

GeoJSON Primer

Format Overview

Topic	Description
File Extension	.geojson
MIME Type	Application, Vendor Tree - application/vnd.geo+json
Structure	Javascript Object Notation
Standard	RFC 7946
Primary fields or areas of use	Geographical Information Systems (GIS), web mapping applications, mobile applications, web APIs, JavaScript web-mapping libraries, JSON-based document databases
Source and affiliation	Internet Engineering Task Force (IETF)
Metadata standards	None specific to GeoJSON
Key questions for curation review	<ul style="list-style-type: none"> • Are coordinates listed in the following format: <i>[longitude, latitude, elevation]</i> • Can the file be opened in a text editor and viewed in QGIS • Does the file pass validation • Is there associated metadata/README.md files
Tools for curation review	QGIS; geojsonlint.com ; geojson.io ; ArcCatalog
Date Created	July 2019
Created by	Nadia Dixson (City Archivist for the City of Somerville) Genevieve Milliken (NYU, Digital Library Technology Services) Keshav Mukunda (SFU Library, Data Services) Reina Murray (Johns Hopkins University, Sheridan Libraries) Rachel Starry (University at Buffalo Libraries)

Suggested Citation: Dixson, Nadia; Milliken, Genevieve; Mukunda, Keshav; Murray, Reina; Starry, Rachel. (2019). GeoJSON Data Curation Primer. Retrieved from the University of Minnesota Digital Conservancy. <http://hdl.handle.net/11299/210208>.

This work was created as part of the “Specialized Data Curation” Workshop #2 held at Johns Hopkins University in Baltimore, MD on April 17-18, 2019. These workshops have been generously funded by the Institute of Museum and Library Services # RE-85-18-0040-18.

Table of Contents

[Description of Format: What is GeoJSON?](#)

[Syntax for GeoJSON](#)

[Existing GeoJSON Documentation](#)

[Example GeoJSON Datasets](#)

[Software for Opening and Viewing GeoJSON Files](#)

[Validating GeoJSON Datasets](#)

[GeoJSON Metadata Standards and README Requirements](#)

[GeoJSON Metadata Standards](#)

[Viewing Metadata in QGIS](#)

[What to Include in a README File](#)

[Preservation Strategy and Recommendations for Transformations](#)

[Shapefile](#)

[GeoPackage](#)

[Conversion Tools](#)

[Additional options for converting GeoJSON](#)

[Checking Transformations](#)

[Creating Metadata using ArcCatalog](#)

[Creating Metadata using mdEditor](#)

[Evaluating GeoJSON Files for FAIRness](#)

[What are the FAIR principles?](#)

[FAIR GeoJSON: Components Evaluated](#)

[Assessing GeoJSON Data for FAIRness](#)

[Findable](#)

[Accessible](#)

[Interoperable](#)

[Reusable](#)

[Appendix A: Example CURATED Checklist for GeoJSON](#)

[CURATED Checklist for GeoJSON](#)

[C - check files and read documentation](#)

[U - understand \(or try to\) the data](#)

[R - request missing information](#)

[A - augment metadata and T - transform file formats for reuse](#)

[E - evaluate for fairness](#)

[D - document all curation activities](#)

[Appendix B: Researcher Survey Results](#)

[Survey Questions](#)

[Survey Results](#)

[Bibliography](#)

Description of Format: What is GeoJSON?

GeoJSON is a geospatial data interchange format for encoding vector geographical data structures, such as point, line, and polygon geometries, as well as their non-spatial attributes. As the name implies, it is based on JavaScript Object Notation (JSON) and continues JSON's lightweight, plaintext, and machine-readable format, making it a versatile file type especially for web-based mapping and applications. The GeoJSON format defines several types of JSON objects to represent the structure and layout of geographic features, their properties, and their area. GeoJSON supports a single geographic coordinate reference system, the World Geodetic System 1984 (WGS84), in units of decimal degrees typically up to 7 decimal places of precision.

In GeoJSON, there are seven case-sensitive “geometry types” as defined in the OpenGIS Simple Features Implementation Specification for SQL: “Point”, “MultiPoint”, “LineString”, “MultiLineString”, “Polygon”, “MultiPolygon”, and “GeometryCollection”. These simple geometries consist of a “type” and a collection of coordinates listed as *longitude*, *latitude*, *elevation*, with elevation being optional. The seven geometry types above, together with the case-sensitive types “Feature” and “FeatureCollection”, constitute the nine “GeoJSON types”. The “Feature” and FeatureCollection” types are geometries together with their descriptive properties.

There are several advantages to GeoJSON that make it a good option for encoding geographical data. GeoJSON is text-based, editable in a text editor, uses common English words, utilizes a very simple data structure, and is easy for both humans and machines to read. Moreover, GeoJSON is a single file, unlike compressed geospatial data formats. As a result of its simplicity and versatility, it is an ideal format both for consuming and producing geospatial data. Some alternatives to GeoJSON that may be encountered include Shapefile, KML/KMZ, GML, and CSV.

Curators may also encounter the EsriJSON file format. This is a proprietary, ESRI standard for encoding spatial data in JSON structured text. It is set as the default conversion option when a spatial dataset is transformed from another vector format to JSON using the software ArcMap and ArcGIS Pro. While the format and syntax are largely similar to GeoJSON, there are some key differences, particularly in the “type” member. An explanation of how to identify an EsriJSON file is provided in the section “Recommendations for File Transformation.” If an EsriJSON file is submitted for curation, consider asking the submitter for the dataset in a different format.

Syntax for GeoJSON

Every GeoJSON object contains a member named “type”, which must be one of the nine GeoJSON types previously mentioned. Almost all of the geometry objects also contain a member named “coordinates” listed in order as longitude, latitude, and (an optional) elevation. Elevation is expressed as the height in meters above or below the WGS84 reference ellipsoid used to approximate the earth. Here are the seven simple geometry objects that can be specified in GeoJSON:

- **Point:** For type “Point”, the coordinates member is a single position. For example, a position with longitude -122.918958, latitude 49.279778, and elevation 340 in WGS84 is specified as:

```
{
  "type": "Point",
  "coordinates": [-122.918958, 49.279778, 340]
}
```

<figure> <figcaption></figcaption> </figure>

- **LineString:** For type "LineString", the coordinates member is an array of two or more positions. This geometry consists of connected line segments:

```
{
  "type": "LineString",
  "coordinates": [
    [-115.650619, 50.869850], [-116.205839, 51.350990], [-119.156539, 53.110808]
  ]
}
```

<figure> <figcaption></figcaption> </figure>

- **Polygon:** For type "Polygon", the coordinates member must be an array of one or more linear ring coordinate arrays, where each array has at least four points and the first and last points must be the same. These linear rings define boundaries of the Polygon. For Polygons with more than one of these linear rings, the first must be the exterior ring, and any others must be interior rings. The exterior ring is the boundary of the polygon, and the interior rings (if present) are the boundaries of holes within the polygon. Here is an example of a four-sided polygon with one triangular hole inside it:

```
{
  "type": "Polygon",
  "coordinates": [
    [
      [-89.771949, 20.359083], [-14.355916, -7.946717], [-3.019922, 54.386109], [-96.741100, 49.071680], [-89.771949, 20.359083],
      [-80.226529, 25.789106], [-64.781380, 32.294887], [-66.1057427, 18.4663188], [-80.226529, 25.789106]
    ]
  ]
}
```

<figure> <figcaption></figcaption> </figure>

- **MultiPoint:** For type "MultiPoint", the coordinates member is an array of positions. This geometry specifies a collection of points.
- **MultiLineString:** For type "MultiLineString", the coordinates member is an array of LineString coordinate arrays. This geometry specifies a collection of LineStrings.
- **MultiPolygon:** For type "MultiPolygon", the coordinates member is an array of Polygon coordinate arrays. This geometry specifies a collection of Polygons.
- **GeometryCollection:** A GeoJSON object with type "GeometryCollection" is a heterogeneous collection of point, line, and polygon geometries grouped together. A GeometryCollection has a member with the name "geometries". The value of "geometries" is an array. Each element of this array is a GeoJSON

Geometry object with its own coordinates. Here is an example of a GeometryCollection that contains a MultiPoint and a Polygon:

```
{  
  "type": "GeometryCollection",  
  "geometries": [  
    {  
      "type": "MultiPoint",  
      "coordinates": [  
        [-76.619409, 39.329061], [-73.997303, 40.729437], [-78.786033, 43.000276]  
      ]  
    },  
    {  
      "type": "Polygon",  
      "coordinates": [  
        [[-80.226529, 25.789106], [-64.781380, 32.294887], [-66.1057427, 18.4663188], [-80.226529, 25.789106]]  
      ]  
    }  
  ]  
}
```

<figure> <figcaption></figcaption> </figure>

The last two types specified in GeoJSON describe **Features**, which are a combination of simple geometry and their descriptive properties.

- **Feature**: An object with type "Feature" has a member with the name "geometry". This "geometry" member must be exactly one of the seven geometry objects defined above, or a null value. A Feature object also contains a member named "properties", whose value can be any JSON object (for example, "name", "marker-size", "marker-color", etc.)
- **Feature Collection**: An object with type "FeatureCollection" has a member with the name "features". This "features" member is a JSON array of Feature objects, which were just described above.

Existing GeoJSON Documentation

[RFC 7946](#) is the official documentation for GeoJSON and is maintained by the Internet Engineering Task Force (IETF). The current documentation, which was put forward by the [GeoJSON Working Group](#) in 2016 and replaced the original documentation 2008 (GJ2008), has undergone public review, is the consensus of the IETF community, and was approved for publication by the Internet Engineering Steering Group (IESG). The documentation provides specifications for GeoJSON, including syntax for GeoJSON objects.

Example GeoJSON Datasets

The GeoJSON format is often a download option in geospatial data repositories, administrative open data portals, and when using certain APIs. While not an exhaustive list, examples of GeoJSON Datasets that are openly available include:

- [Australia Open Data](#)

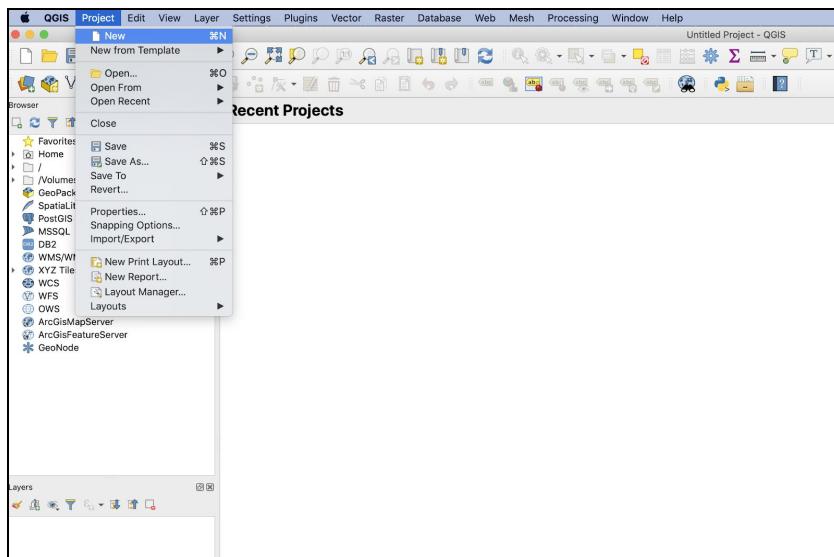
- [Data.gov catalog](#) (United States)
- [Government of Canada](#)
- [NYC OpenData](#)
- [OpenPhilly](#)
- [Plenar.io](#)

Software for Opening and Viewing GeoJSON Files

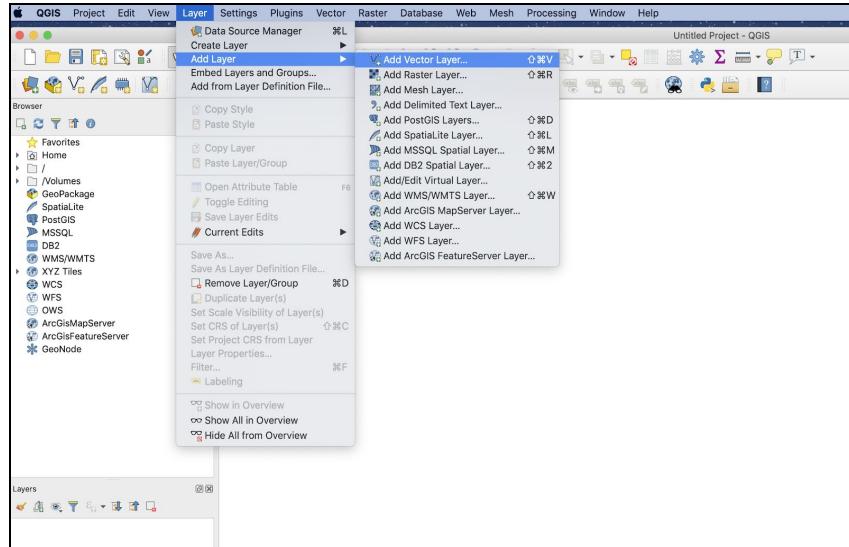
Key questions to ask:

- Can the file first be opened and viewed in a text editor before loading in geospatial software?
- Are coordinates listed in the following format: *[longitude, latitude, elevation]*
- What aspects of the visualization are important? (geometry, color scale, distribution of data, etc.)
- Is the file rendering properly as an individual layer and with a base map?

