**Introduction to Problem and Data**

Obesity is a worldwide epidemic causing pain and suffering to individuals of all ages, as it significantly increases the risk of heart disease, stroke, type 2 diabetes, and cancer (World Health Organization, 2021). Obesity is also becoming ever more prevalent, as worldwide obesity has nearly tripled since 1975 (World Health Organization, 2021), and according to the CDC, over 42% of adults in the U.S. were classified as obese in 2020 (Centers for Disease Control and Prevention, 2020). In addition to the adverse physical health outcomes that obesity contributes to, being obese also carries a stigma that leads to poor mental health (Puhl et al., 2009). Hence, diagnosing obesity is crucial in order to receive needed healthcare services for combating it. It is also important for an obese individual to be aware of their condition so that they can make necessary lifestyle changes, potentially helping them avoid the dangerous physical and mental health effects that this condition entails.

Unfortunately, despite the importance of diagnosing obesity, many low-income and middle-income countries lack basic equipment like weighing scales and stadiometers in their primary care facilities, hindering the routine assessment of body mass index, also known as BMI (World Health Organization, 2017). In Sub-Saharan Africa, for example, obesity is underdiagnosed for these very reasons–there is limited access to standardized tools and anthropometric measurement practices (Popkin et al., 2011). For my final project, I attempted to develop a predictive model that can accurately decipher whether a person is obese or not based on factors other than their height, weight, and BMI. Although building a useful model without these important variables is a difficult task, the goal is for the model to at least improve the ability to diagnose obesity in these regions where resources are scarce. To measure success, I will score the accuracy of each model I create and compare it to a baseline score. In particular, success would be indicated by my best model beating the baseline score.

**Dataset Description**

I used the “Estimation of Obesity Levels Based on Eating Habits and Physical Condition” dataset from the UC Irvine Machine Learning Repository. This dataset includes data for individuals from Mexico, Peru, and Colombia, and the original dataset contained 17 attributes and 2,111 observations. For exploratory data analysis (EDA) purposes, I created an 18th attribute, ‘BMI’, that was not included in the original dataset. During my data exploration, I used the ‘BMI’ attribute as an alternative target variable to ‘Weight\_Level’ because it represents in a relatively precise manner how obese an individual is. Besides adding an extra column to the Data Frame, I made a few other key adjustments:

1. I changed the measurements for the ‘Height’ and ‘Weight’ columns from the metric system to the imperial system.
2. I changed numerous column names to make the Data Frame more readable because the original column names gave little-to-no indication of the variables they represented.
3. I changed the ‘Weight\_Level’ column’s observations from a more-complex set of weight classifications to ‘Obese’ and ‘Not\_Obese’.
4. I rounded the columns that were numerically segmented (eg. 0, 1, 2, 3, 4) to the nearest whole number, as some observations contained decimals, making them difficult to analyze in a categorical sense.

**Exploratory Data Analysis**

The following figures show boxplots of the features I have chosen to include in my binary classification models (‘Age (yrs)’, ‘Family\_History\_Overweight’, ‘High-Caloric\_Diet’, ‘Food\_Between\_Meals’, and ‘Monitors\_Calories’) against either ‘Weight\_Level’ or ‘BMI’.

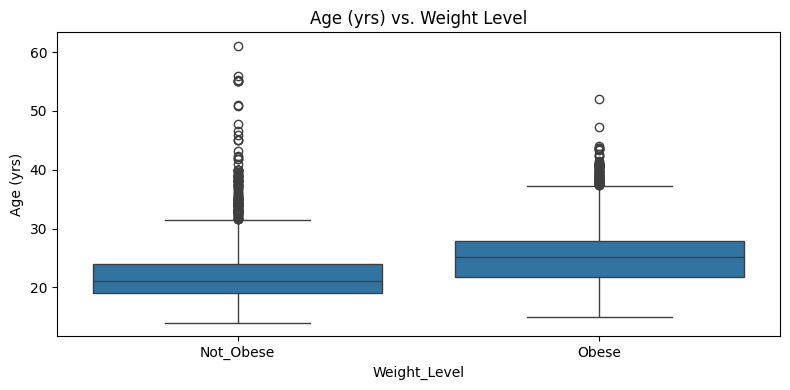
Figure 1:

