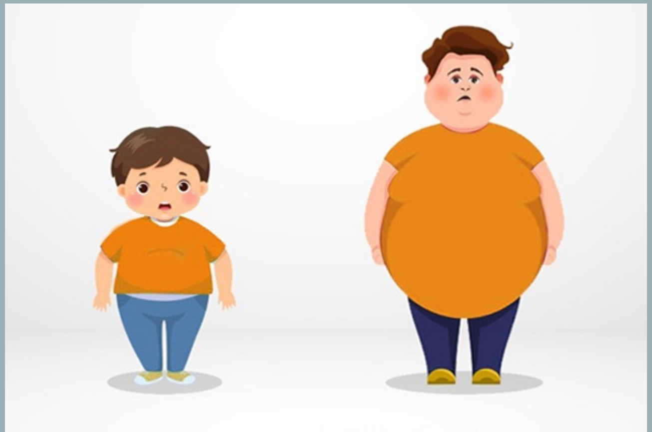
PREDICTION OF OBESITY LEVEL



Siming Yin and Xingzhi Ma

10.25

EXECUTIVE SUMMARY

Analytic Objective(s):

- Obesity Level: (based on BMI)
  - • Underweight Less than 18.5
  - •Normal 18.5 to 24.9
  - • Overweight 25.0 to 29.9
  - • Obesity I 30.0 to 34.9
  - • Obesity II 35.0 to 39.9
  - • Obesity III Higher than 40.

Decisions to be impacted:

- Public Investment on preventing obesity
- Identification of health treatment
- Provide social support to maintain a healthy lifestyle

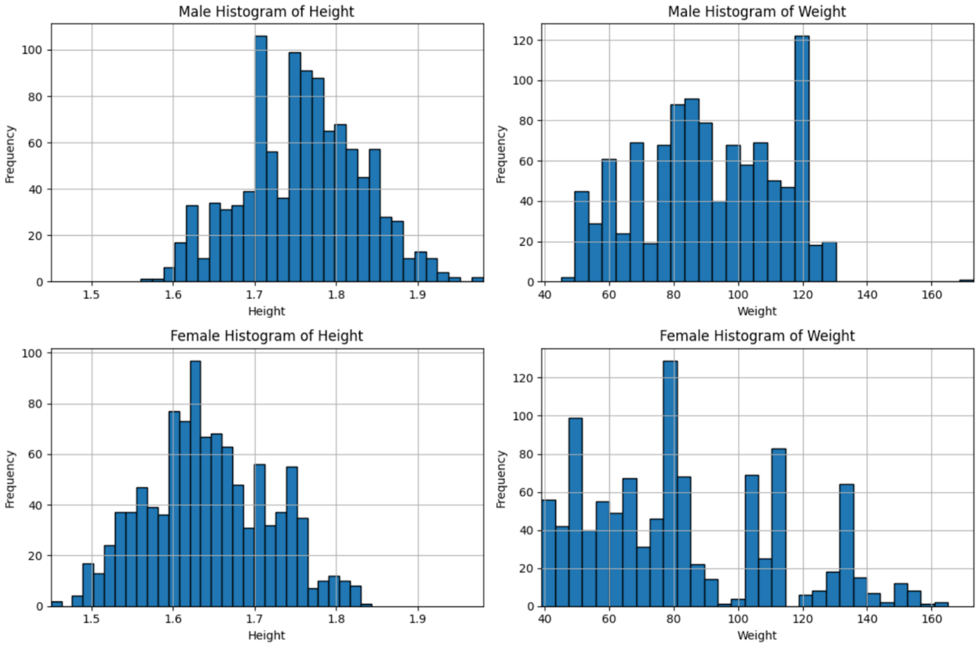
Business Value:

- Personal Health Improvement
- Public Health Improvement
- Research Cost Reduction

Data Assets

- Dataset for estimation of obesity levels based on eating habits and physical condition in individuals from Colombia, Peru and Mexico