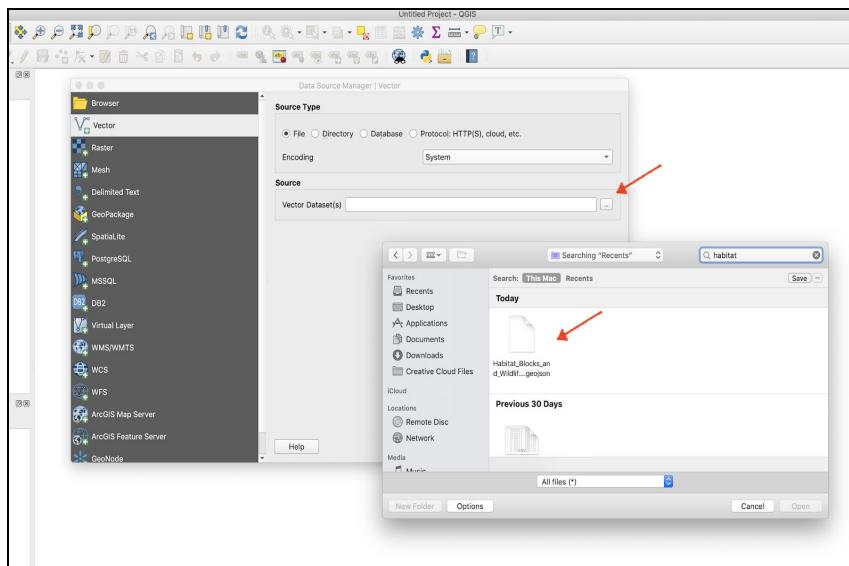
While there are many web-based and desktop software options for viewing GeoJSON, the free and open-source software [QGIS](#) is recommended. QGIS supports MacOS, Windows, and Linux, allows for numerous vector, raster, and database formats, and has a robust user community. Below is a walkthrough for loading and viewing GeoJSON in QGIS 3.8. Supplemental instructions on opening data in QGIS can be found in the [QGIS User Guide](#). This walkthrough uses the [Habitat Blocks and Wildlife Corridors](#) in Vermont dataset.



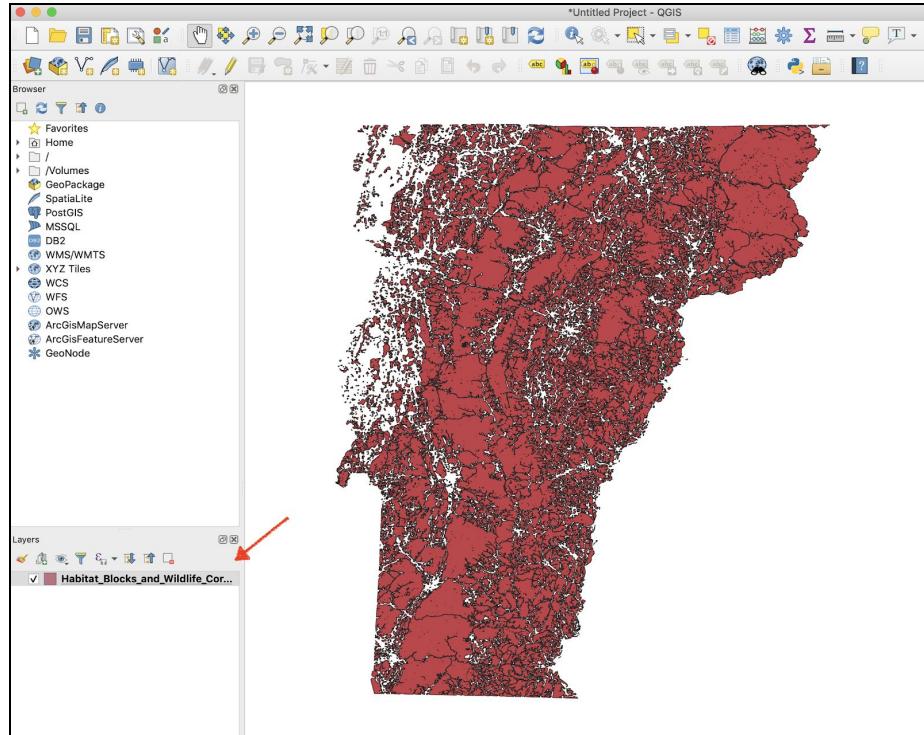
<figure> <figcaption>Fig. 1 Begin by going to “Project” in the QGIS toolbar and selecting “New” from the dropdown menu. </figcaption> </figure>



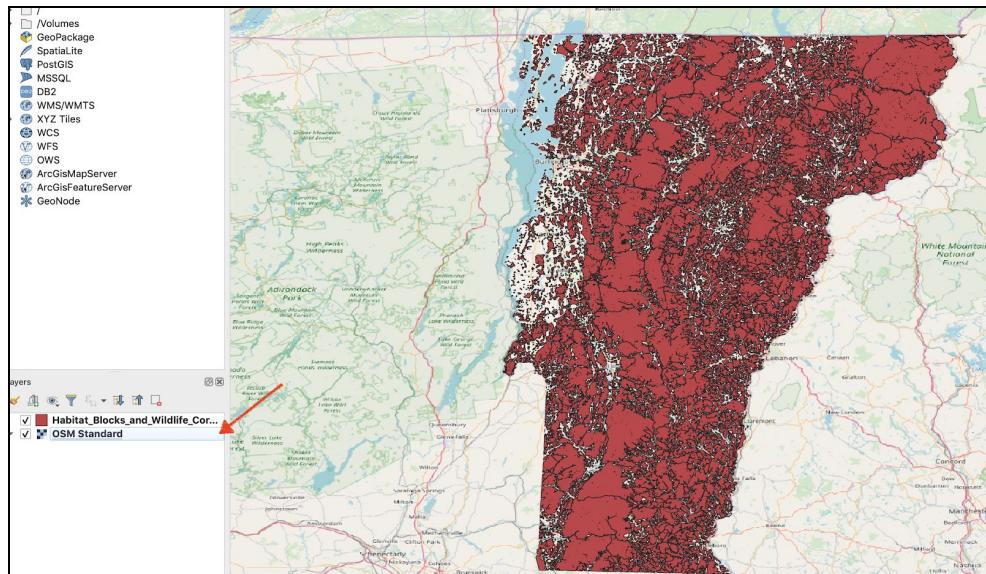
<figure> <figcaption>Fig. 2 After creating a new project, add the GeoJSON as a new layer. This can be done by going to “Layer” in the QGIS toolbar and selecting “Add Vector Layer...”</figcaption> </figure>



<figure> <figcaption>Fig. 3 Select the ellipsis [...] box and then select the GeoJSON you would like to view.</figcaption> </figure>



<figure> <figcaption>Fig. 4 The GeoJSON has been loaded as a layer and is now shown in the “Layers” sidebar located on the bottom left.</figcaption> </figure>



<figure> <figcaption>Fig. 5 A basemap can be added so that the layer is given context. To add a basemap in QGIS, go to “Plugins”, then “Manage and Install Plugins.” In the search box enter “QuickMapServices”. After adding this plugin, a global icon with a plus sign will be available on the toolbar. Use this to add one of the several basemaps available from QMS. [Alternatively, you can add the plugin “OpenLayers”. After adding this plugin, an OpenStreetMap basemap will be a nested option under XYZ Tiles located in the “Browser” sidebar to the left.]</figcaption> </figure>

While QGIS is the preferred software for viewing geospatial data, other software and platforms exist. Some options include, but are not limited to, the following. Please note that some options may be subscription-based:

- [geojson.io](#) (free, browser-based; useful for validating small .geojson files)
- [ArcGIS Pro](#) (proprietary, desktop software; best alternative to QGIS if available)
- [ArcGIS Online](#) (free, browser-based; useful for validating small .geojson files)

- [OpenStreetMap](#) (free, open-source, browser-based)

Validating GeoJSON Datasets

Key questions to ask:

- Is this dataset a valid GeoJSON file?
- Can you map it using an online validation service or desktop GIS software?
- If you come across errors when validating the GeoJSON, are these recoverable errors?

Validating GeoJSON can be conducted on a few web-based validation services. One of the recommended web-based services is [GeoJSONLint](#). Linting is the process of running a program that analyses code for potential syntax errors. To validate using GeoJSONLint, GeoJSON files can be copied and pasted or dropped into the provided text box. Examples of the nine GeoJSON types are also available via the dropdown headers in GeoJSONLint, all which can be used as additional reference examples. Please note that GeoJSONLint closely follows the most recent syntax and returns error codes identifying where problems occur. These errors may be minor and files could operate normally even with these errors. Viewing the file in [QGIS](#) and/or [geojson.io](#) in addition to linting is advisable and recommended.

Use this site to validate and view your GeoJSON. For details about GeoJSON, [read the spec](#).

```
{
  "type": "FeatureCollection",
  "features": [
    {"type": "Feature", "properties": {
      "bikedin": "L", "fromstreet": "W 16 ST", "lanecount": "1", "moddate": "2008-09-25T00:00:00.000Z", "facilitycl": "I", "segment_id": "33547", "tostreet": "W 31 ST", "instate": "2008-09-25T00:00:00.000Z", "tf_facilit": "Protected Path", "allclasses": "I", "onoffst": "ON", "ft_facilit": null, "bord": "1", "street": "9 AV", "comments": null}, "geometry": {
      "type": "MultiLineString", "coordinates": [
        [{"lon": -74.00039850902397, "lat": 40.74776789154442}, {"lon": -73.99993697165543, "lat": 40.74839931726861}]}]
    }
  ]
}
```

Clear Current Features

[Test GeoJSON](#) [Clear](#)

<figure> <figcaption>Fig. 6 Web Interface for validating GeoJSON of New York City Bike Lanes with GeoJSONLint</figcaption> </figure>

GeoJSON Metadata Standards and README Requirements

Key questions to ask:

- Is there a separate metadata file? If yes, is it complete? If not, is there metadata stored in the GeoJSON file?
- Is the GeoJSON filename adequately descriptive?
- Are the recommended minimum requirements for README files met?