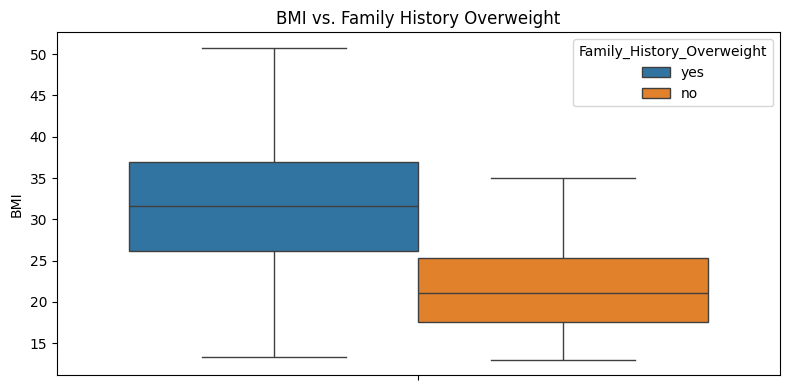
Figure 2:

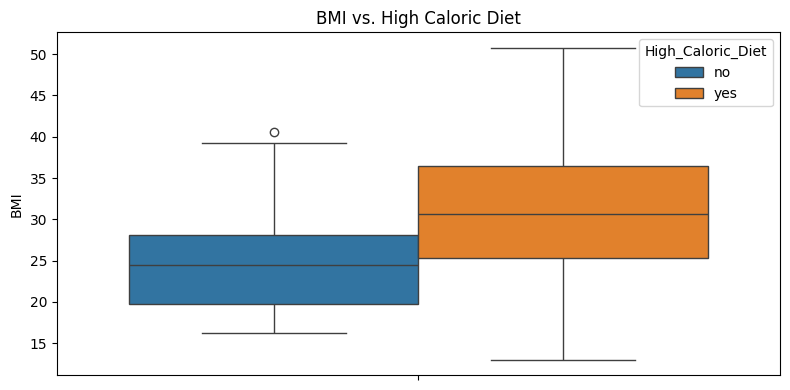
Figure 3:

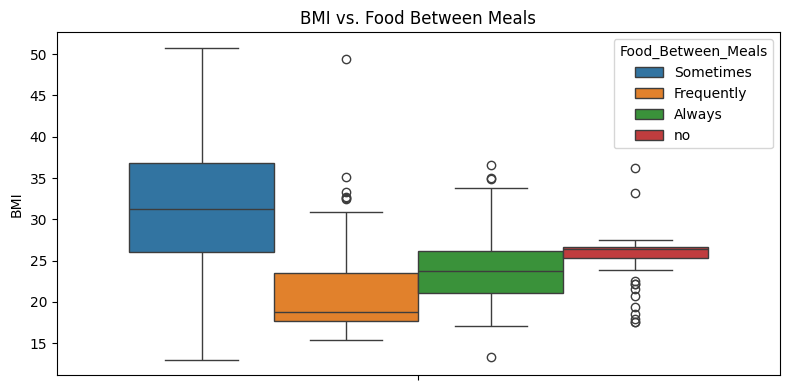
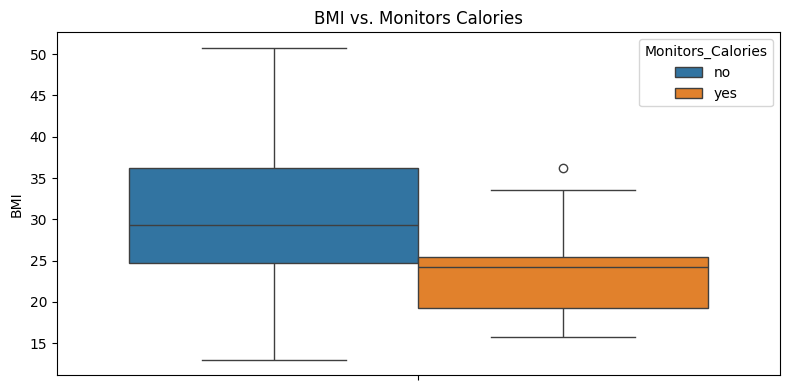
Figure 4:

Figure 5:

Based on these box plots, there seems to be an element of separation present between obese and non-obese individuals with respect to each of these features, making these features ideal to include in my models. Of course, there was a much greater degree of separation between obese and non-obese individuals when comparing ‘Weight\_Level’ to metrics such as ‘Weight’ or ‘BMI’, but including those features in my models would completely undermine my mission of predicting whether an individual is obese or not without the use of certain equipment, like scales.

**Models and Methods**

Before I created any models, I made a column transformer that scaled all numeric columns using StandardScaler (in case I decided to add more numeric features to my models) and encoded my categorical features using OneHotEncoder. I then created individual pipelines for each model using this column transformer. I also train/test split my data in the beginning, but for my grid searched models, I used cross-validation (cv = 5).

