Student name: Chris Doyle

Student number: SBS242034

Class Machine Learning

Lecturer: Dr. Muhammad Iqbal

CCT College, Dublin

April 28th, 2024

Introduction

For this project, I have selected a study on obesity named "ObesityDataSet\_raw\_and\_data\_synthetic.csv." This dataset contains various features that could be used to predict the possibility of a person becoming obese, based on factors such as age, weight, and family history of obesity. According to a 2022 study by the World Health Organization (WHO), approximately 2.5 billion adults worldwide were estimated to be overweight, with 890 million being obese.

The goals for this project are to identify which features from this dataset contribute to an individual becoming obese and to utilize prediction and classification algorithms to determine what percentage of the study sample are most at risk of becoming obese.

Firstly, we will conduct Exploratory Data Analysis (EDA) (Mckinney) to identify any anomalies within the dataset. Then, we will identify the most important feature in the dataset and designate it as our target variable (X). We will explore the best approach for choosing which classification algorithm to use and the reasons behind this decision. If necessary, we will implement data processing techniques such as Principal Component Analysis (PCA). (Rahul Kumar 2019).

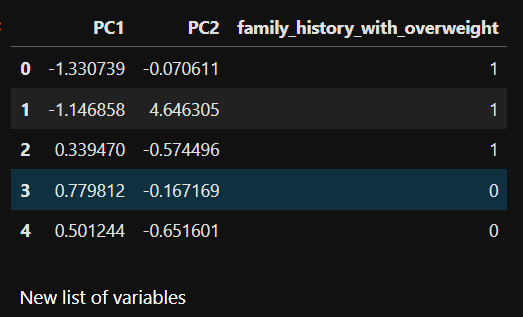
Throughout this project, we will employ various methods for characterizing the data. We will utilize cross-validation techniques and data visualization to explain the variances and argue from a statistical perspective which features are more important than others.

Characterization

**Below are the steps I took to characterize and transform the data.**

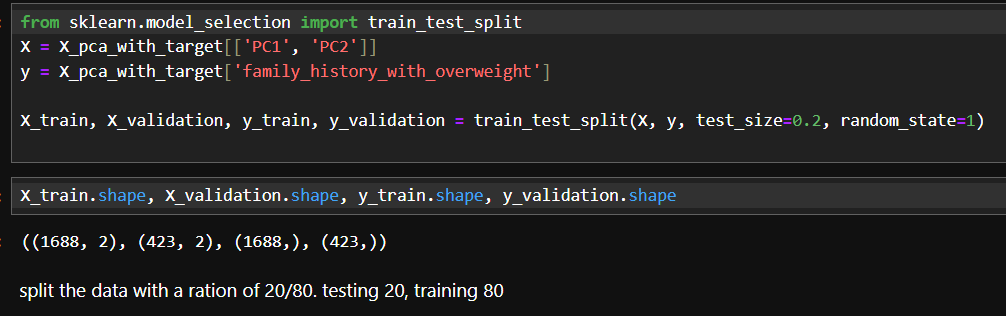
1. **Exploratory Data Analysis (EDA)**:
   * I started by examining the dataset using functions such as **.head()** and **.describe()** to get a sense of its structure and contents.
2. **Variable Selection**:
   * Identified the four main variables I wanted to use: ‘Age’, ‘Height’, ‘Weight’, and ‘SMOKE’, with the target variable named ‘family\_history\_with\_overweight’.
3. **Data Type Conversion**:
   * Converted the features named ‘SMOKE’ and ‘family\_history\_with\_overweight’ to **int64** using a mapping function in Python. This involved encoding categorical variables into numerical format, this process is essential for machine learning algorithms.
4. **Feature Selection**:
   * Dropped the features I didn't want to keep, reducing the dataset from 17 observations to 5. This step required removing irrelevant features that didn’t contribute much to the prediction task.
5. **Feature Scaling**:
   * Used **StandardScaler()** to scale the remaining features. Standardization is a common preprocessing step that involves transforming the data such that it has a mean of 0 and a standard deviation of 1. This can be important for algorithms that are sensitive to the scale of the features, such as support vector machines or k-nearest neighbors.

PCA- Principal Component Analysis



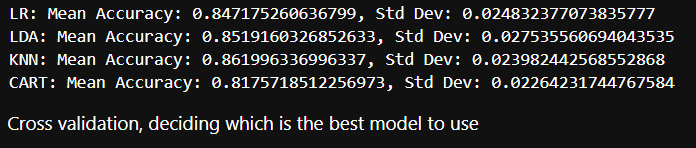
After employing Principal Component Analysis (PCA), I derived two novel features, PC1 and PC2, which effectively encapsulate the essential information from the original dataset while reducing its dimensionality. Subsequently, I reintegrated the target variable into the dataset to ensure its association with the transformed features. This step is crucial for maintaining the context and relevance of the target variable in further analysis or modelling tasks.

