Reproducibility review of: A method to produce metadata describing and assessing the quality of spatial landmark datasets in mountain area

Philipp A. Friese

2022-06-11

This report is part of the reproducibility review at the AGILE conference. For more information see https://reproducible-agile.github.io/. This document is published on OSF at OSF https://osf.io/6s2gp/. To cite the report use

> Friese, Philipp A. (2022, May). Reproducibility review of: A method to produce metadata describing and assessing the quality of spatial landmark datasets in mountain area. https://doi.org/OSF.IO/6S2GP

Reviewed paper

> Van Damme, M.-D., and Olteanu-Raimond, A.-M.: A method to produce metadata describing and assessing the quality of spatial landmark datasets in mountain area, AGILE GIScience Ser., 3, 17, https://doi.org/10.5194/agile-giss-3-17-2022

Summary

The software of the paper under reproduction is publicly available on GitHub. The data sets are publicly available in a Zenodo project. Out of the four Figures and four Tables, two Figures and two Tables are eligible for reproduction. Both eligible Figures have been successfully reproduced. Both eligible Tables have been partially reproduced. The reproduction of Table 3 and 4 was partial as the provided software scripts comprise reproduction of one out of the four data columns. The remaining columns are expected to be reproducible after adjusting the scripts with references to the remaining data sets. The authors showed concern and dedication to improve reproducibility of their work.

Reproduction was *partially successful*.

Reproducibility reviewer notes

The submission contains a Data and Software Availability section which links to a publicly accessible GitHub repository containing software and a publicly available Zenodo repository containing four data sets. The software is published under a CC0-1.0 licence, the data sets are published under a CC-BY license. The software repository comprises a Java project and several SQL scripts for loading and processing data. Initially, the linked software did not contain sufficient explanation for reproduction. After contacting the authors, a step-by-step instruction for reproducing relevant parts of the paper was added to the repository.

All files and images generated during reproduction are available in the OSF repository accompanying this report.

The paper contains two Figures (Figure 3 and 4) and two Tables (Table 3 and 4) eligible for reproduction. The authors provide two procedures to reproduce the required data for Table 3 and 4, and a third procedure to reproduce Figure 4. Figure 3 was reproducible using the linked datasets and QGIS 3.24. The result is displayed in Figure 1.

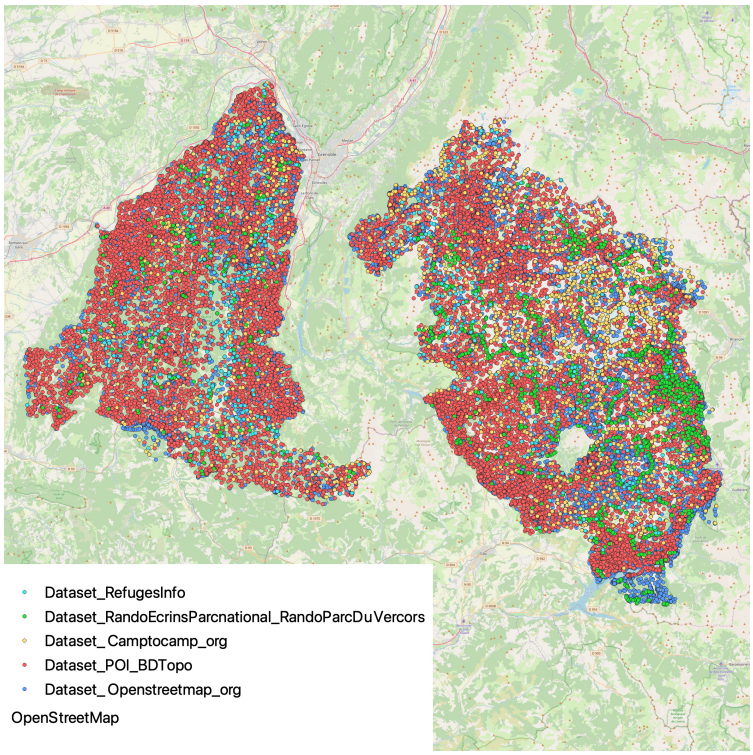

Figure 1: Five data sources for test area located in the French Alps - corresponds to Figure 3 in reproduced paper

Procedure 1

First, the necessary data sets have to be imported into a PostgreSQL database with the extension PostGIS. PostgreSQL version 14.2 and PostGIS version 3.2 was used. Then a total of 7 SQL scripts produce data, which partially have to be processed further in a spreadsheet software.

