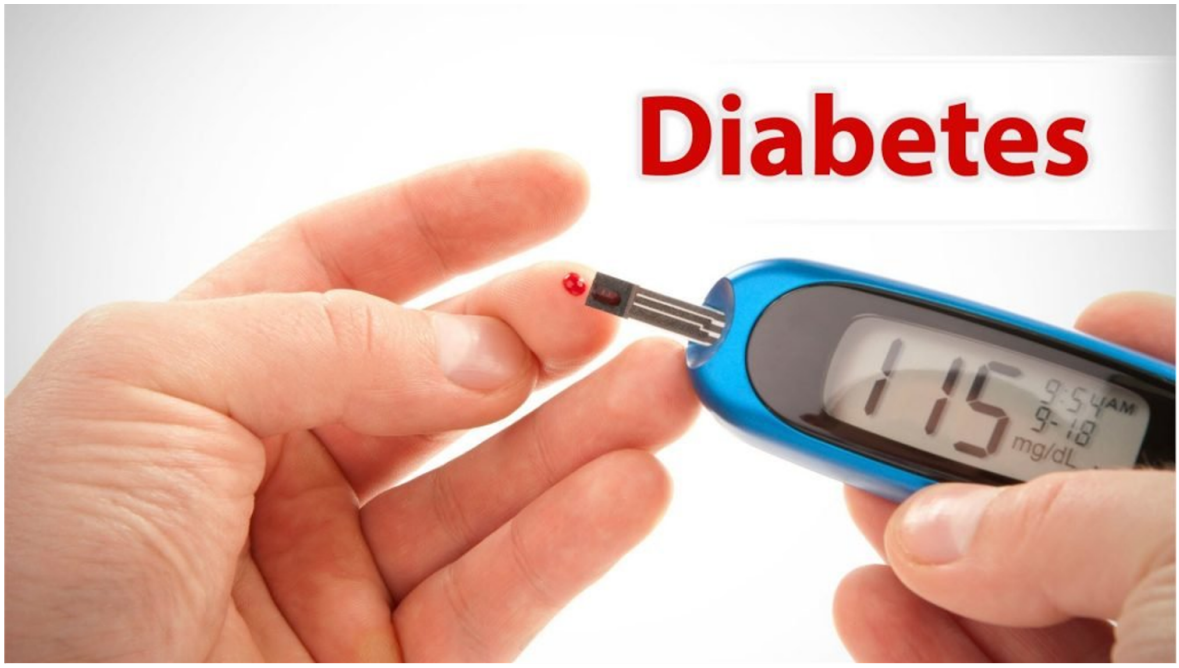


Detecting Diabetes Early

Daniel Kim

Problem Statement

Like any other disease, It is very important to detect the early symptoms of diabetes in order to be cured completely. Reportedly, there are two types of diabetes: type 1 and type 2.
The goal is to classify types of diabetes and their early symptoms so that people can cure diabetes, or prevent diabetes.

(https://archive.ics.uci.edu/ml/datasets/Early+stage+diabetes+risk+prediction+dataset.)

The Data

This data contains 17 attributes which include age, gender, and other symptoms.

1. **Target, y = 'Positive' in the column, 'class'**
2. **Other variables, Xs = all other features than y**
3. **Total number of rows = 521**
4. **Total number of columns = 17**

