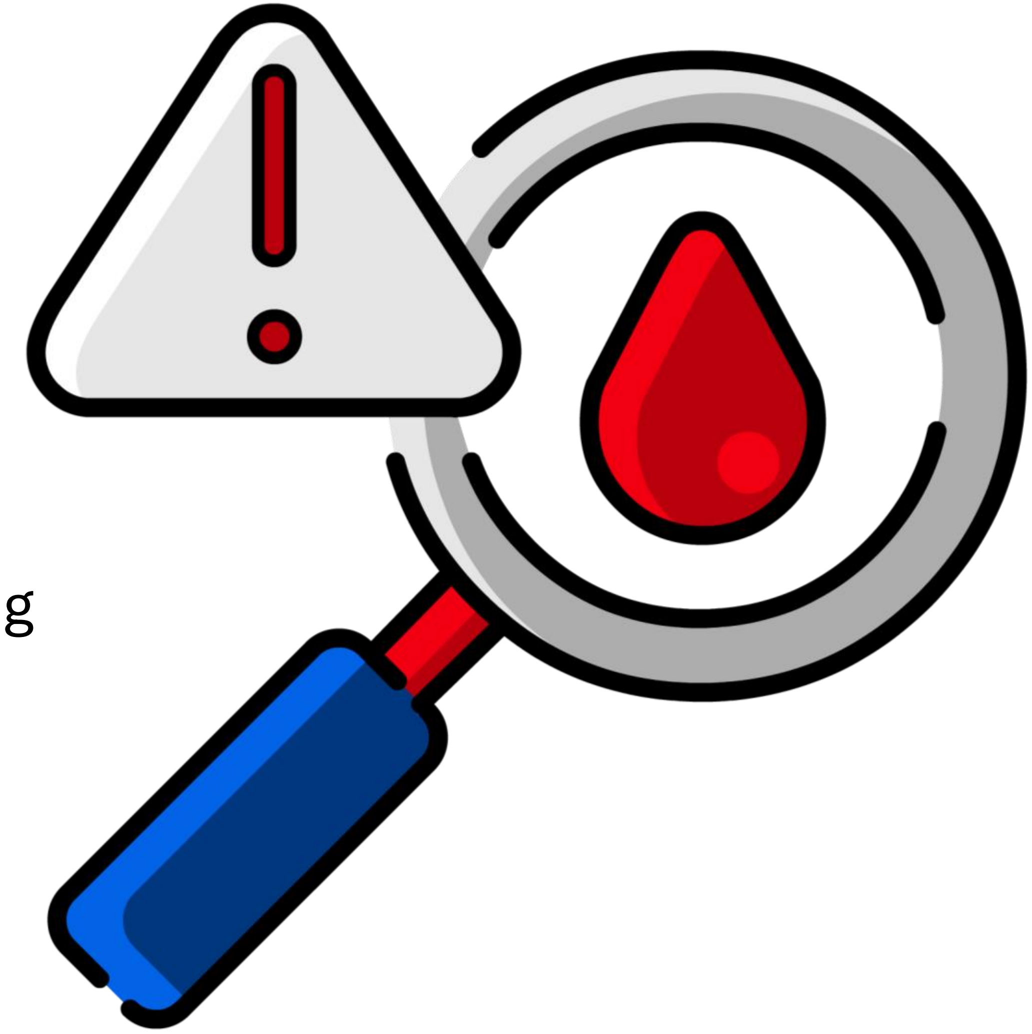
Decoding Diabetes



Predictive Models and Insights Using Machine Learning

Professor: 안용길

Presents

Braian Plaku (브라이언 플라쿠)

Introduction

Diabetes is one of the most prevalent **chronic diseases**, affecting millions globally. This project aims to **predict** the likelihood of an individual having diabetes using the Behavioral Risk Factor Surveillance System **(BRFSS)** dataset from Kaggle.

<u>**The focus**</u> will be on understanding which **factors contribute the most to diabetes risk** and developing machine learning models to predict diabetes **based on survey responses**.