DATA ASSET DESCRIPTION

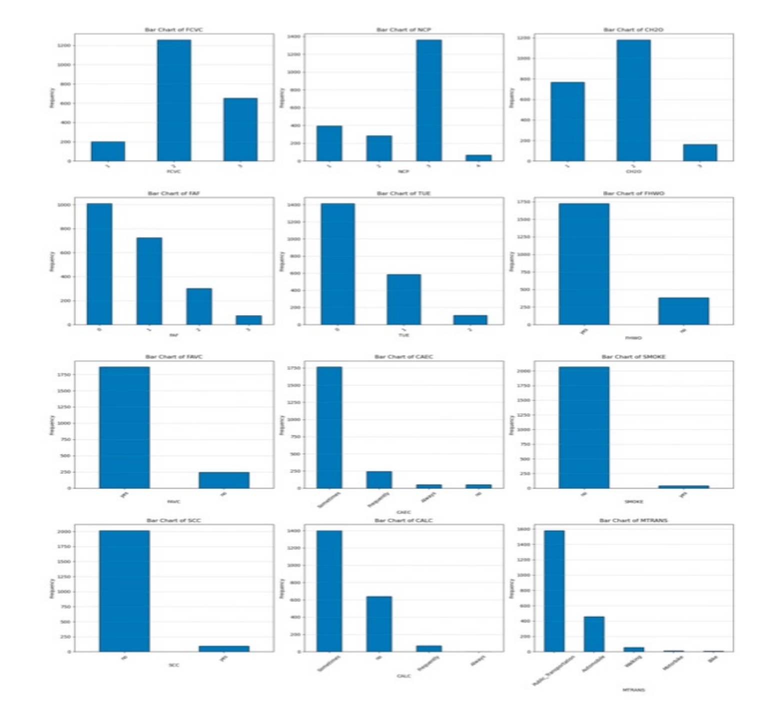


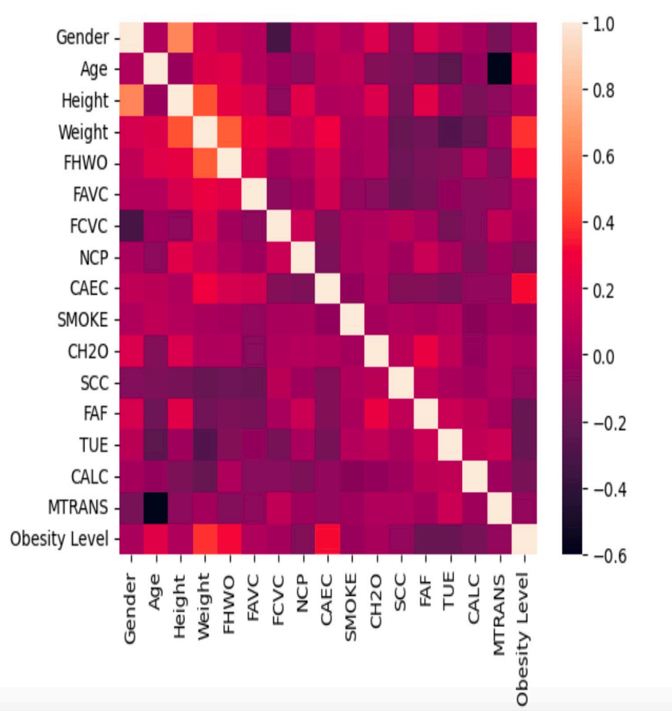
Height
normal-distribution-like shape
Overall, male's height is greater than female's height. Several peaks because of the region difference

Weight
multimodal distribution
Female's weight more spread out than male's weight. Both gender's weight can be divided into small subgroups.

