GCDF 3.0 Administrative Unit Data Files

The administrative (ADM) unit data files provide a top down look at the location of GCDF 3.0 projects. Each ADM file - for ADM1 (typically state or region) and ADM2 (typically district) - links every GCDF 3.0 project that has geospatial information to one or more ADM units that it intersects with at the corresponding ADM level.

Information on how much of the project area overlaps with each intersecting ADM unit as well as how many ADM units the project overlaps with in total are provided to support a range of different types of analysis. The ADM data files can easily be joined to the main GCDF 3.0 project data based on the record IDs to access the full range of attributes and variables available with the GCDF 3.0 dataset.

The ADM level data is a valuable tool for researchers and analysts who are interested in trends at the ADM level, and wish to avoid working with the raw geospatial data on exact, individual project locations. The ADM data can be leveraged for many types of analysis without using GIS or related technical skills, but can also be joined with the GIS features for individual ADM units for more complex spatial analysis, or make use of the included ADM unit centroids to perform basic spatial operations.

File Structure

The following file structure applies to the tabular CSV files for both ADM1 and ADM2 level data.

Each row will contain a unique combination of a project (represented by `id` field) and an ADM unit (represented by `shapeID` field). Projects and ADM units can appear in multiple rows, but the same project and ADM unit will only appear together once.

Column	Description
id	GCDF 3.0 Record ID. Can be used to join data from the main GCDF 3.0 record/activity level dataset. Multiple rows can have the same id when a project spans across multiple ADM units.
shapeID	ID that is unique to the administrative unit associated with the

	row. Multiple rows can have the same shapeID when different projects are in the same ADM unit. Originates from the geoBoundaries data.
shapeGroup	The ISO3 code which identifies the country of the ADM unit for the row. Originates from the geoBoundaries data.
shapeName	The name associated with the ADM unit for the row. These may be blank or an undefined value. Originates from the geoBoundaries data.
intersection_ratio	The percentage of the project area which is within the ADM unit for the row. Expressed as a value between 0 and 1.
even_split_ratio	Ratio based on the total number of ADM units the project intersects with. If a project spans 4 ADM units, the value will be 1 / 4 = 0.25
intersection_ratio_commitment_value	The proportion of the project's commitment value associated with the ADM unit based on the intersection ratio. Equation = intersection_ratio * project commitment value (from main GCDF 3.0 data). Value may be zero if the project commitment value was zero.
even_split_ratio_commitment_value	The proportion of the project's commitment value associated with the ADM unit based on the even split ratio. Equation = even_split_ratio * project commitment value (from main GCDF 3.0 data). Value may be zero if the project commitment value was zero.
centroid_longitude	The longitude of the centroid of the ADM unit for the row
centroid_latitude	The latitude of the centroid of the ADM unit for the row

Methodology

The below methodology applies to both the ADM1 and ADM2 data generation process. The associated Python code which implements this approach can be found on GitHub.

1. Load the GCDF 3.0 project level data and associated geospatial features
2. Download the ADM1 and ADM2 Comprehensive Global Administrative Zones (CGAZ) data from geoBoundaries, version 6. (Permanent link to data here.)
3. For each GCDF 3.0 project containing geospatial features, we identify all ADM units which the project intersects with.
4. For each project, count the total number of ADM units it intersects with to calculate the even split ratio.
5. Expand the project level dataframe containing a list of intersecting ADM units to produce a new dataframe containing a unique row for each project / ADM unit intersection.
6. For each row (an ADM unit a project intersects with) calculate the intersection ratio by dividing the area of the intersection by the total project area.
7. Drop rows where projects intersection is less than 0.01%
8. Calculate the centroid of the ADM unit for each row.
9. Calculate the intersection ratio based commitment value and the even split based commitment value for each row.

Caveats

1. Inconsistent ADM boundaries

Frequency: common

Description: ADM unit boundary definitions often differ in varying amounts between sources of boundary data. The boundary data we use to define ADM units when creating the ADM data files is based on geoBoundaries, while the boundary data used during creation of the Geospatial GCDF Dataset is based on features within OpenStreetMap (OSM). The differences between geoBoundaries and OSM become slightly problematic when projects geocoded to the ADM unit (vs a specific building or road) are then aggregated for our ADM data files based on an alternative boundary definition.

Impact: The difference between boundary definitions, even when very small, results in irregularities during the aggregation process for the ADM data files. Irregularities typically occur in the form of very small percentages of project overlap with ADM units, due to minor differences in the edge of boundaries. We address the vast majority of these cases by thresholding the percentage of overlap we keep, but some still remain. Users can apply a more strict threshold (i.e., filtering out intersection_ratio field values less than the desired value) to produce more conservative ADM unit aggregations). If you find other sources of error in the location of projects and the expected ADM units they exist within, please report them to us.

