Forecasting Diabetes

Note: When this file was created, the saved figures were not copied. After that, I polished my write-up. I'm not able to generate or save figures in this file without codes. I am sorry for the mistake. But ***graphs and figures can be seen in the other file which include codes***. I've also uploaded the figures to the Github resporitory.

Introduction to Problem & Data

Problem Statement:

Diabetes remains a significant global health challenge, impacting millions of lives every year. For my final project, I will optimize multiple models that are capable of estimating whether someone has diabetes based on certain health indicators, and then choose a predictive model with the best performance. This model will identify individuals at higher risk; and thus, it could help individuals in managing their health and minimizing their risk of developing diabetes, support healthcare professionals in diagnosis, especially in early detection. Additionally, since it highlights key risk factors and identify people of higher risks, it may help in promoting healthier dietary habits and improve level of health and quality of life. For instance, given results of data analysis with this model, public health agencies and healthcare professionals can set up targeted programs among people or communities with higher risk.

Dataset Description:

Data for this project is sourced from the National Health and Nutrition Examination Survey (NHANES), administered by the Centers for Disease Control and Prevention (CDC), which collects extensive health and nutritional information from a diverse U.S. population.

In this sub-dataset, the focus is narrowed to predicting respondents' age by extracting a subset of features from the larger NHANES dataset. These selected features include physiological measurements, lifestyle choices, and biochemical markers, which were hypothesized to have strong correlations with age and the level of blood sugar.

The dataset includes variables such as age, gender, physical activity levels, body mass index (BMI), fasting glucose, and insulin levels, which I can use to help predict the risk of having diabetes.

Here is the url of the original survey for reference (More detailed explaination of each variables and their values included!)

https://wwwn.cdc.gov/nchs/data/nhanes/2013-2014/questionnaires/DIQ_H.pdf

Unfortunately, the questionnaire is currently unavailable, but a copy is provided here:

https://drive.google.com/file/d/1d5GChsyDI0WblNhnQtweZlmv5F251brw/view?usp=drivesdk

Preprocessing of the Dataset

Variable Naming: Some of the variable names in the dataset were not intuitive. To make the data more user-friendly, I consulted the NHANES webpage to obtain descriptions for each variable and created a dictionary for short, readable variable names.

I create a table with my new variables and their detailed description for reference:

According to the original questionnaire, 1 represents people with diabetes, 2 represents people on the borderline or those with prediabetes, and 3 represents people who neither have diabetes nor prediabetes. To make these representations more intuitive, I swap the 2's and 3's.

I removed variables like ID numbers that were irrelevant for modeling purposes.

At the end of this preprocessing phase, the cleaned DataFrame was ready for further analysis and modeling.

Preliminary Examination of the Dataset

Descriptive Statistics

The dataset contains two primary age groups: adults and seniors. The age of adult respondents ranges from 12 to 64 years, while seniors span ages 65 to 80 years.

Initial Visualizations

Firstly, check the discrete values, including 'Age_Group', 'Gender', 'Physical_Activity', and 'Oral'. The relationship between diabetes and the variables remains unclear so far.

A pairplot is generated. It's a scatterplot between all combinations of two variables in my dataset, with histograms of single variables on one of its diagonal. The three cases of diabetes are shown with different colors in each plots.

From the histograms we know that **people who have diabetes or prediabetes** tend to **be older, have higher BMI, higher fasting glucose**. In general, people who have prediabetes or are on the boundary have higher insulin level, but the relationship between the rate of having diabetes insulin is not so obvious.

Modeling & Interpretations

Baseline Model

Reference: https://scikit-learn.org/dev/modules/generated/sklearn.metrics.ConfusionMatrixDisplay.html#sklearn.metrics.ConfusionMatrixDisplay.from_predictions

According to the table, the accuracy of predicting that everyone don't have diabetes is approximately 96.53%, which is rather high. So, I'll use this as my base line model - a preferred model should at least reach an accuracy of that of this baseline model, which is 96.53%

Logistics Models

Precision and recall as training metrics

I want to train the model with multiple scoring methods, such as accuracy, recall and precision, evaluate the model's accuracy with the selected scoring method, and choose the scoring method that perfporms the best.

