# wrangle_report

July 13, 2020

## 1 Udacity Data Wrangling Report

### 1.0.1 The Aim of the Project:

The aim of this project was to gather, localized in different sources, twitts data of the famous fan page WeRateDogs. The collected data should be then saved into data frames and assess thoroughly. The goal of the assessment was to identify minimum 8 quality and 2 tidiness issues, hindering the efficiency of data analyzing. After targeting the issues, the cleaning of the created data frames could begin. To, finally finish the project with a few insights and visualizations based on the obtained data.

### 1.0.2 Wrangling Process:

1. `Gathering Part`

- collecting the data from three different sources, namely: local source, URL and twitter API
- using request library to download image-prediction data and writing it to tsv file
- registering on the developer.twitter.com and applying for access to WeRateDogs API data, collecting the data using tweepy library, saving the data into txt file
- writing all three sources of the gathered data's into data frames

2. `Assessment Part`

- performing visual and programatic assessment of the created data frames
- using basic assessment's methods like: `.head()`, `.info()`, `.describe()`, `.duplicated()`, `.isnull`, etc.
- using more advanced assessment's techniques like: list comprehension
- describing 10 quality issues and 2 tidiness issues found among the three data frames

3. `Cleaning Part`

- creating a 'master' copies of each data sets on which cleaning will be performed
- describing a cleaning solution for each of the issues listed in the assessment part
- using basic built-in methods to perform some of the cleaning parts, e.g.: `.drop()`, `.dropna()`, `.to_datetime()`, `.astype()`, `.isin()`, etc.
- applying more complexer approaches to satisfy some of the cleaning goals, e.g.: combining the `groupby()` method with `.transform()` method to drop all of the rows in the data frame with numerator frequencies of 2 and lower
- testing each cleaning action in the seperated `Test` section, using following methods: `.head()`, `.info()`, `.sample()`, `.value_counts`, `.dtypes`, etc.

4. `Analyses and Visualizations`

- plotting histogram depicting counts of the dog stages
- using seaborn to depict the relationship betweet the `rating_numerator` and `dog_stage`
- using basic statistical methods like: `.describe()`, `.mean()`, `.std()`, `.groupby()` to gain some insights on the investigated variables: `rating_numerator` and `dog_stage`