Estimation of Obesity Levels

Using classifiers and regression techniques

Yihe Chen Qianqian Gu

ABSTRACT

This paper provides an overview of the data mining process used to predict the estimation of obesity levels in individuals from the countries of Mexico, Peru, and Colombia, based on their eating habits and physical condition. This analysis was conducted as the final project of CSC240 at the University of Rochester in Fall 2020.

KEYWORDS

data mining, high dimensional data, regression, classification, obesity

1 INRTRODUCTION

Obesity is an increasingly pressing issue in our society. More and more people are living an unhealthy lifestyle unconsciously. The cause of disease involves multiple factors and the disease itself is not just a cosmetic concern. People who are striving to lose weight still get the disease sometimes. Additionally, obesity may lead to heart disease, heart disease, and even certain cancers which can be fatal. Therefore, by analyzing the components that attribute to obesity, we would gain insight into what can possibly help people to levitate the concern.

The whole dataset contains 17 attributes and 2111 records, 23% of the data was collected directly from users through a web platform, and 77% of the data was generated using the SMOTE filter and the Weka tool. The initial recollection of information was made through a web page using a survey where users had evaluated their eating habits and some aspects that helped to identify their physical condition. The survey was accessible online for 30 days and then the original 485 records (composed 23% of the data) were collected[1].

Since our target variable "NObeyesdad" (Level of Obesity) is a class variable and is classified based on a numerical index called Mass Body Index(MBI) (1)

$$Mass\ Body\ Index = weight \div (height * height) \quad (1)$$

After 485 records of target variables were collected and the mass body index for each variable was calculated, the results were compared to a standard provided by WHO and the Mexican

Normativity[2]: Underweight less than 18.5; Normal 18.5 to 24.9; Overweight I 25.0 to 27.0; Overweight II 27.0 to 29.9; Obesity I 30.0 to 34.9; Obesity II 35.0 to 39.9; Obesity III Higher than 40. Since the final class variable is referred to as the numerical variable "MBI", we created this new variable based on original height and weight data. Then, we used both classification and regression techniques to predict "NObeyesdad" and "MBI" respectively and assessed and compared techniques as a whole.

Our approach follows steps:

1.Exploratory Data Analysis: Handled missing values. Conducted to do data visualization and understand distributions of the overall data.

2.Data Preprocessing: Removed outliers; Standardized numerical features and label encoded categorical features.

3.Data Analysis: Implemented regression and enhanced model on linear regression; Constructed different classification methods; Compared different methods in two categories; Tried with PCA and LDA to improve accuracy of model.

2 EXPLORATORY DATA ANALYSIS

First, we wanted to gain an understanding of the dataset by performing exploratory data analysis with Python's numpy and pandas libraries.

2.1 Data Exploration

There are 8 numeric variables and 9 categorical variables with no missing value.

