PHASE 2 PROJECT PRESENTATION

BUILDING RISK PREDICTION MODELS FOR DIABETES USING MACHINE LEARNING

PHASE 2

LESLIE LI, SATYAKI DIXIT,
SHRESHTA PHOGAT

DATA 606 CAPSTONE 14 NOVEMBER 2022

WELCOME!

OVERVIEW

EDA/ Data cleaning step 01

Variable selection and regularization 02

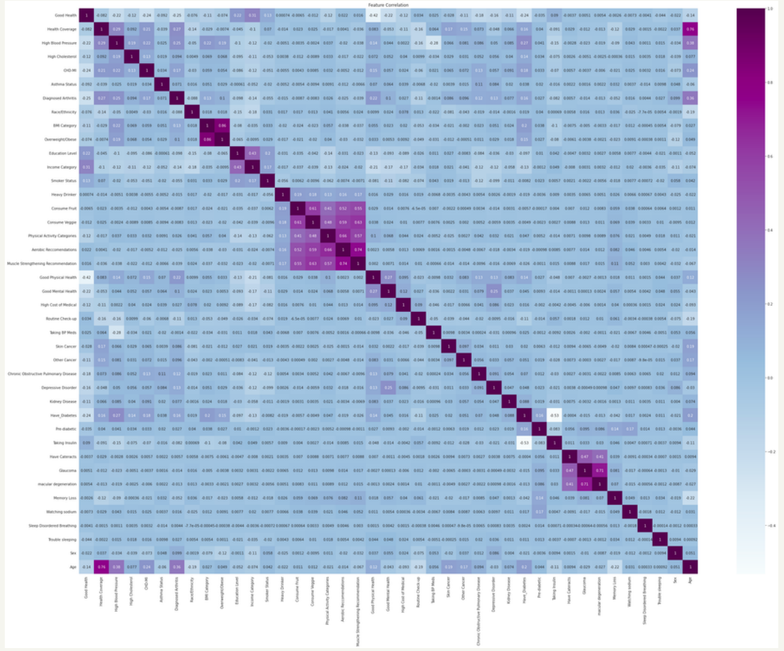
Model Fitting Overview: 03

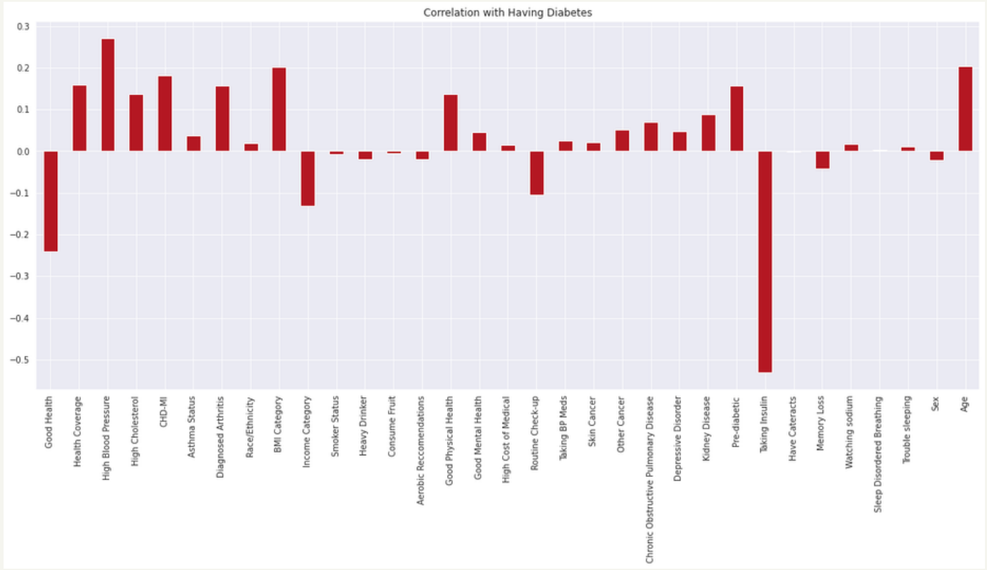
- Logistic Regression
- KNN
- Support Vector Machine
- Decision Tree Classifier
- Random Forest Classifier
- XGBoost Classifier