2. Mislabeled / missing / disputed ADM unit names

Frequency: common

Description: While all ADM units have a unique identifier that should be used for joins or analysis that require precise records, we also include ADM unit names as provided with the boundary source data (geoBoundaries). These names may be missing, incomplete, or mislabeled for varying reasons and are not intended as an authoritative definition of ADM unit names. Issues with ADM unit names may stem from a lack of available data, changes to names since the boundary data was collected, or disputed territories which are known by differing names based on the country claiming the territory.

Impact: Expected names of ADM units may not match user expectations. This is most commonly an issue when users attempt to join other data based on ADM unit names alone. Using ADM unit names for joins is frequently difficult due to variable spellings and even varying official/unofficial names for some ADM units. Clearly wrong names or other issues can be reported directly to the geoBoundaries team.

3. Contested / disputed territories

Frequency: limited

Description: ADM unit boundaries may not be accurate in areas where ownership and definition of boundaries are contested between two nations. In such cases, the underlying source of data used for boundary information in the geoBoundaries data relies on a combination of U.S. Department of State boundary definitions, and isolation of disputed territories as distinct areas.

Impact: Boundaries of ADM units may not align with the user's preferred definitions. While we recognize that may not be ideal, the ADM data files we provide are intended as a general "ease of use" product for users, and cannot support all edge cases. In cases where users wish to utilize alternative boundary definitions, the source code used to produce the ADM data files is publicly available, and does not require unusual technical expertise or computational resources to run. If you believe boundaries are objectively wrong, issues can be reported directly to the geoBoundaries team.

4. No ADM unit intersects with a project

Frequency: rare

Description: A project's geospatial features may exist outside of any ADM units under certain conditions when either the expected ADM unit for that location was ill-defined or does not

exist. Ill-defined boundaries may be the result of ADM unit boundaries that have been simplified for practical purposes (e.g., not every bend along a coastline can be reasonably mapped). Boundaries may not exist for a very small subset of islands or territories that have not been associated with a recognized country.
Impact: This may result in projects not being included in the ADM data files. Given the rarity of these cases, and that they typically can be corrected, if you encounter any of this type of issue, please report it to us and we will work with the geoBoundaries team to debug and correct the issue.

Using the Data

Loading the data

Each individual ADM data file (ADM1 and ADM2) uses a flat, tabular, CSV file format that can easily be opened with any spreadsheet tools such as Excel or be loaded into any programming language environment (e.g., Python, R, Stata). While these are common default settings, make sure when loading the data you have selected options for comma delimited files, which include column headers as the first row.

Merging with GCDF 3.0 project data

Data in the ADM files can be joined with the primary project level data for the GCDF 3.0 dataset. Joining ADM data with project data will allow you to explore trends in project sectors and other project level attributes. The data can be joined on the "id" field from the ADM files, and the "AidData Record ID" field from the project data.

Merging with geoBoundaries

If you wish to perform spatial analysis leveraging the ADM unit boundaries associated with the ADM data files, the raw boundary data from geoBoundaries can be downloaded and joined.

The boundary data used can be downloaded from the geoBoundaries GitHub repository: ADM1 boundary data in GeoPackage format, ADM2 boundary data in GeoPackage format, other file formats for both ADM1 and ADM2 boundary data. Once downloaded, you can load the geospatial boundary data using free GIS software such as QGIS or a variety of open source packages for programming languages such as GeoPandas for Python.

The boundary data and GCDF3.0 ADM data can be joined using the “shapeID” field in both sources of data. Tutorials and examples of merging or joining data are available for QGIS and GeoPandas, and most other popular tools/packages.

Aggregating by ADM unit

A range of approaches exist for aggregating data by ADM units, and specifics may depend on your analysis and goals. In general, pivot tables are a very useful tool in most spreadsheet software and are very well documented with guides and tutorials online. Many approaches can be implemented for aggregation in programming languages and vary depending upon the language used. A common approach in Python is using the Pandas package and the “groupby” method.

Example and more

You can find examples related to using the GCDF 3.0 ADM files and other geospatial data in the examples folder of our GitHub repository.

Reporting Errors

If you find an error relating to the accuracy of a specific boundary, please contact the geoBoundaries team directly via their GitHub Issues page.

For all other issues, you can email us at china@aiddata.org and your inquiry will be directed to a relevant team member.