The README informs that:

> *All the steps described below concern the camptocamp.org data source. To get the results of the other data sources (OpenStreeetMap.org, Refuges.info, rando.ecrins-parcnational.fr and rando.parc-du-vercors.fr), it will be necessary to adapt the link of dataset to download and the table names in the SQL scripts.*

Therefore the provided scripts reproduce the first column "C2C-BDTOPO" of Table 3 and the column "C2C" of Table 4.

For scripts 4 and 7, the use of a spreadsheet software was necessary. LibreOffice has been used for the scope of this reproduction.

Script 0_loading_data

The authors provided the SQL script `0_loading_data.sql` for data import. This import script uses the command `copy <table> from <path/to/file>`, which requires the copied-from file to reside on the server.[1] During reproduction and after copying the dataset files to the server machine, this command failed to access the data set files due to permission issues. The command was altered to `\copy`, which fetches data from the client, and was executed within a `psql` terminal. This resolved the permission issue.

During data import, one of the CSV datasets failed to import due to a non-escaped comma. This issue has been resolved by the authors.

The path to the dataset files had to be changed in the import script, as per instruction.

Script 1_confidence

This script reproduces the row `1:1 DQ confidence` of Table 3 for the column `C2C-BDTOPO`. Executing the script yields three values 82, 98, and 94, which initially caused confusion on the association to Table 3. After contacting the authors, the origin of the three values was explained and the README updated:

> *Run the first request in the script SQL **sql/1_confidence.sql** to get the DQ_confidence for all the scope. Note: the two other scripts compute the DQ_confidence for a subset of the types. This is an example for on demand metadata; for example if the user needs to assess only the confidence of the matching algorithm for a specific types of landmarks (e.g. those corresponding the the ontology class "isolated accommodation")*

The value 82, returned by the first command in script `1_confidence.sql`, matches Table 3.

Script 2_spatial_accuracy

Executing script `2_spatial_accuracy.sql` yields a total of 7 values, which correspond to the rows "PositionalAccuracy" and column "C2C" of Table 4. The values match the paper and are listed in Table 1.

Table 1: Data for rows "PositionalAccuracy", column "C2C" of Table 4 in reproduced paper

scope	measurement	value
all	MeanAbsolute2D	47.045417
all	RootMeanSquareError	70.486361
all	AgreementRate threshold	0.491274
isolated accommodation	MeanAbsolute2D	17.865509
isolated accommodation	RootMeanSquareError	29.904149
landform	MeanAbsolute2D	49.058929
landform	RootMeanSquareError	71.079449

Script 3_confusion_matrix_all

The data produced by script `3_confusion_matrix_all.sql` ultimately reproduce rows "ThematicClassificationCorrectness". The script produces two outputs, which correspond to scope "all" and "isolated accommodation".

The README instructs to execute script `3_confusion_matrix_all.sql`, import the data into a spreadsheet software, and create a cross/pivot table of the data. To calculate the overall accuracy, first all values on the main diagonal of the confusion matrix have to be summed up. In the case of scope "all", the

[1] From the documentation at https://www.postgresql.org/docs/current/sql-copy.html: "Files named in a COPY command are read or written directly by the server, not by the client application. [...] They must be accessible to and readable or writable by the PostgreSQL user (the user ID the server runs as), not the client."

values of 20 additional cells have to be added. This process corresponds to summing the occurrence of correct matchings in the confusion matrix. The overall accuracy is then determined by dividing by the sum of all occurrences.

For the scope "all", the confusion matrix has a shape of 31x31 entries, for the scope "isolated accommodations" 2x2 entries, which match the paper.

Initially, the overall accuracy for scope "all" was calculated to be 68%, which deviates from the value of the paper by 7%. After contacting the authors, additional missing cells were added to the instructions. With the updated instructions, the overall accuracy for scope "all" was calculated to be 75%, which matches the paper.

For the scope "isolated accommodation" the overall accuracy was calculated to be 76%, which matches the paper.

Both the confusion matrices and resulting accuracy values are reproduced and stored in the zip archive `agile-reproreview-2022-015.zip` as file `3_confusion_matrix.ods` in the OSF repository accompanying this report.

Script 4_duplicate_all

Script `4_duplicate_all.sql` reproduces the row "CompletenessCommision Duplicate". It returns a total of 11 rows, which matches the paper.

Script 5_Samal_distance

Script `5_Samal_distance.sql` reproduces rows "NonQuantitativeAttributeAccuracy". The values match the paper and are listed in Table 2.

Table 2: Data for rows "NonQuantitativeAttributeAccuracy", column "C2C" of Table 4 in reproduced paper

scope	measurement	value
all	Mean Samal Distance	0.0482511
all	RootMeanSquareError Samal Distance	0.1347907
all	Agreement Rate Samal Distance	84.6422339
landform	Duplicate	0.0339907
landform	RootMeanSquareError Samal Distance	0.1099512

Script 6_missing_class

Script `6_missing_class.sql` reproduces row "Completeness Missing class nom". It returns the value 1, which matches the paper.

