OpenRefine for Ecology Data

Motivations for the OpenRefine Lesson

1. Data is often very messy and it has to be refined before it is useful. OpenRefine provides a set of tools to allow you to identify and amend the messy data.
2. It is important to know what you did to your data. Additionally, journals, granting agencies, and other institutions are requiring documentation of the steps you took when working with your data.
3. With OpenRefine, you can capture all actions applied to your raw data and share them with your publication as supplemental material.
4. All actions are easily reversed in OpenRefine.
5. If you save your work, it will be to a new file. OpenRefine always uses a copy of your data and does not modify your original dataset.
6. Data cleaning steps often need repeating with multiple files. OpenRefine keeps track of all of your actions and allows them to be applied to different datasets.
7. Some concepts such as clustering algorithms are quite complex, but OpenRefine makes it easy to introduce them, use them, and show their power.
8. Open source.
9. A large growing community, from novice to expert, ready to help.
10. Works with large-ish datasets (100,000 rows). Can adjust memory allocation to accommodate larger datasets.
1. Most importantly, it is not a web service. This is a Java program that runs on your machine (not in the cloud). It runs inside your browser, but no web connection is needed.
11. In other words, you don't have to upload data to use OpenRefine. It is a Desktop application that runs on your computer, even though you interact with it through your web browser. Sensitive data thus never have to leave your computer to get cleaned up.

Launch OpenRefine

If after installation and running OpenRefine, it does not automatically open for you, point your browser at http://127.0.0.1:3333/ or http://localhost:3333 to launch the program.

Getting help for OpenRefine

1. OpenRefine website: check out some great introductory videos.
2. Introductory videos on the website and other videos on OpenRefine can also be found on YouTube.
3. Google Group: answer a lot of beginner questions and problems.
4. OpenRefine Google Plus community: can find a lot of help.
5. OpenRefine libraries are available too, where you can find a script you need and copy it into your OpenRefine instance to run it on your dataset.

Creating a new OpenRefine project

1. Windows: double-click on the openrefine.exe file. Java services will start automatically on your machine, and OpenRefine will open in your browser.
2. Mac: OpenRefine can be launched from your Applications folder.
3. Linux: navigate to your OpenRefine directory in the command line and run ./refine.

Working with OpenRefine

1. OpenRefine can import a variety of file types, including tab separated (tsv), comma separated (csv), Excel (xls , xlsx), JSON, XML, RDF as XML, Google Spreadsheets.
2. We will be using a modified version of the portal rodent data set where several columns have been added: https://ndownloader.figshare.com/files/7823341
3. Once OpenRefine is launched in your browser, the left margin has options to Create Project , Open Project , or Import Project . Here we will create a new project:

- Create Project > Get data from This Computer .
- Choose Files > select the file Portal_rodents_19772002_scinameUUIDs.csv > Open or double-click on the filename.
- Next>>.
- OpenRefine gives you a preview - a chance to show you it understood the file. You can choose the correct separator in the box shown and click Update Preview (bottom left). If this is the wrong file, click <<Start Over (upper left).
- There are also options to indicate whether the dataset has column headers included and whether OpenRefine should skip a number of rows before reading the data.
- If all looks well, click Create Project>> (upper right).

Using Facets

1. Explore data by applying multiple filters.
2. Facets are one of the most useful features of OpenRefine and can help:
    a. get an overview of the data in a project (i.e. seeing a big picture of your data);
    b. bring more consistency to the data;
    c. you can filter data down to just the subset of rows that you want to change in bulk.
3. A 'Facet' groups all the like values that appear in a column, and then allow you to filter the data by these values and edit values across many records at the same time.
4. One type of Facet is called a 'Text facet'. This groups all the identical text values in a column and lists each value with the number of records it appears in.

- Scroll to the scientificName column, click the down arrow and choose Facet > Text facet .
- In the left panel, you'll now see a box containing every unique value in the scientificName column with a number representing how many times that value occurs in the column.
- Try sorting this facet by name and by count.
- Do you notice any problems with the data? What are they?
    - Several near-identical entries due to misspellings.
- Hover the mouse over one of the names in the Facet list. You should see that you have an edit function available.
- You could use this to fix an error immediately, and OpenRefine will ask whether you want to make the same correction to every value it finds like that one. But OpenRefine offers even better ways (clustering) to find and fix these errors, which we'll use instead.