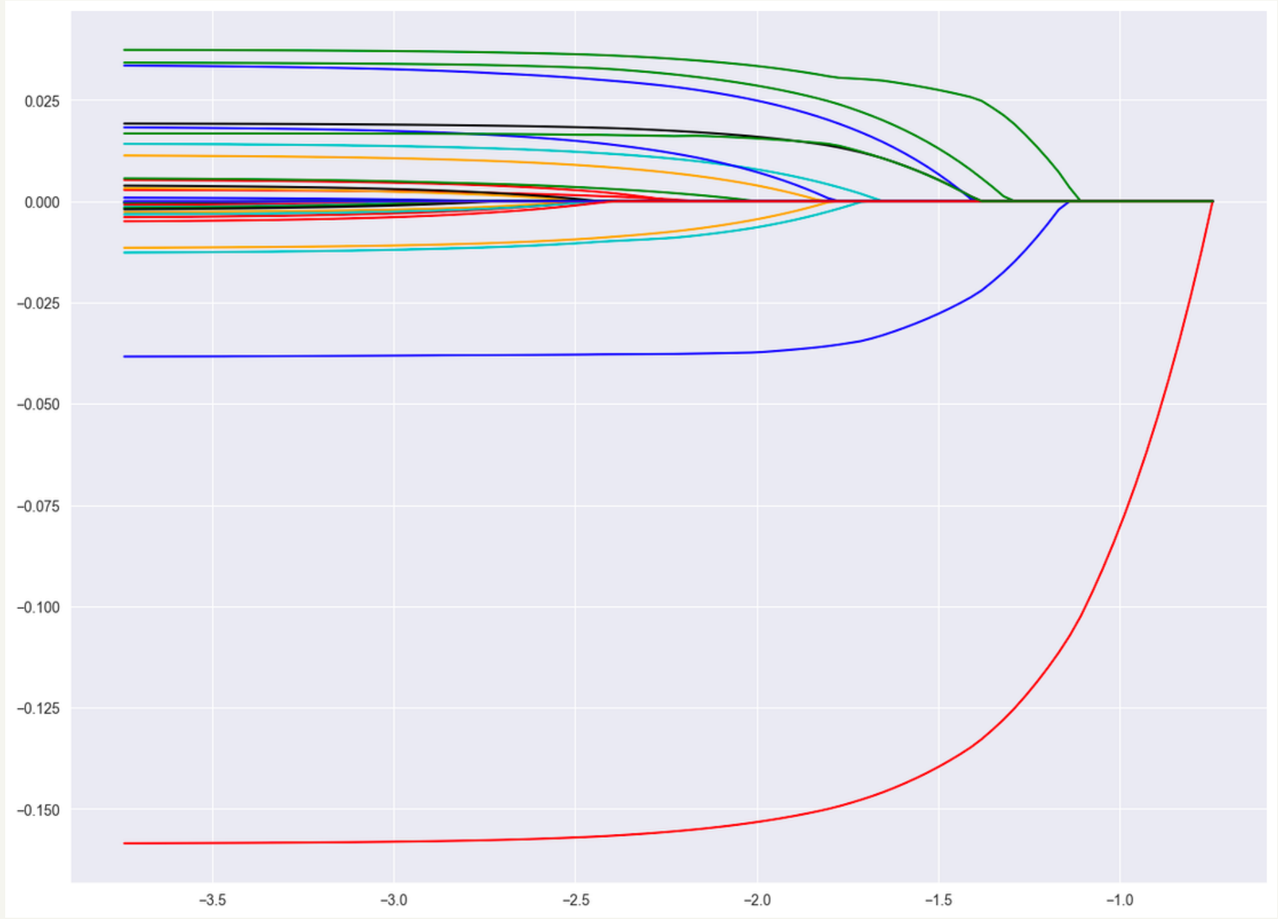
EXPLORATORY DATA ANALYSIS/ DATA CLEANING SUMMARY

