.org/calc/tutorial-visualizing-polygons
```

3. Extracting the Polygon Data

In this part, I will briefly explain how to extract the polygons for (1) languages from existing CLDF datasets and (2) how to extract a set of Glottocodes directly from Glottolog, either through an explicit list or based on language-families. Both tasks are fundamentally the same: We extract a list of Glottocodes from a dataset and then extract the corresponding polygons from Glottography repositories.

3.1 Extracting Polygons from Wordlist Collections

For retrieving the polygon data from Glottography datasets such as the “Atlas of the World’s Languages” (https://github.com/Glottography/asher2007world, Asher & Moseley 2018), we need a CSV-file with a column for Glottocodes. Such a file

representing the LanguageTable of CLDF is standard in all datasets of Lexibank and I will exemplify this with the `blumpanotacana` dataset (Blum et al. 2024a) which we have cloned above. The command to extract the polygon data proceeds in four steps: (1) Identify the Glottocodes in the input file, (2) Remove the header, (3) Extract the polygons of a Glottography dataset based on the provided set of Glottocodes, and (4) Save the file in `.geojson` format.

```
csvcut -c Glottocode blumpanotacana/cldf/languages.csv | \
csvformat -E | \
cldfbench geojson.geojson --no-glottolog queixalos2000linguas/cldf - \
> data/bpt_polygons.geojson
```

One quick way of visualizing the output is to upload the content of this file to https://geojson.io/next/, which provides a basic map using an OpenStreetMap background and some basic customization options. This is great for a quick view on the extracted data, but not sufficient for most publications. I show an example of this in Figure 1.

3.2 Querying Glottolog for a List of Glottocodes

Another option is to extract Glottocodes for specific languages or language families directly from Glottolog using an SQLite query. For example, I was approached by a researcher who approached me about a map of Arawak, Carib, and Tupi languages across the Amazon. Since the point coordinates hide much of the true extension of the families, I recommended going for polygons instead, making use of the newly published datasets.