Data Wrangling

1. Data Cleansing

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 520 entries, 0 to 519
Data columns (total 17 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Age                 520 non-null    int64
 1   Gender              520 non-null    object
 2   Polyuria            520 non-null    object
 3   Polydipsia          520 non-null    object
 4   sudden weight loss  520 non-null    object
 5   weakness            520 non-null    object
 6   Polyphagia          520 non-null    object
 7   Genital thrush      520 non-null    object
 8   visual blurring     520 non-null    object
 9   Itching             520 non-null    object
 10  Irritability        520 non-null    object
 11  delayed healing     520 non-null    object
 12  partial paresis     520 non-null    object
 13  muscle stiffness    520 non-null    object
 14  Alopecia            520 non-null    object
 15  Obesity             520 non-null    object
 16  class               520 non-null    object
dtypes: int64(1), object(16)
memory usage: 69.2+ KB
```

a. Null Value, NaN

There are no Null Values.

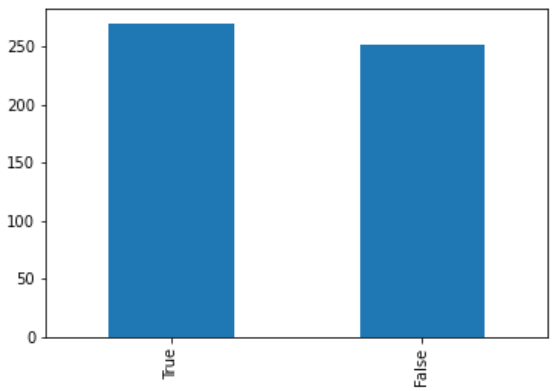
b. Head Samples

