IST 707: Final Project Report on

Yelp Review Analysis

A close up of a logo

Description automatically generated

By

Group 3

Jackson Hett

Immanuel Odarteifio

Shreyas Subhash Tekawade

**Introduction:**

This analysis is embarked upon with a clear and ambitious objective: to navigate the intricate landscape of Yelp reviews and, beyond mere surface-level observations, extract profound insights and discern patterns inherent in the diverse tapestry of user-generated content. Yelp, as a substantial repository of consumer experiences, presents an invaluable opportunity to glean insights that go beyond the immediate and transient, providing a nuanced understanding of user sentiments and preferences.

At the heart of this analysis lies Yelp, a towering presence in the realm of review platforms. Its extensive dataset serves as a digital chronicle, capturing the mosaic of user sentiments, preferences, and feedback across a spectrum of businesses, ranging from local enterprises to global establishments. Yelp, as a platform, mirrors the dynamic and evolving nature of consumer experiences, making it a treasure trove for those seeking to decipher the intricacies of the market.

Beyond the sheer volume of reviews, Yelp encapsulates the essence of diverse industries, acting as a virtual marketplace where users articulate their impressions of services, products, and experiences. This contextual richness provides a unique lens through which businesses can not only assess their current standing but also foresee trends and shifts in consumer expectations.

The significance of this analysis extends beyond academic curiosity; it's a strategic imperative for businesses. In the digital age, where consumer voices resonate globally, Yelp reviews represent a collective narrative that, when decoded, offers businesses a roadmap to customer satisfaction. This is not just about addressing isolated concerns but about crafting an adaptive and responsive business strategy, informed by real-time and authentic feedback.

Understanding Yelp reviews becomes a linchpin for businesses aiming not just for short-term success but for sustained growth. By decoding the collective sentiment within Yelp's extensive dataset, businesses can align their operations with evolving consumer expectations, cultivate positive brand perceptions, and foster a proactive engagement model.

This introduction establishes the groundwork for the ensuing analysis. The objective, context, and significance converge to form a narrative that positions Yelp reviews not just as data points but as strategic keystones for businesses navigating the complex landscape of consumer preferences. As we delve deeper into the subsequent sections, the aim is not only to analyze but to distill actionable insights that can empower businesses to thrive in an environment where consumer voices resonate louder than ever.

**Data Preparation:**

|  |  |  |
| --- | --- | --- |
| Step Number | Topic | Description |
| 1 | Tools Used | The data preparation process was orchestrated using the R programming language, leveraging pivotal libraries such as tidyverse for seamless data manipulation and jsonlite for handling JSON-formatted data. The choice of R bestowed flexibility and efficiency, crucial for the diverse data structures present in the Yelp dataset. |
| 2 | Data Source | Our raw data emanated from Yelp dataset files, encompassing a triad of critical components: user profiles, review specifics, and business information. These files were seamlessly ingested into R, a strategic decision given its innate ability to handle diverse data types and structures. |
| 3 | Sampling | To initiate the exploration, a preliminary random sample of 100 rows was drawn from each of the user, review, and business data frames. This snapshot expedited the familiarization process, providing a quick overview of the dataset's nuances and idiosyncrasies. |
| 4 | Integration | The integration phase saw the amalgamation of user, review, and business data frames through a judicious merging process based on common identifiers—specifically, user\_id and business\_id. This amalgamation birthed a harmonized dataset, weaving together user particulars, review intricacies, and business attributes into a cohesive fabric. |
| 5 | Sorting | To imbue a temporal context, the merged data frame underwent sorting based on the 'date' column, arranged in descending order. This chronological structuring ensured a focus on the most recent interactions, pivotal for understanding the evolution of user reviews and business dynamics over time. |
| 6 | Selection | A focused approach ensued, narrowing our attention to 28 columns of the sorted data frame. This curation targeted key features and variables, laying the groundwork for subsequent analyses and forestalling information overload. |
| 7 | Extraction | Digging deeper into the dataset, a random 25,000 rows were extracted from the sorted data frame. This subset represented a judicious balance—sufficiently sizable to capture diversity yet compact enough to preserve computational efficiency during in-depth exploration. |
| 8 | Output | The fruits of this meticulous process materialized in the form of "yelpfinal.csv"—our curated and consolidated dataset. This output, meticulously crafted, stands as a testament to the diligence invested in structuring the data for nuanced exploration and subsequent analytical pursuits. |