To extract data from Glottolog (v5.3, Hammarström et al. 2026), we clone the data and create an SQLite database. This step presupposes you have `sqlite3` installed to be used on the command line. The usage of SQLite queries with large datasets like Glottolog or Lexibank has proven worthwhile because it is a lot quicker than iterating through a CSV or TSV file. It also allows performing data preprocessing tasks during the data extraction (Blum et al. 2024b). Through `pycldf`, we first create an SQLite database of Glottolog and then open it via `sqlite3` to run the query.

```
# Clone dataset
git clone https://github.com/glottolog/glottolog-cldf

# Create SQLite database
cldf createdb glottolog-cldf/cldf/cldf-metadata.json glottolog.sqlite3

# Open database
sqlite3 glottolog.sqlite3
```

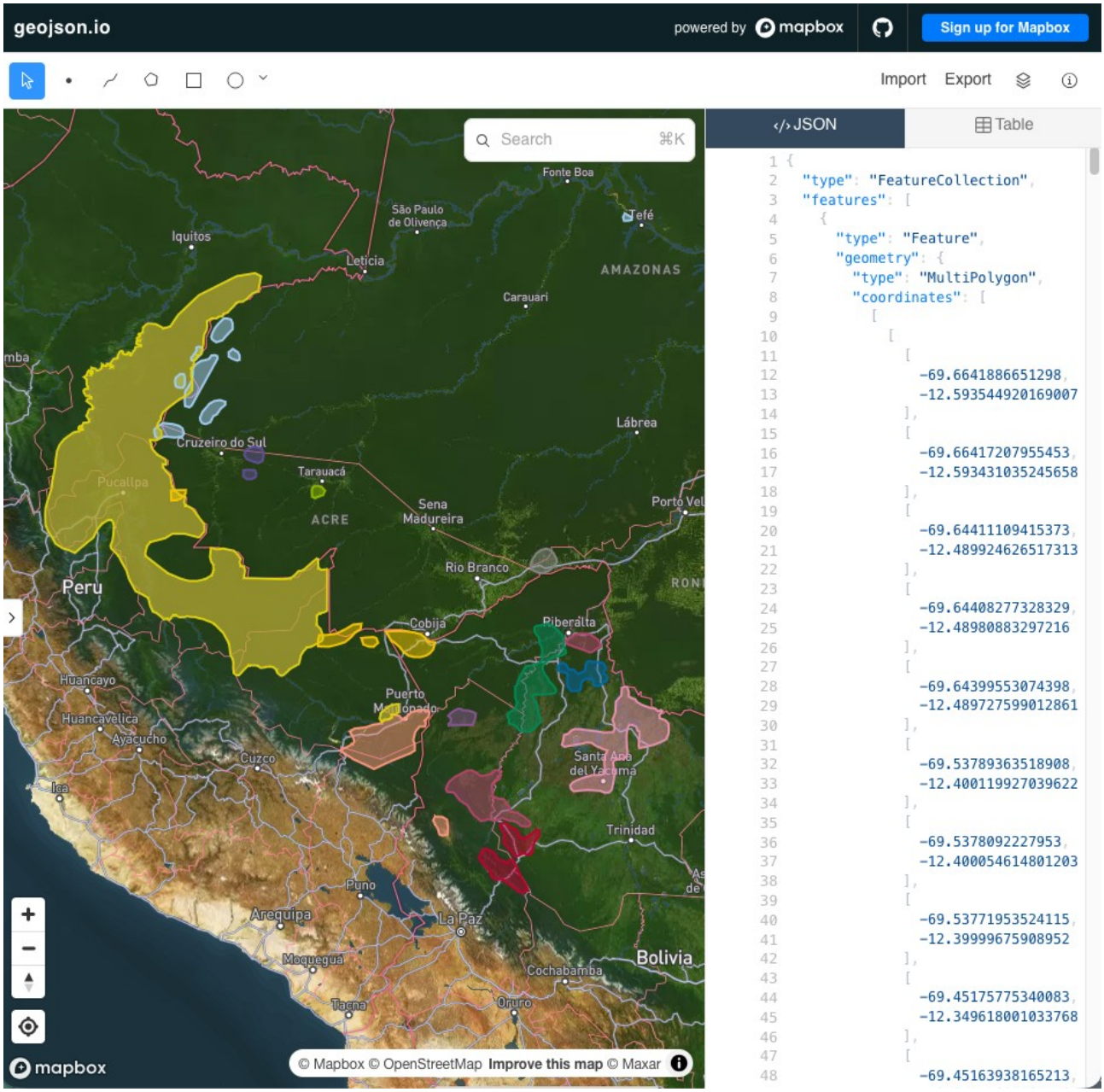

Figure 1: Map of languages in the *blumpanotacana* dataset uploaded to the website for easy visualization. The visualization was automatically created by drag&drop of a .geojson file.

You should be prompted within a `sqlite3` console now:

```
INFO      <cldf:v1.0:StructureDataset at glottolog-cldf/cldf> loaded in
glottolog.sqlite3
SQLite version 3.51.0 2025-06-12 13:14:41
Enter ".help" for usage hints.
Sqlite>
```

In the next step, we build a short SQLite query to retrieve the languages of our choice from Glottolog. In this case, we want to include language families from the Arawak, Carib, and Tupi language families. For the purpose of illustration, I have also added two language isolates to show how we can add individual glottocodes from languages to the query. Within the sqlite3 console, we first choose settings so that the results of the query are stored as CSV with headers.

```
.mode csv
.headers on
.output data/glottocodes.csv
```

Then, we are ready to run a query to extract a CSV-file from three different languages families by providing the family-level glottocode. This extracts all Glottocodes of languages that belong to the respective family. Alternatively, we can also select language-level Glottocodes. You can also extract other columns of tables that are part of Glottolog, like language family, point coordinates, or others, that are then also stored to the output CSV-file. This saving of the list is done in the last line of the console prompt.