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2111 entries, 0 to 2110
Data columns (total 17 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Gender                          2111 non-null   object
 1   Age                             2111 non-null   float64
 2   Height                          2111 non-null   float64
 3   Weight                          2111 non-null   float64
 4   family_history_with_overweight  2111 non-null   object
 5   FAVC                            2111 non-null   object
 6   FCVC                            2111 non-null   float64
 7   NCP                             2111 non-null   float64
 8   CAEC                            2111 non-null   object
 9   SMOKE                           2111 non-null   object
 10  CH2O                            2111 non-null   float64
 11  SCC                             2111 non-null   object
 12  FAF                             2111 non-null   float64
 13  TUE                             2111 non-null   float64
 14  CALC                            2111 non-null   object
 15  MTRANS                          2111 non-null   object
 16  NObeyesdad                      2111 non-null   object
dtypes: float64(8), object(9)
memory usage: 280.5+ KB
```

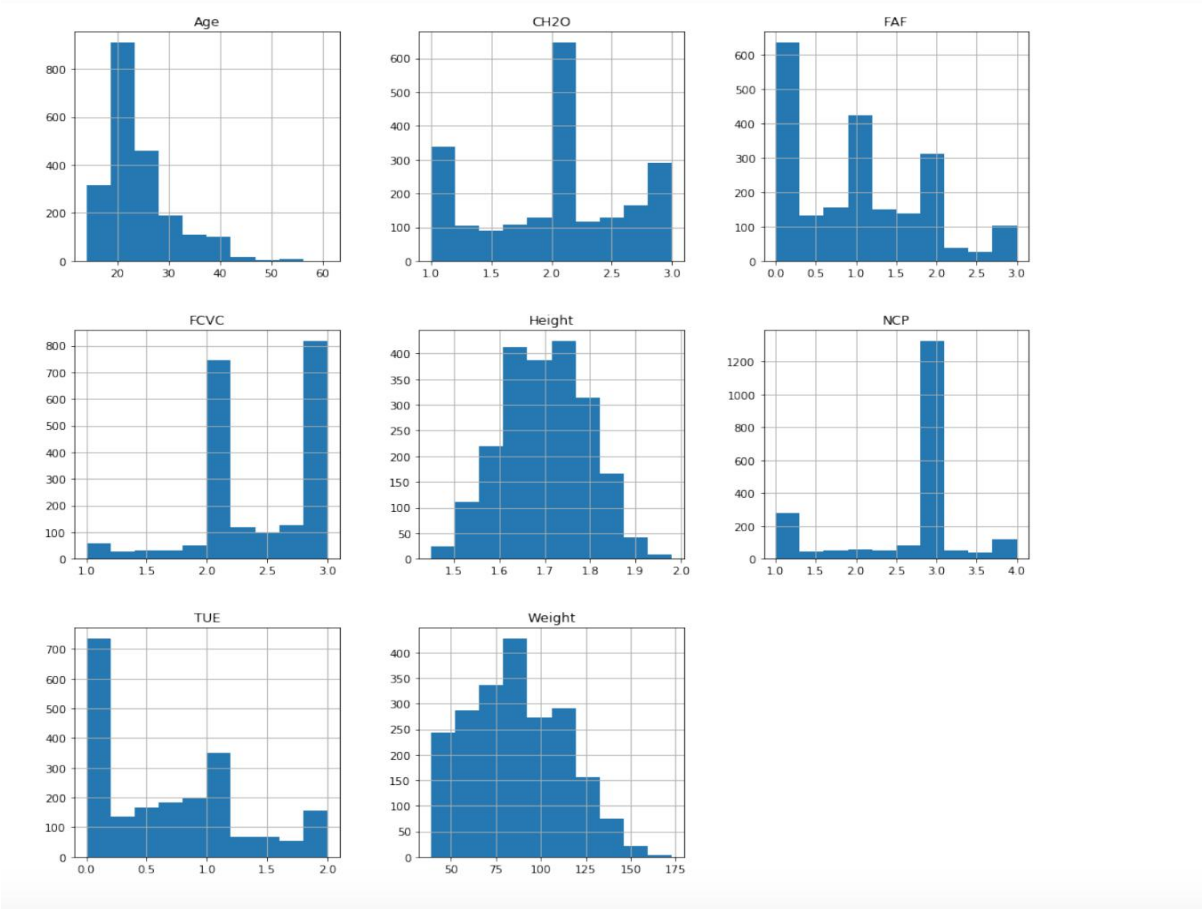
```
df.isnull().any()
```

