Star Rating Classification of Yelp Businesses

Emily Nomura, Brown University, DATA1030 Midterm Project
[https://github.com/emilynomura1/1030MidtermProject]

1. Introduction

The Yelp business dataset is composed of 179,344 unique businesses, the majority of which are located in Montreal, Calgary, Toronto, Pittsburgh, Charlotte, Urbana-Champaign, Phoenix, Las Vegas, Madison, and Cleveland.[1] The data can be downloaded from the Yelp Open Dataset website and converted from JSON to CSV format with the Python converter available on the Yelp Github page.[2] The businesses range from locally owned small shops to widely-known corporate-owned businesses like McDonald's and Starbucks. There are a total of 60 features. Certain features were removed if they were composed of >=95% null values or contained a dictionary as their data type. Likewise, certain observations were removed if they were missing data in the name, city, or categories features. The final cleaned dataset contained 178,372 rows and 43 features before preprocessing.

The dataset includes continuous and categorical numeric features such as the average star rating for each business on a half point scale from one to five, the number of reviews at the time of data collection, latitude, and longitude. The majority of the features are text-based; highlights include a list of categories that the business fits under and various attributes that are specific to each business. The number of attributes that a business has ranges depending on if the business' customers or the business owners themselves input information into Yelp's software.

A detailed description of the variables present in the final cleaned dataset before preprocessing is below. Some business attributes contain the character 'u' at the start of the string value. This means that the category was assigned by Yelp user input rather than the business owner. All variables aside from name, star-rating, review count, categories, and city contain missing data.

Variable Descriptions

Variable	Data structure	Variable type	Description
name	string	categorical	The name of the business. Oftentimes unique, but there are repeated business names if it is a chain e.g. McDonald's, Starbucks, Burger King, etc.
star-rating	integer	categorical	The coerced target variable indicates the average star rating of the business on Yelp. Takes the values (1, 2, 3, 4, 5).
review_count	integer	continuous	The total number of reviews that the business received at the time of data collection. The maximum value is 10129, the minimum is 3.
categories	list of strings	categorical	The categories that the business falls under. E.g. [coffee & tea] or [restaurants, fast food, burgers, chicken wings]
city	string	categorical	The city that the business is located in. The top 3 cities have over 17,000 of the businesses in the dataset located in their city. They are Las Vegas, Toronto, and Phoenix.
state	string	categorical	The state or province that the business is located in. The 3 most common states are, as expected, Arizona, Nevada, and Ontario.

[1] "Frequently Asked Questions." *Yelp Dataset Documentation*, Yelp.
[2] Clark, S., Artem, A., & Chambers, B. *Dataset Examples - Yelp*, Github.