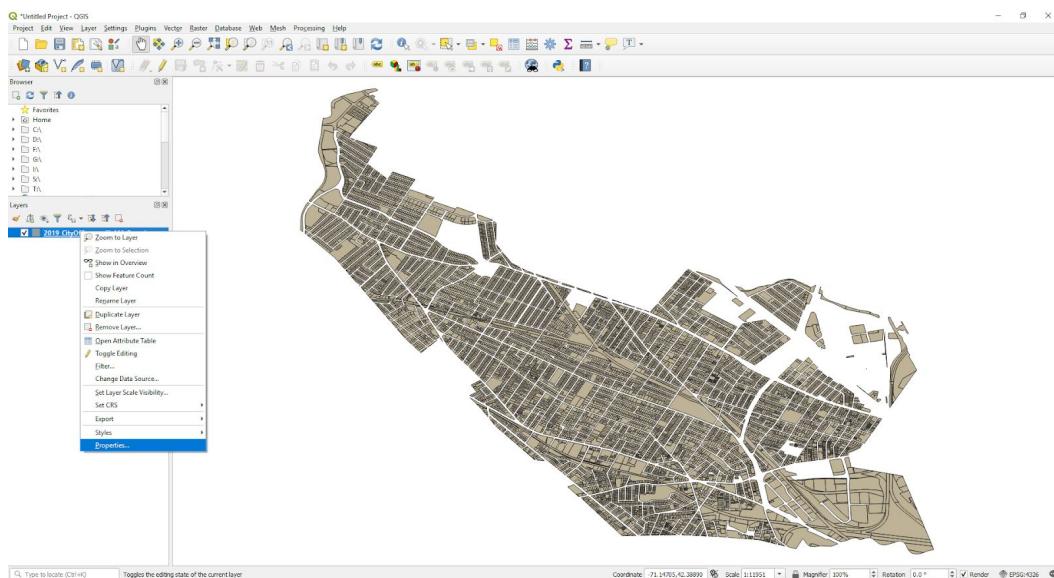
GeoJSON Metadata Standards

Currently, there is no metadata standard for GeoJSON.¹ As there is no particular standard, it is recommended that the curator select a metadata standard that best fits the repository's needs and/or policies. Metadata should be stored in a separate XML metadata file, rather than in the GeoJSON file itself. Finally, a README file should be prepared for each GeoJSON file. Descriptions of the most common geospatial metadata standards are available in the [Geodatabase Data Curation Primer](#).

If a separate XML metadata file exists, it can be viewed using a text editor. If a separate metadata file does not exist and it is unclear what, if any, metadata is in the GeoJSON file itself, QGIS can be used to view a GeoJSON's metadata.

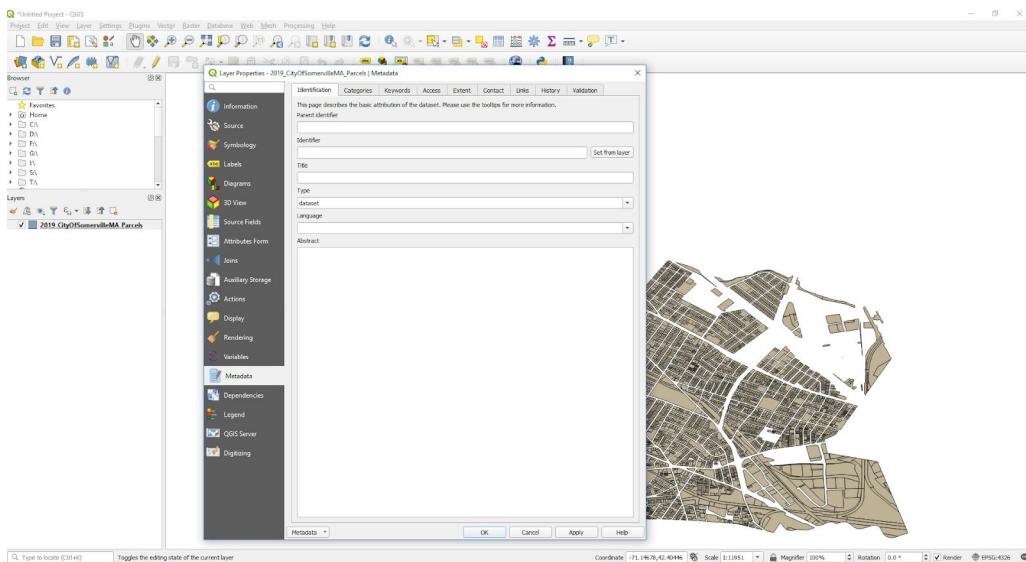
Viewing Metadata in QGIS

After a GeoJSON layer is added to a project in QGIS, metadata can be viewed using the properties menu.



<figure> <figcaption>Fig. 7 In QGIS, right-click on the layer and select "Properties".</figcaption> </figure>

¹ Gillies, S. (2016, August 17). Re: [Geojson] metadata in GeoJSONgeojson. Retrieved from <https://mailarchive.ietf.org/arch/browse/geojson/?q=metadata>



<figure> <figcaption>Fig. 8 The properties menu includes a metadata tab, in addition to tabs providing general information, Coordinate Reference System (CRS) and source fields.</figcaption> </figure>

In QGIS, the completeness of the metadata can be assessed by examining the information in the metadata tab. However, it is important to note that QGIS currently does not offer ways to view metadata using geospatial metadata standards such as ISO 191xx. Additionally, edits made to the metadata will not be preserved when converting the file to another format. The section *Preservation Strategy and Recommendations for Transformations* below gives details on metadata creation and augmentation.

What to Include in a README File

A README file provides information about a data file and ensures that data can be correctly interpreted by others. They are normally written as text files or as markdown. Below is a list of the recommended minimum content need to describe a data file. For more information about README files, please see the Cornell “[Guide to writing "readme" style metadata.](#)”

General Information:

- Title of GeoJSON file
- Name/institution/address/email information of author or PI
- Date of data collection (can be a single date, or a range)
- Information about geographic location of data collection

Data and File Overview:

- a short description of what data it contains
- Date that the file was created
- What other datasets or data sources (if any) were combined to create this GeoJSON file? How are they referenced?

Sharing and Access Information:

- Licenses or restrictions placed on the data

Methodological Information:

- Description of methods used for data processing (describe how the data were generated from the raw or collected data)
- Any software or instrument-specific information needed to understand or interpret the data, including software and hardware version numbers.
- If data pertains to human subjects or endangered species or is otherwise sensitive, have data been sufficiently anonymized?

Preservation Strategy and Recommendations for Transformations

While GeoJSON is a relatively young file type, it is widely used and has become one of the most common formats for spatial vector data, particularly for web-based projects. It is an approved international standard and is well documented. Finally, because GeoJSON is written in plaintext, it is well suited for preservation.

However, GeoJSON is just one of several commonly used geospatial file types. In a survey released in July 2019 for this primer, 100% of respondents stated that they regularly create or use shapefile data. While GeoJSON is an excellent format for preservation, only 11.1% responded that they always reuse data in GeoJSON format. The majority do not—55.6% responding sometimes, 27.8% responding rarely, and 5.5% responding never. It is therefore recommended to convert GeoJSON files into other file formats to ensure broad data access.

Because some of these other file formats run the risk of becoming obsolete over time or are proprietary, it is recommended to keep both the original GeoJSON file and the conversion files such as shapefile and GeoPackage (descriptions below), and document all of this in an accompanying README file.

Shapefile

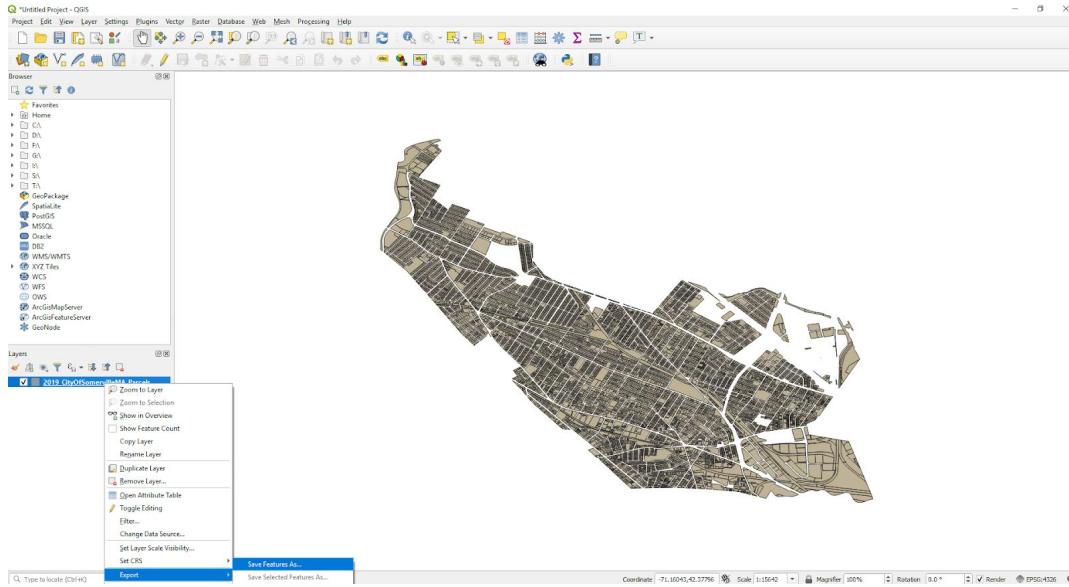
Shapefiles are a proprietary geospatial format created by Esri. While proprietary, it is one of the most widely used file formats for vector spatial data. It is well established and considered a de facto standard in many cases. It can be read by popular GIS desktop software, such as Esri's ArcGIS Pro and ArcMap, as well as the open source GIS desktop software QGIS. However, its file size is restricted to 2 GB, it is proprietary, and it is a multi-file format that is difficult to read and share without using the proper software and precautions. Shapefiles also have a limited number of columns they can hold. It is recommended that a shapefile version be saved along with the original GeoJSON file.

GeoPackage

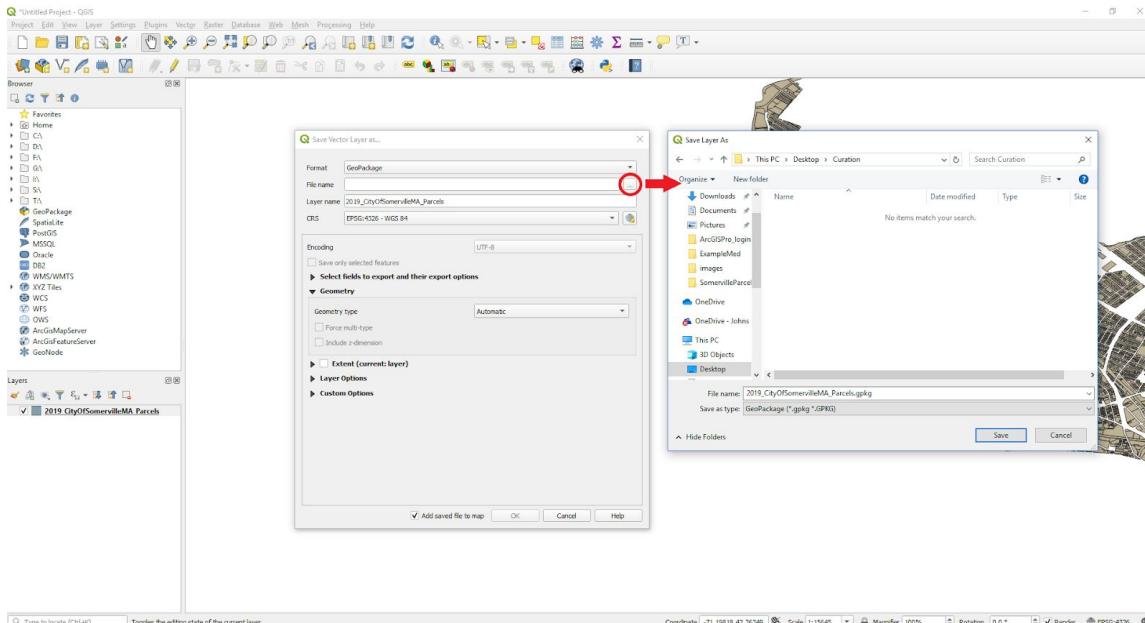
The [GeoPackage](#) is a relatively young file format that is becoming increasingly popular and adopted, particularly among the open source community. Its encoding standard was adopted by the OGC in 2014. It is an open, non-proprietary, platform-independent, and standards-based data format for geographic information systems implemented as a SQLite database container. It is now the default vector spatial file format in QGIS, replacing the shapefile as of QGIS version 3. Because it is platform-independent, it is broadly implemented and can be used in GIS desktop software such as ArcGIS and QGIS, but also with programming languages such as R and Python. It is also lightweight and fast. It is recommended that a GeoPackage version be saved along with the original GeoJSON file.

