ADSI – AT2 – Beer Classification API

Peter Brotherhood – 12875237

# Project artifacts

The following public git repos contain all code for the development and deployment elements of this project, respectively:

[git@github.com:PeRoBr/adv\_dsi\_at\_2.git](mailto:git@github.com:PeRoBr/adv_dsi_at_2.git)

[git@github.com:PeRoBr/adv\_dsi\_at\_2\_api.git](mailto:git@github.com:PeRoBr/adv_dsi_at_2_api.git)

Deployment of the trained model is available at the following url:

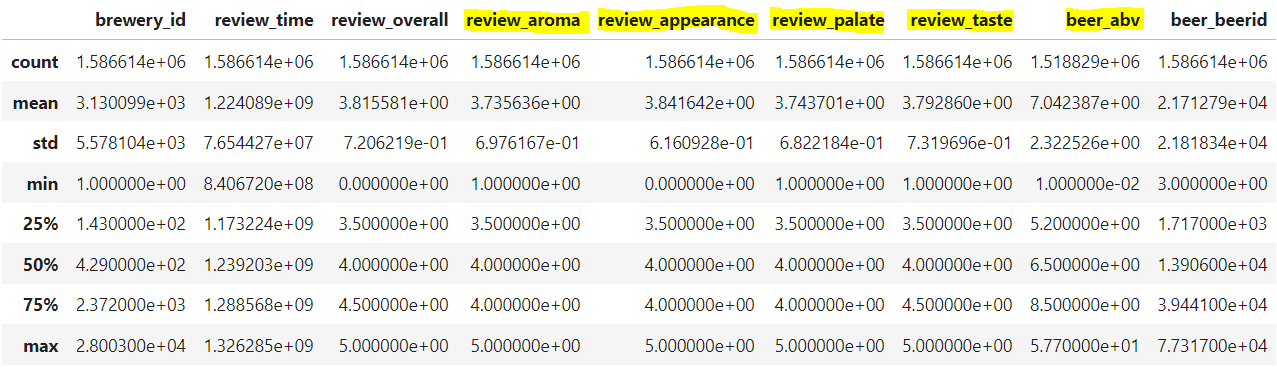
<https://protected-refuge-41634.herokuapp.com/docs>

# Data Understanding

The data available for this project is extensive, containing ~1.5 million reviews from BeerAdvocates. The data set contains 13 variables that describe the brewer, the beer and, and subjective scores from reviewers between 1 and 5 indicating their personal experience of the beer. These subjective variables include aroma, appearance, palate, taste, and an overall score. The alcohol concentration of the beer is also provided. The final output of this study is an API endpoint that can receive a reviewers subjective scores, the alcohol concentration and the brewery name and return to the reviewer a guess as to the style of beer reviewed. Henceforth, the only the variables required by the API endpoint ant the target variable beer style will be discussed. These variables are: brewery\_name, review\_aroma, review\_appearance, review\_palate, review\_taste, beer\_abv.

The data quality is fairly good with a small number of nulls in brewery\_name and beer\_abv and some outliers in beer\_abv (the max alcohol conentration is 57.7%) but these issues ad more than compensated for the sheer volume of data. All the subjective review variables are between either 0 and 5 or 1 and 5 and 1st and 3rd quartile values are all between 3.5 to 4.0 out of five indicating that beer is universally enjoyed. However, an important note for this study there is little variation in scores across the whole dataset and thus beer\_style specific signals might be difficult for an ML algorithm to identify.

**Statistical summary of numerical columns – Those used highlighted in yellow**



There are 5743 brewers represented in the data and 104 different beer styles. In a ranked list of beer\_styles by number of reviews received, American style beers feature very prominently in those most frequently reviewed.

**Top 20 most reviewed beer styles**



There is a challenge posed to modelling in the number of categories in the dependent variable beer\_style (104) and in the number of categories in brewery\_name (5743). The least frequently reviewed beer\_styles and breweries are very rare in the data set and even amongst the most popular breweries and beer\_styles not all combinations are available. For an algorithm to have a good chance of learning all the possible signals that contribute to a classification it must have balanced experience with all the range of factors in the data. Also, if a categorical variable with 5743 levels was OneHotEncoded this would very likely make even a stratified subsample an extremely wide dataset. This motivated the approach to generating our training dataset described below.

# Data Preparation

For the reasons described above the following approach was taken to reduce the dataset to a manageable volume for model training, and still provide a worthwhile proof of concept for the final API.

The dataset was reduced by selecting the top 10 breweries by review count and from those breweries selecting only beer\_styles in the top ten by review count. This yielded ~132 k records from the original 1.5 million. Only 90 of the 100 possible combinations were represented and 15 of these had fewer than 100 reviews. At this stage a small number of missing values were dropped from the data set. To further reduce the volume of the data set and aim to provide a balanced sample of records for each target variable category, a further filter was applied to randomly select 100 records from those brewery-beer\_style combinations with more than 100 reviews.

This yielded a final reduced data set of 7500 records, with ten target categories and 10 (a manageable number of OHE) breweries. The number of records available for the algorithm to train on are fairly well distributed across all target categories and all breweries.

**Distribution of review records across target beer\_style and predictive variable brewery\_name**

 Graphical user interface, table

Description automatically generated with medium confidence

This reduced data set was further processed to prepare it for training a neural network. Numerical values were scaled to mean 0 variance 1 with StandardScaler and the brewery\_name variable was OneHotEncoded. The target variable beer\_style was ordinal encoded and then converted to integer. The order of the categories and thus the mapping to their respective integer values is as per the beer\_style table on the left, above. These processing steps were combined in a sklearn pipeline fit to 60% of the reduced (7500 record dataset) and then used to transform a 20% test and 20% validation dataset.

# Modelling

Modelling to predict beer\_style was conducted with a pytorch neural network set up for multi class classification. The data filtering and processing yielded an input dataset with 14 variables. This was fed to a single 32 neuron layer and subsequently to an output layer with 10 levels, for the ten beer styles in the training dataset. The model was trained over 50 epochs and a sample size of 32. Peak model accuracy of 29.4% on the training data was achieved after ~13 epochs. Accuracy achieved with a reserved validation set was 30% indicating the model was not overfit. A baseline prediction applying the mode beer\_style to all observations in the training data gave an accuracy of 0.14.

The trained model performance is obviously very poor and too poor to provide any real business value, however, we can see that the model has learned from the data and is outperforming basline.

The result here is a proof of concept that the model can be used to make predictions, although poor. No further tuning was attempted, however, this is the obvious first thing to explore when time permits.

# Deployment

The model developed above was successful deployed via fastAPI on Heroku at the following url: <https://protected-refuge-41634.herokuapp.com/docs>

Predictions can be provided on beer reviews from the following ten breweries, when a 0 to 5 rating is provided for the review criteria:

Graphical user interface, table

Description automatically generated with medium confidence

Graphical user interface

Description automatically generated

Predictions are returned as integers 0-9 mapping to beer styles according to the following relation:

