WRANGLE_REPORT: Data Wrangling (ND111)

By Vítor Almeida

The data wrangling process were developed in an Jupyter notebook and sistematically divided in three stages, following the good practices in coding.

After data was cleaned, some brief visualizations and insights were performed. An index was created and cells corresponding to each of these stages were hyperlinked for better navigation. Libraries for data structure, for request web page and twitter API data were imported.

I. GATHER

Data from WeRateDogs tweets were divided into 3 DataFrames gathered with three different methods:

- *tweets_archive*: imported from csv using *read_csv* function.

- *tweets_img*: downloaded using requests, then reading with *read_csv*

- *tweets_api* - extracted using Tweepy (twitter API), using *tweet_id* Series from *tweets_archive*.

II. ASSESS

First, copies of the three DataFrames were made using *copy* function, and then a general assessment were made - visually and/or programmatically – using functions like *info* and *sample*.

The dataset had samples from retweet, what is not aligned with the project motivation. Some data contained NaNs, but in some cases (e.g., *in_reply_to_id* column) the NaNs did not represent missing values, it just meant that tweet were not a reply, for instance. Despite not having NaNs, dog classes, names and races columns had many None and/or inconsistent values, such as “an”, “a”, and “is”.

Finally, columns regarding the same variable were found in multiple columns, and the three distinct DataFrames corresponds to the same observational unity.

After assessment was concluded, quality and tidiness issues were summarized:

IIa. Quality

tweets from archive:

- part of the tweets are retweets;

- *retweeted_status* and *in_reply_to* columns: unsuitable data type (int);

- *datastamp* and *retweeted_status_timestamp* columns: unsuitable datatype (str);

- name column: invalid values ('a', 'an', 'just', 'is', etc);

- dog class columns: unsuitable datatype (str);

- many rows with None dog class;

tweets from image prediction:

- p1 columns: rows with prediction other than dog races;

- p1 columns: non informative columns labels;

- p1 columns: races names without standard;

all:

- tweet id columns: unsuitable datatype (int);

IIb. Tydiness

- same observational unity in 3 different DataFrames;

- tweets from image prediction: one variable in multiple columns (p1, p2, p3);

- tweets from archive: one variable in multiple columns (dog classes);

III. CLEAN

Occasionally, quality and tydiness issues were not cleaned in the same order presented in Assess stage.

The following actions were defined to perfom the cleaning process:

- drop rows in which retweeted_status_id are not NaNs;
- change *tweet_id* columns from int to str object;
- change these columns from float to str object;
- transform timestamp columns dtype to datetime;
- disregard rows with lowercase dog names (invalid names);
- create column *dog_stage* concatening stages from columns;
- drop unclassified dog rows and assign the resulting DataFrame to a separated object;
- drop p2 and p3 columns and maintain p1 as the highest coefficient prediction;
- disregard 'p1_dog' rows with False values;
- rename labels of prediction columns to more informative ones;
- replace races names elements with lowercase and underscore;
- join the three dataframes and keep only *tweet_id* rows that match between them;

Finally, the resulting master DataFrame (*twitter_archive_master*) and the only-classified-dogs DataFrame (*tweets_archive_dog_stage*) were exported to CSV using *to_csv* function.

Visualizations and insights are presented in *act_report.pdf*.