**About the data:**

1. business\_id: Unique identifier for the business where the review was written.
2. user\_id: Unique identifier for the user who wrote the review.
3. review\_id: Unique identifier for the review itself.
4. stars\_review: The rating given by the user in the review.
5. useful\_review: The number of users who found the review useful.
6. funny\_review: The number of users who found the review funny.
7. cool\_review: The number of users who found the review cool.
8. text: The actual content of the review where the user shares their feedback about the business.
9. date: The date when the review was posted.
10. name\_user: The name of the user who wrote the review.
11. review\_count\_user: The total number of reviews written by the user.
12. yelping\_since: The date when the user joined Yelp.
13. useful\_user: The total number of times other users found the user's reviews useful.
14. funny\_user: The total number of times other users found the user's reviews funny.
15. cool\_user: The total number of times other users found the user's reviews cool.
16. elite\_user: Indicates whether the user is part of Yelp's elite program.
17. friends: List of user's friends on Yelp.
18. fans: The number of users who are fans of the reviewer.
19. average\_stars\_user: The average rating given by the user.
20. compliment\_hot to compliment\_photos: Various types of compliments received by the user.
21. name\_business: The name of the business being reviewed.
22. address: The address of the business.
23. city: The city where the business is located.
24. state: The state where the business is located.
25. postal\_code: The postal code of the business.
26. latitude and longitude: The geographical coordinates of the business location.
27. stars\_business: The overall rating of the business.
28. review\_count\_business: The total number of reviews the business has received.

**Exploratory Data Analysis:**

The study aims at exploring the attributes for collinearity, co-relations and certain trends between the attribute independent variables.

The graph below on the left shows the distribution of stars in the 1,000 reviews of the sample that was used. It is clear that a large portion of the reviews, over 500, were given 5 stars. The next most frequent occurrence is 1 star, which was given about 200 times. 4 stars was the next most frequent review, followed by 2 and 3 stars.

The graph below on the right shows the distribution of average ratings for businesses in the sample. The average business rating is centered around 3 or 4 stars. This makes sense because it is exceedingly difficult for a business to please all customers. It is much more common for a business to have some incredibly positive 5-star reviews but also several negative 1- or 2-star reviews, resulting in an average of 3 or 4 stars.

A graph with blue rectangles

Description automatically generatedA graph with numbers and a bar chart

Description automatically generated

**Model Selection:**

The reason we chose an SVM for the model that included text mining is because SVMs are well-suited for high-dimensional data, making them appropriate for text classification tasks where the feature space is large (each word is a feature). They capture non-linear relationships in the data, providing flexibility in modeling complex patterns in the text.

The reason we chose a decision tree model as well for the dataset that included text mining is because in the context of analyzing text data, having an interpretable model allows for a clear understanding of the words or phrases contributing to specific star ratings in reviews. It is also computationally inexpensive, making it easier to run and easily repeatable.

**Results:**

Below are the results of the SVM (Support Vector Machines) model utilizing text mining. This dataset had over 7,000 variables, due to each word being used in a review being transformed into a variable. The accuracy of the SVM model below was 0.5117. The model was trained on 700 observations and tested on a test dataset of 300 observations. Because there are 5 classification categories, if someone randomly guessed the stars for all reviews, we would expect the model to have an accuracy of approximately 0.2. The model struggled to accurately predict 2-star ratings. It also struggled to distinguish 4 star and 5-star ratings. This could be due to the fact that 5-star ratings are clearly the most frequent in our sample.