Script 7_completeness

The README instructs to execute script `7_completeness.sql`, import the data into a spreadsheet software and calculate two ratios of sums. The resulting values reproduce row "Completeness Excess" and "Completeness Missing items".

The calculated values are 0.39 and 0.935 respectively, which matches the paper.

The calculated values are reproduced and stored in the zip archive `agile-reproreview-2022-015.zip` as file `7_completeness.ods` in the OSF repository accompanying this report.

Conclusion

All values of Table 4, column "C2C" have been reproduced by the provided instructions and scripts. Additional columns are expected to be reproducible by modifying the involved script files above with references to the remaining data sources.

Prodecure 2

This procedure reproduces the remaining values of Table 3 by using the Java project QualityMetadataSpatialLandmarkDataset provided in the primary GitHub repository linked to in the paper. Java Temurin 1.8 with Maven 3.8.1 was used.

The provided Java project uses Maven for dependency management and building the project. During reproduction, Maven refused to fetch several dependencies listed in the file `pom.xml`, due to security considerations. These dependencies are given as an insecure `http://` link, which were changed to `https://`. This resolved the dependency issue.

In addition, the listed repository `https://repo.boundlessgeo.com/main` is no longer active and failed to resolve packages during reproduction.[2] The repository link was changed to `https://repo.osgeo.org/repository/release/`, according to instructions at osgeo.org/foundation-news/new-osgeo-repo.

During reproduction, contrary to the instructions the project MultiCriteriaMatching needed to be installed as well, as Maven could not resolve the project from the given repositories. The authors promised to resolve this issue in the future.

Installation of MultiCriteriaMatching required the same replacement of `http://` to `https://` links in file `pom.xml`.

Due to the project MultiCriteriaMatching not being fetched and installed by the main project QualityMetadataSpatialLandmarkDataset, it is presumed that the transitive dependency check from Maven did not locally resolve the dependencies of MultiCriteriaMatching. This resulted in several crashes due to unmet dependencies. A total of seven dependencies had to be added, which are listed below:

- org.apache.jena
- org.slf4j
- fr.ign.cogit
- commons-lang
- org.apache.lucene
- org.apache.commons
- com.ibm.icu

Running the compiled Java project crashed, because the file `Dataset_POI_ALT_NAME_BDTopo.csv` was not found. This file is not present in the Zenodo repository. Creating an empty file with the filename above in resolved this issue.

The program took about 8 minutes to finish execution and returned the following lines:

```
Nb feature c2c = 2289
Nb feature ign = 17769

NB non-app : 403
NB app : 1434
NB d'indécis : 123
NB sans candidat : 329
```

Line "`NB non-app : 403`" corresponds to row "1:0". Line "`NB app : 1434`" corresponds to row "1:1 total". Line "`NB d'indécis : 123`" corresponds to row "Uncertain". All values deviate by ±2. During contact, the authors explained that this deviation might originate from one of the used libraries "*evidence4j*":

> *I suppose it's coming from the version of the library evidence4j. I have, on my local machine, a different version that i can't publish on a maven repository [...].*

The program additionally generated the file `c2c-bdtopo--20220514-142927624.csv`. The last step in the README instructs to install a QGIS plugin VisuValideMultiCriteriaMatching, which involves downloading the repository as a `.zip` file and installing it in QGIS using the integrated plugin manager.

[2] See https://support.planet.com/hc/en-us/community/posts/360009782998-repo-boundlessgeo-com-is-unavailable- for more information.

After opening the plugin in QGIS and loading the file generated by the Java program, an interface for validating the matched data point is displayed. This interface is shown in Figure 2.