```
SELECT
    l.cldf_id AS Glottocode,
    -- Example for additional columns
    l.Family_ID
FROM
    LanguageTable AS l
WHERE
    -- List of language-family glottocodes
    l.Family_ID IN ('araw1281', 'cari1283', 'tupi1275')
        OR
    -- List of language-level glottocodes
    l.cldf_glottocode IN ('taus1253', 'muni1258')
;

.output stdout
```

After extracting our list of desired Glottocodes, we can use this CSV (more than 400 entries!) to query the Glottography repository to extract all available polygon data for the languages in the list. This time, we will be using the “As línguas amazônicas hoje” dataset (Queixalos & Renault-Lescure 2000), which is focused on languages in the Amazon. The command is similar to the one from above, only using different input, output, and geojson file names.

```
csvcut -c Glottocode polygons.csv | \
csvformat -E | \
cldfbench geojson.geojson --no-glottolog asher2007world/cldf/traditional -
\
> data/amazon_polygons.geojson
```

If we only want data from a specific language family, we can also use a prompt that is based on `csvkit`, which we have installed previously. This is a bit slower than the SQLite query, but if we are only interested in a single family, this can be a reasonable alternative.

```
csvgrep -c Parameter_ID -r "^classification$" \
glottolog-cldf/cldf/values.csv | \
csvgrep -c Value -m cari1283 | \
csvcut -c Language_ID | \
csvformat -E | \
cldfbench geojson.geojson --no-glottolog asher2007world/cldf/traditional -
\
> data/carib_polygons.geojson
```

Until now, I have shown different ways of extracting geojson data from Glottography projects, using either data from a single CLDF dataset, or extracting data for language families directly from Glottolog. Now we are ready to turn to the visualization of the extracted data.

4. Using GIS for Visualizing Language Polygons

4.1 Setting up QGIS

We will be using GIS for the map making, which stands for “geographic information system”. This acronym refers to systems that allow the editing of geospatial data. For this tutorial, I have used the OpenSource application QGIS (v4.0.1). Some configurations might look slightly different in older versions, but the basic functionality is the same. When opening the application, you can choose between a blank template or an OpenSteetMap Basemap. We choose the latter. You can also add map data from other resources (e.g. https://www.tracestrack.com), but this will not be covered as part of the tutorial.

4.2 Loading the Data

Within a QGIS project, you have several toolbars. We will only use a small amount of functions for the moment. I will visualize the basic tools for our purpose using the

`amazon_polygons.geojson` file with the data from Arawak, Carib, and Tupi languages. In the very first step, you can simply drag&drop the geojson-file of your choice to QGIS. You should now already see the distributions of those languages on the map after the drag&drop operation.

4.3 Basic Visualization Tools

The first customization is to colour the distributions according to language family. To do this, we double-click the layer-name in the bottom left pane to open the layer properties. Here, we select “Symbology” to change the representation of each polygon. At the top, currently “Single Symbol” is chosen by default. We change this to “Categorized” and select “family” as “Value”. A click on “Classify” at the bottom created a random colour ramp based on the different values on the “family” variable on the input data. This setting changes the fill-colour of all polygons. The three steps are shown in Figure 2.