A screenshot of a computer program

Description automatically generated

f(x) = β' \* x + b

Where:

* f(x) is the predicted output of the model (stars\_review)
* x is the input vector containing all features (i.e the words mined from the text reviews)
* β' is a combined coefficient representing the weighted influence of all features
* b is the bias term

The above is the model specification for the SVM model used. The train2 dataset included all words as a column and the number of occurrences as the values.

A screenshot of a computer screen

Description automatically generatedA screenshot of a computer screen

Description automatically generated

Next, we decided to move on to decision tree models. This model was run with the same sample as the SVM model above. The results have been inserted below. The accuracy of this model was 0.6187. This is approximately 10% more accurate than the SVM model previously shown. Unfortunately, this model had an issue in which it predicted a vast majority of the reviews as 5 stars. This could be due to the fact that there were more 5-star reviews in the training data than any other class.

A table with text on it

Description automatically generated

T(x) = argmax\_ {t ∈ T}\* P(y | t(x))

Where:

* T(x) is the predicted class label i.e., stars\_review
* t ∈ T is a decision tree in the space of all possible decision trees T.
* P(y | t(x)) is the conditional probability of class label y (stars\_review) given data point x and the decision tree t(x).

The above is the model specification for the decision tree classifier used.

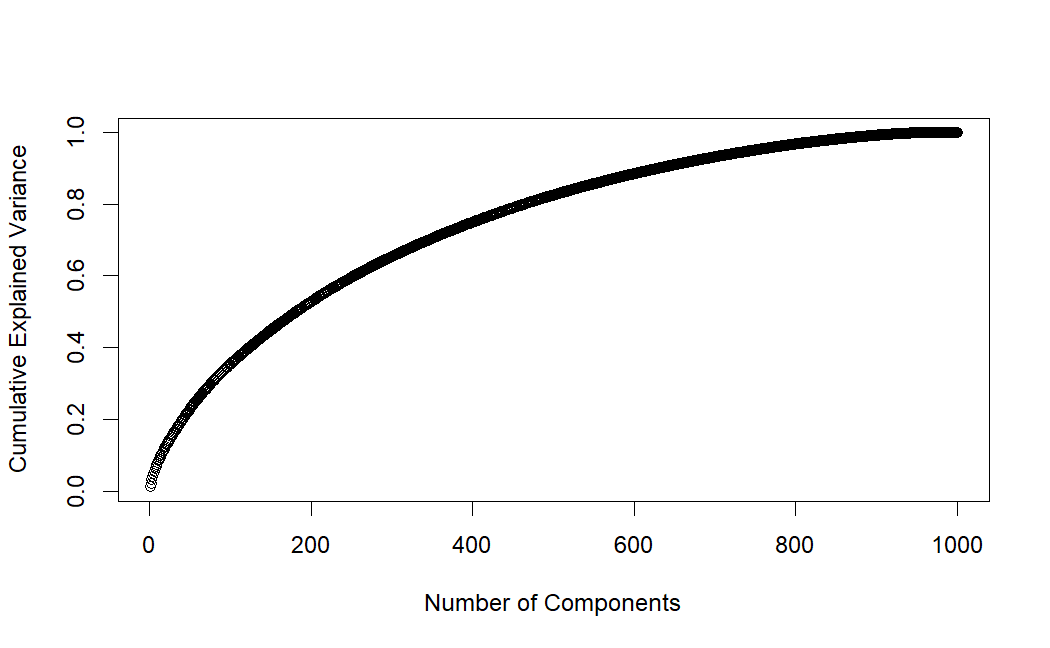
A screenshot of a computer

Description automatically generatedA screenshot of a computer screen

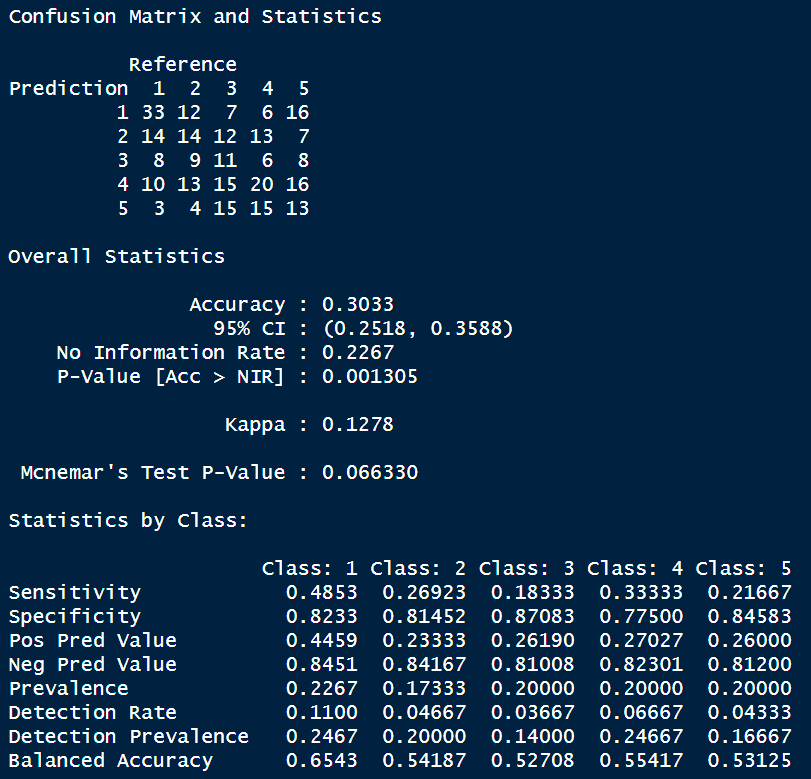
Description automatically generated

After this, we decided to try stratified sampling to see if this would improve model predictions. Using stratified sampling, a new sample of 1,000 reviews was selected. This included 200 5-star reviews, 200 4-star reviews, 200 3-star reviews, 200 2-star reviews, and 200 1-star reviews.

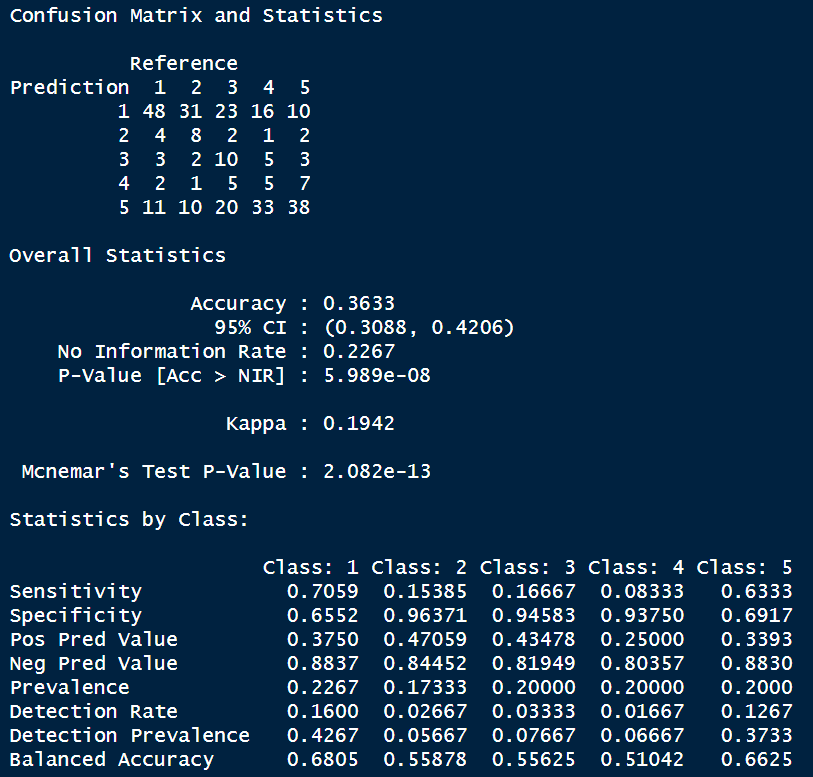
Next, principal component analysis (PCA) was used to reduce the dimensionality of the dataset due to the fact that there were over 7,000 variables. The results were plotted, which can be seen below. These results show that nearly all of the variation can be explained by the 800 most important variables. Therefore, those 800 variables were kept.