MAJOR PATTERNS AND TRENDS IN THE DATA SET

- Data was cleaned into a readable format with 41 columns and 315853 rows.
- Clean data set binary with the target variable having diabetes
- Feature correlation was done for all features.
- Correlation with having diabetes







STANDARD SCALING

VARIABLE SELECTION AND REGULARIZATION

DRAW A LASSO PATH- PLOT CONVERGENCE TO ZERO

- The features with the most correlation to Y will remain above Zero longer. Those crashing to zero quickly did not have much correlation.
- Top 5 Strongest features to the target:
  - Taking Insulin
  - Good Health
  - High Blood Pressure
  - BMI Category
  - Pre-diabetic
- Top 5 Weakest features to the target
  - High Cost of Medical
  - Chronic Obstructive Pulmonary Disease
  - Sleep Disordered Breathing
  - Trouble Sleeping
  - Have Cateracts

MODEL FITTING OVERVIEW

LOGISTIC REGRESSION

KNN

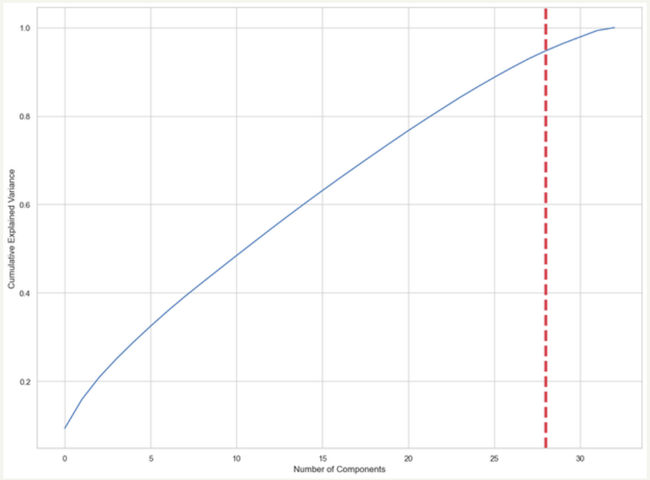
SUPPORT VECTOR MACHINE

DECISION TREE CLASSIFIER

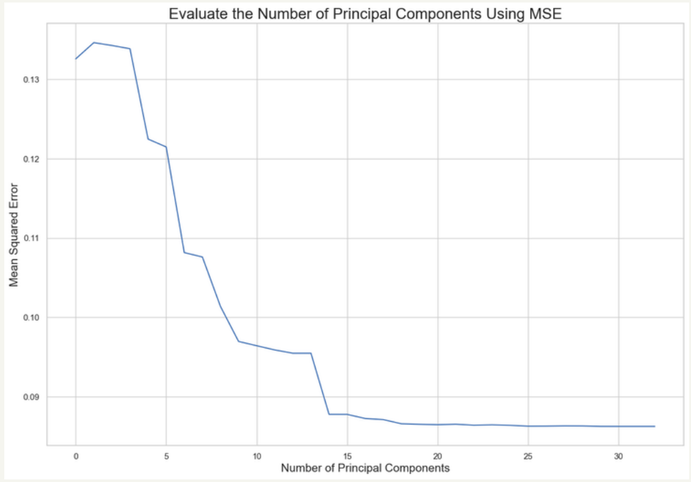
RANDOM FOREST CLASSIFIER

XGBOOST CLASSIFIER

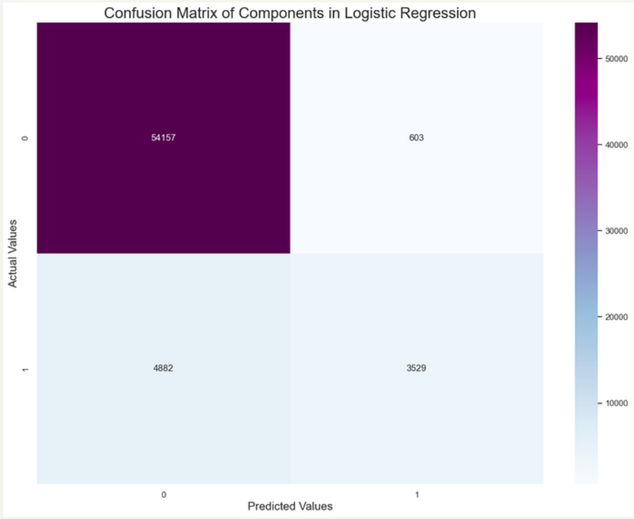
LOGISTIC REGRESSION WITH PCA



- Used 3-fold cross-validation to determine the number of dimensions
- Validated how many components needed for 95% cumulative variance
- The graph shows after more than 28 components, we don't gain much-explained variance



- Evaluate the number of principal components using mean square error
- The MSE plummeted after 14 principal components