```
df.head()
```

	Age	Gender	Polyuria	Polydipsia	sudden weight loss	weakness	Polyphagia	Genital thrush	visual blurring	Itching	Irritability	delayed healing	partial paresis	muscle stiffness	Alopecia	Obesity	class
0	40	Male	No	Yes	No	Yes	No	No	No	Yes	No	Yes	No	Yes	Yes	Yes	Positive
1	58	Male	No	No	No	Yes	No	No	Yes	No	No	No	Yes	No	Yes	No	Positive
2	41	Male	Yes	No	No	Yes	Yes	No	No	Yes	No	Yes	No	Yes	Yes	No	Positive
3	45	Male	No	No	Yes	Yes	Yes	Yes	No	Yes	No	Yes	No	No	No	No	Positive
4	60	Male	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Positive

As seen above, each column shows binary categorical responses like ‘Male’ or ‘Female’, or ‘Yes’ or ‘No’, or ‘Positive’ or ‘Negative’ except for age(Numerical).

c. Duplicate Rows



As seen left, the number of duplicated rows outnumbers the number of NOT duplicated; this can be interpreted as there are many people who share the exact same symptoms.

Thus, it is better not to remove duplicate rows

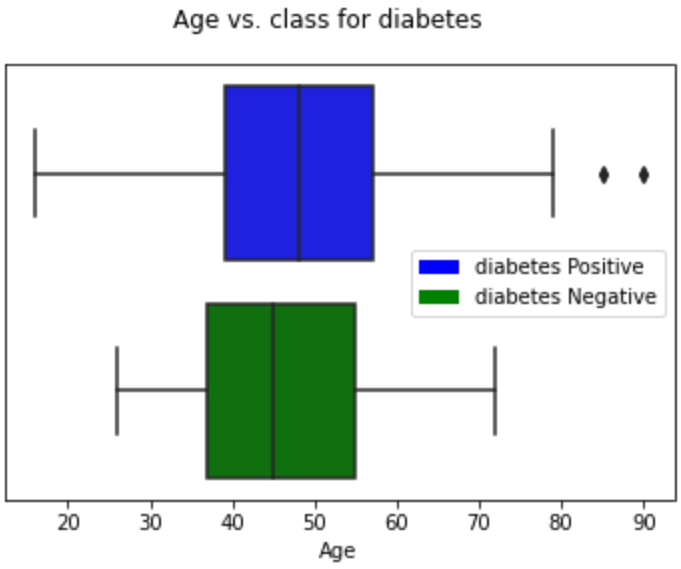
Exploratory Data Analysis

1. Value Counts

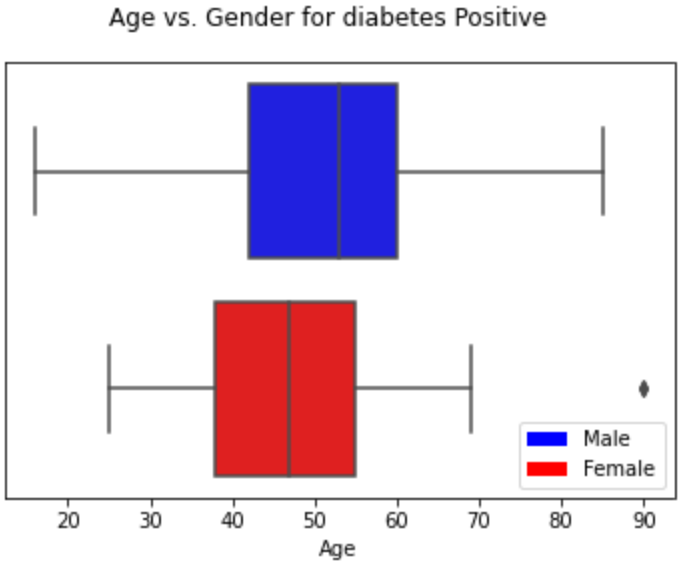
<Overall> <Diabetes Positive Only>

As seen above two value_counts bar charts, people with diabetes positive show different symptoms.

2. Box Plots



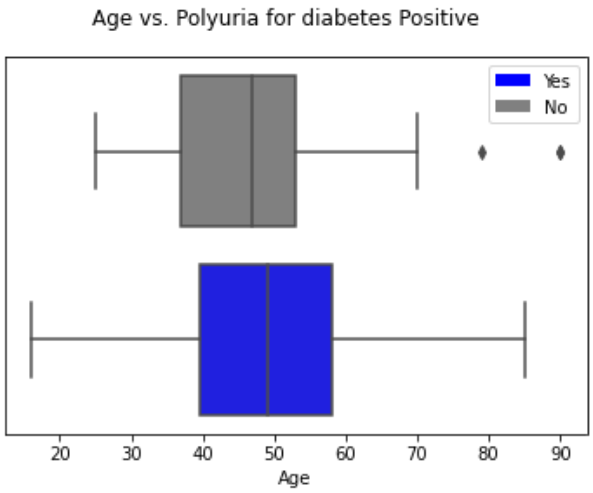
Older ‘Age’ group tends to have diabetes a little bit more.

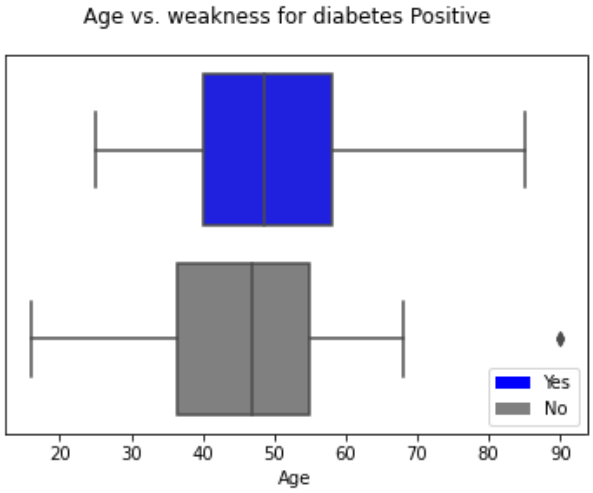


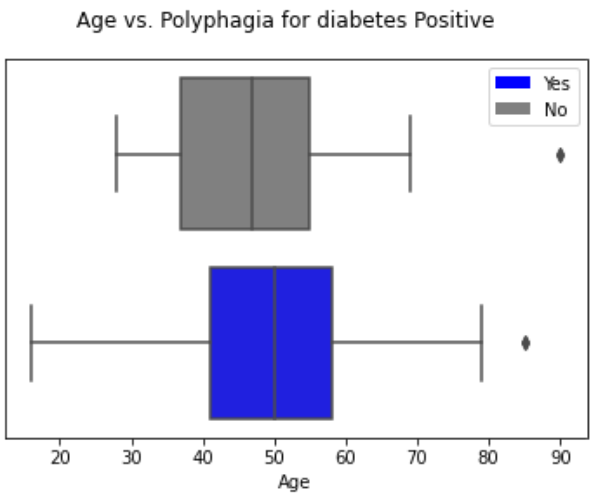