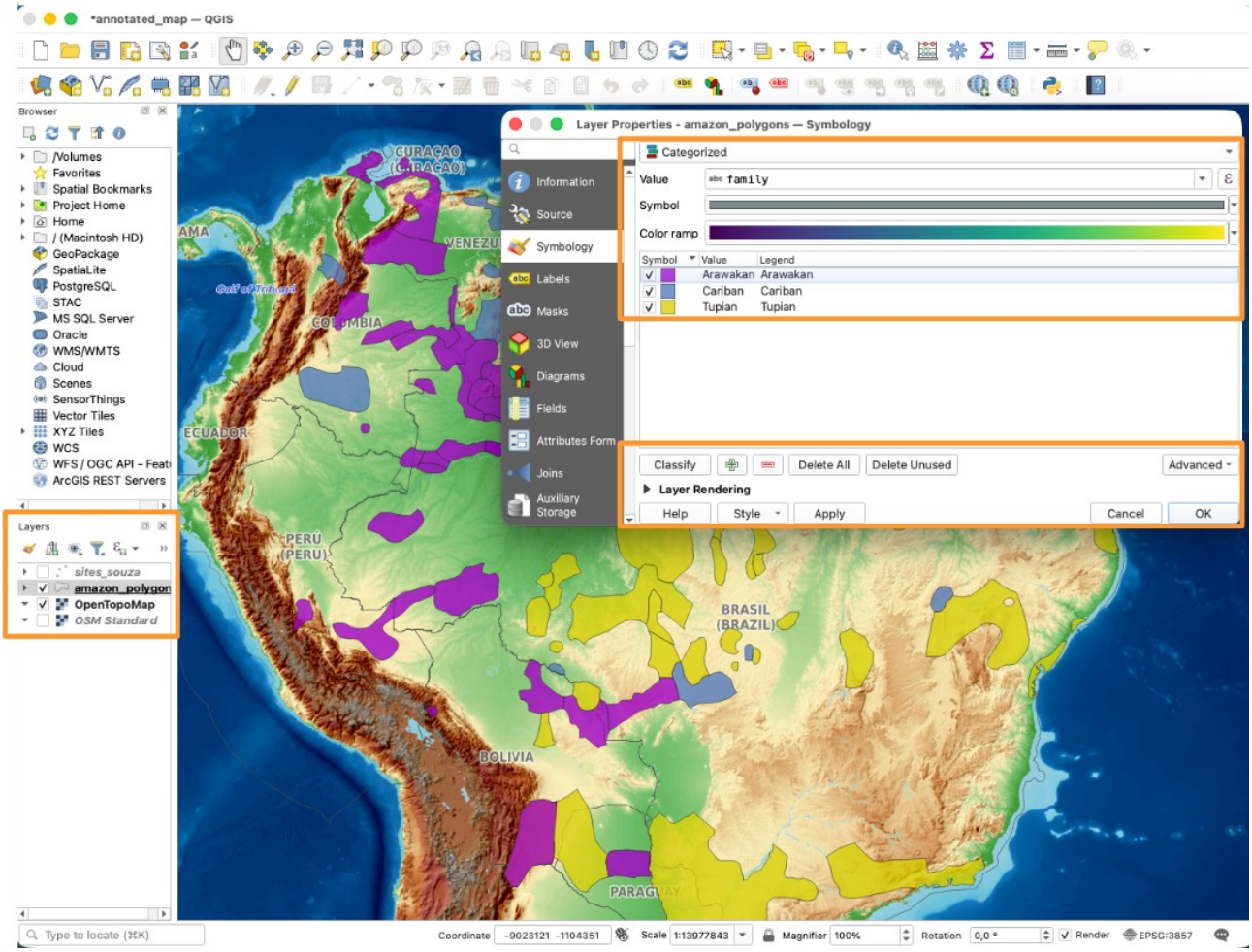

Figure 2: Distribution of Arawak, Carib, and Tupi languages across the Amazon as visualized in QGIS. The application interface shows how the fill-colour for each language family can be customized.

4.4 Bonus: Adding Archaeological Sites in QGIS

QGIS begins to excel once we combine different data types. For example, we can add data for archaeological sites that come in TSV or CSV-files. For this purpose, I have created a CSV-file from the archaeological sites published in the article “Archaeological expansions in tropical South America during the late Holocene: Assessing the role of demic diffusion” (Gregorio de Souza, Alcaina Mateos & Marco 2020) which originally came in ODS-Format (link). The CSV-file could be made part of the tutorial repository, since the data is thankfully published under a CC-license.