latitude	float	continuous	Latitude measure of the business to the 6th decimal place.
longitude	float	continuous	Longitude measure of the business to the 6th decimal place.
attributes.BYOB	boolean	categorical	Business attribute answering the question: Is this business BYOB (bring your own beer)? Takes the values True or False.
attributes.RestaurantsGoodForGroups	boolean	categorical	Business attribute answering the question: Is this restaurant good for groups? Takes the values True, False, or None.
attributes.RestaurantsAttire	string	categorical	Categorical business attribute answering the question: What kind of attire is recommended for this restaurant? Takes the values u'casual', casual, u'dressy', dressy, u'formal', formal, or None.
attributes.GoodForDancing	boolean	categorical	Business attribute answering the question: Is this business good for dancing? Takes the values True, False, or None.
attributes.WheelchairAccessible	boolean	categorical	Business attribute answering the question: Is this business wheelchair accessible? Takes the values True, False, or None.
attributes.RestaurantsTakeOut	boolean	categorical	Business attribute answering the question: Does this restaurant offer take-out? Takes the values True, False, or None.
attributes.BusinessAcceptsCreditCards	boolean	categorical	Business attribute answering the question: Does this business accept credit cards? Takes the values True, False, or None.
attributes.ByAppointmentOnly	boolean	categorical	Business attribute answering the question: Is this business by appointment only? Takes the values True, False, or None.
attributes.CoatCheck	boolean	categorical	Business attribute answering the question: Does this business have a coat check? Takes the values True, False, or None.
attributes.HasTV	boolean	categorical	Business attribute answering the question: Does this business have a TV? Takes the values True, False, or None.
attributes.HappyHour	boolean	categorical	Business attribute answering the question: Does this business have a happy hour? Takes the value True, False, or None.
attributes.Smoking	string	categorical	Business attribute answering the question: Does this business allow smoking? Takes the values u'no', no, u'outdoor',outdoor, u'yes', yes, or None.
attributes.OutdoorSeating	boolean	categorical	Business attribute answering the question: Does this business have outdoor seating? Takes the value True, False, or None.
is_open	binary	categorical	Binary value indicating whether the business is open or not. 1 indicates yes, 0 no.
attributes.NoiseLevel	string	categorical	Business attribute answering the question: What is the noise level of this business? Takes the values u'average', average, u'quiet', quiet, u'loud', loud, u'very_loud', very_loud, or None.
attributes.DriveThru	boolean	categorical	Business attribute answering the question: Does this business have a drive through? Takes the values True, False, or None.
attributes.RestaurantsReservations	boolean	categorical	Business attribute answering the question: Does this restaurant take reservations? Takes the values True, False, or None.
attributes.RestaurantsTableService	boolean	categorical	Business attribute answering the question: Does this restaurant offer table service? Takes the values True, False, or None.
attributes.RestaurantsPriceRange2	integer	ordinal	Business attribute indicating the price range of a restaurant. Takes the values 1, 2, 3, 4, or None. 1=$10 or less, 2=$11-$30, 3=$31-$60, 4=$61+
attributes.DogsAllowed	boolean	categorical	Business attribute answering the question: Does this business allow dogs? Takes the values True, False, or None.

attributes.BusinessAcceptsBitcoin	boolean	categorical	Business attribute answering the question: Does this business accept bitcoin? Takes the values True, False, or None.
attributes.Alcohol	string	categorical	Business attribute indicating what kind of alcohol the business serves, if any. Takes the values u'full_bar', full_bar, u'none', none, u'beer_and_wine', beer_and_wine, or None.
attributes.Caters	boolean	categorical	Business attribute answering the question: Does this business offer catering services? Takes the values True, False, or None.
attributes.WiFi	string	categorical	Business attribute indicating what kind of WiFi the business has, if any. Takes the values u'free', free, u'no', no, u'paid', paid, or None.
attributes.BYOBCorkage	string	categorical	Business attribute indicating what kind of corkage services the business offers, if any. Takes the values u'no', no, u'yes_free', yes_free, u'yes_corkage', yes_corkage, or None.
attributes.Corkage	boolean	categorical	Business attribute answering the question: Does this business offer corkage? Takes the values True, False, or None.
attributes.AcceptsInsurance	boolean	categorical	Business attribute answering the question: Does this business accept insurance? Takes the values True, False, or None.
attributes.BikeParking	boolean	categorical	Business attribute answering the question: Does this business have bike parking? Takes the values True, False, or None.
Monday.hrs.open	integer	continuous	Number of hours the business is open on Monday. Created by splitting on the old time range, coercing to datetime, and subtracting the open and close time. Ranges from (0, 23).
Tuesday.hrs.open	integer	continuous	Number of hours the business is open on Tuesday. Ranges from (0, 23).
Wednesday.hrs.open	integer	continuous	Number of hours the business is open on Wednesday. Ranges from (0, 23).
Thursday.hrs.open	integer	continuous	Number of hours the business is open on Thursday. Ranges from (0, 23).
Friday.hrs.open	integer	continuous	Number of hours the business is open on Friday. Ranges from (0, 23).
Saturday.hrs.open	integer	continuous	Number of hours the business is open on Saturday. Ranges from (0, 23).
Sunday.hrs.open	integer	continuous	Number of hours the business is open on Sunday. Ranges from (0, 23).