```
Gender                            False
Age                               False
Height                            False
Weight                            False
family_history_with_overweight    False
FAVC                              False
FCVC                              False
NCP                               False
CAEC                              False
SMOKE                             False
CH2O                              False
SCC                               False
FAF                               False
TUE                               False
CALC                              False
MTRANS                            False
NObeyesdad                        False
dtype: bool
```

2.2 Distribution of Numeric Features

The numeric attributes are: Age, Height, Weight, Frequency of consumption of vegetables (FCVC), Number of main meals (NCP), Consumption of water daily (CH20), Physical activity frequency (FAF), Time using technology devices (TUE).

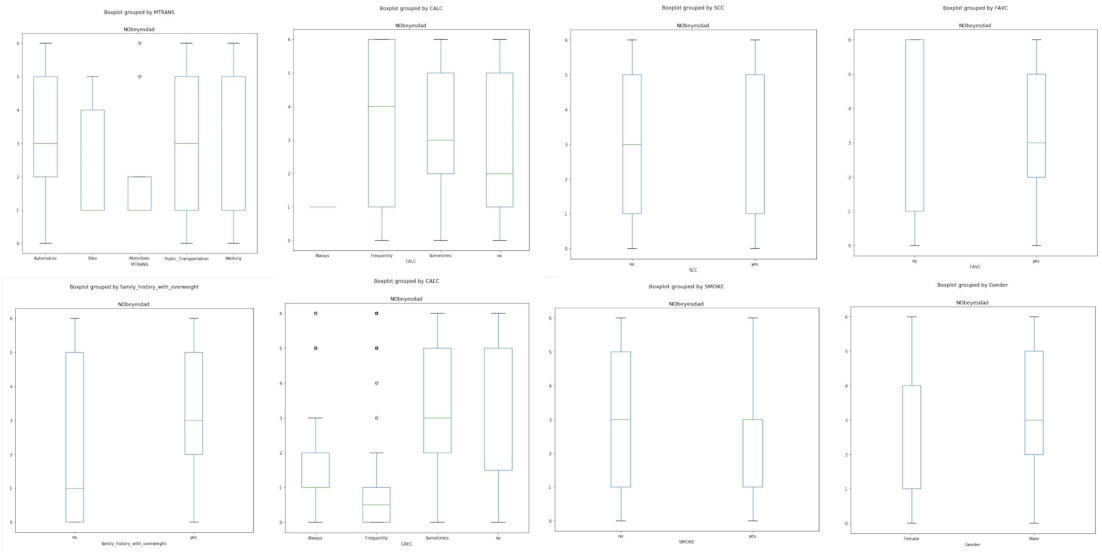
For each numeric variable, the frequency of occurance was plotted against its value so that we were able to visualize its distribution.



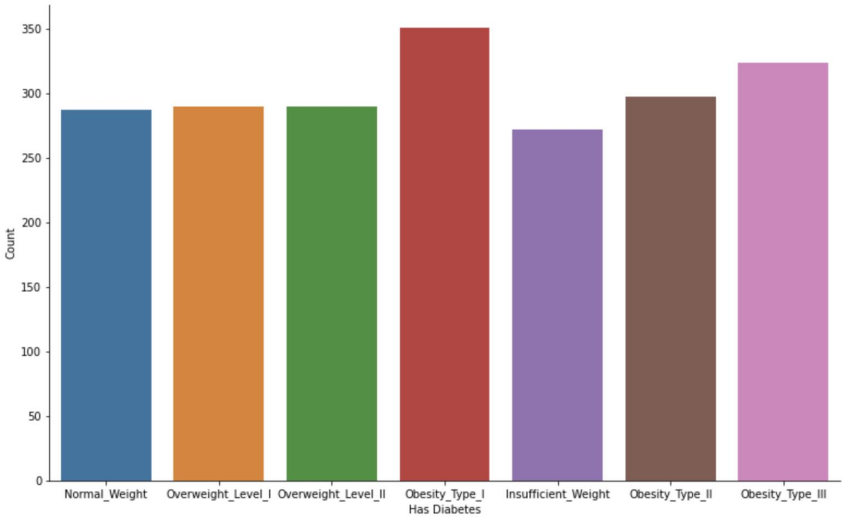
2.3 Distribution of Categorical Features

The categorical features include: Gender, Family history with overweight, Smoking habit (SMOKE), Frequent consumption of high caloric food (FAVC), Consumption of food between meals (CAEC), Consumption of alcohol (CALC), Calories consumption monitoring (SCC), Transportation used (MTRANS).

For each categorical feature, we plotted obseity level category against its levels. This allowed us to visualize how each level of the each variable was distributed across the target varible.



And finally we have our target variable, Obseity level category (NObeyesdad). Its distribution was plotted as below. We were glad to see a mostly balanced distribution across classes, which provided us a good foundation for model fitting.



3 DATA PREPROCESSING

We needed to clean data by removing outliers and to appropriately deal with both numerical values and categorical variables.

3.1 Data Cleaning

```
from scipy import stats
z_scores = stats.zscore(df1)
abs_z_scores=np.abs(z_scores)
abandoned_entries=(abs_z_scores>=3).all(axis=1)
print(abandoned_entries)
```