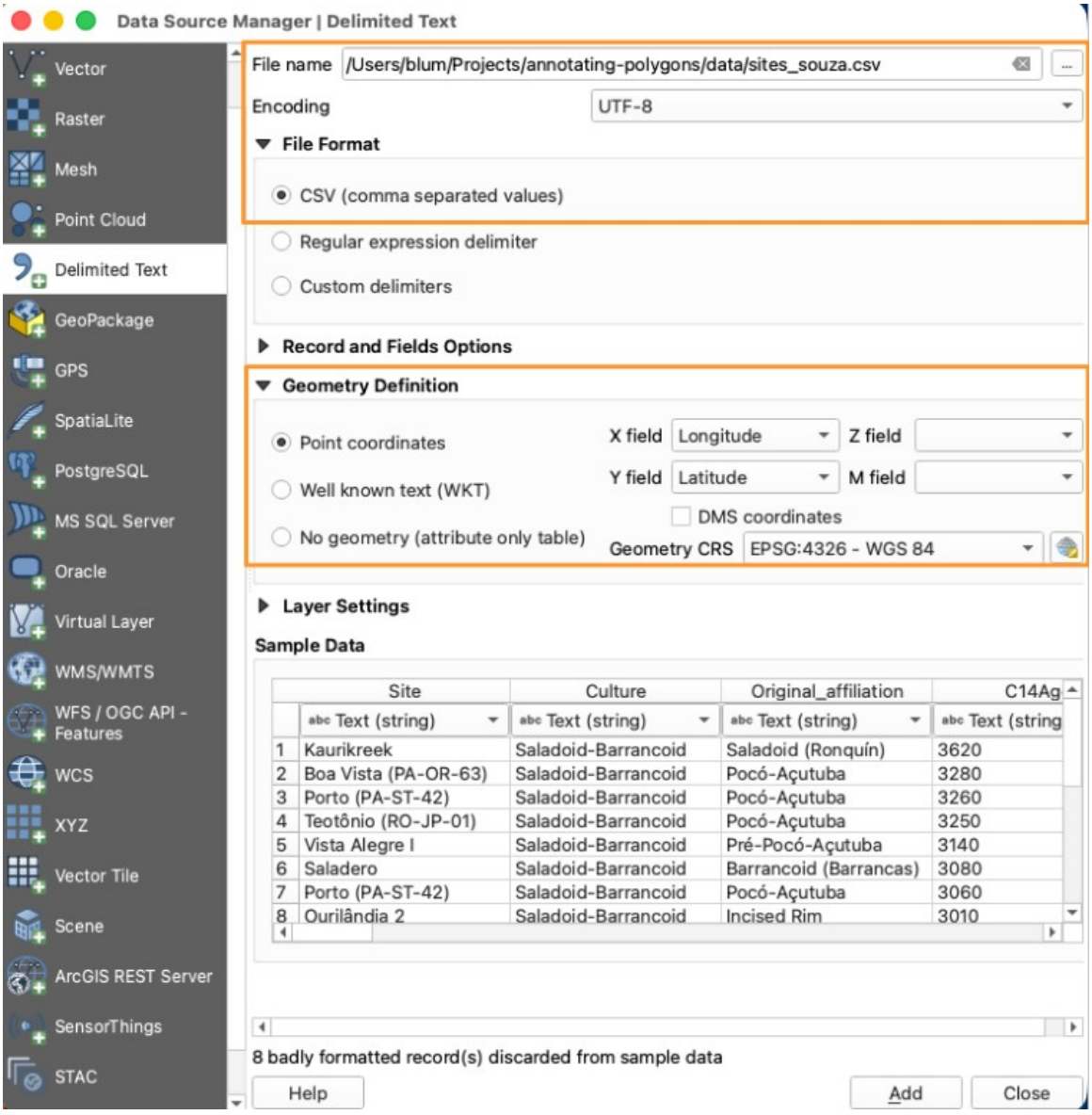

Figure 3: Importing CSV data within QGIS. It is important to set the Geometry Definition for coordinates and the CRS (coordinate reference system).