DATA ASSET DESCRIPTION

1. Bar plots for Categorical Data:
   smoke (SMOKE):
   97% of the data are in the category of No
   Calories Consumption Monitoring (SCC):
   95% of the data in the category of Yes
2. Correlation Plot:
   The relationship between different features
   Weigh and Family History With Overweight (FHWO) are most correlated features with Obesity level
   Target Value (Obesity level) is calculated by BMI, which is related to weight and height. We will remove these two features in our predictive model. We will use all the features except weight and height in our model.

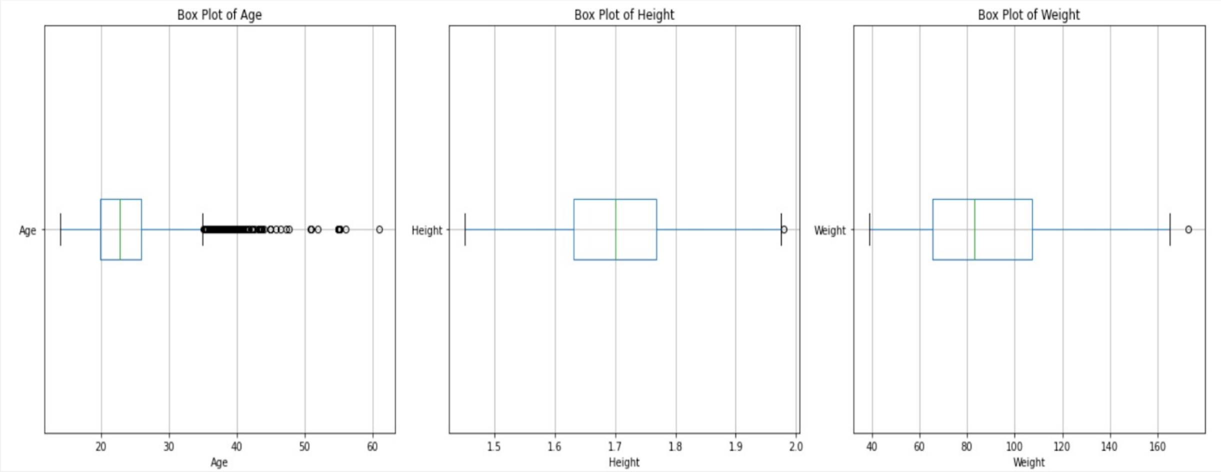




Maximum Frequency Information for Categorical Columns:

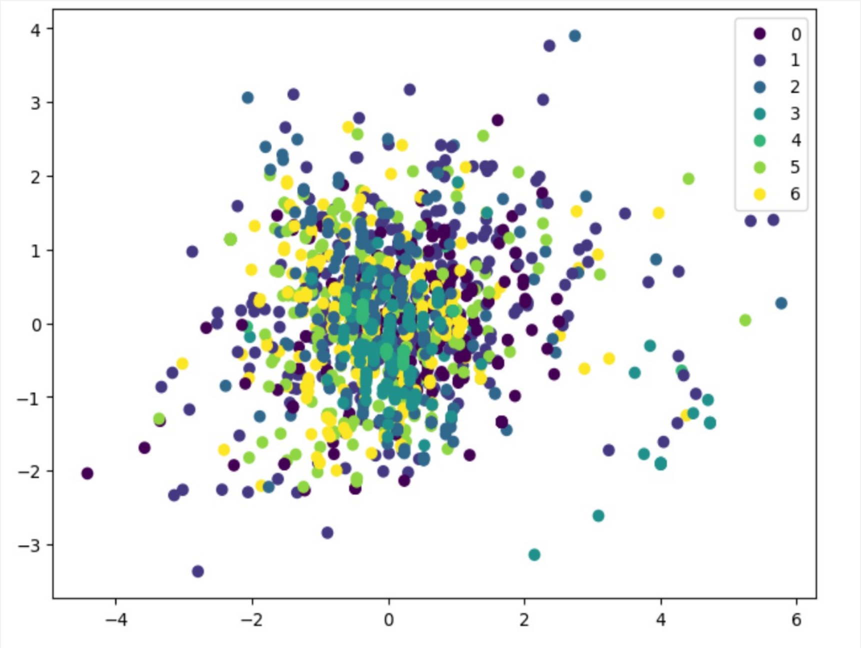
	Column	Max Category	Max Frequency	Max Percentage
0	FCVC	2	1257	59.545239
1	NCP	3	1362	64.519185
2	CH2O	2	1180	55.897679
3	FAF	0	1011	47.891994
4	TUE	0	1415	67.029844
5	FHWO	1	1726	81.762198
6	FAVC	1	1866	88.394126
7	CAEC	2	1765	83.609664
8	SMOKE	0	2067	97.915680
9	SCC	0	2015	95.452392
10	CALC	2	1401	66.366651
11	MTRANS	3	1580	74.846045