This smaller dataset to once again run a decision tree model.



Unfortunately, the accuracy decreased significantly to 0.3033. Next, we once again used the smaller dataset to create a model. This time, it was an SVM model. The results are shown below.



Once again, the performance of this model was much worse than the previous iteration. This time, the accuracy shown was 0.3633.

Both of these results lead us to the conclusion that utilizing random sampling was a better method for our objective than the stratified sampling strategy. In addition to this, the PCA analysis allowed us to reduce dimensionality and speed up the computations, but they did cause a slight decrease in performance as well.

**Quantitative feature analysis:**

The data had 22 columns with useful quantitative data specifically related to user reactions to a reviewer's review (stars\_review, useful\_review, funny\_review) and a reviewer's general status. A random sampling gave about 12,000 rows from the original data and later split it into 6718 to train and 5724 for testing. We ran our data through four models to see how accurately we could predict the business stars. We used a grid search on three of the models (decision tree, random forest and multi-class ordinal logistic regression).

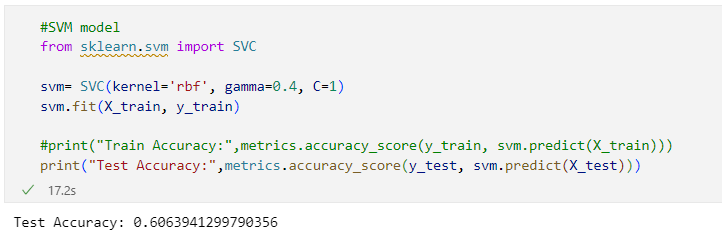
SVM: This was a pretty simple model to build and predict with. Using a gamma of 0.4, and a C of 1, we got a final test accuracy of 60.6%. This was the highest accuracy we could get from the model. Tweaking around with gamma and C had no increasing effect on final accuracy just decreased it but not by a significant amount. One thing we found interesting was the drop off in accuracy between the train set and the test data which wasn’t the same for the other models we built. We did not run a grid search for the SVM model because it was very computationally intensive.

SVM model specification:

f(x) = β' \* x + b

Where:

* f(x) is the predicted output of the model (stars\_review)
* x is the input vector containing all features (i.e the quantitative data of the columns)
* β' is a combined coefficient representing the weighted influence of all features
* b is the bias term