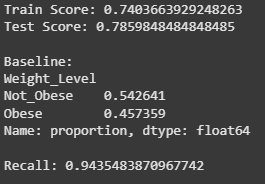
I decided to create four different models: 1) Logistic Regression since it works well on relatively small datasets (such as the one I am using for this project), 2) K-Nearest Neighbors (KNN) since classifications are based on the ‘closeness’ to other data points which means that no assumptions are made about the distribution of the data, 3) Decision Tree since it produces rules that are easy to interpret and handles both numerical and categorical features well, and 4) Random Forest since it combines many decision trees to reduce overfitting and highly-varied results.

Although there is a rationale that makes each of these 4 models attractive to use, I will be looking for the model that is differentiated by having the highest score. However, another important point of differentiation is having the lowest number of false negative predictions (predicting that someone is not obese when they actually are obese). If someone is misdiagnosed as not being obese, then they will not receive the necessary treatment for their condition, posing significant health risks. Therefore, we want to limit false negative predictions as much as possible, and so that will also be considered to differentiate models with similar accuracy scores.

**Baseline Model**

To get my baseline model, I simply took the normalized value counts of the ‘Weight\_Level’ column. The higher proportion out of the two possible values in that column was ‘Not\_Obese’ (~54.26%). Hence, I used 54.26% as the baseline score.

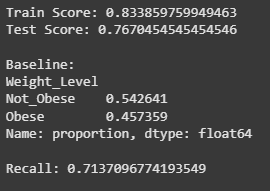
**Results and Interpretation #1: Logistic Regression**

Figure 6:

Overall, my Logistic Regression model performed better than my baseline. Both the training data and testing data outperformed the baseline, and the training data performed slightly lower than the testing data. This model also has an outstandingly high recall, meaning that there is a relatively small number of false negatives. I believe the Logistic Regression model particularly performed well due to its stronger performance with smaller datasets, as I mentioned earlier. Also, and perhaps even more importantly, Logistic Regression treats categorical features as binary flags after they are OneHotEncoded, making it a strong model for categorical features. Since my features are predominantly categorical, Logistic Regression was not only accurate, but had very strong recall.

The ‘Family\_History\_Overweight’ feature was by far the most important feature in this scenario, and some less-important but still influential features were ‘High\_Caloric\_Diet’, ‘Food\_Between\_Meals’, and ‘Monitors\_Calories’. The least effective feature in this model was ‘Age (yrs)’.

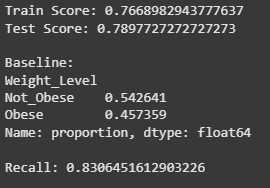
**Results and Interpretation #2: KNN**

Figure 7:

My KNN model also performed better than the baseline. Both the training and testing data had higher scores than the baseline model, with the train score being a decent amount higher than the test score, which may suggest slight overfitting. Also worth noting is the significantly lower recall compared to the Logistic Regression model, meaning that the KNN model did a worse job preventing false negatives. To mitigate the overfitting, I made another KNN model using GridSearchCV to find the optimal number of neighbors to use, and while it lowered the degree of overfitting, it did not produce any other meaningful results, as the test score and recall were highly similar to the original test score and recall, respectively. The KNN model may have performed worse overall since my data appears sparser after using OneHotEncoder. Since encoding my categorical features creates many 1s and 0s in my dataset, KNN views 0 and 1 as equidistant, which may slightly distort the overall results since many of my features were categorical. Also, KNN does not learn patterns, which could have contributed to its worse performance compared to the logistic regression model.

Again, ‘Family\_History\_Overweight’ was the most important feature. ‘High\_Caloric\_Diet’ and ‘Food\_Between\_Meals’ followed.

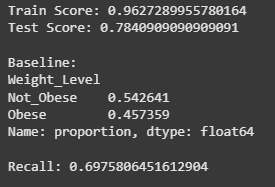
**Results and Interpretation #3: Decision Tree**

Figure 8: 

Like the other models thus far, my decision tree model outperformed the baseline, both in terms of the train data and the test data. The train score was very slightly lower than the test score. The recall on this model was significantly improved compared to the KNN model, but it was still lower than the recall on the Logistic Regression model. Hence, despite the scores of the Decision Tree model and the Logistic Regression model being similar, I would still lean towards the Logistic Regression model since it does a better job limiting false negative predictions. I tried making another Decision Tree model using GridSearchCV to see if the model would perform better, but it actually ended up performing worse in terms of the test score and the recall, making it less desirable compared to the original model. Still, overall, the Decision Tree model performed well. Since decision trees split based on differences in features, and categorical features are already naturally split (eg. monitoring calories vs. not monitoring calories), the model probably made more-accurate predictions with my dataset since I am mostly using categorical features.

Yet again, this model was mostly influenced by ‘Family\_History\_Overweight’, but ‘Food\_Between\_Meals’ and ‘Age (yrs)’ were also influential–the remaining 2 features weren’t.