- After transformation, the data set has 28 components that captured around 95% variability in the original dataset, and multicollinearity is eliminated



- Fit the logistic regression model, make predictions, and measure model performance
- shown above is the confusion matrix
- train accuracy: 0.91
- test accuracy: 0.91

LOGISTIC REGRESSION WITH PCA

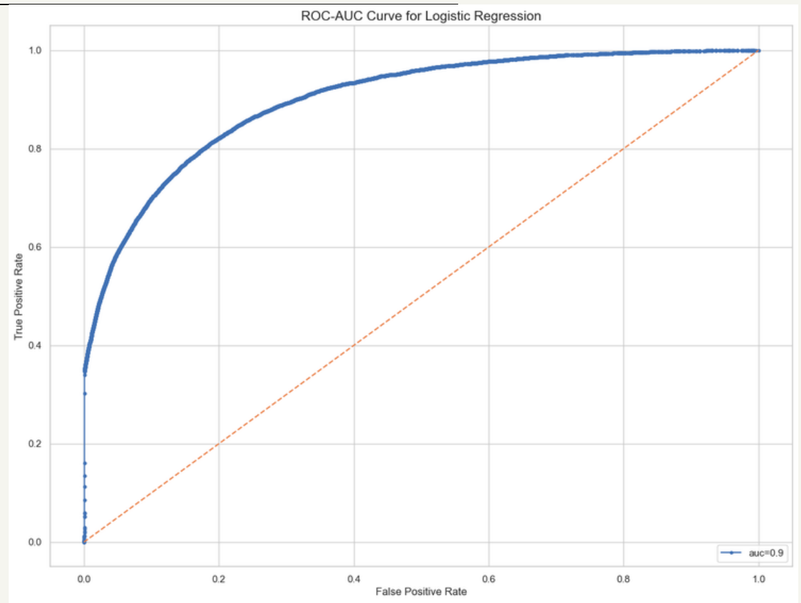
Classification report

	precision	recall	f1-score	support
0.0	0.92	0.99	0.95	54760
1.0	0.85	0.42	0.56	8411
accuracy			0.91	63171
macro avg	0.89	0.70	0.76	63171
weighted avg	0.91	0.91	0.90	63171

- Precision (Accuracy of positive predictions): Out of all survey respondents that the model predicted would have diabetes, 85% did.
- Recall (Fraction of correctly identified positive predictions): Out of all survey respondents with diabetes, our model only predicted correctly for 42% of them.
- F1 Score (Harmonic mean of precision and recall): This value is not close to 1, indicating that the model performance may not be quite good at predicting whether the respondent has diabetes. Since we have imbalanced data, we will consider the F1 score as a major evaluation metric for capturing Precision and Recall. And we'll use this to compare classifier models as we move forward.

Apply ROC-AUC Plot to evaluate binary classification problem

As the AUC score is at 0.9, close to 1, the classifier did a great job distinguishing all the positive and negative points correctly, namely those who have diabetes and those who don't.