The problem I want to solve in this project is classification. More specifically, can the model accurately predict the star rating of a business given the predictors we supply it? This type of problem is useful to answer because when any food-ordering or business-reviewing software is supplied with new business data, there is no information about the business other than what the business owners input themselves. Therefore, once customers start to visit the business, write reviews, and answer questions about the business' attributes, this type of prediction model can help to categorize and label businesses in order to better aid consumer and user decisions.

2. Exploratory Data Analysis

It is often believed that Yelp users only write reviews for businesses they either "really like" or "really dislike." This would imply that the most common star ratings given out to businesses are 1 and 5. Evidently, by visualizing the counts of each star rating group in a simple bar chart, this is actually a misconception. In the Yelp business dataset, an average star rating of 3 or 4 are the most common, while a star rating of 1 is least common. Nevertheless, around 10,000 businesses still have an average star rating of 1. The distribution of the target variable is indeed imbalanced. Thus, caution should be taken when performing data splitting and data preprocessing.

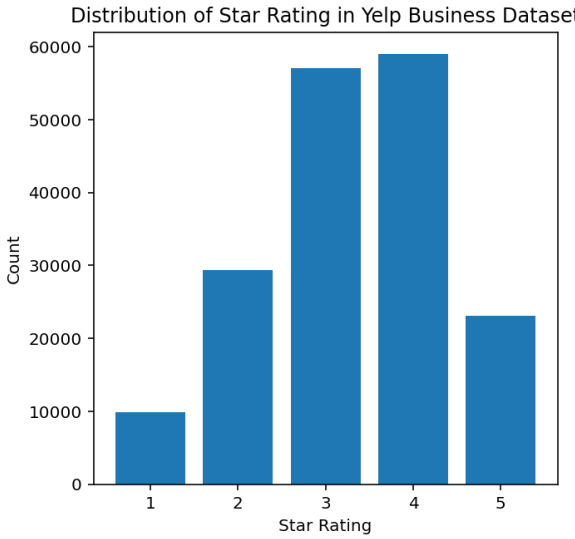

Fig. 1. The star rating variable was coerced from the original stars variable, reducing the number of total classes in the target variable from 9 to 5. The bar plot was created using matplotlib and the base function value_counts().

The average number of reviews for a business in the Yelp business dataset is 37. When examining the boxplots identified by star rating, it is clear that review count of each business varies greatly regardless of star rating. There doesn't seem to be any evidence that the number of reviews a business has has any indication of their star rating. Although the average number of reviews for each star rating group is relatively low (between 5 and 50 reviews), there are a great deal of outliers in each group. One particular business with an average star rating of 4 has over 10,000 total reviews.

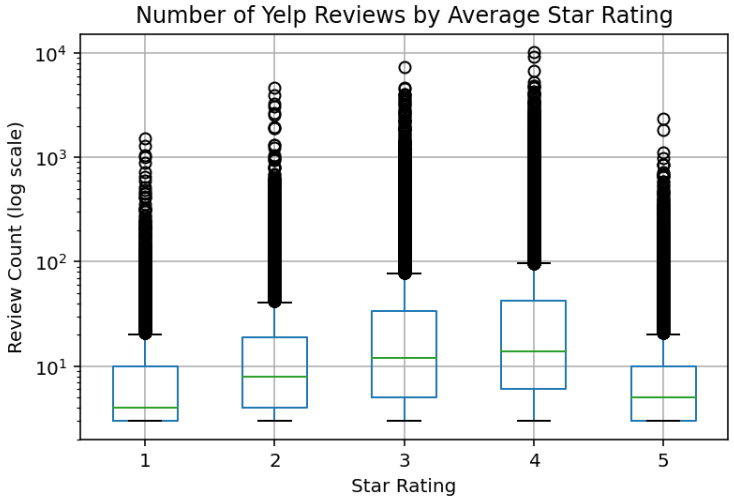

Fig. 2. The boxplots were created using matplotlib. These show helpful information about the review count of businesses when grouped by star rating like the minimum, median, maximum, interquartile range, and potential outliers.