Test, Train, Split



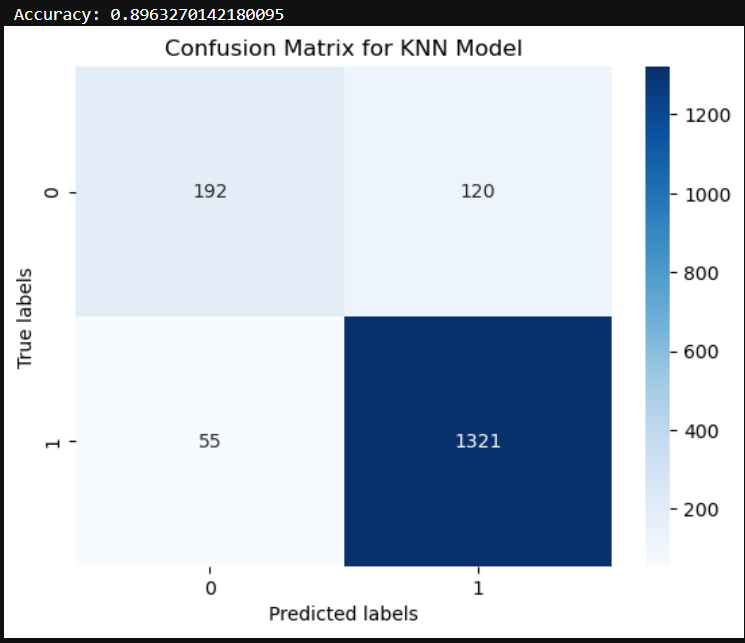
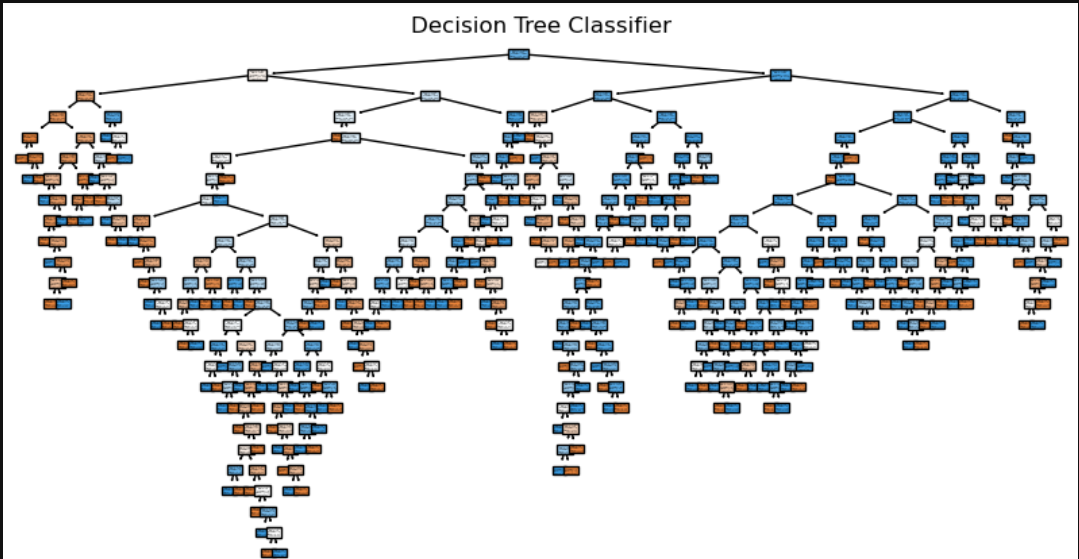
I then began to split my dataset into training and validation sets. ((1688, 2), (423, 2), (1688,), (423,)) this divides the feature matrix X and the target variable y into four subsets: X\_train, X\_validation, y\_train, and y\_validation. The split is performed with 80% of the data allocated for training and 20% for validation, while setting a random seed (random\_state=1). This approach facilitates model training on the training data and evaluation on the validation data to assess generalization performance.

Cross Validation



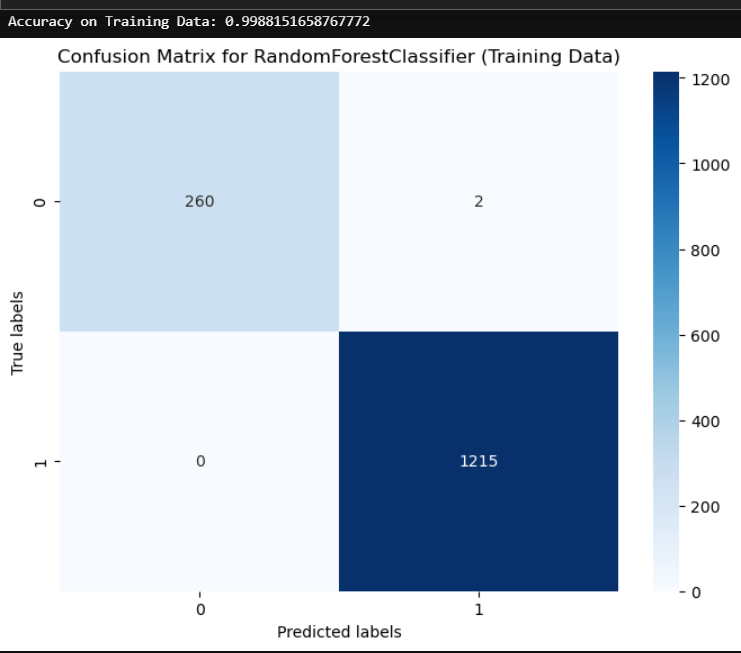
I started exploring various machine learning models for this project, employing cross-validation to find the most suitable one. This process involved importing essential libraries such as *StratifiedKFold* and *cross\_val\_score* and then using models like Logistic Regression, Linear Discriminant Analysis, k-Nearest Neighbors, and Decision Trees. By utilizing cross-validation techniques, I aimed to rate and compare these models objectively to determine their effectiveness to the project's objectives.