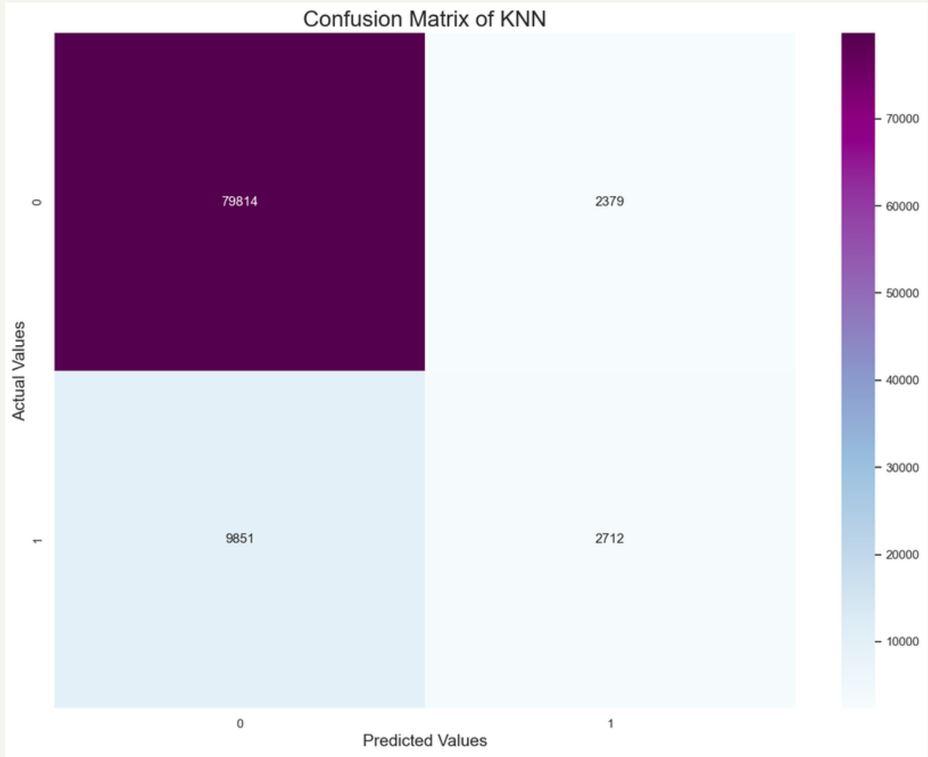


K-NEAREST NEIGHBORS

	precision	recall	f1-score	support
0.0	0.89	0.97	0.93	82193
1.0	0.53	0.22	0.31	12563
accuracy			0.87	94756
macro avg	0.71	0.59	0.62	94756
weighted avg	0.84	0.87	0.85	94756

- KNN Score for test set: 0.870932
- KNN Score for training set: 0.902432
- Precision (Accuracy of positive predictions): Out of all survey respondents that the model predicted would have diabetes, 53% did.
- Recall (Fraction of correctly identified positive predictions): Out of all survey respondents with diabetes, our model only predicted correctly for 21% of them.
- F1 Score (Harmonic mean of precision and recall): This value is not close to 1, indicating that the model performance may not be quite good at predicting whether the respondent has diabetes.

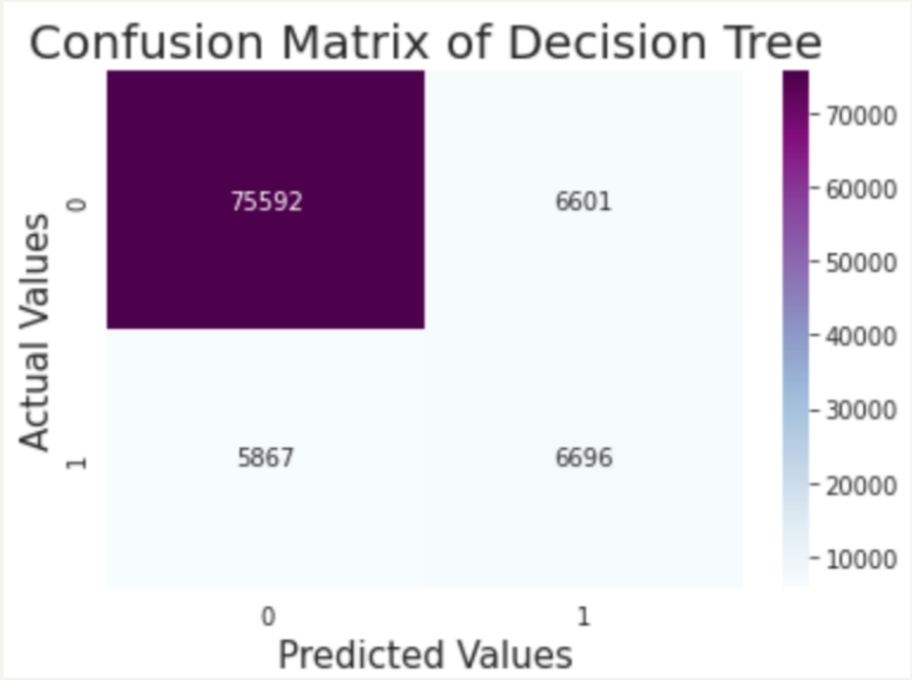
Confusion matrix:



DECISION TREE CLASSIFIER

- Evaluating the model with best estimator found from Grid Search

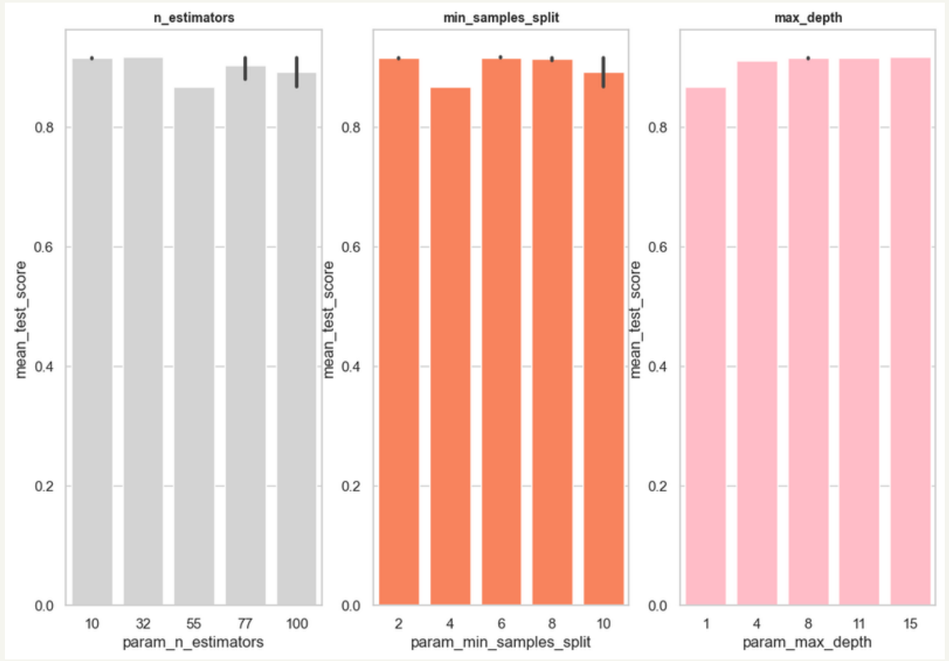
```
Train Accuracy: 0.9177962613694441
Test Confusion Matrix: [[189526   2257]
 [ 15918  13396]]
--------------------
Test Accuracy: 0.9139790620119043
Test Confusion Matrix: [[81142  1051]
 [ 7100  5463]]
```