When deciding where to eat, one of the most significant factors one takes into consideration is often price. Yelp provides a price range variable in the dataset for businesses that fall into the restaurant category. Users can indicate how much they've spent at a particular business on Yelp's mobile application, which affects the price group a restaurant is assigned to. Price group 2 ($11-$30) contains the largest proportion of 5 star reviews and the smallest proportion of 1 star reviews compared to other price range groups. We must ingest this information with a grain of salt because the stacked bar graph does not show the value counts of each price range group. However, as it turns out, price range group 2 contains the largest number of observations of any other group, with a total of 53,571 restaurants.

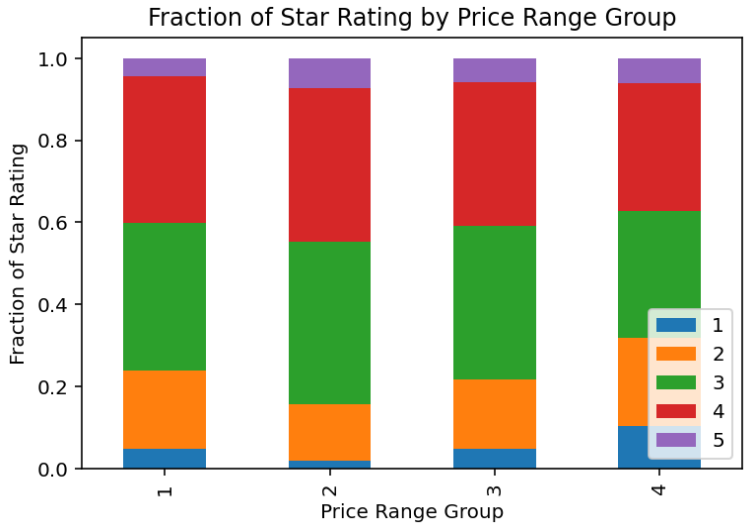

Fig. 3. Stacked bar plots were created using matplotlib. Any observations with 'None' as the value in the price range feature was omitted before plotting. As noted in the Variable Descriptions section, price group 1 indicates that a visit to the business costs $10 or less, group 2: $11-$30, group 3: $31-60, and group 4: $61 or more.

It may be helpful to visualize the locations of the businesses in the Yelp dataset. There are a variety of geographic features included such as city, state, latitude, longitude, and postal code. During the data cleaning process, businesses in states with less than 1000 businesses were filtered out. Therefore, many of the businesses are clustered in certain North American cities. By referring back to the variable description of the 'city' feature, it can be verified that the majority of businesses are located in Arizona, Nevada, and Ontario.