Dependent Variable Y: Diabetes_012

- (0) no diabetes or only during pregnancy
- (1) prediabetes
- (2) diabetes

Dependent Variable X: 21 variables in the dataset

- BMI
- Age
- Income
- PhysHlth
- GenHlth
- Education
- Education
- MentHlth
- HighBP
- Smoker
- Fruits
- Sex
- HighChol
- Veggies
- PhysActivity
- DiffWalk
- HeartDiseaseorAttack
- NoDocbcCost
- Stroke
- AnyHealthcare, CholCheck

Digging deeper into the Data

Source:

CDC (Centers for Disease Control) Behavioral Risk Factor Surveillance System, [LINK]

ANNUAL, uniform, state-specific data on preventive health practices and risk behaviors

Year 2011, extracted on Kaggle from CDC itself [LINK]

Original Data, in ‘csv’ format, 253,680 survey responses

Alex Teboul (Data Scientist), based on Building Risk Prediction Models for Type 2 Diabetes Using Machine Learning Techniques on Kaggle [LINK]

The data was **cleaned** into a useable format for machine learning alogrithms, reduction was made from **330 features** (dependent variables) onto **21 variables**

Link to his Notebook can be found here: [LINK]

Alex Teboul

Shaping the Data: Preprocessing & Preparation

Target Variable **Binarization**

Diabetes_012

- (0) no diabetes or only during pregnancy
- (1) prediabetes
- (2) diabetes

(0) no diabetes or only during pregnancy

(1) prediabetes or diabetes

allows the models to **focus** on a simpler task!

Perform **univariate logistic regression** for each feature

after **standardizing** the data

Feature selection based on their **p-values**

Shaping the Data: Preprocessing & Preparation

Feature selection based on their **p-values**

	coefficient	p-value	odds_ratio
HighBP	0.780275	<0.001	2.182072
Age	0.579303	<0.001	1.784794
DiffWalk	0.552433	<0.001	1.737476
PhysHlth	0.444809	<0.001	1.560193
GenHlth	0.914087	<0.001	2.494498

Cross-Validation

using **5 splits**

Data Undersampling

using the `RandomUnderSampler` from **imblearn-undersampling** Python Library

Key Traits of the three chosen Models

three models implemented

Random Forest

- Builds multiple **decision trees** and combines their predictions
- Provides **feature importance**, which tells us which factors are most important in predicting diabetes
- Better suited for capturing complex **patterns** and **interactions** between features

Logistic Regression

- Assigns **weights (coefficients)** to each feature, making it easy to interpret how each feature affects diabetes risk
- **Simple and more interpretable**, but may not capture complex patterns as well as Random Forest

Gradient Boosting

- Builds an ensemble of **weak learners**, each focusing on correcting errors made by previous models.
- Combines predictions in a sequential manner, leading to **higher accuracy over time**.
- More complex than Logistic Regression.

Key Traits of the three chosen Models

three models implemented

Random Forest

- Splitted Training and Testing Sets
- Built Random Forest Model
  - estimating **100** trees
- Fit & trained the model
- Predicted the data
- Evaluated the model
- Analyzed Feature Importance
- Plotted the results visually

Logistic Regression

- Splitted Training and Testing Sets
- Built Logic Regression Model
  - with **1000** iterations
- Fit & trained the model
- Predicted the data
- Evaluated the model
- Plotted the results visually