Conversion Tools

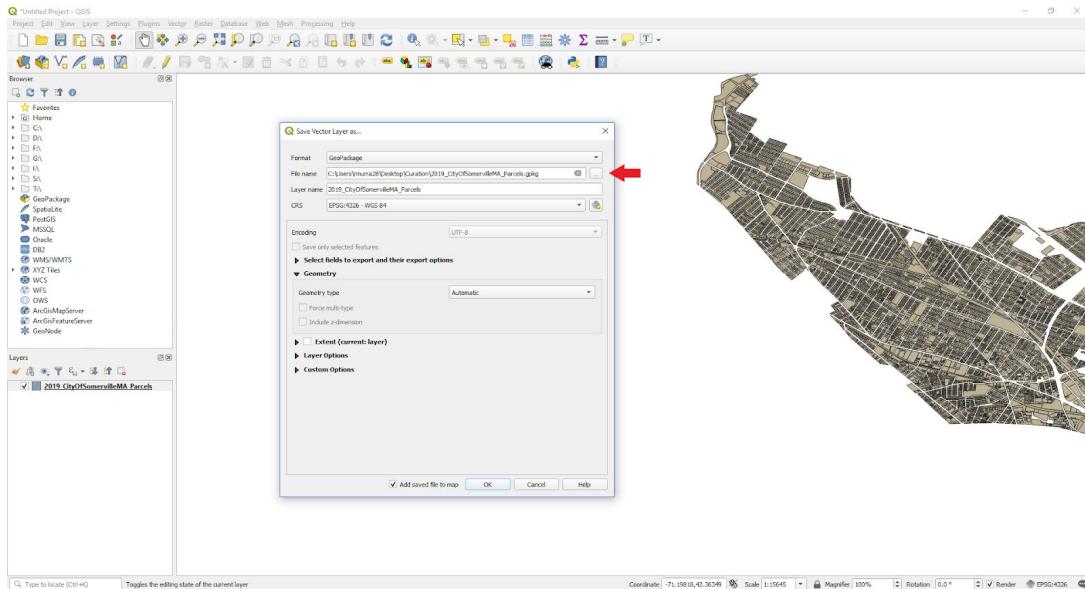
While there are a number of conversion tools available to transform a GeoJSON file to another format, it is recommended to use QGIS because of its ability to convert to both data formats listed above. A walkthrough for converting GeoJSON to a shapefile in QGIS is provided below. Note: an additional explanation of saving a layer as a shapefile is provided in the [Geodatabase Data Curation Primer](#).



<figure> <figcaption>Fig. 9 Right-click on layer and select “Export” and “Save Features As...”.</figcaption> </figure>



<figure> <figcaption>Fig. 10 Change format to “Esri Shapefile” or “GeoPackage”. Click on “...” and navigate to a folder to save the file. Enter a file name in the new window prompt. Click OK.</figcaption> </figure>



<figure>
 <figcaption>Fig. 11 Notice that the file name includes the full path. The full path is required to avoid generating an error. Leave encoding as UTF-8. Leave CRS as EPSG:4326 - WGS84. Do not resize the data. Click OK to finish conversion. </figcaption>

While QGIS is recommended for transforming file formats, other software and platform options exist:

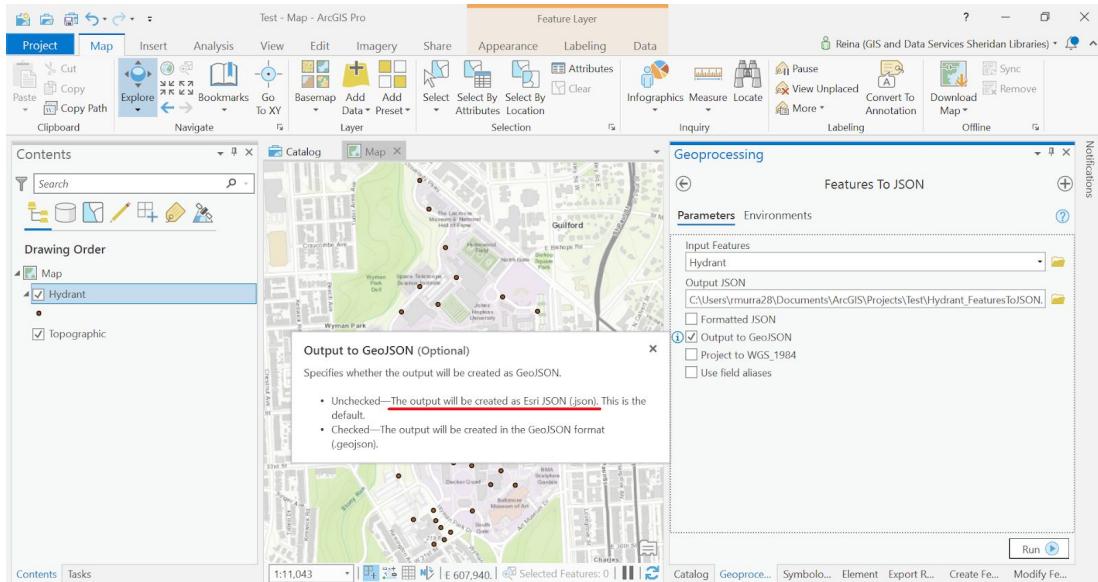
- ArcGIS Pro and ArcGIS Desktop: [JSON to Features conversion tool](#)

ArcGIS Pro and ArcGIS Desktop (also known as ArcMap) are part of Esri's suite of geospatial desktop software. Both offer a geoprocessing tool, JSON to Features, which can convert JSON and GeoJSON files to feature layers.

Note on Esri's [Features to JSON conversion tool](#): Esri also provides a geoprocessing tool, Features to JSON, which converts spatial data to JSON or GeoJSON format. It is important to note that Esri's proprietary version of a GeoJSON, known as EsriJSON, is created if the "Output to GeoJSON" parameter is not checked.

While similar to a GeoJSON file, an EsriJSON has its own categories for certain syntax, such as "esriGeometryType" for geometryType and "esriFieldTypeOID" for type. The full list of field types can be found on Esri's documentation for [ArcGIS REST API services](#).

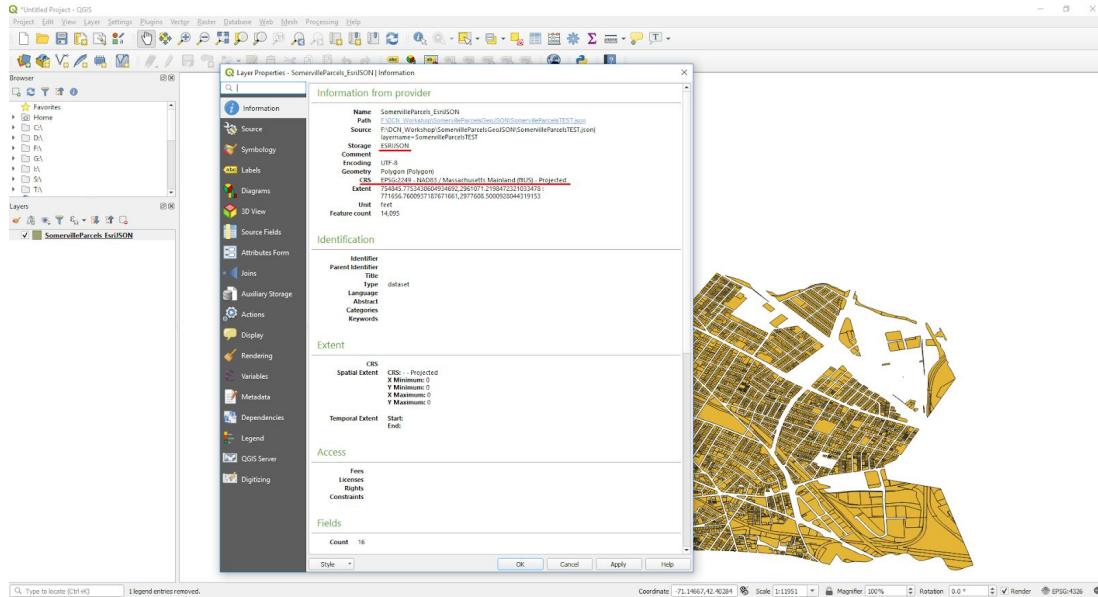
EsriJSON supports any coordinate reference system and preserves other Esri-proprietary file format structures, like the GlobalID in geodatabases. The file name also ends with '.json', as opposed to '.geojson'.



<figure> Fig. 12 Leaving "Output to GeoJSON" unchecked in ArcGIS Pro will result in the creation of an EsriJSON file.

<figure> <figcaption>Fig. 13
Example of EsriJSON syntax.</figcaption> </figure>

When validating a file, online validation platforms like GeoJSONlint will throw an error if an EsriJSON file is pasted in. The file should also be examined in QGIS. To see if a file is an EsriJSON file as opposed to a GeoJSON file, check the layer properties in QGIS; the CRS could be something other than EPSG:4326 - WGS84, and the storage type will be "EsriJSON".



<figure> <figcaption>Fig. 14 In Layer Properties, the storage type is set to "ESRIJSON", and for this particular file, the CRS is EPSG:2249, a CRS not supported by GeoJSON.</figcaption> </figure>

While similar to GeoJSON, EsriJSON is a proprietary format developed and managed by Esri. If an EsriJSON file is submitted for curation, the data owner should be contacted and asked to provide the original dataset in another file format if possible.

Additional options for converting GeoJSON

- [Mapshaper](#)

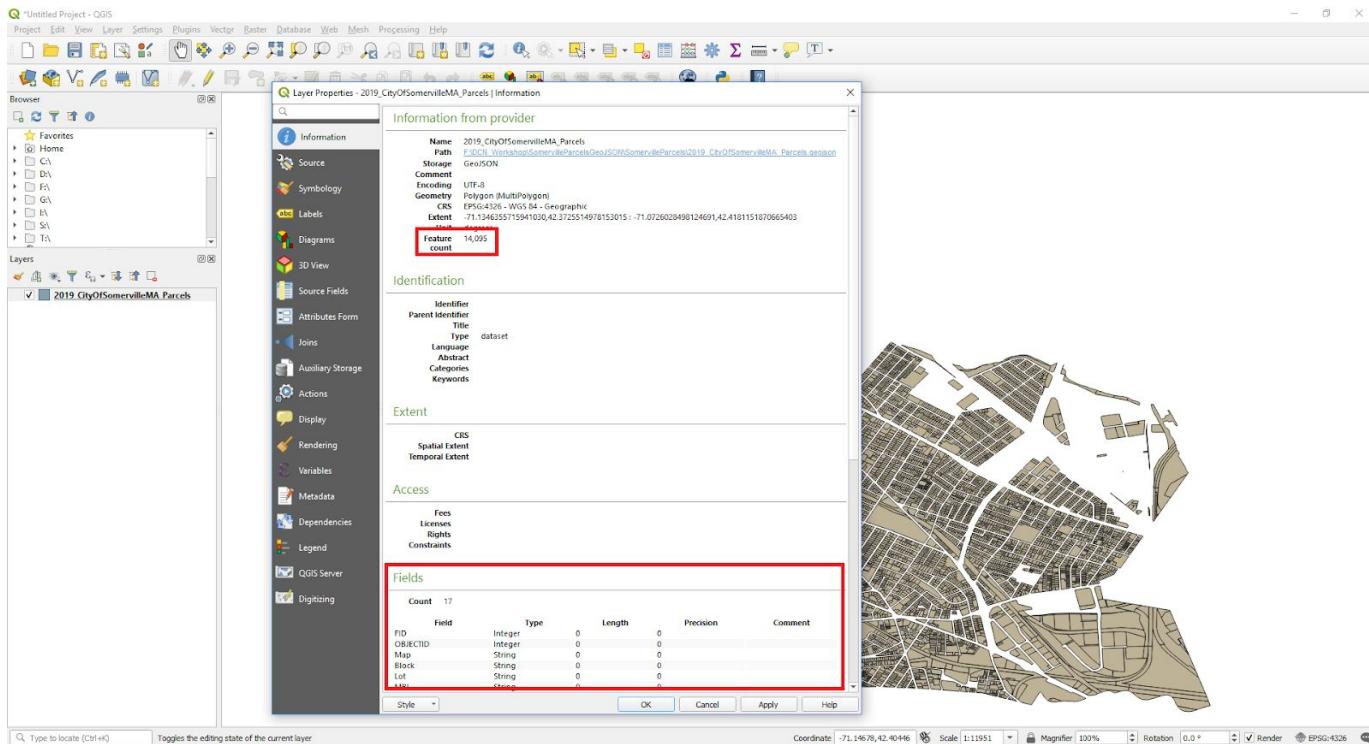
Mapshaper is a free software for editing shapefile, GeoJSON, TopoJSON, CSV and other data formats. It can be used to convert easily from one file format to another, and can be operated from the command line, but also through a web user interface. Instructions on converting a GeoJSON file to shapefile is [documented by StatSilk](#).