PREPROCESSING – OUTLIER DETECTION



- For the three numerical data, we used box plots
- One outlier for *Height* and *Weight*
- Majority *Age* data points are at younger age
- There are a few extreme values

PCA on Obesity level with the two highest correlated features (Family History with Obesity and Consumption of Food between Meal).
A major cluster in the center with some scatter around



OUTLIER DETECTION – ISOLATION FOREST

Isolation forest is an unsupervised learning algorithm that identifies anomalies by isolating outliers in the data.

	Gender	Age	FHWO	FAVC	FCVC	NCP	CAEC	SMOKE	CH2O	SCC	FAF	TUE	CALC	MTRANS	anomaly	scores
18	0	30	1	1	3	4	1	1	1	0	0	0	3	0	-1	-0.0104834
21	0	52	1	1	3	1	2	1	2	0	0	0	3	0	-1	-0.0245264
25	1	20	1	0	2	4	1	1	2	0	3	2	3	3	-1	-0.0112556
30	1	29	0	1	1	4	1	0	3	0	0	1	3	2	-1	-0.00732673
68	1	30	1	1	1	3	3	1	2	1	0	0	1	0	-1	-0.0532807
92	1	55	1	0	3	4	1	0	3	1	3	0	1	4	-1	-0.0466905
119	0	19	1	0	3	3	1	1	3	0	2	1	2	0	-1	-0.00817113
132	0	19	1	1	3	3	1	1	3	1	1	2	1	3	-1	-0.0250775
133	0	61	0	1	3	3	0	0	2	0	1	1	1	3	-1	-0.00607298
142	1	23	0	1	2	3	1	1	1	0	1	1	1	0	-1	-1.97072e-05
152	0	38	1	1	2	1	0	1	2	0	0	0	2	0	-1	-0.00580758
188	1	35	1	1	3	1	3	0	3	0	3	1	1	0	-1	-0.029988
191	1	26	1	1	3	1	1	1	2	1	2	0	2	3	-1	-0.0118799
200	0	23	1	0	3	1	2	1	3	0	1	2	2	3	-1	-0.01063
217	1	21	0	0	2	3	1	0	3	1	3	1	1	0	-1	-0.00317801
232	0	51	1	0	3	3	2	1	3	1	2	0	3	3	-1	-0.0239557
236	0	21	0	1	1	3	0	0	2	1	3	0	3	0	-1	-0.00738452
245	0	20	0	0	3	3	2	1	2	0	2	1	2	0	-1	-0.00376463
252	1	56	1	0	2	3	2	1	2	0	1	0	1	0	-1	-0.00592272
277	1	21	0	1	2	4	0	1	3	0	3	2	2	4	-1	-0.0394712
333	0	23	0	0	3	4	0	0	3	1	3	0	3	0	-1	-0.0236606
495	1	19	1	1	3	1	0	0	1	1	0	0	3	2	-1	-0.00784573

```
The number of Outlier (Isolation Forest): 22
```

1. Define and fit the model

- model. IsolationForest(n_estimators, max_samples, contamination, max_features)