Accuracy will be used as the way to evaluate my model, since it aligns with the dataset's real-world application, where the overall performance matters.

Precision Precision measures how many of the predicted positive cases are actually correct. The quality of positive predictions is measured with precision as a metric.

Formula: Precision=True Positives / (True Positives+False Positives)

Recall measures how many of the actual positive cases the model correctly identifies. Recall as a training metrics would emphasize how many true positives the model find.

Formula: Recall= True Positives/ (True Positives + False Negatives)

I won't use recall as a metric because a high recall for one class could ignore the correct predictions of the majority class, especially when there are many factors to be analyzed, resulting in a model that appears to perform well on recall but fails overall.

Reference: https://drbeane.github.io/python_dsci/pages/grid_search.html#grid-search-with-logistic-regression

References:

https://scikit-learn.org/1.5/modules/generated/sklearn.metrics.make_scorer.html

https://scikit-learn.org/1.5/modules/generated/sklearn.metrics.precision_score.html

As we can see, the score of the model trained with accuracy is about 96.49%, and that of precision is 98.81%. Accuracy is not so good as precision. It even does not perform as well as the baseline model, whose accuracy is about 96.53%.

Why does precision perform better than accuracy as a metric? In a word, the distribution of the dataset is skewed. As we have discussed in the section of baseline model, the majority class (people who don't have diabetes or prediabetes) dominates the whole dataset. Precision focuses on the model's ability to correctly identify positive cases while ignoring the majority class, and it fits our dataset very well.

Decision Tree Regression Model

I also chose to build a decision tree regression model because it can capture non-linear relationships within the data. In addition, it makes it easier to understand how predictions are made by providing a clear decision making process.

Random Forest Regression Model

To further enhance the performance of my model, I extended my decision tree into a random forest classification model. Random forests include multiple decision trees to improve predictive accuracy and robustness.

The best number of trees is 20, meaning that the model with the best performance has a depth of 20. However, from the graph we know that its accuracy is approximately 96.67%, which is less than that of logistics model. So, it is not so good as our logistics model.

It doesn't perform so good as expected. A reason might be that there are too many variables that don't matter so match. Since all variables have the same weight in this model, it's bad for generating an accurate model. Let's use some variables that might be important, as discussed when doing initial visualizations: 'Age', 'BMI', 'Fasting_Glucose', and 'Insulin'.

It also doesn't not as good as the logistics model. The random forest model may have overfitted or been impacted by irrelevant features.

Next Steps & Discussion

Summary of Findings

The models ranked in terms of performance accuracy are as follows:

1. Logistic Regression (precision as training metric) – 98.81%
2. Decision Tree Regression – 96.92%
3. Random Forest Regression – 96.67%
4. Baseline Model – 96.53%
5. Random Forest (selected features) – 96.49%
6. Logistic Regression (accuracy as training metric) – 96.49%

The logistic regression model trained with precision as the evaluation metric has the best performance. Its focus on identifying true positive cases while avoiding the influence of the majority class made it particularly effective for this imbalanced dataset.

Next Steps/Ways of Improvements

To enhance the predictive capabilities of the models, I could improve my model or the selection of model in the following ways:

- Increase the time of training.
- Identify the set of most impactful features or appropriately weighting variables. This could help the model focus on the variables that contribute most to accurate predictions, especially for forests and trees, since variables have the same weights in tree-based models. It prevents the model from the impact of less related variables.
- Adjust steps of increasing maximum depth in the random forest model.
- Exploring other datasets or including external factors might be helpful.
- Trying entropy or other criterions for trees and/or forests will probably improve the models' ability.

- Trying other models might also be helpful.

By integrating these additional factors into the analysis, I would be able to refine these models even more and identify individuals at higher risk, potentially helping with diagnosis and prevention of diabetes.