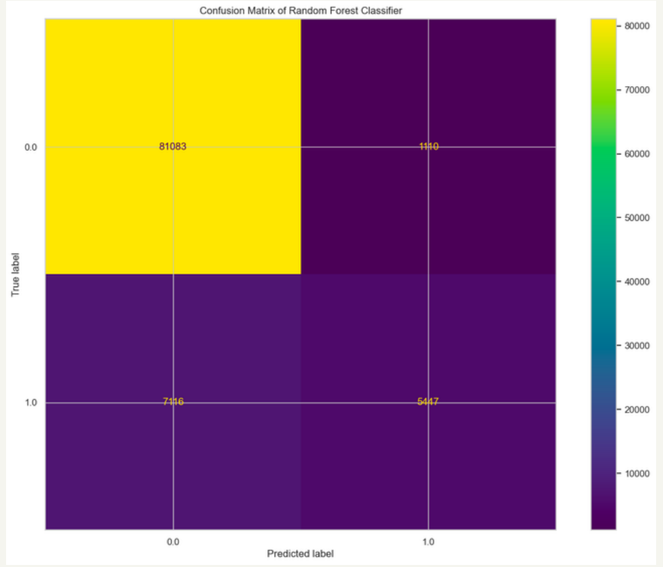
	precision	recall	f1-score	support
0.0	0.92	0.99	0.95	82193
1.0	0.84	0.43	0.57	12563
accuracy			0.91	94756
macro avg	0.88	0.71	0.76	94756
weighted avg	0.91	0.91	0.90	94756

- Precision (Accuracy of positive predictions): Out of all survey respondents that the model predicted would have diabetes, 84% did.
- Recall (Fraction of correctly identified positive predictions): Out of all survey respondents with diabetes, our model only predicted correctly for 43% of them.
- F1 Score (Harmonic mean of precision and recall): This value is not close to 1, indicating that the model performance may not be quite good at predicting whether the respondent has diabetes.
- Since we have imbalanced data, we will consider the F1 score as a major evaluation metric for capturing Precision and Recall. And we'll use this to compare classifier models as we move forward.

RANDOM FOREST CLASSIFIER



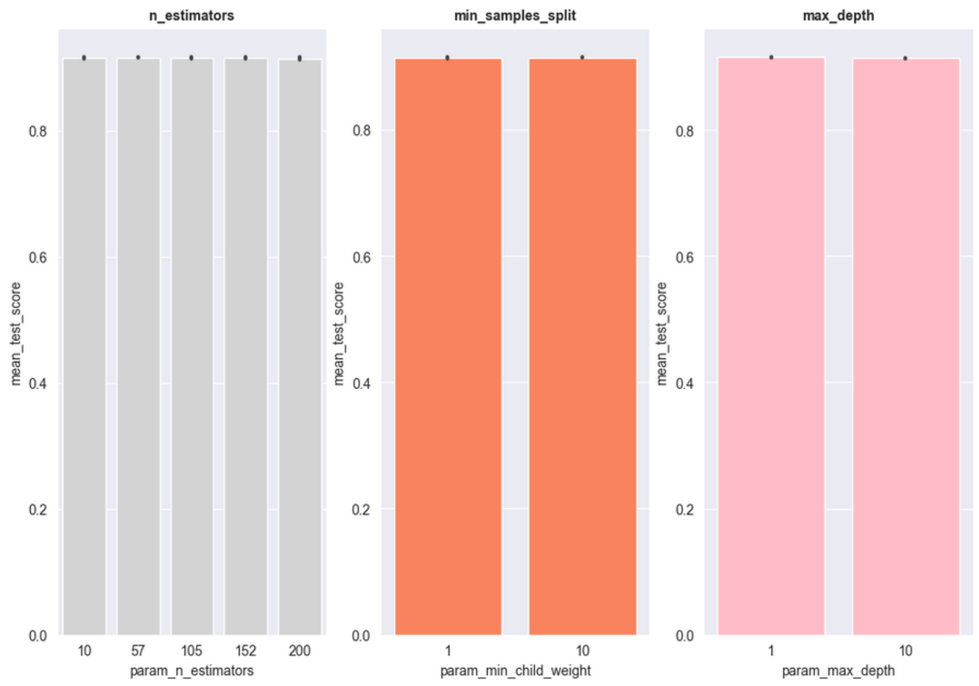
- n_estimators: 10, 32, 77, and 100 seem to have the highest average scores.
- min_samples_split: 2, 6, 8, 10 seem to perform the best.
- max_depth: values from 4 to 15 seem to perform well.