More on Facets

1. OpenRefine Wiki: Faceting (https://github.com/OpenRefine/OpenRefine/wiki/Faceting)
2. 'Text facets' Refine also supports a range of other types of facet. These include:
    a. Numeric facets
    b. Timeline facets (for dates)
    c. Custom facets
    d. Scatterplot facets
3. Numeric and Scatterplot facets display graphs instead of lists of values. The numeric facet graph includes 'drag and drop' controls you can use to set a start and end range to filter the data displayed.
4. Custom facets are a range of different types of facets. Some of the default custom facets are:
    a. Word facet - this breaks down text into words and counts the number of records each word appears in.
    b. Duplicates facet - this results in a binary facet of 'true' or 'false'. Rows appear in the 'true' facet if the value in the selected column is an exact match for a value in the same column in another row.
    c. Text length facet - creates a numeric facet based on the length (number of characters) of the text in each row for the selected column. This can be useful for spotting incorrect or unusual data in a field where specific lengths are expected (e.g. if the values are expected to be years, any row with a text length more than 4 for that column is likely to be incorrect).
    d. Facet by blank - a binary facet of 'true' or 'false'. Rows appear in the 'true' facet if they have no data present in that column. This is useful when looking for rows missing key data.

Exercise

1. Using faceting, find out how many years are represented in the survey results.
2. Is the column formatted as Number, Text or Date?
3. How does changing the format change the faceting display?
4. Which years have the most and least observations?

Solution

1. For the column yr do Facet > Text facet . A box will appear in the left panel showing that there are 26 unique entries in this column.
2. By default, the column yr is formatted as Text.
3. You can change the format by doing Edit cells > Common transforms > To Number. Notice the values in the column turn green. Doing Facet > Numeric facet creates a box in the left panel that shows a histogram of the number of entries per year. If you instead transform the column to a date, the program will assume all entries are on January 1st of the year.
4. Click Sort by count in the facet box. The year with the most observations is 1997. The least is 1977.

Clustering

1. Clustering means “finding groups of different values that might be alternative representations of the same thing”. Example: New York and new york – capitalization differences.
2. Clustering is a very powerful tool for cleaning datasets which contain misspelled or mistyped entries.

- In the scientificName Text Facet we created in the step above, click the Cluster button.
- In the resulting pop-up window, you can change the Method and the Keying Function. Try different combinations to see what different mergers of values are suggested.
- Select the key collision method and metaphone3 keying function. It should identify three clusters.
- Click the Merge? box beside each cluster, then click Merge Selected and Recluster to apply the corrections to the dataset.
- Try selecting different Methods and Keying Functions again, to see what new merges are suggested.
- You may find there are still improvements that can be made, but don’t Merge again; just Close when you’re done. We’ll now see other operations that will help us detect and correct the remaining problems, and that have other, more general uses.
- Important: If you Merge using a different method or keying function, or more times than described in the instructions above, your solutions for later exercises will not be the same as shown in those exercise solutions.

Different clustering algorithms

The technical details of how the different clustering algorithm work can be found at the link below:
https://github.com/OpenRefine/OpenRefine/wiki/Clustering-In-Depth

Split

If data in a column needs to be split into multiple columns, and the parts are separated by a common separator (say a comma, or a space), you can use that separator to divide up the pieces into their own columns.

- Click on the down arrow at the top of the scientificName column. Choose Edit Column > Split into several columns…
- In the pop-up, in the Separator box, replace the comma with a space.
- Uncheck the box that says Remove this column.
- Click OK. You'll get some new columns called scientificName 1, scientificName 2, and so on.
- Notice that in some cases scientificName 1 and scientificName 2 are empty. Why is this? What do you think we can do to fix this?
    - Leading and trailing white space > very difficult to notice when cleaning data manually

Exercise

1. Try to change the name of the second new column to "species". How can you correct the problem you encounter?

Solution

1. On the scientificName 2 column, click the down arrow and then Edit column > Rename this column. Type "species" into the box that appears. A pop-up will appear that says Another column already named species. This is because there is another column where we've recorded the species abbreviation. You can choose another name like speciesName for this column or change the other species column name to speciesAbbreviation.