- [ogr2ogr package](#)

The ogr2ogr package is part of the Geospatial Data Abstraction Library and provides a way to convert data between common geospatial file formats, including GeoJSON, shapefiles, and GeoPackages. It can be downloaded as part of the [FW Tools Toolkit](#), and is usually run from the command line with the following syntax: `ogr2ogr -f "file_format" destination_data source_data`

Checking Transformations

After converting a GeoJSON dataset to both shapefile and GeoPackage format, confirm that both transformations were completed successfully using QGIS. These new data formats can be opened following the steps outlined in the earlier section *Software for Opening and Viewing GeoJSON Files*. Verify that the newly created shapefile and GeoPackage can be opened successfully. Compare the number of features in the original dataset with the new version to ensure that no data was lost in the process of file transformation.

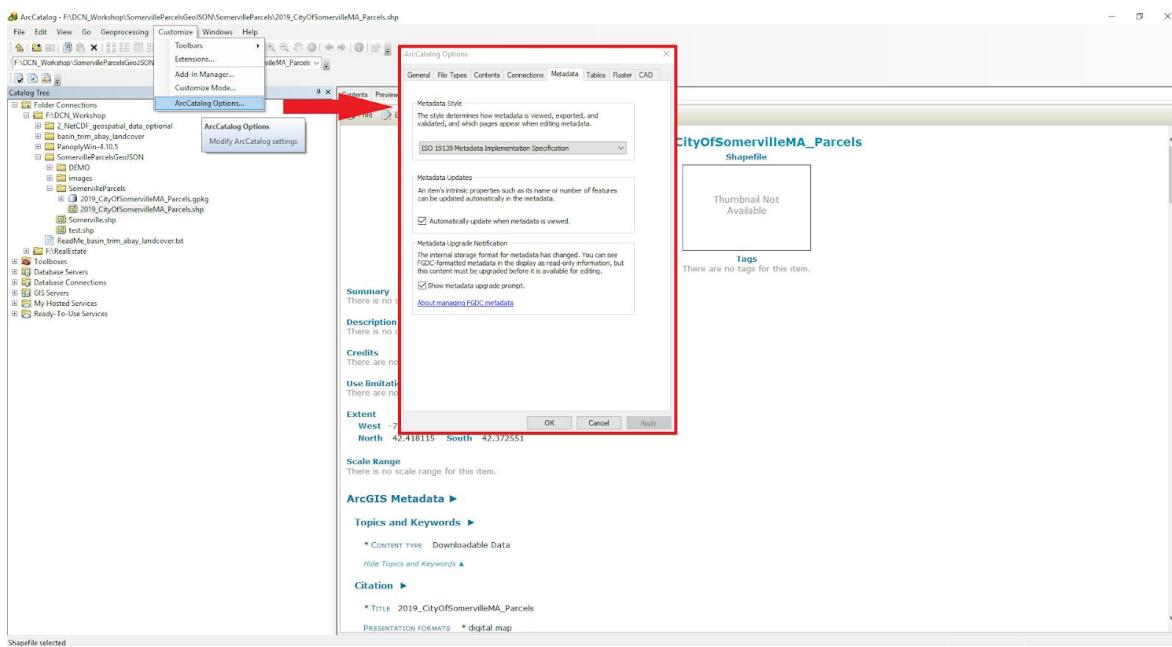


<figure> <figcaption>Fig. 15 In layer properties, the Information tab provides details on the feature count and fields. Inspect the feature count and the fields in the original dataset and in the newly converted datasets and confirm that they are the same.</figcaption> </figure>

Creating Metadata using ArcCatalog

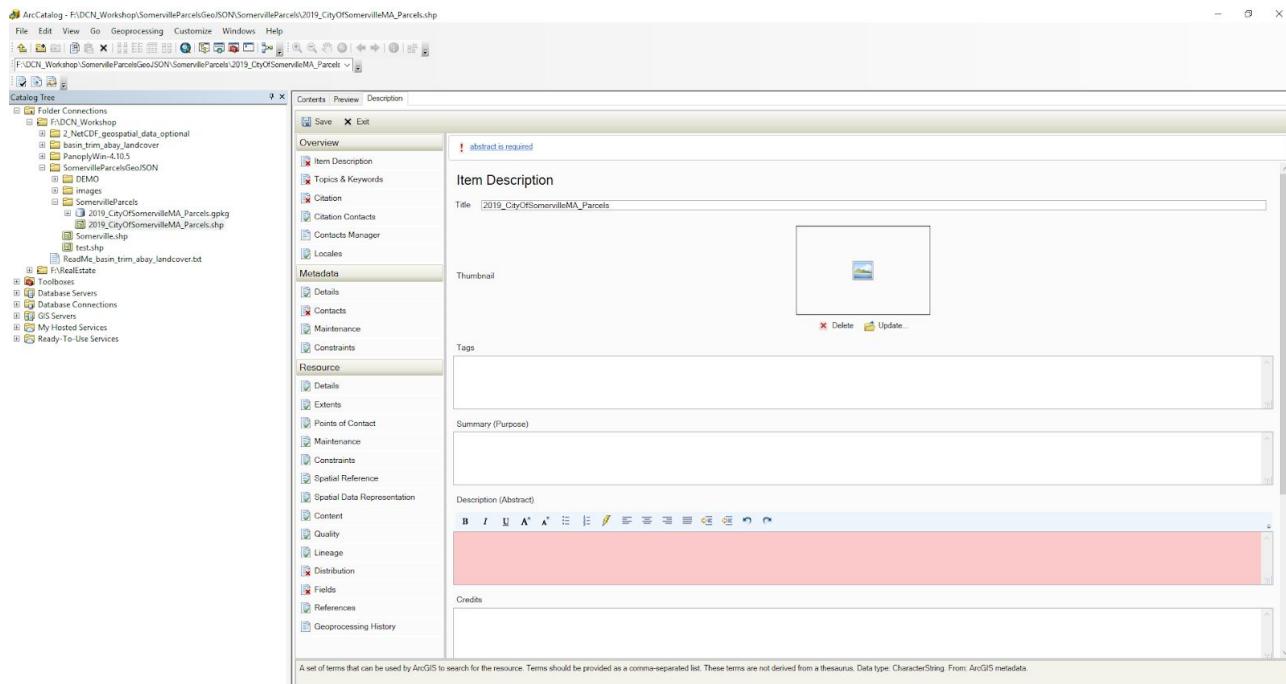
One option for creating metadata for geospatial files involves using ArcCatalog, which is part of the Esri ArcGIS for Desktop suite. The program, however, does not recognize or allow previewing GeoJSON files. As a result, it is necessary to first convert a GeoJSON file to a shapefile using the steps above. The shapefile is then opened in ArcCatalog, which allows for the creation of a separate metadata XML file that can be stored alongside the original GeoJSON file.

In ArcCatalog, the metadata style can be set by going to “ArcCatalog Options”, and clicking on the “Metadata” tab as demonstrated below. More information on the metadata styles and standards supported in ArcCatalog can be found on [Esri's metadata styles and standards documentation page](#).



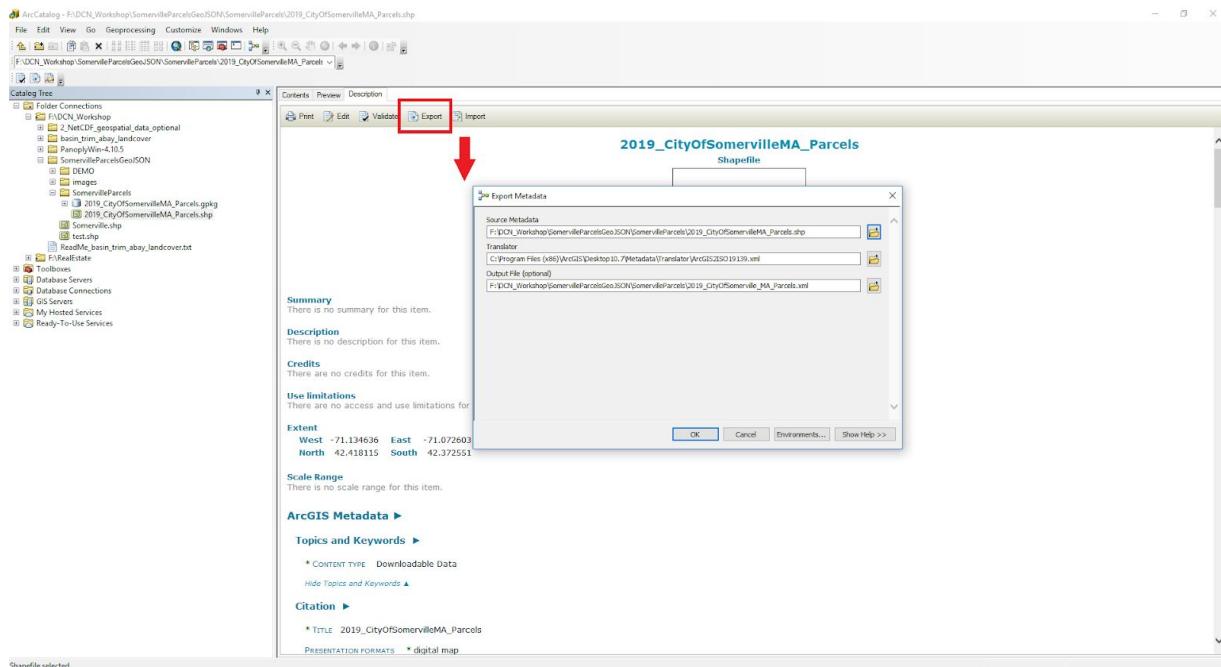
<figure> <figcaption>Fig. 16 Metadata styles can be set through the ArcCatalog Options pane.</figcaption> </figure>

After selecting a metadata style, establish a connection to the folder where the converted shapefile is located. To view the metadata for a specific file, click on the description tab. To edit the metadata, click “Edit”. This will display the metadata for the file, using the metadata style selected. Items that are required will be identified with a red ‘x’ in the left-hand navigator, and across the top for each section.



<figure> <figcaption>Fig. 17 Metadata for a dataset can be edited through the Description tab. Required items will be marked by the software.</figcaption> </figure>

After editing/augmenting a file’s metadata, hit “Save”. Then, hit the “Export” button to export the metadata as a separate XML file. Saving the metadata as a separate XML file allows it to be associated with the original GeoJSON as well as the other created spatial formats: the shapefile and the GeoPackage.



<figure> <figcaption>Fig. 18 Use the Export tab to create a separate metadata file in XML format..</figcaption> </figure>

Creating Metadata using mdEditor

If an XML metadata file already exists for the GeoJSON file in question, or ArcCatalog is not available to use, [mdEditor](#) may be a viable option. An open source web application, mdEditor is designed for both the creation and editing of metadata, and accommodates geospatial metadata standards such as ISO 191XX and CSDGM. However, as of October 2019, it is only available in beta.

Evaluating GeoJSON Files for FAIRness

What are the FAIR principles?

"In 2016, the '[FAIR Guiding Principles for scientific data management and stewardship](#)' were published in *Scientific Data*. The authors intended to provide guidelines to improve the findability, accessibility, interoperability, and reuse of digital assets. The principles emphasise machine-actionability (i.e., the capacity of computational systems to find, access, interoperate, and reuse data with no or minimal human intervention) because humans increasingly rely on computational support to deal with data as a result of the increase in volume, complexity, and creation speed of data." (Source: [Go-Fair.org](#))

FAIR GeoJSON: Components Evaluated

To meet the standards for (meta)data documentation required by the [FAIR principles](#), multiple entities must be evaluated for FAIRness: (1) the data stored in GeoJSON format, (2) the GeoJSON data's associated metadata, and (3) the repository infrastructure in which the data is deposited. The following table indicates

which entities (data, metadata, and/or infrastructure) must be evaluated to meet the standards for each FAIR principle.