To include this file, add a new layer in QGIS and import this file by selecting “Layer > Add Layer > Add Delimited Text Layer”. Here, you browse through your file-system to choose the input file and select the correct encoding. In the next step, you have to select the Longitude (X) and Latitude (Y) columns and choose a Geometry system (CRS). Now, you have a second layer with the archaeological sites and can again choose a colouring system of your choice, as well as other customizations like opacity, size, and stroke for a symbol of your choice. To give just one example, you could use different SVG markers for sites related to different archaeological traditions. It is also possible to import custom SVG’s. The import of the CSV file is illustrated in Figure 3.

4.5 Print Layout

The final task is a print layout. Doing this for a first time, it seemed a bit unintuitive and unnecessarily complex to me. But soon I realized that QGIS is worth it, since it actually offers you a lot of layout options that make it easy to include a map cutout and a legend to the final version of your map. Also, the richness of customization options opens up many possibilities of building high-quality maps. To begin with this process, either select “Cmd + P”, or select “Project > New Print Layout” from the main toolbar. This opens a second window, which will be your print layout.

First, we select “Add Map” from the symbols at the left and drag across the whole layout page to include our main map. The map gets added exactly as you see it in your map interface. Second, we select “Add Legend”, and again drag the area in the layout where we want to put the legend. The legend automatically accesses our selected symbology (shape, colour) for the respective layers of language families and archaeological sites. Now, we already have map and legend ready for print. Great!

Towards the right, you find a lot of customization options for all added items. For the legend, we want to change the names of the layers to something more insightful. For this, we first change to “Manual” under “Legend Items” and select new names for each legend to “Archaeological Traditions” and “Language Families”. You can also customize background, frames, and all other kinds of data that is shown. This allows you to change the text and icons of all the symbology visualized in the legend.

In a final step, we want to add a bigger map as a cutout in a top corner. For this, we first need to select the Map 1 from the items panel and lock the layers, so that they don’t get changed when we change the original map. Then, we switch back to said main map, deactivate the language and archaeology layers by unticking the box before their name and zoom out to the desired view of the world. We can now return to the Print Layout and add a second map. Again we can select the area where we want to include the map,