Transforming data

- Click the down arrow at the top of the JSON column. Choose Edit Cells 1. > Transform...
- This will open up a window into which you can type a GREL expression. GREL stands for General Refine Expression Language.
- If you want to perform "find and replace" operations, type the following expression in the Expression box: value.replace("{", "") and click OK .
- What the expression means is this: Take the value in each cell in the selected column and replace all of the "{" with "" (i.e. nothing - delete).
- Click OK. You should see in the JSON column that there are no longer any left curly brackets.
- Replace statements can be combined to perform all transformation steps simultaneously. The command is: value.replace("{", "").replace("}", "").replace("[", "").replace("]", "").
- To reuse a GREL command, click the History tab and then click Reuse next to the command you would like to apply to that column.

Using undo and redo

1. It's common while exploring and cleaning a dataset to discover after you've made a change that you really should have done something else first.
2. OpenRefine provides Undo and Redo operations to make this easy.

Exercise

1. Click where it says Undo / Redo on the left side of the screen. All the changes you have made so far are listed here.
2. Click on the step that you want to go back to, in this case go back several steps to before you had done any text transformation.
3. Notice that you can still click on the later steps to Redo the actions. Leave the dataset in the state in which the scientificNames were clustered, but not yet split.

Trim Leading and Trailing Whitespace

1. Words with spaces at the beginning or end are particularly hard for we humans to tell from strings without, but the blank characters will make a difference to the computer.
2. We usually want to remove these.
3. OpenRefine provides a tool to remove blank characters from the beginning and end of any entries that have them.

- In the header for the column scientificName, choose Edit cells > Common transforms > Trim leading and trailing whitespace.
- Notice that the Split step has now disappeared from the Undo / Redo pane on the left and is replaced with a Text transform on 3 cells
- Perform the same Split operation on scientificName that you undid earlier. This time you should only get two new columns. Why?
    a. Removing the leading white spaces means that each entry in this column has exactly one space (between the genus and species names). Therefore, when you split with space as the separator, you will get only two columns.
    b. Important: Undo the splitting step before moving on to the next lesson.

Filtering

1. There are many entries in our data table. We can filter it to work on a subset of the data in the list for the next set of operations.

- Click the down arrow next to scientificName > Text filter. A scientificName facet will appear on the left margin.
- Type in bai and press return. There are 48 matching rows of the original 35549 rows (and these rows are selected for the subsequent steps).
- At the top, change the view to Show 50 rows. This way you will see all the matching rows.

Exercise

1. What scientific names (genus and species) are selected by this procedure?
2. How would you restrict this to only one of the species selected?

Solution

1. Do Facet > Text facet on the scientificName column after filtering. This will show that two names match your filter criteria. They are Baiomys taylori and Chaetodipus baileyi.
2. To restrict to only one of these two species, you could make the search case sensitive or you could split the scientificName column into species and genus before filtering or you could include more letters in your filter.

Excluding entries

1. In addition to the simple text filtering we used above, another way to narrow our filter is to include and/or exclude entries in a facet.
2. You will see the include or exclude options if you hover over the name in the facet window.
3. If you still have your facet for scientificName, you can use it, or use drop-down menu > Facet >Text facet to create a new facet. Only the entries with names that agree with your Text filter will be included in this facet.
4. Faceting and filtering look very similar. A good distinction is that faceting gives you an overview description of all of the data that is currently selected, while filtering allows you to select a subset of your data for analysis.

Exercise

1. Use include / exclude to select only entries from one of these two species.

Solution

1. In the facet (left margin), click on one of the names, such as Baiomys taylori. Notice that when you click on the name, or hover over it, there are entries to the right for edit and include.
2. Click include. This will explicitly include this species, and exclude others that are not explicitly included. Notice that the option now changes to exclude.
3. Click include and exclude on the other species (Chaetodipus baileyi) and notice how the two entries appear and disappear from the table.