Principle	Entities Evaluated for FAIRness
F1: globally unique, persistent identifier	data; metadata
F2: data described with rich metadata (see R1)	metadata
F3: metadata explicitly reference data identifier	metadata
F4: data indexed in searchable resource	data; metadata; infrastructure
A1: data/metadata retrievable by identifier	data; metadata; infrastructure
A2: metadata accessible beyond life of data	infrastructure
I1: data/metadata represented in standard format	data; metadata
I2: data/metadata use standard vocabularies	data; metadata
I3: data/metadata cross-reference other data/metadata	data; metadata
R1: metadata richly described (license, provenance, domain-relevant community standards)	metadata

Assessing GeoJSON Data for FAIRness

The following questions can assist researchers in evaluating the FAIRness of their GeoJSON data. The questions are organized according to each FAIR principle, and most questions pertain to both the data itself and its accompanying metadata: both intrinsic metadata such as variable names and controlled vocabularies as well as external metadata that describes the GeoJSON data file.

Findable

- F1.** Does each GeoJSON and accompanying metadata file have a globally unique, persistent identifier?
- F2. (a)** Does each GeoJSON have an accompanying metadata file? **(b)** Does the filename of each accompanying metadata file explicitly reference the globally unique, persistent identifier for the GeoJSON it describes?
- F3.** Does each metadata file accompanying a GeoJSON include a field explicitly referencing the file's globally unique, persistent identifier?
- F4. (a)** Are the GeoJSON and accompanying metadata file(s) stored in a repository with full index and search functionality? **(b)** Does the title of the repository item accurately and fully describe the key components of the file's contents, including at least the name of the dataset, general geographical coverage of the dataset, name of the dataset creator, and date(s) of temporal coverage of the dataset?
(c) Does the repository metadata include the globally unique, persistent identifiers for GeoJSON and accompanying metadata file(s)?

Accessible

A1. Is each GeoJSON and accompanying metadata file(s) able to be retrieved using the file's globally unique, persistent identifier? If the data files are not public, does the repository in which the data and metadata are stored permit retrieval of the data and metadata via secure authentication protocols such as HTTPS or FTPS?

A2. Does the repository in which the GeoJSON is stored permit the perpetual storage of metadata about the file in the event that the data itself becomes unavailable? (According to a [2017 study by Dunning et al.](#), many repositories do not have clear policy statements on the perpetual availability of metadata.)

Interoperable

I1. (a) To meet filetype standards, GeoJSON must be organized as JSON data structures representing a *geometry*, *feature*, or *collection of features*. Does each GeoJSON meet the [RFC 7946](#) standard? **(b)** Does the accompanying metadata file for each GeoJSON include standard metadata fields for geospatial information? (see [ISO 19115:2003](#) for the metadata standards for geospatial data)

I2. (a) To meet file type standards, GeoJSON should use the following vocabulary for values in at least the starting name/value pair: *Point*, *MultiPoint*, *LineString*, *MultiLineString*, *Polygon*, *MultiPolygon*, *GeometryCollection*, *Feature*, *FeatureCollection*. Does each file meet this vocabulary standard? **(b)** Does the accompanying metadata file for each use the standard vocabularies for the chosen metadata schema?

I3. Does each GeoJSON and accompanying metadata file(s) properly cite any related publications or datasets (using those publications' or datasets' globally unique and persistent identifiers)?

Reusable

R1. The metadata file(s) accompanying each GeoJSON should include, at a minimum, information about the dataset's provenance, ownership, licensing, and any other contextual information that is necessary for the data to be reused. Do(es) the metadata file(s) meet these minimum metadata standards?

Appendix A: Example CURATED Checklist for GeoJSON

CURATED Checklist for GeoJSON

This curation workflow is meant to serve as an example, demonstrating the recommendations made within this primer. It follows the Data Curation Network's recommended [CURATED](#) steps:

- C - check files and read documentation
- U - understand (or try to) the data
- R - request missing information
- A - augment metadata
- T - transform file formats for reuse
- E - evaluate for fairness

- D - document all curation activities

Dataset: SomervilleParcels

Description of dataset: This dataset was provided by the City of Somerville, MA for this data curation primer and represents the City of Somerville, MA's parcels.

File size: 10.4 MB

C - check files and read documentation

1. Does the file open as expected?

Yes. The file was opened using notepad++, a text editor.



```
<figure>  <figcaption>Fig. 19  
SomervilleParcels GeoJSON file opened in notepad++.</figcaption> </figure>
```

2. Is the dataset valid?

At first glance, the dataset appears to be a valid GeoJSON file – in notepad++, one can see that the dataset has a type of “FeatureCollection”, with coordinates that appear to be in the format [longitude, latitude]. However, to properly validate the dataset, the dataset was opened in both an online validation service ([GeoJSONLint](#)) and QGIS, an open source GIS desktop software program.

To validate using online services, the full text in notepad++ was copied and pasted into the website. Note that with online validation services, the size of the file in question can cause the site to stall and/or become unresponsive. At 10.4 MB, this file took GeoJSONLint approximately 3 minutes to load:

GeoJSONLint Point ▾ LineString ▾ Polygon ▾ Feature FeatureCollection GeometryCollection

Use this site to validate and view your GeoJSON. For details about GeoJSON, [read the spec.](#)

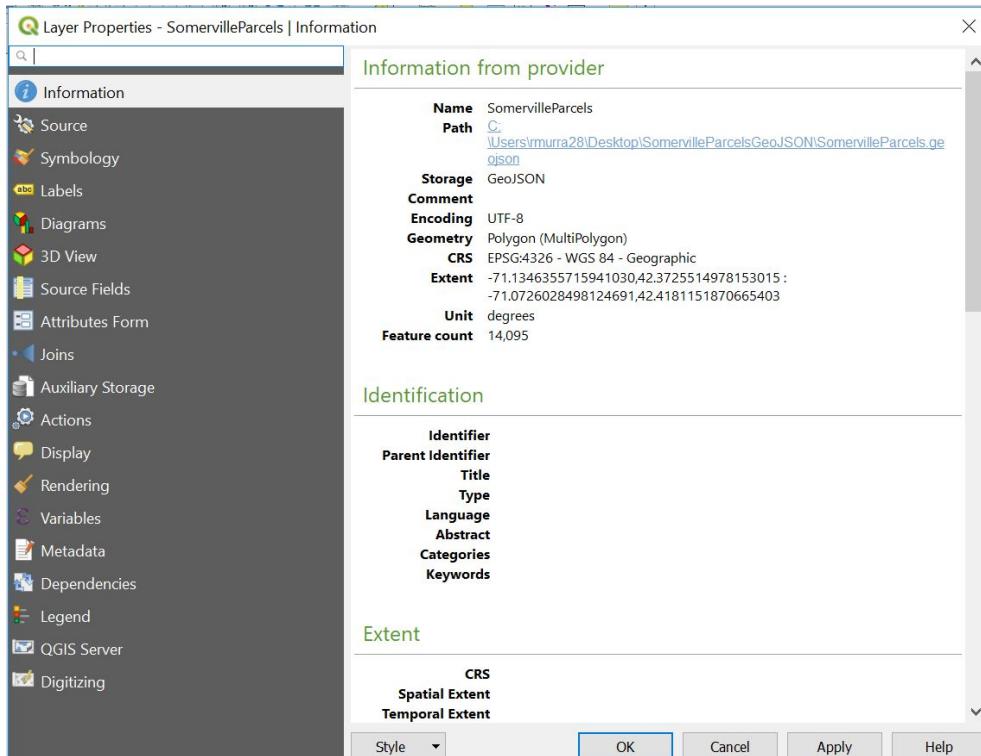
```
"{"MBL": "Pending", "PolyType": "PARCEL", "AddNum": "106", "Street": "A", "AddNum2": "106", "Street2": "South St", "AddNum3": "106", "Street3": "George St", "SublotOf": "106", "TaxParMBL": "106", "Shape_Leng": 45.22773800000002, "Shape_Area": 106.39885700000001}, {"type": "Feature", "id": 14094, "geometry": {"type": "Polygon", "coordinates": []}, "properties": {"FID": 14094, "OBJECTID": 14585, "Map": "99", "Block": "A", "Lot": "1"}, {"MBL": "Pending", "PolyType": "PARCEL", "AddNum": "3923", "Street": "A", "AddNum2": "3923", "Street2": "Huron Ave", "AddNum3": "3923", "Street3": "Brattle St", "SublotOf": "3923", "TaxParMBL": "3923", "Shape_Leng": 1121.417539999999, "Shape_Area": 3923.076185999998}}]
```

Clear Current Features [Test GeoJSON](#) [Clear](#)

<figure> <figcaption>Fig. 20 SomervilleParcels dataset validated using GeoJSONLint.</figcaption> </figure>

The dataset was also brought into QGIS, where it could be verified that the dataset appears to be what it is stating to be: a collection of polygons representing parcels.

The dataset's property details panel was used to discover further information. The first tab, "Information" is read-only and serves as a place to quickly grab summarized information and metadata on the layer in question. Here several key pieces of information were verified for the dataset, such as its title (SomervilleParcels), its storage file format (GeoJSON), its geometry (Polygon -- MultiPolygon), its coordinate reference system, or CRS (EPSG:4326), the extent or bounding box of the dataset, and the feature count (14,095). The storage file format and CRS help us verify that this dataset is a valid GeoJSON file.



<figure>
<figcaption>Fig. 21 Layer properties panel for SomervilleParcels dataset.</figcaption> </figure>

3. Is there a separate metadata file?

No. The GeoJSON file did not come with an XML metadata file.

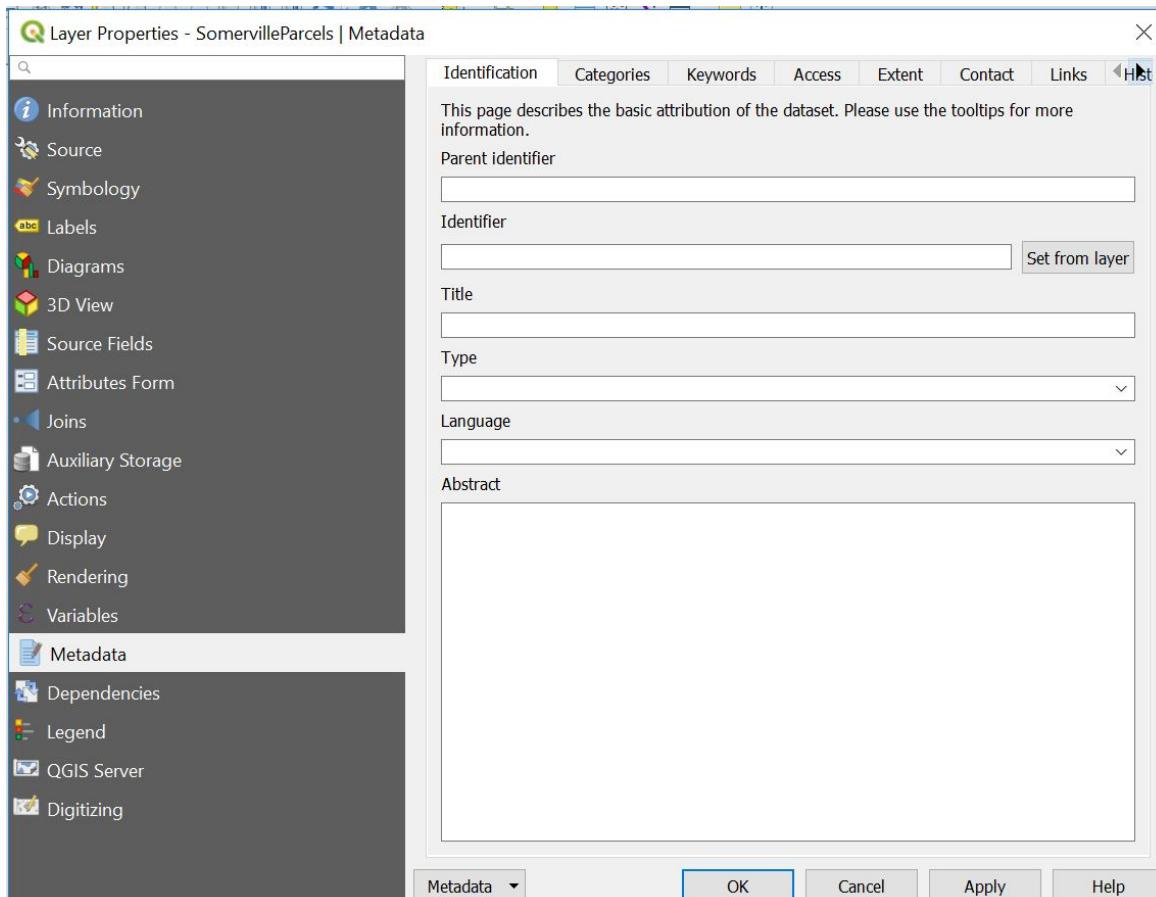
4. Is the metadata quality rich, accurate, and complete?