for which I chose the top right corner. Through selecting “Add Shape” in the left item panel we can add a rectangle that indicates the area of our map we have visualized. Simply drag the rectangle above your little world map and move it to the corresponding area. There is probably also some automated way of referencing the extent of the cut-out map of which I am not currently aware. But considering the offered complexity of QGIS, I’d be very surprised if this was not the case.

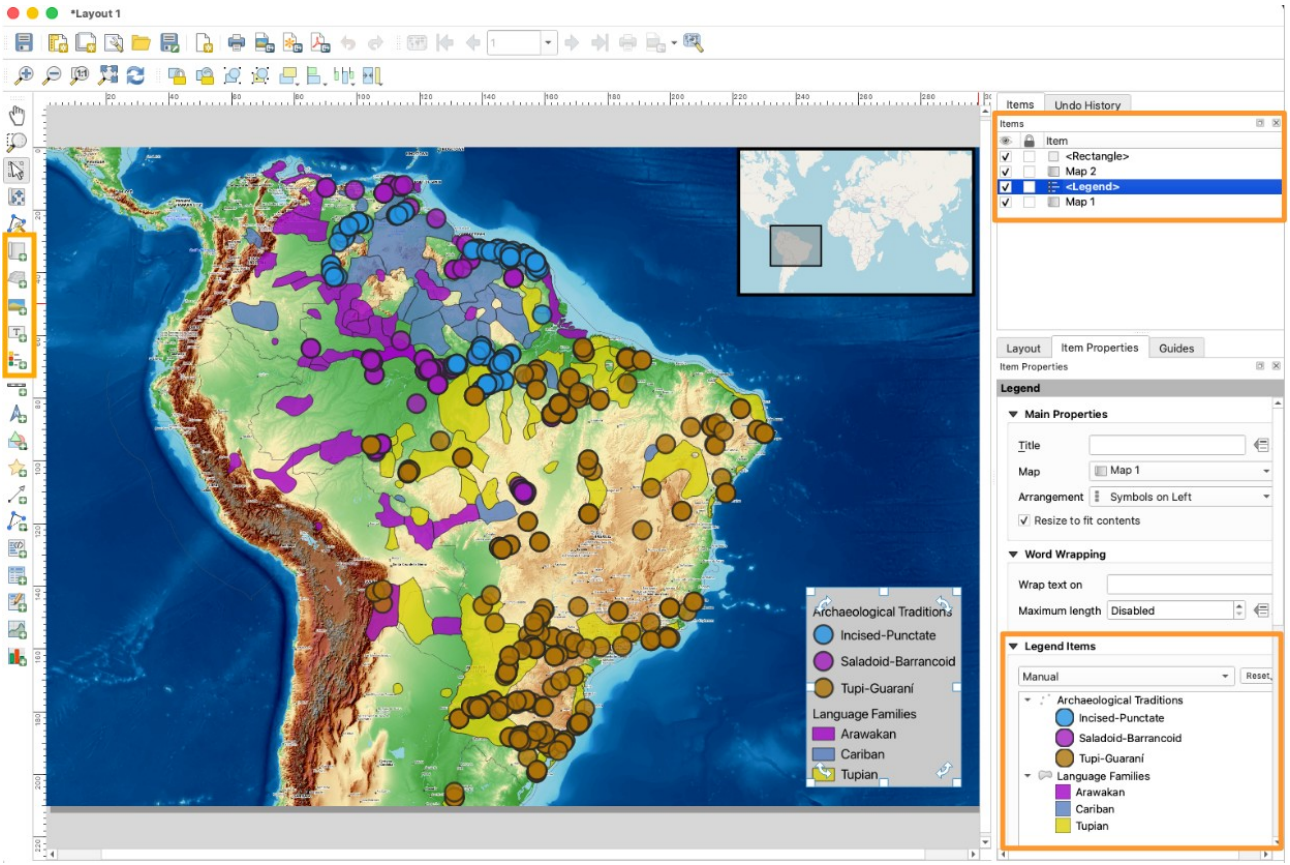

Figure 4: Final print layout including the main map, a legend, and a world-map indicating the extent of the selected map.

The final map now has the original language distributions, the archaeological sites, a legend, and a world-map in the top corner. Through “Cmd + P” or by clicking the “Export PDF” symbol in the toolbar, you can now export a PDF file of your print layout for dissemination and publishing. The options for the print layout are shown in Figure 4.

Such visualizations can help us to build hypotheses about historical expansions of language and culture, putting linguistic data into spatial context. In the case of the three Amazonian language families, we see interesting overlaps of the distribution of language families and different archaeological traditions. These links have been proposed in the archaeological literature and often rely on such spatial overlays of

linguistic and archaeological distributions (Iriarte 2024). Other possibilities involve topographic maps, which put more emphasis into geological information such as mountains and rivers and could be used to reference possible migration or contact routes. To understand the spatial element behind such distributions is the necessary first step for a more sophisticated modeling of space.

5 Conclusion

This tutorial provided an introduction to annotating language distributions with QGIS and showed how to combine different kinds of input data. The tool-set offered by QGIS seems complex at first, but offers a wide variety of customization options that make it possible to create high-quality maps. The visualization of languages as polygons provides a more realistic account of the distribution of language families and can lead to a better understanding of contact, expansion, and diversification. The polygon data is also the necessary first step for incorporating language distributions in spatial models for computational analysis, potentially linking up the analysis with other kinds of geospatial data like rivers or mountains. Again, QGIS can be a valuable tool for the visualization of such information.

References

Asher, R. E. & Christopher Moseley. 2018. *Atlas of the world's languages*. 2nd edn. London: Routledge. https://doi.org/10.4324/9781315829845.

Blum, Frederic, Carlos Barrientos, Johannes Englisch, Robert Forkel, Simon J. Greenhill, Christoph Rzymski & Johann-Mattis List. 2025. Lexibank 2: Pre-computed features for large-scale lexical data. *Open Research Europe* 5(126). 1–27. https://doi.org/10.12688/openreseurope.20216.2.