Gradient Boosting

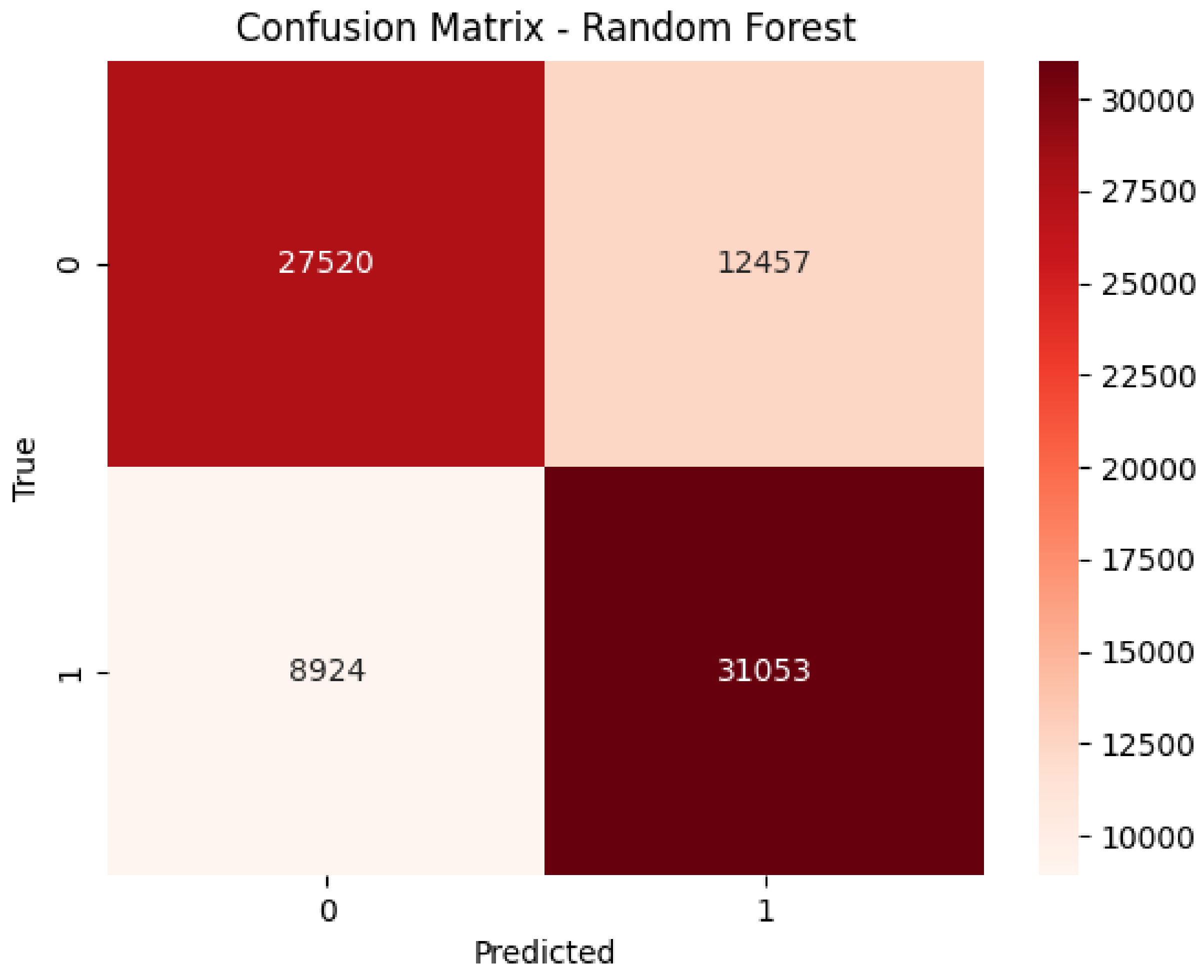
- Splitted Training and Testing Sets
- Built Gradient Boosting Model
  - using 100 estimators and a learning rate of 0.1
- Fit & trained the model
- Predicted the data
- Evaluated the model
- Analyzed Feature Importance
- Plotted the results visually

Outcome Review

Random Forest: Confusion Matrix

- Breakdown of true and false positives and negatives

- **Correctly predicted**
  - 27,520 negatives
  - 31,053 positives
- **Incorrectly flagged**
  - 8,924 negatives
  - 12,457 positives