```
[False False False ... False False False]
```

Our analysis shows that there is no outliers present in dataset. This is already a well-cleaned dataset.

3.2 Data Preprocessing

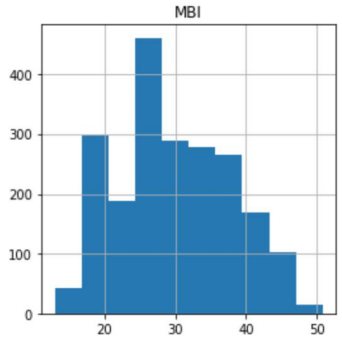
Although the distributions of numerical values are mostly normal, we standardized every numerical variable to ensure the models are not favoring one feature over another due to their magnitude.

We encoded every categorical attribute into a dummy variable. In this way, we were able to prevent cases in which models may mistake a nominal variable as ordinal and thus deduce a non-existing trend.

4 ANALYSIS AND MODELING

The final processes were performing data analysis and assessing models. Firstly, we split the labeled training data into a training set and a test set, while the training set composes 80 percent of the data and the test set composes the rest. The purpose is to avoid overfitting and to test the accuracy of the model against data that was not used to train the model.

Secondly, we considered a log transformation on the target variable MBI. But since the sk ewness of this variable is 0.154 and the histogram exhibits a normal distribution, as show n below, log transformation is not necessary.



We intended to do Principal Component Analysis(PCA) for regression to reduce dimensionality and to do feature selection but found the explained variance ratio for this method is relatively low(only 22.6%), so we dropped this method.

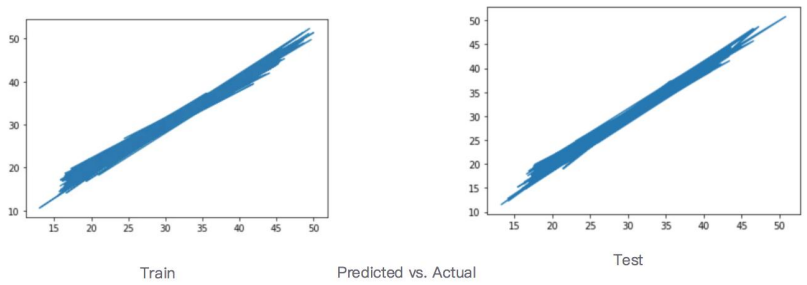
For model assessment, we mainly used the accuracy score for classification and mean squared error and R2-score for regression. The accuracy score is similar to precision and is given by the ratio of the number of correct predictions divided by the number of samples. In regression, because the predicted value is a continuous variable, we instead wanted to quantify how overall we mispredict the target variable. So we used metrics such as mean absolute error, mean squared error, R2-score for either trained set or test set.

4.1 Regression

For regression techniques, we first implemented linear regression as our baseline model, and then we implemented three enhancement models on linear regression to see if there is improvement over the baseline model and to see if it is necessary to change the baseline model. It turned out that Ridge Regression has the highest R2-score(0.9919775) and lowest RMSE value(0.7509505) for test set among all the models. After we analyzed the score for each coeffcient, we found for Ridge Regression, four most features in determing the value of MBI(thus the obesity level) are Weight (+), Height (-), FAVC_no: Frequent consumption of high caloric food(No)(-), and GAEC_Always: Consumption of food between meals(Do you eat any food between meals-Always) (+) (with direction indicated). Four most significant features in Linear Regression are Weight (+), Height (-), GAEC_Always: Consumption of food between meals(Do you eat any food between meals-Always) (+), and Gender_female (+).

4.1.1 Baseline Model: Linear Regression

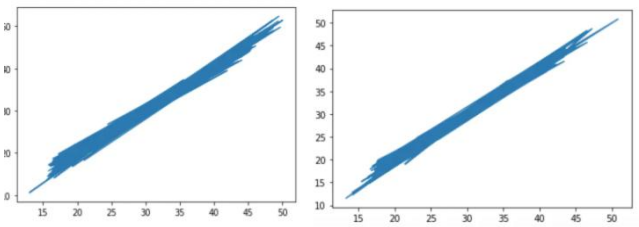