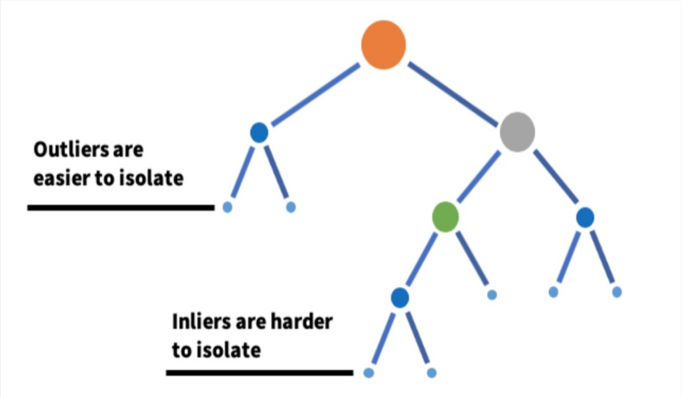
2. Find the scores and anomaly (1 is normal; -1 is outlier)
3. Print Anomalies

Isolation forest algorithm:

Step 1: When given a dataset, a random sub-sample of the data is selected and assigned to a binary tree.

Step 2: Branching of the tree starts by selecting a random feature first. And select a random threshold.

Step 3: Continued recursively till each data point is completely isolated or the defined max depth is reached.



MODEL UPDATE

KNN(K-nearest Neighbors)

The k-nearest neighbors algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point

- Minkowski Distance $= \sum_{i=1}^{n} |x_i - y_i|^{1/p}$
- Mahalanobis Distance $= (x-\mu)^T \Sigma^{-1} (x-\mu)$
- k-values

Random Forest

Random forests or random decision forests is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees.

- n_estimators, max_depth, train_test_split
- grid search, random search, bayesian optimization

KNN Result

Accuracy: 0.79905
Train score: 1.0000
Test score: 0.7991

Classification Report:

	precision	recall	f1-score	support
0	0.78	0.83	0.80	64
1	0.61	0.44	0.51	45
2	0.72	0.91	0.80	64
3	0.90	0.87	0.88	60
4	0.96	0.99	0.97	70
5	0.69	0.74	0.72	58
6	0.88	0.69	0.77	62
accuracy			0.80	423
macro avg	0.79	0.78	0.78	423
weighted avg	0.80	0.80	0.79	423

Accuracy: 0.80851
Train score: 1.0000
Test score: 0.8085

Classification Report:

	precision	recall	f1-score	support
0	0.74	0.83	0.78	64
1	0.53	0.53	0.53	45
2	0.82	0.88	0.85	64
3	0.91	0.87	0.89	60
4	1.00	0.99	0.99	70
5	0.75	0.78	0.76	58
6	0.83	0.69	0.75	62
accuracy			0.81	423
macro avg	0.80	0.79	0.79	423
weighted avg	0.81	0.81	0.81	423

Random Forest Result

Train score: 1.0000
Test score: 0.8842
Classification Report of Random Forest Classifier :

	precision	recall	f1-score	support
0	0.9107	0.9107	0.9107	56
1	0.6444	0.7250	0.6824	40
2	0.8857	0.8732	0.8794	71
3	0.9420	0.9559	0.9489	68
4	0.9855	1.0000	0.9927	68
5	0.8361	0.8361	0.8361	61
6	0.9057	0.8136	0.8571	59
accuracy			0.8842	423
macro avg	0.8729	0.8735	0.8725	423
weighted avg	0.8869	0.8842	0.8850	423

NEXT STEPS

- For the next steps, based on the performance, we will perform feature selection and tunning the hyperparameters with Grid search method or Random search method to improve our model.
- We will build another model for weight control recommendation system based on the correlation between weight and other features in order to provide suggestions for people weight control.

10/23 - 10/27	Midterm Presentation
10/30 - 11/3	Perform KNN and Random Forest algorithm
11/6 - 11/10	Fix model
11/13 -12/1	Try to build weight control recommendation system
Rest of semester	Review the project