M. Buczynski, S. Fortier

Professor Rattigan

CICS 397A

19 December 2022

Final Project Write Up

I. Introduction

The FoodData API provides access to FoodData Central ("FDC") as developed by the United States Department of Agriculture's Agricultural Research Service. FDC is a database featuring foundation, experimental, and branded foods among other things. This comprehensive library provides information regarding national and international branded foods and their nutritional labels, nutritional contents and portions per 'What We Eat in America' from NHANES, data published in peer-reviewed journals in collaboration with the USDA, and individually sampled minimally processed foods. There was no preprocessing that the data had to undergo in order to begin analysis. We decided to investigate the data about breads, specifically looking at nutrients and whether one can determine the healthiness of a bread based on nutrients.

II. API Access

In order to obtain a usable dataset, we had to make GET requests to the FoodData API. We were able to use Homework 2 as a reference since we had previously accomplished similar tasks, importing similar modules such as "csv", "requests", and "json". In terms of the write_to_csv function, there were many rows of data that were missing values. We wrote a few for loops checking which nutrients were present and subsequently filled the missing cells with a null value in order to proceed with a full dataset. When run, the program takes approximately

five to ten minutes to output the requested data as a csv, since traversing the response from the search by FDA ID endpoint is quite slow.

III. Clustering

We decided to cluster by “unhealthy” nutrients (fat, sodium, and sugar) when eaten in large amounts. We performed agglomerative clustering with eight clusters and ward linkage, which “minimizes the variance of the clusters being merged” per scikit-learn’s documentation. We attempted other linkage methods, though found that ward gave the most evenly dispersed groupings compared to the other methods. StandardScaler standardizes scores, similar to a Z statistic, which is helpful when doing clustering because the data becomes more normalized.

The clusters appeared relatively random. However, some clusters contained similar bread types, like ciabatta and baguette in Cluster 2. Other clusters, such as Cluster 6, had many different types of breads that would presumably not be grouped together like white bread, multigrain bread, and ciabatta. This was likely due to variation in the amount of nutrients within different types of bread by brand or style.

IV. Predictive Modeling

We choose to predict healthiness based on nutrients (fat, sodium, carbohydrates, fiber, sugars, calories, and protein). We predetermined which breads were healthy through Internet research. After adding a new column called ‘category’, which gave each bread either a “healthy” or “not healthy” value, we used a decision tree classifier, analyzing performance through accuracy, recall, and precision. The model’s performance varied everytime it ran, though it

remained in the range of .6 to .8. Therefore, we could conclude that the model is fairly accurate in predicting healthiness in bread, though it could be improved upon.

V. Visualization

One way that we visualized the data was through a heat map of every nutrient. The graph showed the correlation between every variable and gave a lighter colored box the more related two variables were. Evidently, fiber had very little (~.2) correlation with every other variable. The nutrients that had the strongest association were calories and carbohydrates as well as sodium and carbohydrates.

We also chose to visualize each nutrient in the form of faceted boxplots in an effort to see the difference in summary statistics for “healthy” and “unhealthy” breads. “Unhealthy” have a higher concentration of “unhealthy” nutrients, such as fat, carbohydrates, and sugars. Protein looks fairly similar between the two categories, while calories and sugars have the largest differences in means with respect to their scales.

The final tree plot helps visualize the previously created decision tree model. We can see that fiber is the initial variable that the data splits on, though several runs featured carbohydrates as the first node. We can see these same nutrients showing up repeatedly throughout the tree in an effort to refine previous decisions

VI. Challenges & Insights

The FoodCentral Data was quite difficult to work with given that we had to use two endpoints. The search by keyword endpoint provided some nutritional information and was much faster than the search by FDA ID, making the latter impractical to use. It was also

extremely difficult to test given its lagging speeds and the inconsistent information being provided in each response.

When working with the data, we found difficulty in adding our own columns for supervised and unsupervised purposes. While it was not included in the final product, we had made a column of simplified descriptions that classified each bread as one of the main eleven types. It was difficult simplifying our goals into pertinent functions. The same could be said for our attempt to pivot the data frame into a longer format for graphing purposes.

VII. Future

In the future, this data could be used as a determinant for what is deemed healthy. Our methods can also be applied to many different food groups, not just bread. It would be interesting to use this data to classify if the type of bread or other foods one is eating is healthy in hopes of making dietetic improvements. If we further developed our current work, we would like to see how algorithms decide to classify foods in terms of healthiness if they receive health data in response to specific diets.

VIII. Contributions

Mary helped write the sections about clustering, predictive modeling, visualization, and future. Mary mainly coded the sections about clustering and supervised learning, with help from Sophia, and graphed the heat plot and decision tree.

Sophia coded the usda_api.py file and worked with Mary in developing the boxplots as well as cleaning the initial data. She assisted in writing the Introduction, API Access, and Visualization sections. Both participated in editing each file to its final form.

IX. Works Cited

Buczynski, Mary. "Homework 7." 21 Nov. 2022. CICS 397A: Predictive Analytics with Python, University of Massachusetts Amherst, student assignment.

Fortier, Sophia. "Homework 2." 21 Sept. 2022. CICS 397A: Predictive Analytics with Python, University of Massachusetts Amherst, student assignment.

U.S. Department of Agriculture, Agricultural Research Service. FoodData Central, 2019. Fdc.nal.usda.gov.

Waskom, Michael. “Seaborn: Statistical Data Visualization.” *Journal of Open Source Software*, vol. 6, no. 60, 2021, p. 3021, https://doi.org/10.21105/joss.03021.