Simple linear regression was selected as our baseline model to predict the continuous variable "MBI" based on a set of observed features. The regression coefficients here solved by the method of Ordinary Least Squares(OLS), which minimizes the error between the actual data and the estimate of the regression line. The results from the test set indicate that the OLS regression model does a good job of predicting and fitting the data. The results for both train and test set are shown below.



Train RMSE	0.7383157526389027
Train R-sq	0.9915053025150218
Test RMSE	0.7509519549924226
Test R-sq	0.9911977106290695

4.1.2 Ridge Regression

Our next step was to use Ridge Regression, which is a technique for analyzing multiple regression data. It would make our model more robust if it suffered from multicollinearity. When multicollinearity occurs, least squares estimates are unbiased, but their variances are large so they may be far from the true value. OLS predictions are also highly sensitive to feature beta. Ridge imposes a penalty on the size of beta, therefore reducing the overfitting issue. It is hoped that the net effect will be to give estimates that are more reliable. We used the scikit-learn package to train the model. The results show an improvement over the OLS regression model.

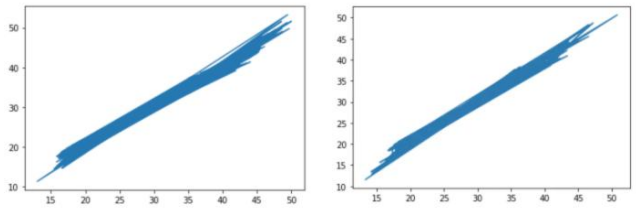


Ridge Regression

Train RMSE 0.7383157632134331
Train R-sq 0.9915053022716911
Test RMSE 0.7509504847268145
Test R-sq 0.9911977450964959

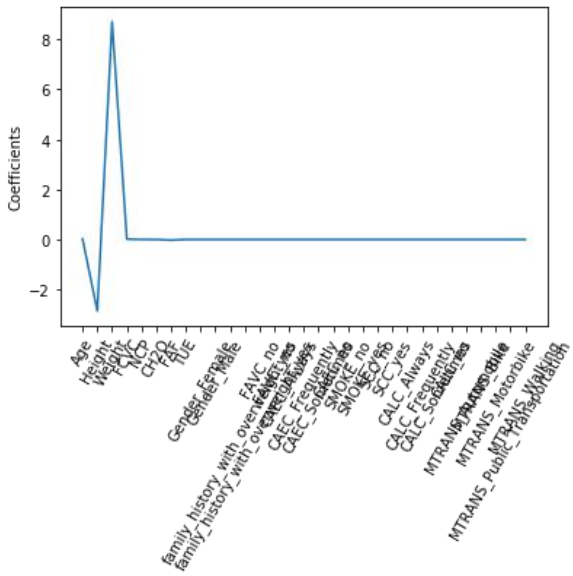
4.1.3 Lasso Regression

Lasso Regression was the next enhancement we explored. Unlike Ridge Regression which imposes a penalty on the size of the coefficients (L2), Lasso Regression imposes a penalty on the number of coefficients in the model(L1). Thus, this regularization may help us do feature selection in lieu of Principal Component Analysis's covariance matrix decomposition. But similar to Ridge Regression, Lasso Regression also minimizes the residual sum of squares with a penalty. The accuracy of this model and the feature selection results are shown below. The score of each coefficient was plotted with Lasso Regression. Other insignificant variables all have scores around zero, indicating that Weight and Height are almost the two only significant variables in determining MBI.



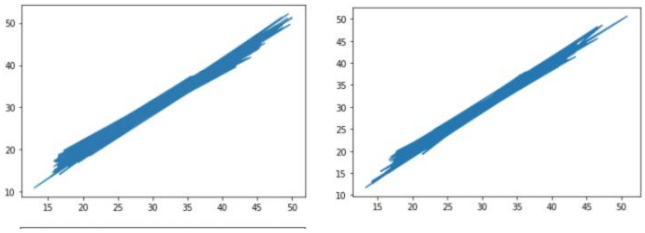
Lasso Regression

Train RMSE 0.8505151264999712
Train R-sq 0.9887273056376994
Test RMSE 0.8168996864882547
Test R-sq 0.9895838123323265



4.1.4 Elastic-Net

The Elastic-Net Regression combines the L1 regularization of the Lasso regression and the L2 regularization of the Ridge regression. It overcomes the limitations of the Lasso Regression (least absolute shrinkage and selection operator) and also optimizes the hyperparameter of Ridge Regression. The results of the regression are shown below.



Elastic Net Regression

Train RMSE 0.7477098458312271
Train R-sq 0.9912877596704566
Test RMSE 0.7512987105988517
Test R-sq 0.9911895797550173

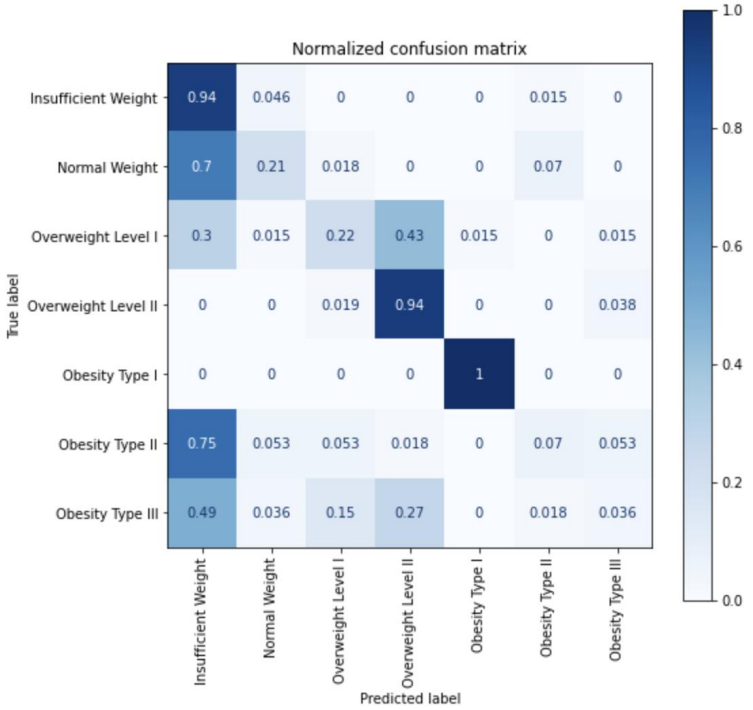
4.2 Classification

The three classification model we chose were: Naive Bayes, Decision Tree, and Support Vector Machine. We first wanted to see how each of the three models perform with the preprocessed data.

4.2.1 Baseline model: Naive Bayes