No. The GeoJSON file was opened in QGIS, and the metadata tab was examined. The metadata is incomplete. Critical information is missing, such as the formal title; parent identifier (collections this dataset is a part of); abstract; access policies and/or use restrictions; contact information; etc.

In particular, for geospatial data, the following are required by both of the two most common geospatial metadata standards (ISO 191xx and FCGC CSDGM):

- Bounding box
- Geographic location (place keywords)
- Spatial representation type (vector, raster, etc)
- Projection/coordinate system

Due to the definition of a GeoJSON file, the spatial representation type is always a vector, and the projection/coordinate system will always be WGS84, or EPGS:4326. The extent of the dataset, provided in the layer's properties information tab, serves as the bounding box. However, geographic location, or place keywords, is missing for this dataset.



<figure> <figcaption>Fig. 22 Blank metadata tab in SomervilleParcels geojson file.</figcaption> </figure>

5. What documentation exists?

There is no associated README.md file, or other document providing additional context. There is also no codebook or description for the attribute table column names.

6. Is additional documentation necessary for interpretation?

Yes. Additional documentation will help to provide context to this dataset.

7. Is the filename adequately descriptive?

No – the filename, “SomervilleParcels”, provides only some key information on this dataset. We know the dataset in question concerns parcel data in Somerville, but from the title, one cannot ascertain which city/town/township/area/local administration the dataset is referring to. The filename was changed to “2019_CityOfSomervilleMA_Parcels.”

8. Were any tools used to create the dataset? Software? Hardware?

It is unclear what tools, software, or hardware were used to create this dataset, as well as what other datasets or data sources were combined, if any, to create this file.

U - understand (or try to) the data

1. View the data on a map. Does the location of the parcels match the general description of the dataset?

Yes. The dataset is located in Somerville, Massachusetts. A basemap can be added to confirm the general location of the layer in QGIS.

2. Ascertain number of rows and columns. Are the column names understandable?

In QGIS, the Information tab in Layer Properties provides a count of the total number of rows, as well as a list and count of the field names in the dataset. This dataset has 14,095 rows of data and 17 attribute fields:

- FID
- OBJECTID
- Map
- Block
- Lot
- MBL
- PolyType
- AddNum
- Street
- AddNum2
- Street2
- AddNum3

- Street3
- SublotOf
- TaxParMBL
- Shape_Leng
- Shape_Area

The column names do not have aliases. A code book is not included with the dataset, making it difficult to know for sure what the columns describe. The documentation of the data is not sufficient for reuse. For example, what does the column “Map” or “Block” refer to?

3. Inspect the attribute table.

The attribute table can be accessed by right-clicking on the layer and selecting “Open Attribute Table”. Viewing the attribute table, we can assess that the dataset in question does have a populated attribute table; while there are some rows with missing values for some columns (such as Street2, AddNumb3, SublotOf), the dataset overall appears to be complete. There are no empty rows. Each item appears to have a unique ID (FID and OBJECTID). The data in the table is readable, with no strange encodings or text.

	FID	OBJECTID	Map	Block	Lot	MBL	PolyType	AddNum	Street	AddNum2	Street2	AddNum3	Street3	SublotOf	TaxParMBL	Shape_Leng	Shape_Area
	14002	0110	0120 90	01	21	90-00-01	PARCEL	129-121	PEARL ST	01-09	CROSS ST	68	OLIVER ST			110-410-109	1020-121294
14086	11136	11157 92	F	1	92-F-1	PARCEL	114	CROSS ST	68					92-F-1	294.063726	5360.466164	
14087	12600	12628 83	E	4	83-E-4	PARCEL	26	OAK ST	7	BOLTON ...				83-E-4	369.406155	6839.286253	
14088	13736	14024 92	C	1	92-C-1	PARCEL	124-126	PEARL ST	72-78	CROSS ST				92-C-1	408.250743	7956.36367	
14089	11561	11586 104	E	23	104-E-23	PARCEL	85	WASHINGTON ST	78	FRANKLI...				104-E-23	805.126707	26464.089343	
14090	2157	2163 61	F	2	61-F-2	PARCEL	93	HIGHLAND AVE	81	HIGHLAN...	61	HIGHLAN...		61-F-2	3486.025342	573327.964851	
14091	6513	6520 33	B	2	33-B-2	PARCEL	2	BERWICK ST	81	HINCKLE...				33-B-2	194.363484	2076.827462	
14092	10747	10767 51	F	21	51-F-21	PARCEL	81	SUMMER ST	8-10	AVON ST				51-F-21	552.513904	13670.189692	
14093	11885	11912 74	C	15	74-C-15	PARCEL	38	UNION SQ	8-10	BOW ST	301	SOMERVI...		74-C-15	363.453158	7700.875378	
14094	11634	11661 106	A	4	106-A-4	PARCEL	17	INNER BELT RD	9	NEW WA...				106-A-4	931.159717	32278.706698	
14095	10413	10433 91	F	1	91-F-1	PARCEL	132	PERKINS ST	93	MYRTLE ST				91-F-1	409.637168	10168.193899	

<figure> <figcaption>Fig. 23 Attribute table for SomervilleParcels geojson file.</figcaption> </figure>

R - request missing information

1. Solidify list of questions and missing information

The first two steps result in a list of questions and missing information, the bulk of which is needed in order to adequately curate this dataset. These should be gathered and used to formulate a list of recommendations/questions to communicate to the data author. This can include clarifying assumptions about the dataset, as well as missing information:

- Who created this dataset?
- What year was it created?
- What year of data does it show?
- What is the primary purpose of this dataset?
- How was this dataset created?
- How often is it updated?

- Are there any restrictions on accessing and sharing this dataset?
 - Is this dataset related to other data we should be aware of?
 - Is there a codebook or detailed descriptions for the columns in the attribute table?
 - Is there a README file available? Other relevant documentation?
2. From the list of questions/recommendations, identify the top 3-5 questions/recommendations to ask the data author:
 - Is there any documentation, such as a README file, that provides some information on this dataset (year of creation, who created it, how often it is updated, how it was created, etc)?
 - Are there any restrictions on access to this data, and how should it be cited?
 - Is there a codebook for the attribute table headers? If not, can you provide a brief description for the headers?
 - Please verify if you should be listed as the contact for this dataset and provide your contact information.
 - Are there other GIS data this dataset is related to?
 3. Incorporate recommendations into email to the data author.

The Data Curation Network provides a sample letter for requesting changes from data authors in their [manual on the CURATED steps](#).

A - augment metadata and T - transform file formats for reuse

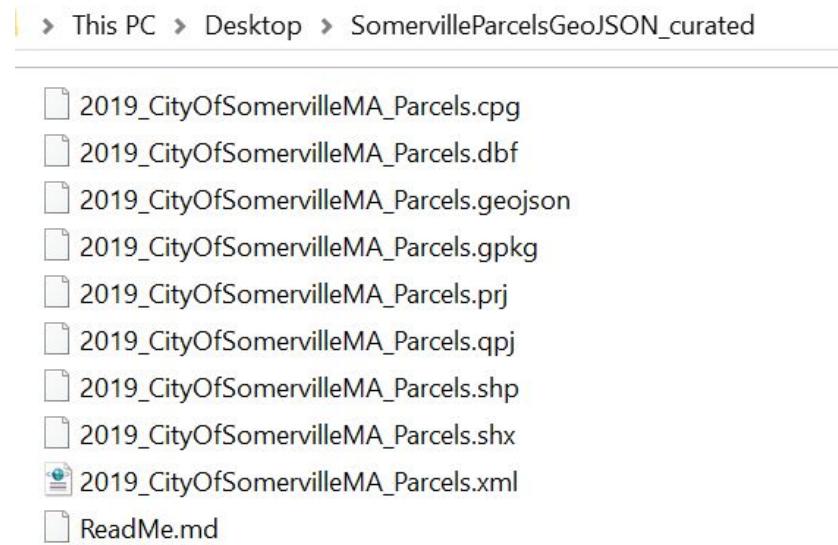
Due to the lack of a metadata XML file, and the metadata editing limitations in QGIS, the steps to augment metadata and transform file formats for reuse are done concurrently for this dataset. The following steps are taken:

1. [Draft a README file](#).
2. [Convert GeoJSON to shapefile](#).
3. [Edit metadata in ArcCatalog](#):
 - a. Change the metadata style to ISO 19139
 - b. Edit metadata sections
 - c. Export metadata as a separate XML file.
4. [Convert GeoJSON to GeoPackage](#).
5. [Check conversions](#).

Both the newly created shapefile and GeoPackage were opened in QGIS to confirm that 1) the files opened properly, and 2) the files overlapped with the original GeoJSON file. The attribute table was opened and examined to verify that both have the same number of rows and columns as the original dataset.

6. Inspect final file folder.

After metadata creation, file conversions, and verifications are complete, the resulting file folder for the dataset in question should contain the original GeoJSON file (.geojson), a metadata XML file (.xml), a ReadMe file (.md), a GeoPackage file (.gpkg), and a shapefile (.cpg, .dbf, .prj, .qpj, .shp, and .shx):



<figure> <figcaption>Fig. 24 File folder with renamed geojson file, converted files, metadata and ReadMe file. </figcaption> </figure>

E - evaluate for fairness

1. [Go through the FAIR principles](#) and assess both the metadata and the GeoJSON file:

Note: this dataset was provided as an example for this primer. It has not been curated/preserved in a repository. Principles that could not be assessed are marked as "N/A".

Principle		data	metadata
Findable	F1. globally unique, persistent identifier	N/A	N/A
	F2. data described with rich metadata	yes	
	F3. metadata explicitly reference data identifier		N/A
	F4. data indexed in searchable resource	N/A	N/A
Accessible	A1. data/metadata retrievable by identifier	N/A	N/A
	A2. metadata accessible beyond life of data		yes
Interoperable	I1. data/metadata represented in standard format	yes	yes
	I2. data/metadata uses standard vocabularies	yes	yes
	I3. data/metadata cross-reference other data/metadata	N/A	N/A

Reusable	R1. metadata richly described		yes
----------	-------------------------------	--	-----

D - document all curation activities

Documentation should be created throughout the CURATE process. Going through each of the steps above, take notes on the following, making sure to also note who was responsible for each step and when it was done:

1. Original dataset and documentation
 - a. Software and tools were used to assess and validate data
 - b. Questions/concerns that came up through validation process
 - c. State of metadata and other documentation files
2. Conversions
 - a. Software and tools used to transform data
 - b. Transformations done to data (shapefile, GeoPackage)
3. Metadata and documentation augmentation/creation
 - a. Creation of shapefile and exporting of metadata to XML
 - b. Changes made to metadata and documentation
4. Correspondence with data owner on dataset
 - a. Request for missing information
5. Evaluation for fairness

This example did not include the final step of depositing a dataset into a repository. Additional notes to document when undertaking this step include:

6. Submission records
7. Repository collection metadata
8. Any additional requirements at repository in question

Appendix B: Researcher Survey Results

In July 2019, the GeoJSON primer authors created and deployed a short online questionnaire, intended to gather information about researchers' creation, use, and reuse of geospatial vector data in GeoJSON format. The survey - delivered as a Google Poll - was shared with various email lists and GIS working groups and received a total of 18 responses, as of August 8, 2019.