Sort

1. You can sort the data by a column by using the drop-down menu in that column.
2. There you can sort by text , numbers , dates or booleans (TRUE or FALSE values).
3. You can also specify what order to put Blanks and Errors in the sorted results.
4. If this is your first time sorting this table, then the drop-down menu for the selected column shows Sort... . Select what you would like to sort by (such as numbers). Additional options will then appear for you to fine-tune your sorting.

Exercise

1. Sort the data by plot. What year(s) were observations recorded for plot 1 in this filtered dataset.

Solution

1. In the plot column, select Sort... > numbers and select smallest first. The years represented are 1990 and 1995.

Sorting by multiple columns

1. You can sort by multiple columns by performing sort on additional columns.
2. The sort will depend on the order in which you select columns to sort.
3. To restart the sorting process with a particular column, check the sort by this column alone box in the Sort pop-up menu.
4. If you go back to one of the already sorted columns and select > Sort > Remove sort , that column is removed from your multiple sort. If it is the only column sorted, then data reverts to its original order.

Exercise

1. You might like to look for trends in your data by month of collection across years.
2. How do you sort your data by month?
3. How would you do this differently if you were instead trying to see all of your entries in chronological order?

Solution

1. For the mo column, click on Sort... and then numbers. This will group all entries made in, for example, January, together, regardless of the year that entry was collected.
2. For the yr column, click on Sort > Sort... > numbers and select sort by this column alone. This will undo the sorting by month step. Once you've sorted by yr you can then apply another sorting step to sort by month within year. To do this for the mo column, click on Sort > numbers but do not select sort by this column alone. To ensure that all entries are shown chronologically, you will need to add a third sorting step by day within month.

Numbers

1. When a table is imported into OpenRefine, all columns are treated as having text values.
2. We saw earlier how we can sort column values as numbers, but this does not change the cells in a column from text to numbers. Rather, this interprets the values as numbers for the purposes of sorting but keeps the underlying data type as is.
3. We can, however, transform columns to other data types (e.g. number or date) using the Edit cells > Common transforms feature. Here we will experiment changing columns to numbers and see what additional capabilities that grants us.
4. Be sure to remove any Text filter facets you have enabled from the left panel so that we can examine our whole dataset. You can remove an existing facet by clicking the x in the upper left of that facet window.
5. To transform cells in the recordID column to numbers, click the down arrow for that column, then Edit cells > Common transforms… > To number. Depending on the version of OpenRefine, the recordID values might change from left-justified to right-justified, and from black to green color.

Exercise

1. Transform three more columns, including period, from text to numbers. Can all columns be transformed to numbers?

Solution

1. Only observations that include only numerals (0-9) can be transformed to numbers. If you apply a number transformation to a column that doesn't meet this criteria, and then click the Undo / Redo tab, you will see a step that starts with Text transform on 0 cells. This means that the data in that column was not transformed.

Examining Numbers in OpenRefine: Numeric facet

Sometimes there are non-number values or blanks in a column which may represent errors in data entry and we want to find them. We can do that with a Numeric facet.

Exercise

1. For a column you transformed to numbers, edit one or two cells, replacing the numbers with text (such as abc) or blank (no number or text).
2. Apply a numeric facet to the column you edited. The facet will appear in the left panel.
3. Notice that there are several checkboxes in this facet: Numeric, Non-numeric, Blank, and Error. Below these are counts of the number of cells in each category. You should see checks for Non-numeric and Blank if you changed some values.
4. Experiment with checking or unchecking these boxes to select subsets of your data.

When done examining the numeric data, remove this facet by clicking the x in the upper left corner of its panel. Note that this does not undo the edits you made to the cells in this column. If you want to reverse these edits, use the Undo / Redo function.

Scatterplot facet

1. Now that we have multiple columns representing numbers, we can see how they relate to one another using the scatterplot facet.
2. Select a numeric column, for example recordID, and use the pulldown menu to > Facet > Scatterplot facet.
3. A new window called Scatterplot Matrix will appear. There are squares for each pair of numeric columns organized in an upper right triangle. Each square has little dots for the cell values from each row.

Examine pair of columns in detail

We can examine one pair of columns by clicking on its square in the Scatterplot Matrix. A new facet with only that pair will appear in the left margin.