Naive Bayes is a supervised learning method based on applying Bayes' theorem with strong (naive) feature independence assumptions. In theory, it has the smallest error compared with other classification methods. However, that is rarely the case when applying to a real life dataset, because the independence assumption cannot always be met. When the number of features is large or when the correlations between features are high, naive bayes usually doesn't deliver the best classification. With that in mind, we were not surprised to see an average performance.

The predictions were not very consistent with the true classes, and the accuracy was only 50% given our data.

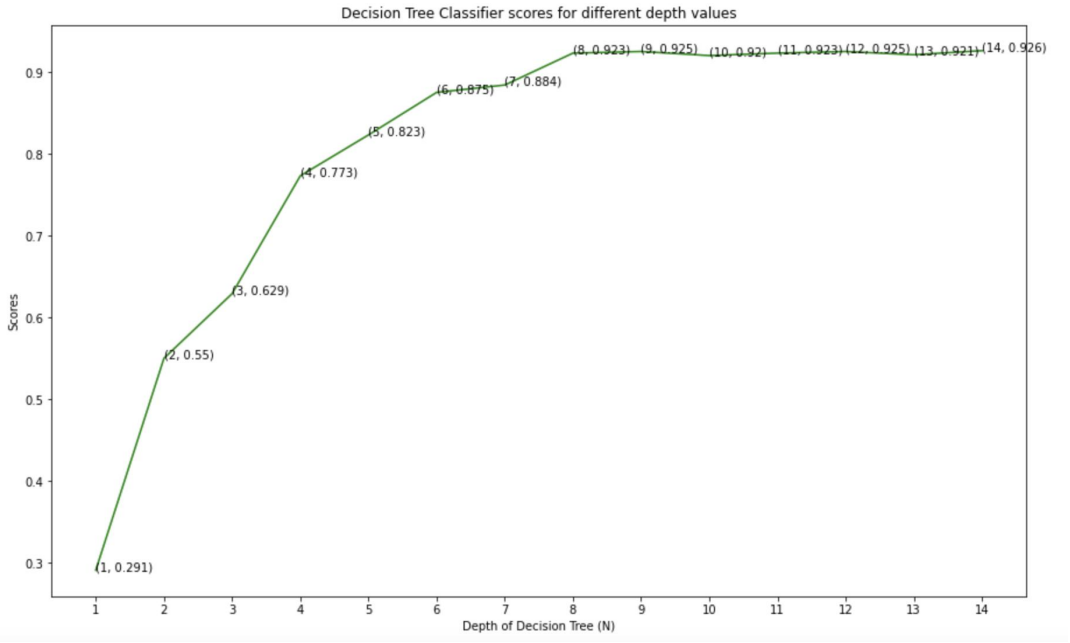


	precision	recall	f1-score	support
Insufficient_Weight	0.32	0.94	0.48	65
Normal_Weight	0.57	0.21	0.31	57
Obesity_Type_I	0.54	0.22	0.32	67
Obesity_Type_II	0.53	0.94	0.68	53
Obesity_Type_III	0.99	1.00	0.99	69
Overweight_Level_I	0.40	0.07	0.12	57
Overweight_Level_II	0.25	0.04	0.06	55
accuracy			0.50	423
macro avg	0.51	0.49	0.42	423
weighted avg	0.52	0.50	0.44	423

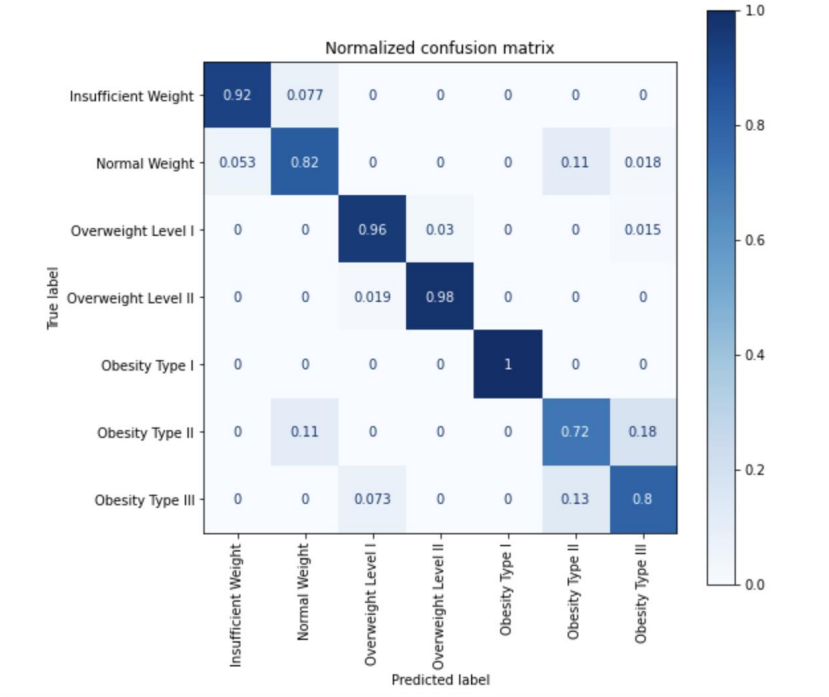
4.2.2 Decision Tree

Decision trees classify the examples by sorting them down the tree from the root to some leaf or terminal node, with the terminal node providing the classification of the example.
Each node in the tree acts as a test case for some attribute, and each edge descending from the node corresponds to the possible answers to the test case. This process is recursive in nature and is repeated for every subtree rooted at the new node.

To find the optimal depth for our decision tree classifier, we plotted the results of decision_scores over a range of depth of 1 to 14 as shown below.



We observed an increase of score as depth increases, until the score reached a plato at depth 8, indicating that the optimal depth catered to our dataset was most likely to be 8. Therefore we trained the decision tree classifier with maximum depth 8 and obtained the following result.

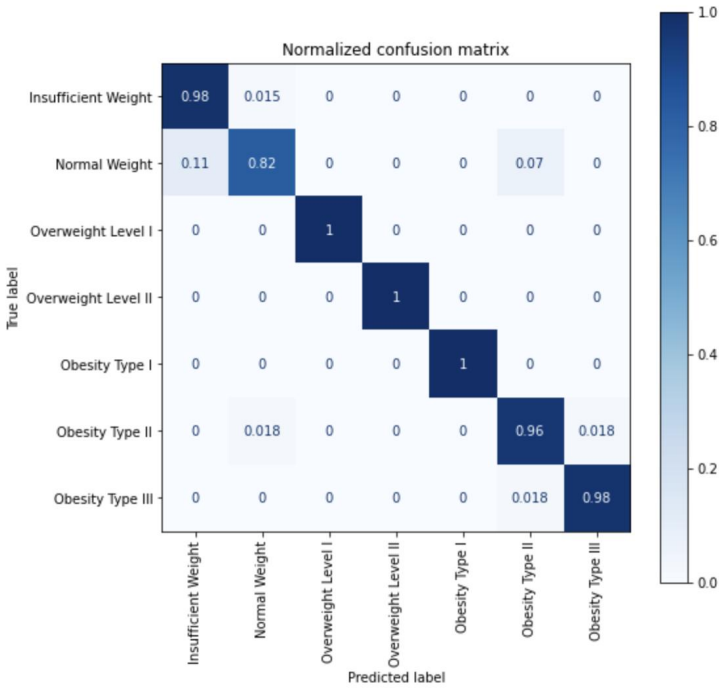


	precision	recall	f1-score	support
Insufficient_Weight	0.95	0.94	0.95	65
Normal_Weight	0.85	0.82	0.84	57
Obesity_Type_I	0.91	0.96	0.93	67
Obesity_Type_II	0.96	0.96	0.96	53
Obesity_Type_III	1.00	1.00	1.00	69
Overweight_Level_I	0.75	0.75	0.75	57
Overweight_Level_II	0.80	0.80	0.80	55
accuracy			0.90	423
macro avg	0.89	0.89	0.89	423
weighted avg	0.90	0.90	0.90	423

There was a huge improvement from the naive bayes classifier. We obtained 90% accuracy with the decision tree classifier in predicting class labels, compared with the 50% accuracy from naive bayes. While naive bayes performed poorly with correlated attributes, decision tree favors highly relevant attributes. The nature of our attributes became an advantage, giving us much better predictions.

4.2.3 Support Vector Machine

The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space that distinctly classifies the data points. To separate the two classes of data points, it is the most helpful to find a plane that has the maximum distance between data points of both classes, and that is, we want to maximize the margin between the data points and the hyperplane. Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane. Using these support vectors, we maximize the margin of the classifier. Here we utilized the linear kernel, giving us 97% accuracy, which is the best performance observed. We believe that it is because SVM is effective in high dimensional spaces and favors small and balanced data sets.