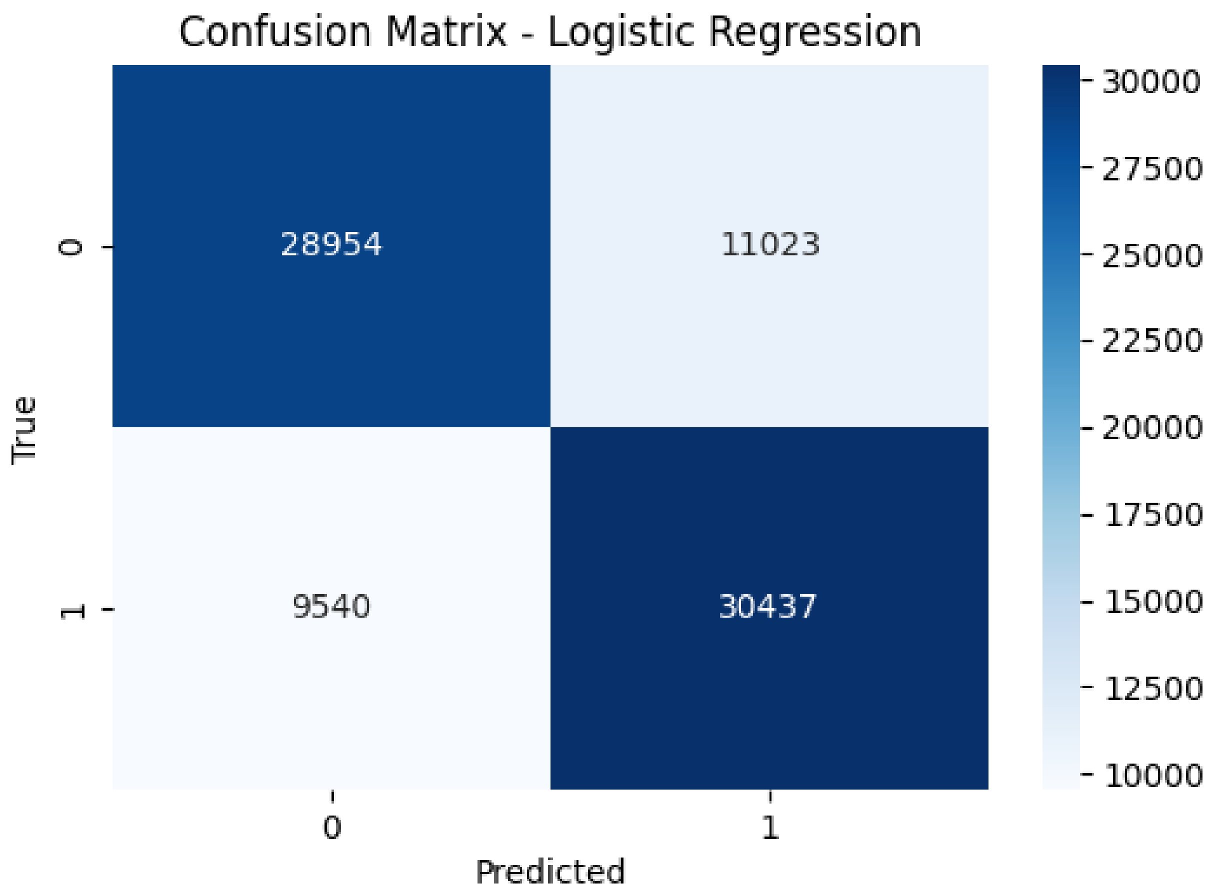


Outcome Review

Logistic Regression: Confusion Matrix

- Breakdown of true and false positives and negatives

- **Correctly predicted**
  - 28,954 negatives
  - 30,437 positives
- **Incorrectly flagged**
  - 9,540 negatives
  - 11,023 positives

