# Wrangle and Analyze Data

REVIEW

HISTORY

## Meets Specifications

Congratulations 🎓
You've made it in the first attempt!!
This is an excellent submission!

You took a lot of variables from the dataset and did an excellent job of systematically exploring it, assessing it, cleaning it and then coming up with some interesting findings. I've really liked the way you've structured your project! Please do check specific comments.

Be a lifelong learner.
Stay Safe and Stay Udacious!

**Further Readings:**
Data Wrangling Cheatsheet
Twitter Analytics: "WeRateDogs"
Better, Faster, Stronger Python Exploratory Data Analysis (EDA)
Data Wrangling and Why Does it Take So Long
Data Preprocessing vs. Data Wrangling in Machine Learning Projects

## Code Functionality and Readability

**All project code is contained in a Jupyter Notebook named wrangle_act.ipynb and runs without errors.**

All the code is present in the `wrangle_act.ipynb` notebook and run without errors. Well done!

### Suggested Readings

- How to use shortcuts with Jupyter Notebook
- Cheat sheet for data wrangling using python
- Use functions to avoid code repetition

The Jupyter Notebook has an intuitive, easy-to-follow logical structure. The code uses comments effectively and is interspersed with Jupyter Notebook Markdown cells. The steps of the data wrangling process (i.e. gather, assess, and clean) are clearly identified with comments or Markdown cells, as well.

Every step of the wrangling process (gather, assess, and clean) is well documented with markdown text and the code is often commented. This helps anyone to follow and understand easily the wrangling workflow, that's a very good practice useful when you work in a team with other developers and also for ourselves. Well done!

### Suggested Readings

- 10 Tips for Writing Cleaner & Better Code

## Gathering Data

Data is successfully gathered:

- From at least the three (3) different sources on the Step 1: Gathering Data page.
- In at least the three (3) different file formats on the Step 1: Gathering Data page.

Each piece of data is imported into a separate pandas DataFrame at first.

Data is successfully gathered from three different sources and in three different formats. Nice job!

## Assessing Data

Two types of assessment are used:

- Visual assessment: each piece of gathered data is displayed in the Jupyter Notebook for visual assessment purposes. Once displayed, data can additionally be assessed in an external application (e.g. Excel, text editor).
- Programmatic assessment: pandas' functions and/or methods are used to assess the data.

Well done using `info()`, `describe()`, `isnull()`, `head()`, `duplicated()` and other useful functions to explore the data.

### Suggestion

Remember that once data is displayed, data can additionally assess data using external applications like Microsoft Excel, Google Sheet or text editor.

**At least eight (8) data quality issues and two (2) tidiness issues are detected, and include the issues to clean to satisfy the Project Motivation. Each issue is documented in one to a few sentences each.**

The issues pointed out in the dataset are appropriate. And good work on documenting them in a consistent way by numbering them 👍🏼
It helps a reader keep track while reading. Brownie points for identifying multiple dog stages in some of the tweets.

## Suggestion

There is no definitive rule by WeRateDogs to not accept wild ratings, such as >10 or non 10 denominators. Notice this passage in the Project Motivation page:

**The fact that the rating numerators are greater than the denominators does not need to be cleaned. This unique rating system is a big part of the popularity of WeRateDogs.**

---

### Data Quality issues you can consider

**ID fields:** The ID fields, like `tweet_id` , `in_reply_to_status_id` etc. should be objects, not integers or floats because they are not numeric and aren't intended to perform calculations.

**Ratings:** Considering that the data analysis part is based on the rating columns we should mention among the data quality issues that the most appropriate data type for the `rating_numerator` column should be float and also it should be correctly extracted. Indeed there are some anomalous values in the `twitter_archive_enhanced.csv` , as written in the project pages is explained that the ratings are extracted incorrectly. There are many more such issues in the dataset. So it should be extracted and cleaned correctly. The `rating_denominator` column is acceptable as an integer but is preferred as float since there is nothing stopping future dog ratings from having a number with a decimal in the denominator.

**Dog Names** : In the name column, there are several values that are not dog names, like 'a', 'the', 'such', etc. Notice that all of these observations have lowercase characters, an important pattern that could be used to clean up this field. Another way is to drop duplicated values.

**Retweets and Favorite Count**: retweet_count and favorite_count should be integers, not floats.

### Suggested Readings

A useful resource to know how to classify correctly each type of issue: Data Quality & Tidiness

# Cleaning Data

The define, code, and test steps of the cleaning process are clearly documented.

Good Work on following the `Define`, `Code` and `Test` sequence in the data cleaning process. This process will go a long way in understanding the steps involved in cleaning process.

Copies of the original pieces of data are made prior to cleaning.

All issues identified in the assess phase are successfully cleaned (if possible) using Python and pandas, and include the cleaning tasks required to satisfy the Project Motivation.

A tidy master dataset (or datasets, if appropriate) with all pieces of gathered data is created.

Well done using the copy() function, making a copy of the datasets, before cleaning the data! This is a very good practice to follow before starting the cleaning process.