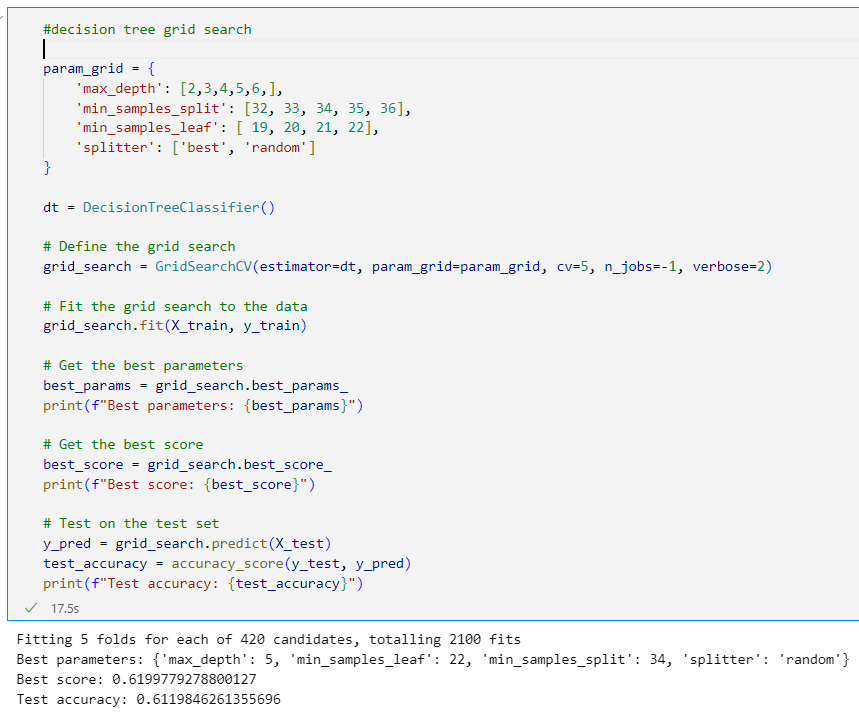
Decision Tree: Grid search came up with the best parameters, max\_depth:5, min\_sample\_leaf:22, min\_sample\_split:34 and splitter: random. We ended with 61% accuracy in the test data, close to what we had for the SVM model.

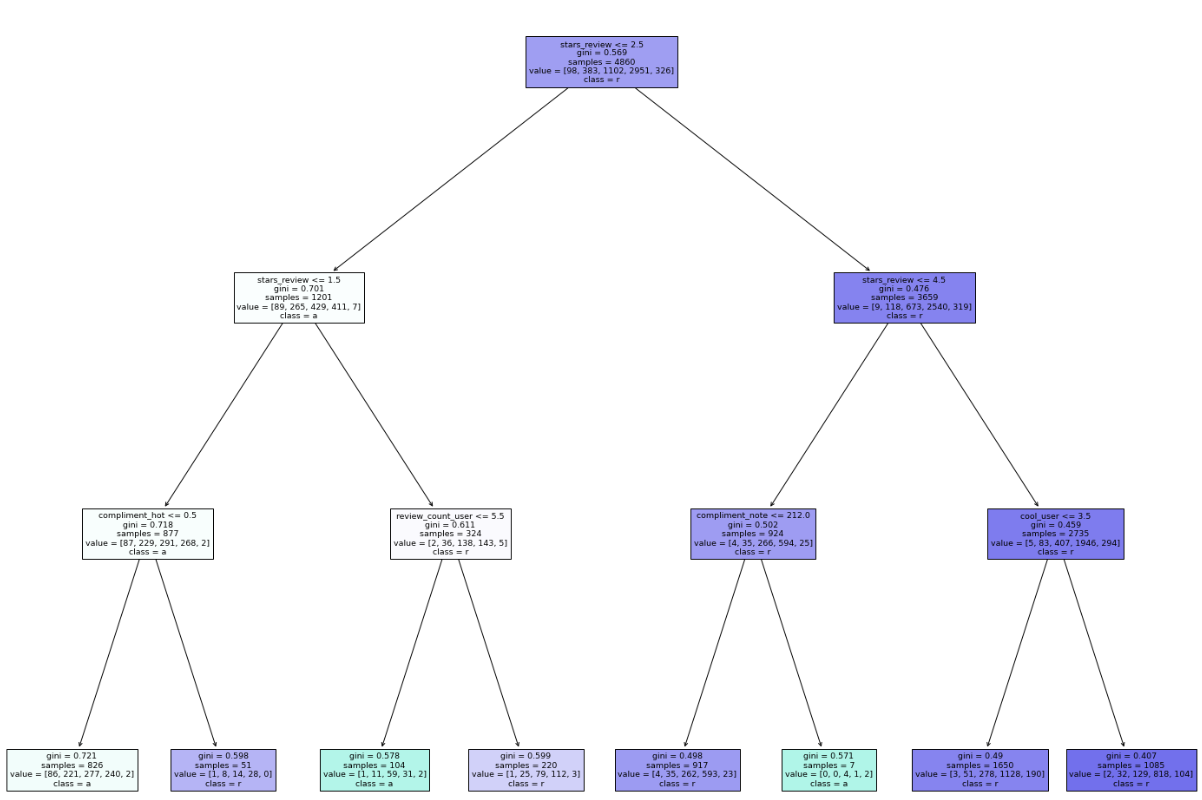
Decision tree model specification:

T(x) = argmax\_ {t ∈ T}\* P(y | t(x))

Where:

* T(x) is the predicted class label i.e., stars\_review
* t ∈ T is a decision tree in the space of all possible decision trees T.
* P(y | t(x)) is the conditional probability of class label y (stars\_review) given data point x and the decision tree t(x).



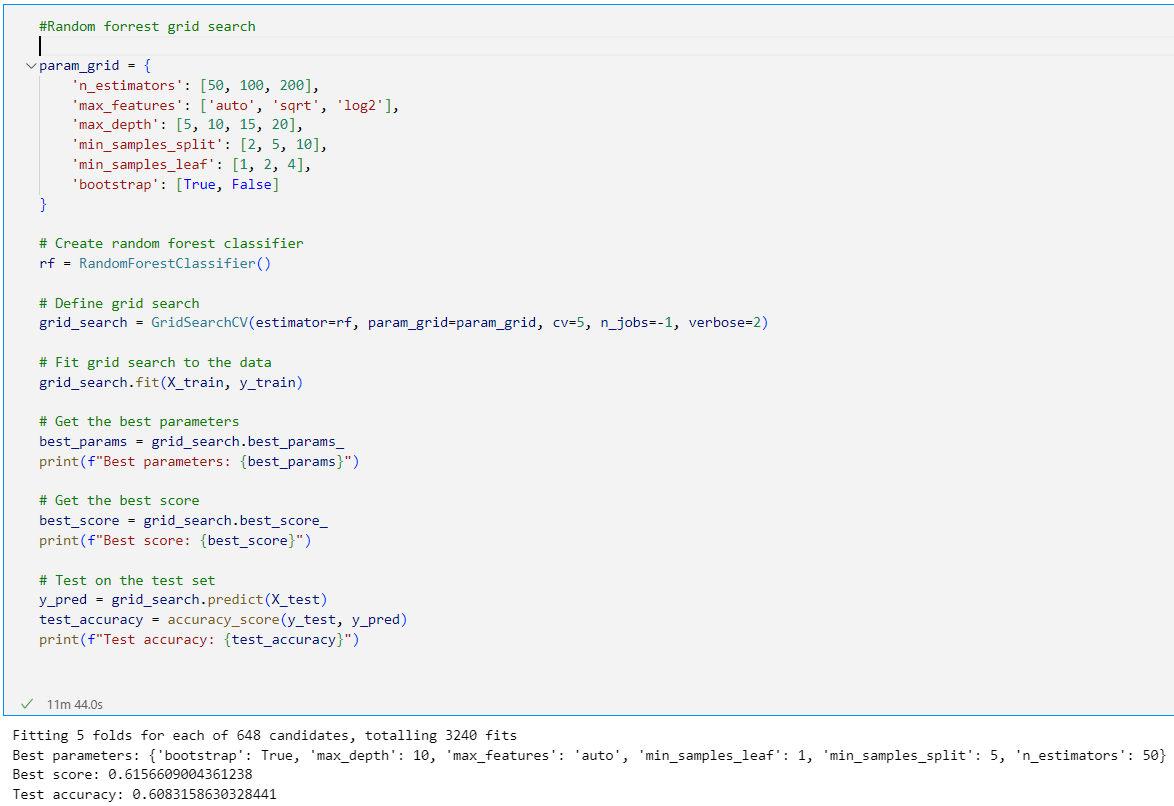


Random Forest: The random forest is supposed to be a combination of numerous decision trees, so it was a surprise that the test accuracy came out around 60.8% lower than the decision tree. Grid search gave a max\_depth: 10, max\_features: auto, min\_samples\_leaf:1 min\_samples\_split:5, and n\_estimators:50.