How OpenRefine records what you have done

1. As you conduct your data cleaning and preliminary analysis, OpenRefine saves every change you make to the dataset.
2. These changes are saved in a format known as JSON (JavaScript Object Notation).
3. You can export this JSON script and apply it to other data files.
    a. If you had 20 files to clean, and they all had the same type of errors (e.g. misspellings, leading white spaces), and all files had the same column names, you could save the JSON script, open a new file to clean in OpenRefine, paste in the script and run it.
    b. This gives you a quick way to clean all of your related data.

Saving your work as a script

- In the Undo / Redo section, click Extract... , and select the steps that you want to apply to other datasets by clicking the check boxes.
- Copy the code from the right hand panel and paste it into a text editor (like NotePad on Windows or TextEdit on Mac). Make sure it saves as a plain text file. In TextEdit, do this by selecting Format > Make plain text and save the file as a .txt file.

Importing a script to use against another dataset

Let's practice running these steps on a new dataset. We'll test this on an uncleaned version of the dataset we've been working with.

- Start a new project in OpenRefine using the messy dataset you downloaded before. Give the project a new name.
- Click the Undo / Redo tab > Apply and paste in the contents of .txt file with the JSON code.
- Click Perform operations. The dataset should now be the same as your other cleaned dataset.
- For convenience, we used the same dataset. In reality you could use this process to clean related datasets. For example, data that you had collected over different fieldwork periods or data that was collected by different researchers (provided everyone uses the same column headings).

Saving and Exporting a Project

1. In OpenRefine you can save or export the project.
2. This means you're saving the data and all the information about the cleaning and data transformation steps you've done.
3. Once you've saved a project, you can open it up again and be just where you stopped before.

Saving

1. By default, OpenRefine is saving your project continuously.
2. If you close OpenRefine and open it up again, you'll see a list of your projects.
3. You can click on any one of them to open it up again.

Exporting

1. You can also export a project, which allows you to send your raw data and cleaning steps to a collaborator, or share this information as a supplement to a publication.
2. Click the Export button in the top right and select Export project.
3. A tar.gz file will download to your default Download directory.
    a. The tar.gz extension tells you that this is a compressed file.
    b. The downloaded tar.gz file is actually a folder of files which have been compressed.
    c. Linux and Mac machines will have software installed to automatically expand this type of file when you double-click on it. For Windows based machines you may have to install a utility like '7-zip' in order to expand the file and see the files in the folder.
4. After you have expanded the file look at the files that appear in this folder. What files are here? What information do you think these files contain?

Solution

1. You should see:
    a. a history folder which contains a collection of zip files. Each of these files itself contains a change.txt file. These change.txt files are the records of each individual transformation that you did to your data.
    b. a data.zip file. When expanded, this zip file includes a file called data.txt which is a copy of your raw data.
    c. You may also see other files.
2. You can import an existing project into OpenRefine by clicking Open... in the upper right > Import Project and selecting the tar.gz project file. This project will include all of the raw data and cleaning steps that were part of the original project.

Exporting Cleaned Data

1. You can also export just your cleaned data, rather than the entire project.
2. Click Export in the top right and select the file type you want to export the data in. Tab-separated values (tsv) or Comma-separated values (csv) would be good choices.
3. That file can then be opened in a spreadsheet program or imported into programs like R or Python.
4. Remember from our lesson on Spreadsheets that using widely-supported, non-proprietary file formats like tsv or csv improves the ability of yourself and others to use your data.

Key Points

1. OpenRefine is a powerful, free and open source tool that can be used for data cleaning.
2. OpenRefine will automatically track any steps allowing you to backtrack as needed and providing a record of all work done.
3. OpenRefine can import a variety of file types.
4. OpenRefine can be used to explore data using filters.
5. Clustering in OpenRefine can help to identify different values that might mean the same thing.
6. OpenRefine can transform the values of a column.
7. OpenRefine provides a way to sort and filter data without affecting the raw data.
8. OpenRefine also provides ways to get overviews of numerical data.
9. All changes are being tracked in OpenRefine, and this information can be used for scripts for future analyses or reproducing an analysis.
10. Cleaned data or entire projects can be exported from OpenRefine.
11. Projects can be shared with collaborators, enabling them to see, reproduce and check all data cleaning steps you performed.
12. Other examples and resources online are good for learning more about OpenRefine.