## Hints

Some data cleaning hints in order to clean properly the above suggested issues:

- In the `name` column, there are several values that are not dog names, like 'a', 'the', 'such', etc. Notice that all of these observations have lowercase characters, an important pattern that could be used to clean up this field.
  Here is how you can print out all lowercase names:

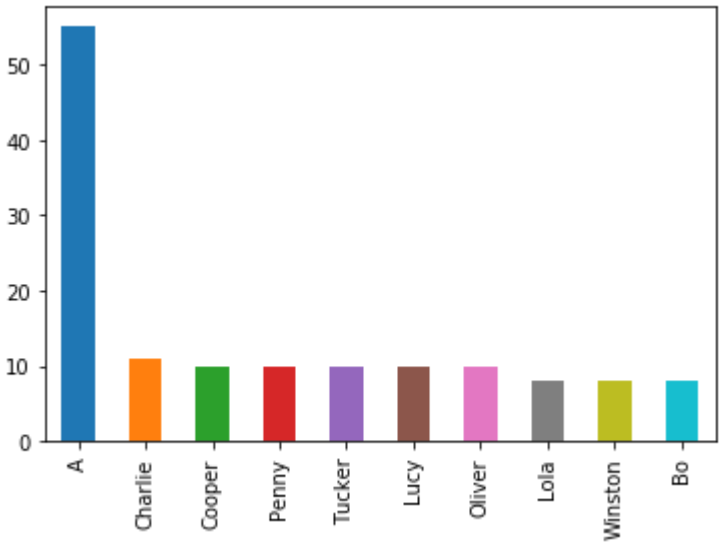
  ```
  mask = archive_clean.name.str.contains('^[a-z]', regex = True)
  archive_clean[mask].name.value_counts().sort_index()
  ```

  Which will give you the following output:

```
a              54
actually        2
all             1
an              6
by              1
getting         2
his             1
incredibly      1
infuriating     1
just            3
life            1
light           1
mad             1
my              1
not             2
```

Note that cleaning the dog names completely will prove to be a very challenging task. Even after cleaning the issues as mentioned above, there will be other invalid names in the dataset. For instance, dog names have

an invalid value `A` too:




---

- To remove the dog names lowercase values from the dataset, you can index of rows where the dog names were lower case.

```
lower_dog_name_index = archive_df[archive_df.name.str.islower()].index
```

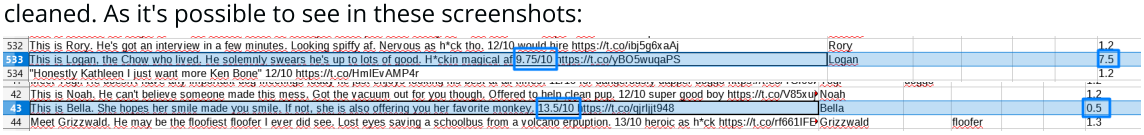
Then drop the rows using this index

```
archive_df.drop(lower_dog_name_index, inplace=True)
archive_df.shape
```

Now you can see the rows being removed.

---

- Ratings: As mentioned in the review section dedicated to the assessment part, since the data analysis part is based on the ratings we should properly clean these values otherwise we risk altering the results of the data analysis, weakening our findings. Indeed, as mentioned above and in the project instructions, there are some rating values that are not correctly extracted and should be properly cleaned. As it's possible to see in these screenshots:




after the data cleaning the ratings are transformed but are still not correctly extracted.

We can pick the decimals in the ratings using a regex, like the one in the code snippet below:

```
rating = df_clean.text.str.extract('((?:\d+\.)?\d+)\/(\d+)', expand=True)
rating.columns = ['rating_numerator', 'rating_denominator']
```

### Suggested Readings

- How To Convert Data Types in Python 3
- Python Regular Expression HOWTO
- How do I remove letters from a string in Python?
- Strings and Character Data in Python
- Missing data with pandas
- Python, delete rows
- Pandas drop
- Introduction to manual feature engineering
- Tidy Data and How to Get It

## Storing and Acting on Wrangled Data

**Students will save their gathered, assessed, and cleaned master dataset(s) to a CSV file or a SQLite database.**

A master dataset is correctly saved to a CSV file. Great job setting the Index Parameter to False in order to avoid an "unnamed" index column to the dataset!

**The master dataset is analyzed using pandas or SQL in the Jupyter Notebook and at least three (3) separate insights are produced.**

**At least one (1) labeled visualization is produced in the Jupyter Notebook using Python's plotting libraries or in Tableau.**

**Students must make it clear in their wrangling work that they assessed and cleaned (if necessary) the data upon which the analyses and visualizations are based.**

You did a good job on your analyses, producing interesting insights and meaningful data visualizations.

**Comments:**

One of the most important steps in creating an impactful visualization is making sure all of its elements are labeled appropriately. The text components of a graph give your reader visual clues that help your data tell a story and should allow your graph to stand alone, outside of any supporting narrative.

### Suggested Readings

Some useful resources to improve your fantastic skills and produce always better visualizations 🚀

- How to make beautiful data visualizations in Python with Matplot
- 5 quick and easy data visualizations in Python
- The Best Python Data Visualization Libraries
- 10 Useful Python Data Visualization Libraries for Any Discipline

## Report

**The student's wrangling efforts are briefly described. This document (wrangle_report.pdf or wrangle_report.html) is concise and approximately 300-600 words in length.**

You have made a very nice `wrangle_report` document, concise, clear and interesting to read. Well done!

**The three (3) or more insights the student found are communicated. At least one (1) visualization is included.**

**This document (act_report.pdf or act_report.html) is at least 250 words in length.**

You made a nice `act_report`, effectively communicating the insights and including at least 1 visualization.

### Suggestion

Please, consider making your report even more interesting enriching your insights adding your personal thoughts and everyday experiences. You can also include some pictures for aesthetic and additional context purposes on top of the visualizations. (for example, a picture of a specific dog breed or a tweet's screenshot). Anything to get the reader engaged. Picture this external report like a blog post or magazine article. All these details help people to be engaged and have fun while reading.

#### Suggested Readings

Actionable Insights: How data presentation skills create value
Communicating data science: A guide to presenting your work

## Project Files

**The following files (with identical filenames) are included:**

- **wrangle_act.ipynb**
- **wrangle_report.pdf or wrangle_report.html**
- **act_report.pdf or act_report.html**

**All dataset files are included, including the stored master dataset(s), with filenames and extensions as specified on the Project Submission page.**

DOWNLOAD PROJECT