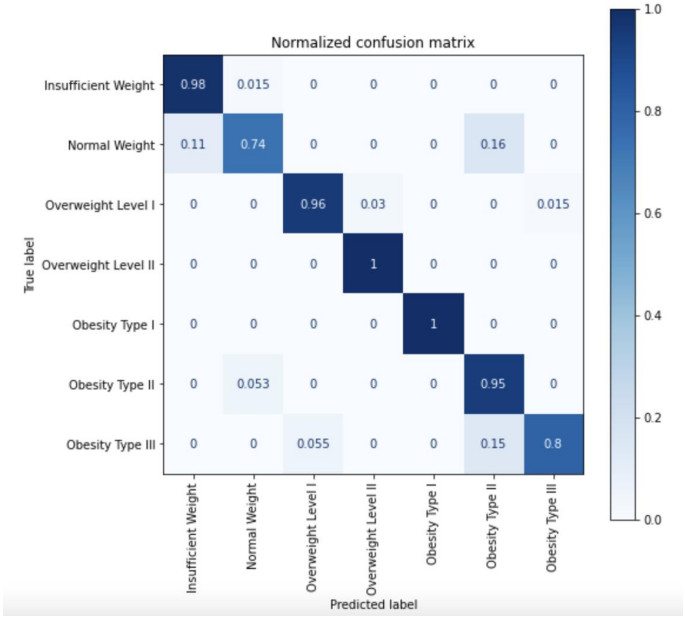
	precision	recall	f1-score	support
Insufficient_Weight	0.91	0.98	0.95	65
Normal_Weight	0.96	0.82	0.89	57
Obesity_Type_I	1.00	1.00	1.00	67
Obesity_Type_II	1.00	1.00	1.00	53
Obesity_Type_III	1.00	1.00	1.00	69
Overweight_Level_I	0.92	0.96	0.94	57
Overweight_Level_II	0.98	0.98	0.98	55
accuracy			0.97	423
macro avg	0.97	0.97	0.97	423
weighted avg	0.97	0.97	0.97	423

4.2.4 Enhancement with LDA

Since the performance of the naive bayes classifier was below optimal, we wanted to explore ways to enhance its prediction accuracy, and one of the ways was linear discriminant analysis (LDA). LDA is a supervised dimensionality reduction technique that tries to reduce dimensions of the feature set while retaining the information that discriminates output classes. It strives to find a decision boundary around each cluster of a class, then projects the data points to new dimensions in a way that maximizes the distance between the clusters and minimizes the distance between the data points within a cluster and their centroid. Then we have the linear discriminants of the feature set formed with these new dimensions.

By performing LDA on our feature set, the conditional independence assumption held by the naive bayes classifier was finally met. Hence we would expect the performance of naive bayes to be improved, which was exactly what we observed. The accuracy was increased from 0.5 to 0.92. We were able to improve the performance of naive bayes classifier with LDA.

	precision	recall	f1-score	support
Insufficient_Weight	0.91	0.98	0.95	65
Normal_Weight	0.91	0.74	0.82	57
Obesity_Type_I	0.96	0.96	0.96	67
Obesity_Type_II	0.96	1.00	0.98	53
Obesity_Type_III	1.00	1.00	1.00	69
Overweight_Level_I	0.76	0.95	0.84	57
Overweight_Level_II	0.98	0.80	0.88	55
accuracy			0.92	423
macro avg	0.93	0.92	0.92	423
weighted avg	0.93	0.92	0.92	423



5 COMPARISON OF TECHNIQUES

As we have done analysis both on regression and classification, we also wanted to find ways better comparing two sets of techniques directly. Hence, for regression techniques, we split labeled data for training and testing with "NObesyesdad" as part of the independent variable and then label encoded the value of it in the test set individually for future use. Then, for the predicted result of MBI in each regression model, we classified the value of MBI into labels from zero to six based on WHO's standard of obesity to match the previous label encoded value. Therefore we were able to compare the predicted results of regression based on MBI with the original test set

that has obesity levels as the target variable and get the accuracy score for each regression model. This is our method finding the accuracy score of classification based on regression results, which more directly combined two sets of techniques. The results for each regression's "accuracy score" are shown below.