**Results and Interpretation #4 - Random Forest**

Figure 9:

My fourth and final model–the Random Forest model–had train and test scores that both outperformed the baseline. However, something immediately concerning was the disparity between the train score and test score: the train score was significantly higher, suggesting overfitting. I tried to mitigate this by determining the optimal number of estimators to use through GridSearchCV, but grid searching only made the model worse in every metric, and the gap between the train and test scores remained roughly the same. The recall on this model was also worse than any other model, despite the comparable test score, making this model closest in performance to the KNN model. I originally assumed that the Random Forest model would be strongest since it ensembles, which should have reduced overfitting and improved overall accuracy. It may not have performed this way due to my dataset being relatively small. Since Random Forest performs best with large datasets, my model probably could not benefit from ensemble diversity. Also, since Random Forest essentially combines many decision trees, a large number of useful features would have probably been necessary, but my dataset only had 5 features that at least somewhat separated weight levels, which was likely insufficient.

Unlike all other models, ‘Family\_History\_Weight’ was not the most important feature. Instead, it was second in importance (by a pretty wide margin) to ‘Age (yrs)’. The next most influential feature was ‘Food\_Between\_Meals’, and in relativity, the remaining features were not very important.

**Conclusion and Next Steps**

When all is said and done, each one of the models I constructed was an improvement over the baseline model, demonstrating their success and usefulness. Based on my assessment of each model’s scores and recall, here is how I would rank them from best to worst: Logistic Regression, Decision Tree, KNN, and Random Forest.

Key Findings:

1. Success of Logistic Regression Model: The Logistic Regression model turned out, holistically, to be the best-performing model when considering its accuracy and recall compared to those of the other models.
2. Impactful Feature: ‘Family\_History\_Overweight’ proved to be the most impactful feature overall, as it had the heaviest influence in 3 out of the 4 models. In the only model where it did not have the highest influence, it was the second most significant feature. The emphasis on this feature aligns with a person’s genetic makeup being a key contributor to their weight level.
3. Importance of Size of Data and Feature Types: The success of each model, in this case, primarily depended on 2 things: 1) its ability to handle a relatively small dataset and 2) its ability to work with categorical variables. The two better models (Logistic Regression and Decision Tree) were stronger in these respects, while the two worse models (KNN and Random Forest) struggled to handle smaller datasets and categorical variables.

To further improve these models’ capability to predict whether or not an individual is obese, I would like to incorporate additional features into my future models:

1. Additional Family Data
   1. I would like to further analyze data about individuals’ families, such as the eating habits of family members, given that family history was such a significant factor that contributed to the predictability of my models.
2. Socioeconomic Data
   1. Data indicating an individual’s socioeconomic standing, such as their income, could potentially help improve the predictability of my models, as it is typically more expensive to maintain a healthy lifestyle due to the higher prices of fruits and vegetables compared to unhealthy foods. Also, living a healthy life may entail incurring expenses such as obtaining a gym membership and hiring a personal trainer.
3. Data Indicating Stress Level
   1. For many individuals, food is a comforting escape from stressful situations, which prompts them to eat more, thus making them obese. Hence, I would like to incorporate metrics that indicate how stressed an individual generally is, such as the number of hours they work per week, the intensity level of their job, the number of children they have, etc. This could potentially improve the predictability of my models and give us insight into the effects of a stressful environment on an individual’s weight level.

These additional variables would help make future models more complex, contributing to their accuracy since they would be grounded not only in more data, but more highly-influential data. This approach would significantly lessen the impact of the lack of diagnostic resources in regions like Sub-Saharan Africa. With effective predictive models, millions of individuals around the world could gain access to much-needed obesity-related healthcare, improving not only health, but also quality of life.

**References**

Centers for Disease Control and Prevention. 2020. *Adult Obesity Facts*.https://www.cdc.gov/

obesity/data/adult.html

Popkin, Barry M., and Linda S. Adair. 2011. "Global nutrition transition and the pandemic of

obesity in developing countries." *Nutrition Reviews* 70(1): 3–21. https://doi.org/10.1111/j.

1753-4887.2011.00456.x

Puhl, Rebecca M., and Chelsea A. Heuer. 2009. “The Stigma of Obesity: A Review and Update.”

*Obesity* 17(S2): S41–S64. https://doi.org/10.1038/oby.2009.511

World Health Organization. 2021. *Obesity and Overweight*. https://www.who.int/news-room/fact

-sheets/detail/obesity-and-overweight

World Health Organization. 2017. *Tackling NCDs: 'Best buys' and other recommended*

*interventions for the prevention and control of noncommunicable diseases*.https://apps[.](https://apps.who.int/iris/handle/10665/259232)

who.int/iris/handle/10665/259232