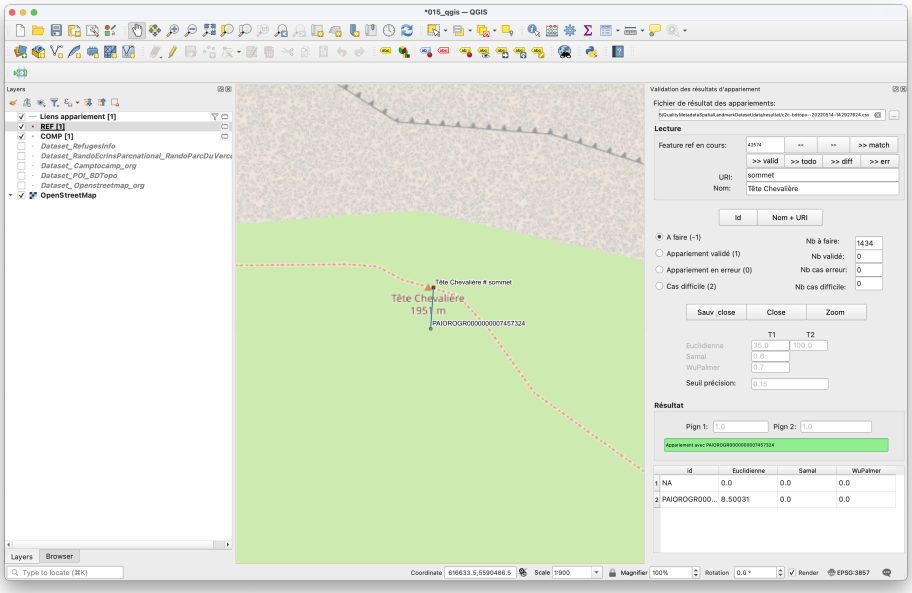

Figure 2: Interface of QGIS plugin VisuValideMultiCriteriaMatching

Full validation of the data set, which contains over 6.000 rows, was omitted for the scope of this reproduction.

Prodecure 3

Reproducing Figure 4 requires downloading and importing the result of data matching between data sources "Refuges.info" and "BDTOPO". After import, the README instructs to execute the SQL file `p3_1_boxplot_samal_distance.sql` and visualise the data with a snippet of R code given below. R version 3.6.3 was used.

```
x <- read.csv("/<path>/<to>/<output>.csv",header=T, sep=",")
boxplot(x, xlab="Refuges.info",
        ylab="Samal distance",
        main="Boxplot of Samal distance for names in Refuges.info source")
```

The result reproduces Figure 4 in the paper, which is displayed in Figure 3 in this report.

Comments to the authors

The process of reproduction was time-consuming and tedious. Without the help of the authors, this reproduction would not have been possible. Throughout the reproduction, the authors showed concern and dedication to improve reproducibility of their work. Outlined below are several recommendations to further improve reproducibility of this and future submissions.

Procedure 1 involved many manual steps which could have been performed automatically using e.g. a scripting language or another automation tool. Reproducing procedure 1 took at least one hour and reproduced one out of four data sets. If the steps outlined in procedure 1 where automated and easily adaptable to reproduce the other three data sets, then this paper would be "fully reproducible". *I strongly recommend simplifying and automating the data processing pipeline*, especially for future submissions.

In addition, procedure 1 involved deploying and populating a PostgreSQL database. If operating directly on the data files (in this case the .csv files) is not an option, for example because PostGIS is required, then it may be advisable to look into using Docker for managing the PostgreSQL+PostGIS instance for

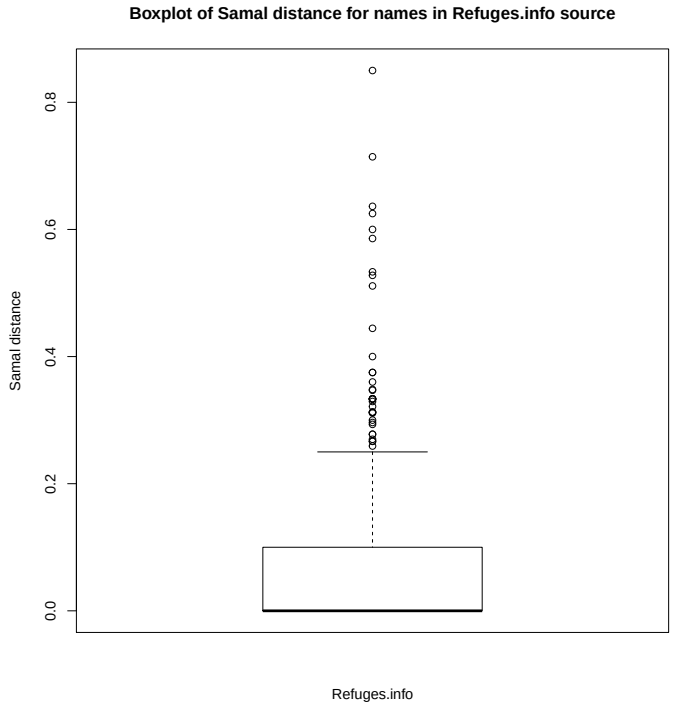

Figure 3: Boxplot of Samal distance for names in Refuges.info source - corresponds to Figure 4 in reproduced paper

future submissions. This would greatly simplify reproduction. *I recommend simplifying the data storage* for future submissions.

Procedure 2 required several modifications to the dependency management file and required the installation of a separate Java project. *I recommend testing the software pipeline on an unrelated computer, ideally by an independent party* (e.g. a colleague) prior to submission to check for these issues.