Survey Questions

1. What discipline or field do you work in? [short answer]
2. What is your general level of familiarity or expertise with geospatial data? [multiple choice (select one) + "other" (short answer)]
3. (Optional) What is your position or title? [short answer]
4. How often do you CREATE data in GeoJSON format? [multiple choice (select one) + "other" (short answer)]
5. How often do you (RE)USE data that is in GeoJSON format? [multiple choice (select one) + "other" (short answer)]

6. What tools or platforms do you use to create, view, or manipulate GeoJSON data? [multiple choice (select many) + “other” (short answer)]
7. Aside from GeoJSON, what other vector data formats do you create or use regularly? [multiple choice (select many) + “other” (short answer)]
8. What online resources do you consult for creating or using GeoJSON data? [multiple choice (select many) + “other” (short answer)]
9. Where do you search for and find reusable GeoJSON data? Please mention specific repositories or platforms if applicable. [long answer]
10. In what ways do you share your geospatial data, including data in GeoJSON format? Please mention specific repositories or platforms if applicable. [long answer]
11. Do you create metadata files to accompany your GeoJSON data files? [multiple choice (select one) + “other” (short answer)]
12. If you store or share your geospatial data in a data repository, do you primarily submit it in GeoJSON format? [multiple choice (select one) + “other” (short answer)]
13. If you do share your data, what other file formats do you submit to data repositories? [long answer]

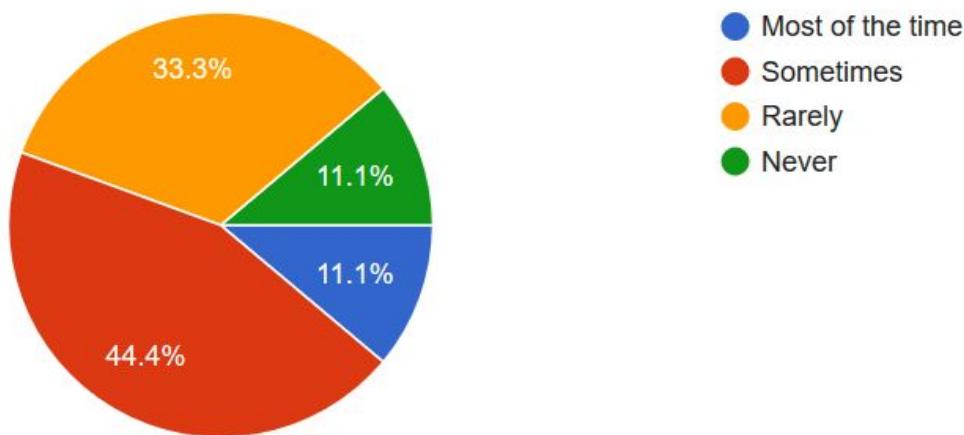
Survey Results

Preliminary results indicate that survey participants primarily represented researchers and librarians. About 83% of respondents said they possess advanced familiarity or expertise with geospatial data (Question 2), with about 17% stating that they have intermediate familiarity.

The following charts summarize responses regarding respondents' use of GeoJSON data (Questions 4 and 5):

How often do you CREATE data in GeoJSON format?

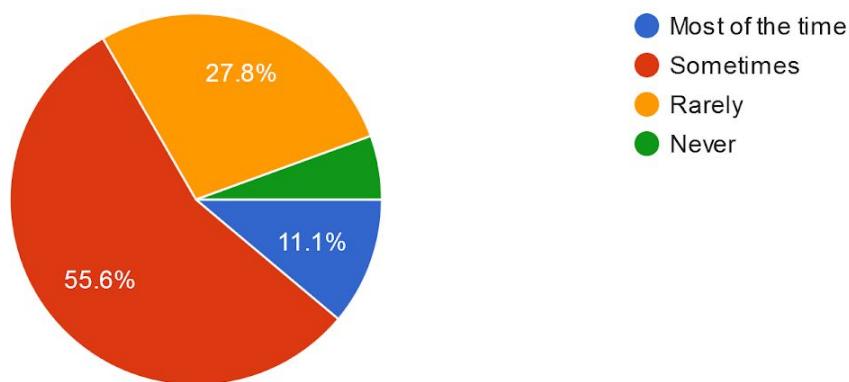
18 responses



<figure> <figcaption></figcaption> </figure>

How often do you (RE)USE data that is in GeoJSON format?

18 responses



<figure> <figcaption></figcaption> </figure>

In response to the question regarding which tools or platforms respondents use to create or use GeoJSON data (Question 6), 94.4% of respondents stated that they use commercial or open-source GIS software. The second most common response was scripting languages and packages such as R or python (66.7% of respondents), and browser-based tools were used by 55.6% of respondents.

A full 100% of respondents stated that, aside from GeoJSON, they regularly create or use shapefile data (Question 7), and 50% stated that they use XML-based GIS standard data such as GML or KML files.

The most common response to the question regarding what online resources respondents consult regarding GeoJSON data (Question 8) was "I do not typically consult online resources or documentation for GeoJSON," although 27.8% of respondents said they consult GeoJSON help pages on software guides and 27.8% of respondents said they consult the official GeoJSON documentation ([RFC 7946](#)).

Among the most useful responses to Question 9 on repositories and platforms where respondents search for GeoJSON data were the following statements:

"I don't search specifically for GeoJSON. I search for the data I need (for example, sidewalks in Oklahoma City), and once I find it, consider the format. Many government agencies rely on the ArcGIS platform, which has far more robust GeoJSON support in recent versions."

"When searching for vector data, I'm not usually searching for a specific vector format. Most repositories or platforms I frequent for my work (NED, GADM) do not provide GeoJSON. If I need GeoJSON, I'll convert it myself."

"I frequently see it as an option for exporting from the Socrata platform, which is the infrastructure behind many cities' open data portals."

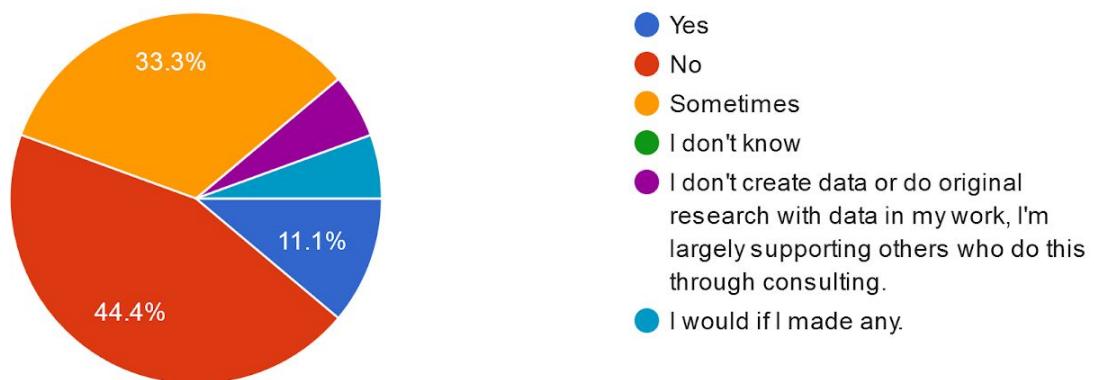
Other responses to Question 9 include: OpenIndexMaps, commercial vendors, Data.gov, GitHub, Geonames, Zenodo, Geoservers, city data portals, earthworks.stanford.edu, and data.humdata.org.

Among the most common responses to Question 10 on ways respondents share GeoJSON data were the following: OpenIndexMaps, geodatabase files, Zenodo, GitHub, static HTML pages, institutional repositories, cloud storage, and ArcGIS Online.

The following charts summarize responses regarding respondents' creation of metadata and sharing of GeoJSON files (Questions 11 and 12).

Do you create metadata files to accompany your GeoJSON data files?

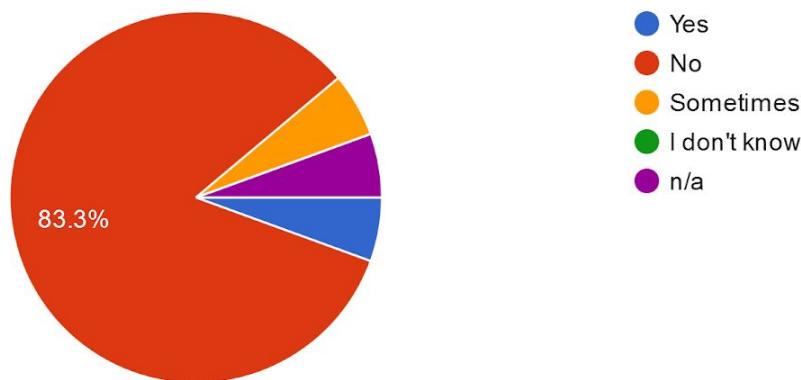
18 responses



<figure> <figcaption>Fig. 26 </figcaption> </figure>

If you store or share your geospatial data in a data repository, do you primarily submit it in GeoJSON format?

18 responses



<figure> <figcaption></figcaption> </figure>

The most frequent responses to Question 13 on other file formats respondents commonly use to submit geospatial data to repositories include: shapefiles, geodatabase, CSV, JSON, geotiff, and kmz/kml.

Bibliography

- Arora, S. K. (2018, February 3). A primer on GeoJSON standard and visualization tools. Retrieved June 17, 2019, from Medium website:
<https://medium.com/@sumit.arora/what-is-geojson-geojson-basics-visualize-geojson-open-geojson-using-qgis-open-geojson-3432039e336d>
- Esri. (n.d.). ArcGIS Online Reference: GeoJSON [Reference]. Retrieved from ArcGIS Online website:
<https://doc.arcgis.com/en/arcgis-online/reference/geojson.htm>
- Gillies, S. (n.d.). geojson: Python bindings and utilities for GeoJSON (Version 2.4.1) [Python, OS Independent]. Retrieved from <https://github.com/frewsxcv/python-geojson>
- Gillies, S. (2016, August 17). Re: [Geojson] metadata in GeoJSONgeojson. Retrieved from
<https://mailarchive.ietf.org/arch/browse/geojson/?q=metadata>
- Gillies, S., Butler, H., Daly, M., Doyle, A., & Schaub, T. (n.d.). The GeoJSON Format. Retrieved April 18, 2019, from <https://tools.ietf.org/html/rfc7946>
- Hanson, B. A., & Seeger, C. J. (2015). Validate GeoJSON: GeoJSONLint. *Extension and Outreach Publications*, 183, 3.
- Lampros, M. (n.d.). Processing of GeoJSON data in R. Retrieved April 18, 2019, from
https://cran.r-project.org/web/packages/geojsonR/vignettes/the_geojsonR_package.html
- Library of Congress. (2014). GeoJSON, Version 1.0 [Web page]. Retrieved April 18, 2019, from
<https://www.loc.gov/preservation/digital/formats/fdd/fdd000382.shtml>
- MacWright, T. (2015). More than you ever wanted to know about GeoJSON. Retrieved August 9, 2019, from macwright.org website: <https://macwright.org/2015/03/23/geojson-second-bite.html>
- MacWright, T. (2019). *GeoJSON utilities that will make your life easier*. Retrieved from
<https://github.com/tmcw/awesome-geojson> (Original work published 2015)
- Mapbox. (n.d.). GeoJSON [Text]. Retrieved April 18, 2019, from Mapbox website:
<https://www.mapbox.com/help/glossary/geojson/>
- MapBox. (n.d.). geojson.io. Retrieved April 18, 2019, from geojson.io website: <https://geojson.io/>
- MongoDB. (n.d.). GeoJSON Objects [Text]. Retrieved April 18, 2019, from
<https://github.com/mongodb/docs/blob/v4.0/source/reference/geojson.txt> website:
<https://docs.mongodb.com/manual/reference/geojson>
- SpatiaLite. (n.d.). SpatiaLite: Supporting GeoJSON. Retrieved April 18, 2019, from
<https://www.gaia-gis.it/fossil/libspatialite/wiki?name=Supporting+GeoJSON>
- Switch from Shapefile. (n.d.). Retrieved from Switch from Shapefile website:
<http://switchfromshapefile.org/#alternatives>
- Wikipedia. (2019). GeoJSON. In *Wikipedia*. Retrieved from
<https://en.wikipedia.org/w/index.php?title=GeoJSON&oldid=900774444>