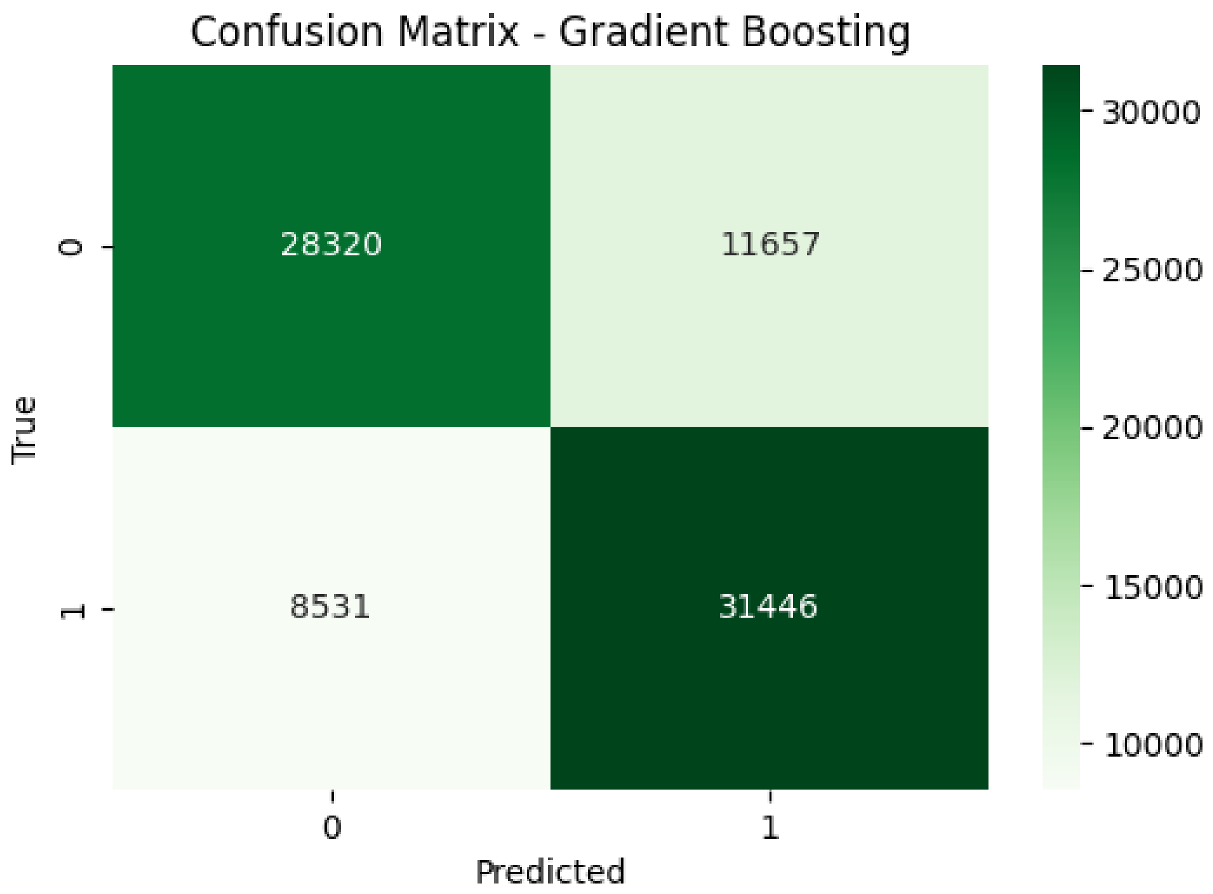


Outcome Review

Gradient Boosting: Confusion Matrix

- Breakdown of true and false positives and negatives

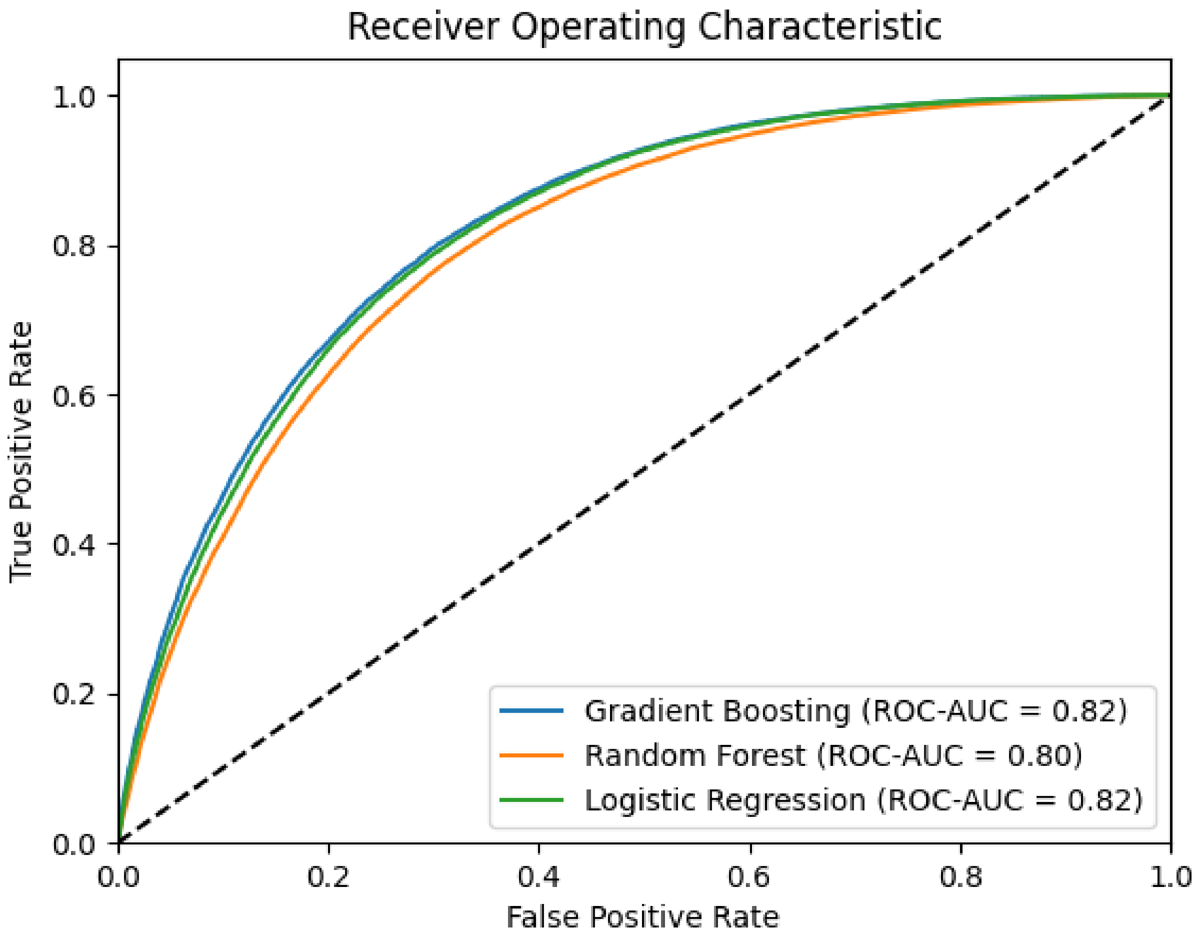
- **Correctly predicted**
  - 28,320 negatives
  - 31,446 positives
- **Incorrectly flagged**
  - 8,531 negatives
  - 11,657 positives



Outcome Review

Conclusion: Model Evaluation

- **Random Forest:**
  - Lower accuracy (73.2%)
  - Good feature importance
  - Low precision (71.3%)
- **Logistic Regression:**
  - Higher accuracy (74.2%)
  - More interpretable
  - Slightly better precision (73.4%)
- **Gradient Boosting:**
  - Best accuracy (74.7%)
  - Best ROC-AUC (82.4%)
  - Good precision (73.0%)



	Acccuracy	Precision	ROC-AUC
Random Forest	73,2%	71,3%	80,2%
Logistic Regression	74,2%	73,4%	81,8%
Gradient Boosting	74,7%	73,0%	82,4%

Literature

From the Web

Dataset:

Centers for Disease Control and Prevention:

- https://www.cdc.gov/brfss/annual_data/annual_data.htm

Kaggle Datasets:

- https://www.kaggle.com/datasets/cdc/behavioral-risk-factor-surveillance-system
- https://www.kaggle.com/datasets/alexteboul/diabetes-health-indicators-dataset

Material:

Statistical Learning with Python, Stanford Online

- https://www.youtube.com/playlist?list=PLoROMvodv4rPP6braWoRt5UCXYZ71GZIQ

Python Libraries:

- https://scikit-learn.org/
- https://imbalanced-learn.org/stable/under_sampling.html

More on Random Forest:

- https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/

Thank you

for your attention!