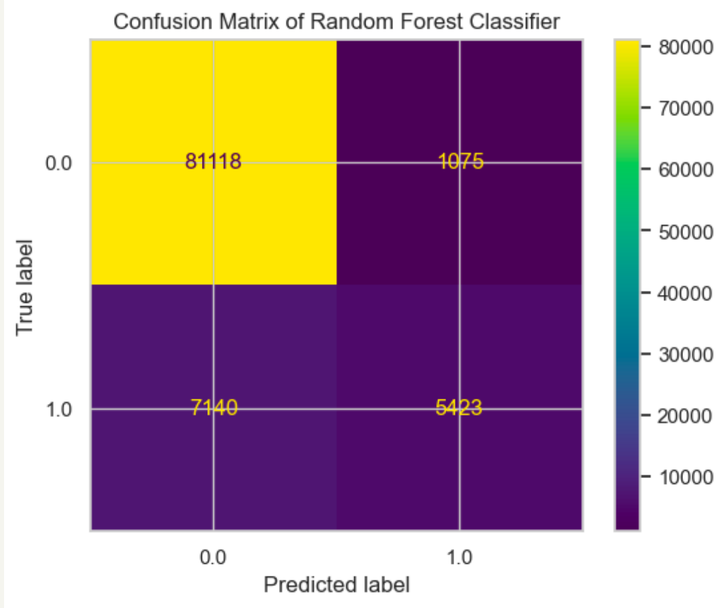

```
Train Accuracy: 0.9562002198130233
Test Confusion Matrix: [[191497    286]
 [  9398  19916]]
--------------------
Test Accuracy: 0.9133036430410739
Test Confusion Matrix: [[81118  1075]
 [ 7140  5423]]
```

	precision	recall	f1-score	support
0.0	0.92	0.99	0.95	82193
1.0	0.83	0.43	0.57	12563
accuracy			0.91	94756
macro avg	0.88	0.71	0.76	94756
weighted avg	0.91	0.91	0.90	94756

XGBOOST CLASSIFIER



- n_estimators: 10, 57, 105, 152, and 200 seem to have the highest average scores.
- min_samples_split: 1, 10 seem to perform the best.
- max_depth: values from 1 to 10 seem to perform well.




```
Train Accuracy: 0.916575077906982
Test Confusion Matrix: [[189824   1959]
 [ 16486  12828]]
--------------------
Test Accuracy: 0.9148233357254422
Test Confusion Matrix: [[81355   838]
 [ 7233  5330]]
```

	precision	recall	f1-score	support
0.0	0.92	0.99	0.95	82193
1.0	0.86	0.42	0.57	12563
accuracy			0.91	94756
macro avg	0.89	0.71	0.76	94756
weighted avg	0.91	0.91	0.90	94756

NEXT STEPS

Project GitHub: https://github.com/les1smore/DATA606-Capstone-Project

Tune models to have high recall

- The recall is the measure of our model correctly identifying True Positives. Thus, for all the patients who actually have diabetes disease, recall tells us how many we correctly identified as having diabetes.
- We want to avoid the case when the patient has diabetes, but no treatment is given to them as the models predicted.

Threshold moving for imbalanced classifications

- The default threshold for interpreting probabilities to class labels is 0.5, and such a default threshold for imbalanced classification can result in poor performance.
- A simple and straightforward approach is to tune the threshold used to map probabilities to class labels.

Apply resampling techniques to deal with imbalanced classes

- Over-sampling: Duplicating random records from the minority class can cause overfishing.
- Under-sampling: removing random records from the majority class can cause a loss of information.
- Extra concern: Long running time for models with increased rows.

References:

1. https://towardsdatascience.com/precision-and-recall-a-simplified-view-bc25978d81e6#:~:text=Models%20need%20high%20recall%20when,cover%20false%20negatives%20as%20well.
2. https://machinelearningmastery.com/threshold-moving-for-imbalanced-classification/
3. https://www.analyticsvidhya.com/blog/2020/07/10-techniques-to-deal-with-class-imbalance-in-machine-learning/

THANK YOU!