```
[[43 12  0  0  0  0  0]
 [ 2 54  0  0  0  2  0]
 [ 0  0 69  0  0  0  1]
 [ 0  0  0 60  0  0  0]
 [ 0  0  0 30 45  0  0]
 [ 0  5  0  0  0 46  5]
 [ 0  0  1  0  0  2 46]]
0.8581560283687943
```

Linear regression

```
[[39 16  0  0  0  0  0]
 [ 1 55  0  0  0  2  0]
 [ 0  0 68  0  0  0  2]
 [ 0  0  0 60  0  0  0]
 [ 0  0  0 37 38  0  0]
 [ 0  0  0  0  0 55  1]
 [ 0  0  1  0  0  4 44]]
0.8486997635933806
```

Lasso Regression

```
[[43 12  0  0  0  0  0]
 [ 2 53  0  0  0  3  0]
 [ 0  0 69  0  0  0  1]
 [ 0  0  0 60  0  0  0]
 [ 0  0  0 30 45  0  0]
 [ 0  5  0  0  0 45  6]
 [ 0  0  1  0  0  2 46]]
0.8534278959810875
```

Elastic Net

```
[[43 12  0  0  0  0  0]
 [ 2 54  0  0  0  2  0]
 [ 0  0 69  0  0  0  1]
 [ 0  0  0 60  0  0  0]
 [ 0  0  0 30 45  0  0]
 [ 0  5  0  0  0 46  5]
 [ 0  0  1  0  0  2 46]]
0.8581560283687943
```

Ridge Regression

We can see that the accuracy score result is consistent with the original regression analysis results in that Linear regression and Ridge regression are two models with the highest score.

6 CONCLUSIONS

In this analysis, we estimated obesity levels in individuals from the countries of Mexico, Peru, and Colombia, based on their eating habits and physical condition. In achieving this, we explored various regression and classification models and experimented on their enhancement. We also tried to directly compare the result of regression with classification by transforming the predicted values. Eventually, the model with the highest accuracy in estimating obesity levels was SVM (0.97), followed by LDA-enhanced Naive Bayes (0.92). The best-fitted regression techniques were Ridge Regression and Linear Regression.

7 Reference

[1]Palechor, F. M., & de la Hoz Manotas, A. 2019. Dataset for estimation of obesity levels based on eating habits and physical condition in individuals from Colombia, Peru and Mexico. *Data in Brief*, 104344

[2]DO, NORMA Oficial Mexicana NOM-008-SSA3-2010, Para el tratamiento integral del sobrepeso y la obesidad, *Diario Oficial* (2010)