

Hands on Data Cleaning of Messy Data

June 28, 2018

Motivation of the Lab

- ▶ to present some of the data quality issues in real-world datasets.
- ▶ to explain how one can improve quality of such messy data.
- ▶ to offer a hands on practice for data cleaning.

A Real-World Scenario: The rodents dataset

- ▶ data on rodents during a survey (over 25 years).
- ▶ each row denotes the information collected on an individual rodent.
- ▶ useful for studying population dynamics and species interactions.
- ▶ useful links:
 - ▶ the original data is provided by Ernest et. al. [?].
 - ▶ meta-data is also available at http://esapubs.org/archive/ecol/E090/118/Portal_rodent_metadata.htm.

Data Quality

- ▶ **inconsistencies:**

- ▶ domain violation: a date entry of 4.31.2000, which is an invalid value (the intention might be 5.01.2000).

- ▶ **missing data:** empty cells (some of them!), -99, etc.

- ▶ **format variabilities**¹:

- ▶ typos: “Amph**e**spiza bilineata” for “Amph**i**spiza bilineata”.
- ▶ record linkage: “UNITED STATES” and “United States of America”.
- ▶ abbreviations: “US”.
- ▶ leading and trailing whitespace: “ Amphisipiza bilineata” and “ Amphisipiza bilineata ”.

¹several columns added for teaching purposes (see

<http://www.datacarpentry.org/OpenRefine-ecology-lesson/01-working-with-openrefine/>).

Data Quality

- ▶ more issues for you to explore.
- ▶ repetitive tasks taking lots of time.
- ▶ various tools that helps to transform such data: Trifacta [?], OpenRefine [?], etc.

Cleaning with OpenRefine

- ▶ a tool for working with messy datasets.
- ▶ see [?] for details.
- ▶ useful links:
 - ▶ the software at <http://openrefine.org>.
 - ▶ the documentation at <https://github.com/OpenRefine/OpenRefine/wiki/Documentation-For-Users>.
- ▶ we will now install OpenRefine and describe its features for various data cleaning tasks.

Installation

- ▶ **long answer:**

- ▶ detailed installation instructions at <http://openrefine.org/download.html>.

- ▶ **short answer:**

- ▶ download the file depending on the OS at <https://github.com/OpenRefine/OpenRefine/releases/tag/2.8>.
- ▶ install OpenRefine as follows:
 - ▶ Linux: extract.
 - ▶ Mac: open, drag icon into the Applications folder.
 - ▶ Windows: unzip.

Running OpenRefine

- ▶ **run OpenRefine** depending on the operating system:
 - ▶ Linux: `./refine` in your installation folder
 - ▶ Mac: OpenRefine in your Applications folder
 - ▶ Windows: `.exe` file in your installation folder

Loading Data

- ▶ **get the dataset:**

- ▶ clone the git repository at <https://github.com/tahaceritli/acm-summer-school-2018>.
- ▶ use the file at `datasets/Portal_rodents_19772002_scinameUUIDs.csv`.
- ▶ the corresponding meta-data at `meta-data/E090-118-D1-Rodent metadata.htm`.

- ▶ **import the data:**

- ▶ click “Create Project”.
- ▶ click “Choose Files”.
- ▶ select `Portal_rodents_19772002_scinameUUIDs.csv`.
- ▶ click “Next”.

Data Preview

- ▶ configuration page for importing.
- ▶ a subset of the data is shown.
- ▶ use the defaults.
- ▶ click “**Create Project**” on the top-right corner.

View Range of Values

- ▶ click the drop-down arrow in the “scientificName” column.
- ▶ select “Facet>Text Facet”.
- ▶ lists the values and their counts.
- ▶ any problems with the data?

Updating Cell Values

- ▶ notice the spelling errors, e.g. “Amph**e**spiza bilineata” for “Amph**i**spiza bilineata”.
- ▶ hover over the former and select “**e**dit” to update its value.

Trimming Whitespace

- ▶ leading and trailing whitespace: “ Amphisiza bilineata” and “ Amphisiza bilineata ”.
- ▶ click the drop-down arrow in the “scientificName” column.
- ▶ select “Edit cells>Common transforms>Trim leading and trailing whitespace”.