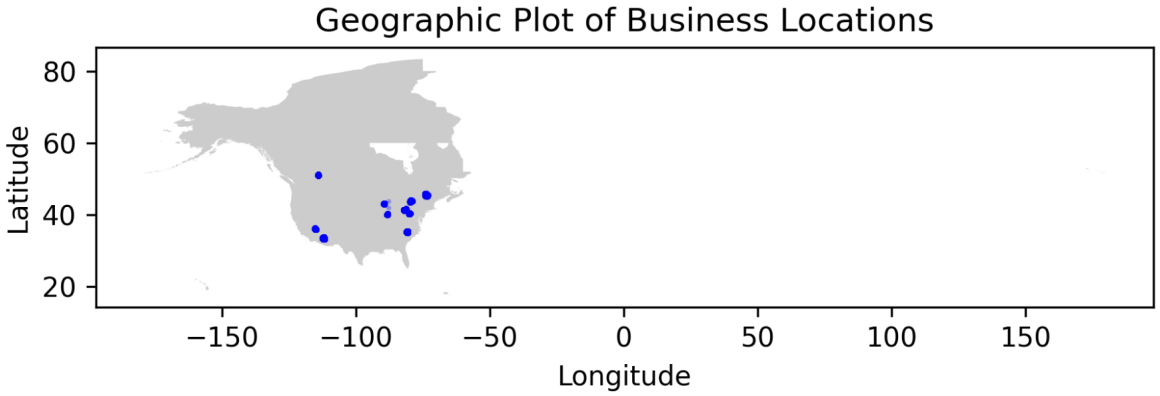

Fig. 4. Latitude and longitude coordinates along with the background map were pointed with matplotlib, geopandas, and the geometry package from shapely. The outline of the North America land mass requires a shapefile .zip that can be found in the Data folder. Most of the businesses in the Yelp dataset are located in large cities on the mainland U.S. and Canada.

3. Data Preprocessing

Each observation represents a unique business. Although there are repeated names for large chains such as McDonald's and Starbucks, the feature information about each business varies because it represents an entirely different location. Thus, the Yelp business data is independent and identically distributed. It does not have a group structure or consist of any type of time-series. A random state of 77 was specified during data splitting and preprocessing for the sake of reproducibility. The data was split using the train test split function from the scikit-learn preprocessing package. 80% of the feature matrix was split into training, 10%

in validation, and 10% in testing. Because the data is imbalanced, the stratification argument was specified on the target variable during splitting to ensure stratified splits on the 5 classes of star rating.

The feature matrix contained missing data in almost every categorical feature, but no missing data in any of the continuous features. A new category for each categorical feature was imputed to replace the NaN values before the data was split. It is difficult to define whether this pattern of missingness is missing at random (MAR) or missing not at random (MNAR). For example, if a user decides not to answer an attribute-related question regarding a business they just visited, it could be because they don't know exactly how to use the different features of Yelp's mobile application or simply because they don't have the time. This would imply that some of the missing data in the business attribute features is MAR. However, if a user decides not to answer an attribute-related question regarding a business they just visited because they are upset about the experience they had at the business, this would imply that the missingness patterns in certain attribute features are MNAR.

A semi-custom one hot encoder was deployed in order to deal with the messy categories feature. The one hot encoder used the multi label binarizer function from scikit-learn's preprocessing package to create a separate dummy feature for each business category. The classes were defined by looping through each observation in the categories feature and splitting the list of strings by comma.

A scikit-learn preprocessing pipeline was initialized and included all four main preprocessing functions: ordinal encoder, one hot encoder, min max scalar, and standard scalar. There was only one ordinal feature - the business attribute indicating restaurant price range. This feature is ordinal because it takes the values (1, 2, 3, 4), where 1 indicates a low spending value and 4 represents a high spending value. All other business attribute features as well as city, state, and the boolean feature 'is_open' were transformed with the one hot encoder. The one hot encoder features are all categorical string variables, which is necessary for the one hot encoder transformation. The review count feature was transformed with standard scalar because the number of reviews for each business was not well-bounded and varied greatly between businesses. The rest of the continuous, numeric features were relatively well bounded, so they were supplied the min max scalar. These features consisted of latitude, longitude, and the seven variables describing the total number of hours a business is open for each day.

The final data after preprocessing had a total of 2525 features. 1196 of those features were created during the preprocessing pipeline, while the other 1329 were defined during the semi-custom one hot encoder transformation of the categories feature.

References

1. "Frequently Asked Questions." *Yelp Dataset Documentation*, Yelp. https://www.yelp.com/dataset/documentation/faq.
2. Clark, S., Artem, A., & Chambers, B. *Dataset Examples - Yelp*, Github. November 7, 2014. https://github.com/Yelp/dataset-examples.
3. "Recommended Reviews." *Support Center*, Yelp. https://www.yelp-support.com/Recommended_Reviews?l=en_US.