Blum, Frederic, Carlos Barrientos, Roberto Zariquiey & Johann-Mattis List. 2024a. A comparative wordlist for investigating distant relations among languages in Lowland South America. *Scientific Data* 11(1). https://doi.org/10.1038/s41597-024-02928-7.

Blum, Frederic, Ludger Paschen, Robert Forkel, Susanne Fuchs & Frank Seifart. 2024b. Consonant lengthening marks the beginning of words across a diverse sample of languages. *Nature Human Behaviour* 8. 2127–2138. https://doi.org/10.1038/s41562-024-01988-4.

Nyall Dawson, Jürgen Fischer, Matthias Kuhn, Alessandro Pasotti, Denis Rouzaud, mhugent, Alexander Bruy, et al. 2026. qgis/QGIS: 4.0.1. Zenodo. https://doi.org/10.5281/zenodo.19401447.

Forkel, Robert, Johann-Mattis List, Simon J. Greenhill, Christoph Rzymski, Sebastian Bank, Michael Cysouw, Harald Hammarström, Martin Haspelmath, Gereon A. Kaiping & Russell D. Gray. 2018. Cross-Linguistic Data Formats, advancing data sharing and re-use in comparative linguistics. *Scientific Data* 5(1). 1–10. https://doi.org/10.1038/sdata.2018.205.

Gregorio de Souza, Jonas, Jonas Alcaina Mateos & Madella Marco. 2020. Archaeological expansions in tropical South America during the late Holocene: Assessing the role of demic diffusion. *PLOS ONE* 15(4). 1–32. https://doi.org/10.1371/journal.pone.0232367.

Guzmán Naranjo, Matías & Gerhard Jäger. 2024. Euclide, the crow, the wolf and the pedestrian: Distance metrics for linguistic typology. *Open Research Europe* 3(104). 1–32. https://doi.org/10.12688/openreseurope.16141.2.

Hammarström, Harald, Robert Forkel, Martin Haspelmath & Sebastian Bank. 2026. *Glottolog database (v5.3)*. Leipzig: Max Planck Institute for Evolutionary Anthropology. https://doi.org/10.5281/zenodo.18840935.

Iriarte, José. 2024. *The archaeology of Amazonia: A human history*. London: Bloomsbury Academic. https://doi.org/10.5040/9781350270770.

Queixalos, Francisco & Odile Renault-Lescure. 2000. *As línguas amazônicas hoje*. São Paulo: IRD/Instituto Socioambiental/MPEG. http://www.cartographie.ird.fr/linguas2.html.

Ranacher, Peter, Robert Forkel, Nour Efrat-Kowalsky, Matthias Urban, Antonia Hehli, Micha Franz, Gregory Biland, et al. 2025. A global and interoperable dataset of linguistic distributions derived from the Atlas of the World's Languages. *Scientific Data* 12(1466). 1–10. https://doi.org/10.1038/s41597-025-05828-6.

Ranacher, Peter, Robert Forkel, Nour Efrat-Kowalsky, Matthias Urban, Antonia Hehli, Micha Franz, Gregory Biland, et al. 2026. Glottography: An Open-Source Geolinguistic Data Platform for Mapping the World's Languages. Journal of Open Humanities Data 12(47). 1–16. https://doi.org/10.5334/johd.459.

Skirgård, Hedvig, Hannah J. Haynie, Damián E. Blasi, Harald Hammarström, Jeremy Collins, Jay J. Latarche, Jakob Lesage, et al. 2023. Grambank reveals the importance of genealogical constraints on linguistic diversity and highlights the impact of language loss. *Science Advances* 9(16). 1–15. https://doi.org/10.1126/sciadv.adg6175.

Snee, David. 2024. Using CLDFBench and PyLexibank on Windows. University of Passau 7(2). 103–109. https://doi.org/10.15475/CALCIP.2024.2.6.

Acknowledgements
Special thanks to Robert Forkel, who provided much of the code of the command-line applications and to Antonia Lieschke, who answered all my QGIS questions.
Funding Information
This project has received funding from the Max Planck Society. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.