T(x) = Σ (a\_i \* y\_i) / N

Where:

* T(x) is the predicted class label i.e., stars\_review
* a\_i is the target variable.
* y\_i is the conditional probability of class label
* N is the number of decision trees



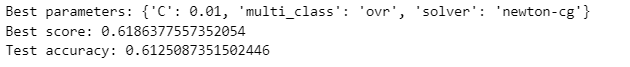
Multi-Class Ordinal Logistic Regression (MCOLR): This is a form of logistic regression that considers the ordinal form of the dependent categorical variables. In using this model, we achieved an accuracy of 61% . Grid search found the best parameters of C: 0.01, multi\_class: ovr, solver:newton-cg.

P(Y≤k)=1/(1+exp(−(β0(k) +β1 x1 +β2 x2 +…+βp xp )))

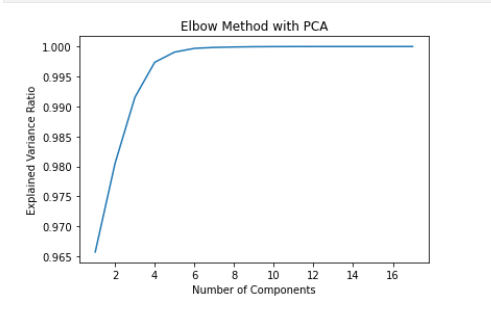
Where:

* P(Y≤k) is the predicted class label i.e., stars\_review
* x1, x2,…, xp represents the independent variable.
* Β1, β2 ,…, βp represents the coefficients associated with the independent variables

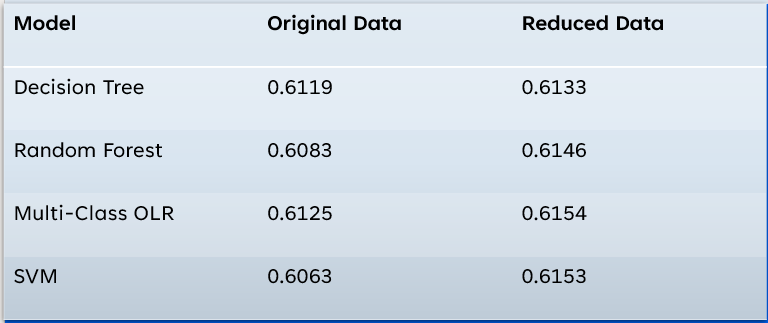




22 independent variables are a lot, so we ran PCA to reduce the dimensions with hopes of reducing the runtime and increase the final model accuracy. We used the elbow method to find the ideal number of components to run the PCA reduction. From the graph it looks like the ideal number will be 4 components.



Below is a table showing the final accuracies of the original data and the reduced data for the four models we ran. There was in increase in the final accuracy across the board on all models used.



**Limitations:**

One of the major limitations is that the dependent variable, review stars, is ordinal. This limited the available models that were used. Another limitation is that the data is dealing with humans who are inherently biased and hard to predict. Finally, most of the data is about food, however there was not a way to only filter to these restaurants.

Yelp reviews might be highly imbalanced, with a disproportionate number of positive or negative reviews. This can affect the model's ability to learn from both classes and may lead to biased predictions. Some reviews may express sentiments that are ambiguous or context dependent. For example, a review with a high rating might contain negative comments. Handling such nuances can be challenging for the model. The sentiment or nature of reviews may change over time due to various factors, such as changes in management, menu updates, or external events. A model trained on historical data might not perform well on more recent reviews.