Selecting the Model

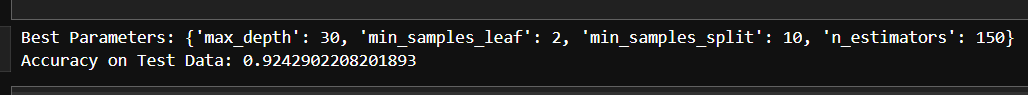
I explored different models for my dataset. Initially, I applied KNeighborsClassifier (KNN) and visualized its performance using a heatmap to display the confusion matrix. Then, I employed DecisionTreeClassifier (CART). Finally, I utilized RandomForestClassifier to assess the model's accuracy.

Visualisation



For this project, I utilized a range of data visualizations to elucidate data variances and depict data classifiers. I employed techniques such as heatmaps, decision trees, and confusion matrix visualizations to delve into accuracy levels and discern the optimal course of action. Through these methods, I pinpointed crucial metrics, including the accuracy level derived from training data using random forest classifier on a confusion matrix. This information is pivotal in articulating the outcomes of our model.

hyperparameter tuning.



I employed GridSearchCV to facilitate hyperparameter tuning. This allowed me to optimize the performance of my model by systematically searching through a grid of hyperparameters. Before conducting the grid search, I split my data into training and testing sets using train\_test\_split with a ratio of 70-30, different to the previous test ratio and a random state of 40. Then, using rf\_classifier RandomForestClassifier I defined a parameter grid containing various values for hyperparameters such as 'n\_estimators', 'max\_depth', 'min\_samples\_split', and 'min\_samples\_leaf' these parameter ranges will be investigated during the grid search process.

Conclusions

For this CA1 project, I began by selecting a dataset named "ObesityDataSet\_raw\_and\_data\_synthetic.csv." This dataset provided numerous features for exploration, enabling me to identify essential information for my model. I specifically focused on features such as 'SMOKE,' 'Height,' 'Weight,' and 'Age' due to their significant statistical relevance. My objective was to ascertain whether analysing this data could facilitate predicting whether individuals with a history of obesity were more likely to be overweight based on our feature list.

Subsequently, I conducted exploratory data analysis (EDA) on the dataset and identified two features as strings. I converted them into int64 format, transitioning from 'Yes/No' to '1/0'. Following this, I removed my target variable and proceeded with principal component analysis (PCA) to reduce the feature size from 4 to a completely new set of features, which I named PC1 and PC2.

After reintroducing my target feature, I divided my dataset using an 80/20 ratio, allocating 20% for testing and 80% for training the model. I then applied cross-validation to assess which model would yield the most accurate results. Through K-fold validation, I determined that models such as KNN, DecisionTreeClassifier, and RandomForest would provide the most accurate results for utilization.

When testing both the training and test datasets, I observed that my model performed well on both, indicating it was neither underfitted nor overfitted. Typically, if one dataset's performance significantly surpasses the other, it suggests either underfitting or overfitting. For instance, if the model performs better on the training data than the test data, it suggests underfitting, while superior performance on the test data compared to training data signals overfitting.

On the test data, I applied hyperparameter tuning to conduct a more in-depth analysis. By tuning hyperparameters such as 'n\_estimators,' 'max\_depth,' 'min\_samples\_split,' and 'min\_samples\_leaf,' I refined the accuracy of my model on the test data to 92%.

Overall, this project has presented significant challenges. While I acknowledge that I might have made mistakes in some of the coding aspects, I believe that completing this CA has deepened my understanding of machine learning. I look forward to advancing to the next stages of this module with newfound insights and skills.

References

<https://www.who.int/news-room/fact-sheets/detail/obesity-and-overweight#:~:text=In%202022%2C%201%20in%208,million%20were%20living%20with%20obesity>.

Mckinney, Wes. Python for Data Analysis : Data Wrangling with Pandas, NumPy, and IPython. O’reilly Uuuu-Uuuu, 2017. (Mckinney)

(Rahul Kumar 2019). Machine learning quick reference : quick and essential machine