Clustering

- ▶ data often contains more complex inconsistencies due to data collection procedures.
- ▶ “Clustering” helps to find cells in a column, that refers to the same entity with different values.
- ▶ various methods to determine clusters.

Clustering

- ▶ click the drop-down arrow in the “scientificName” column.
- ▶ select “Edit cells>Cluster and edit...”.
- ▶ change the method to nearest neighbor.
- ▶ you can now check boxes and merge the clusters.

Reconciliation

- ▶ external knowledge to clean messy data.
- ▶ reconciliation refers to the process of matching data to databases.
- ▶ popular knowledge bases:
 - ▶ Wikidata at https://www.wikidata.org/wiki/Wikidata:Main_Page.
 - ▶ Google Knowledge Graph at <https://www.google.com/intl/es419/insidesearch/features/search/knowledge.html>.
- ▶ OpenRefine uses Wikidata. details at <https://github.com/OpenRefine/OpenRefine/wiki/Reconciliation>.
- ▶ note that Google Knowledge Graph comes with an API to query <https://developers.google.com/knowledge-graph/>.

Reconciliation with OpenRefine

- ▶ click the drop-down arrow in the “state” column.
- ▶ select “**Reconcile**>**Start reconciling...**”.
- ▶ select “**Wikidata**” and click “**Start Reconciling**”.
- ▶ can take up to 1-2 minute to complete.
- ▶ lists the matches with their confidence score.
- ▶ two matching options:
 - ▶ with one tick to update the given cell only.
 - ▶ with two ticks to update all the cells with the same value.

Filtering Rows

- ▶ click the drop-down arrow in the “scientificName” column.
- ▶ select “Text Filter”.
- ▶ type “bai”, which lists the first 10 matching rows.
- ▶ searching rows based on multiple columns:
 - ▶ click the drop-down arrow in the “scientificName” column.
 - ▶ select “Facet>Custom text facet...”.
 - ▶ using General Refine Expression Language (GREL).
 - ▶ `and(cells[“scientificName”].value == “Amphispiza bilineata”, cells[“country”].value == “AUSTRALIA”)`.

Deliverables

1. **inconsistencies:**

- ▶ report the number of rows that match 4, 31, 2000 in mo, dy, yr columns respectively.

2. **format variabilities:**

- ▶ report how many unique values exist in the “country” column of the original dataset.
- ▶ is there any problems with the data? if so, explain how did you solve them?
- ▶ report how many unique values you have after fixing the potential inconsistencies.
- ▶ hint: “clustering” or “reconciliation” feature of OpenRefine could be useful for this task.

3. missing data:

- ▶ for the columns stake, species, county and nestdir:
 - ▶ report the columns that have missing data and the encodings used to denote missing entries.
 - ▶ check whether they are actually missing (hint: you may want to check meta-data!).
 - ▶ hint: commonly used encodings: "NA", "N/A", "Null", "-1", "-99", etc.

Deliverables - Bonus

- ▶ Traffic Violations dataset at <https://catalog.data.gov/dataset/traffic-violations-56dda>.
- ▶ import the subset of the original data given in `datasets/traffic_violations_subset.csv`.
- ▶ meta-data at `meta-data/traffic_violations.json`.
- ▶ look at the columns named “Make” and “Driver City”.
- ▶ report the data quality issues.
- ▶ use OpenRefine to clean the data and explain which feature you used.

Submissions

- ▶ 1 page of either .txt or .pdf (not a .doc file!).
- ▶ answer the four questions mentioned earlier.
- ▶ send your report to `acm2018datacleaninglab@gmail.com` with the title `ACMReport-NAME-SURNAME`.
- ▶ attach the exported file at step 2 to the email.

References I

- [1] M. Ernest, J. Brown, T. Valone, and E. P. White, “Portal Project Teaching Database,” https://figshare.com/articles/Portal_Project_Teaching_Database/1314459 [Accessed on 27/06/2018].
- [2] “Trifacta.” <https://www.trifacta.com/> [Accessed on 27/06/2018].
- [3] R. Verborgh and M. De Wilde, *Using OpenRefine*. Packt Publishing Ltd, 2013.