Male ‘Gender’ group with diabetes Positive tends to be older than Female one.

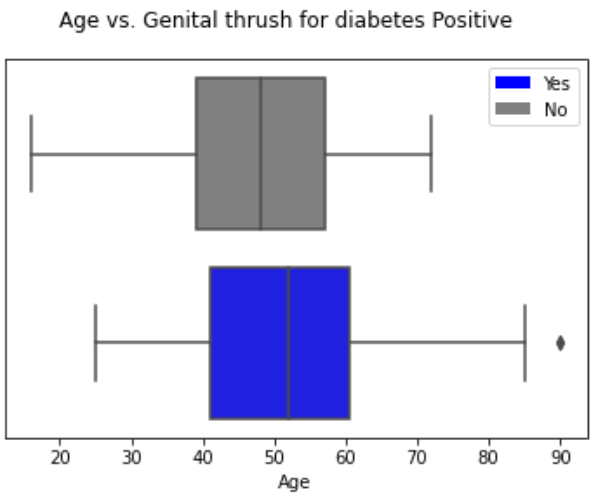
a. Difference in Age Group per Symptom in Diabetes Positive group

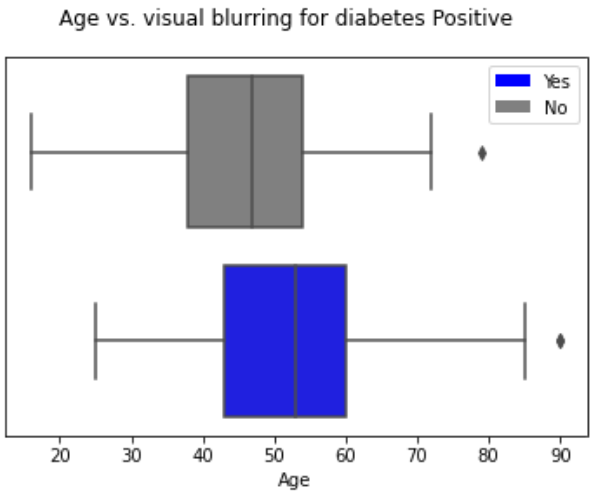
Within the Diabetes Positive group, the below symptoms shows a tendency of differences in age group.

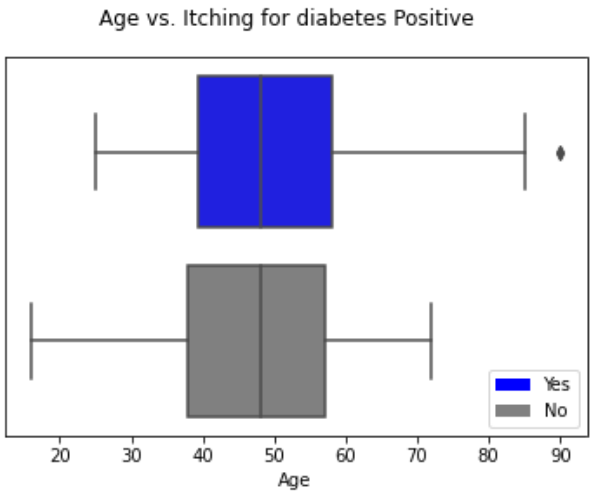


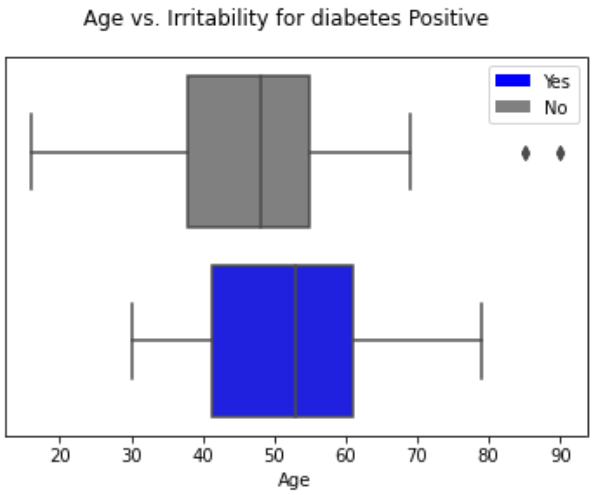


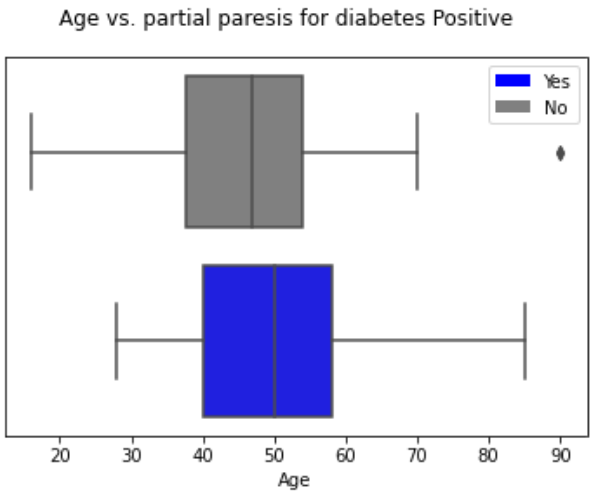


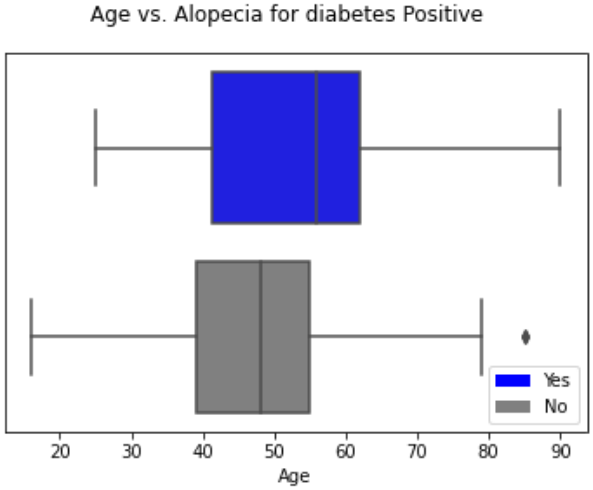






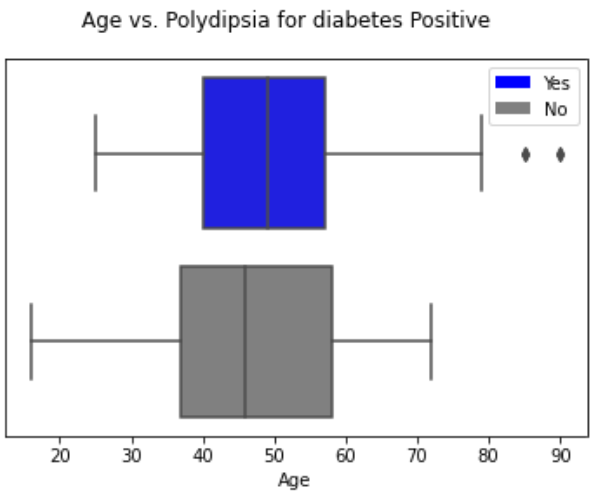

Age vs. Irritability for diabetes Positive
Yes
No
20
30
40
50
60
70
80
90
Age

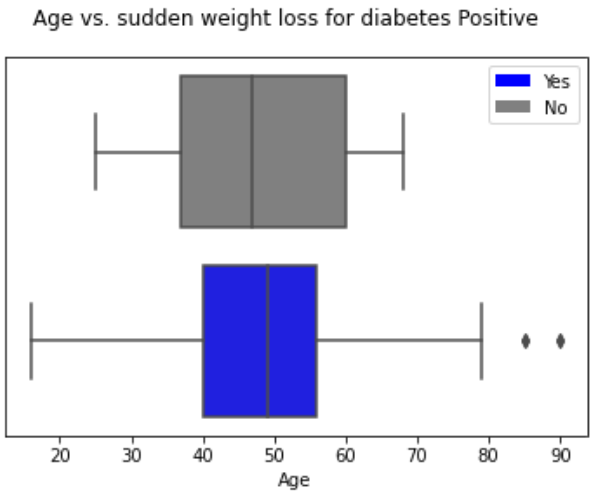

Age vs. partial paresis for diabetes Positive
Yes
No
20
30
40
50
60
70
80
90
Age

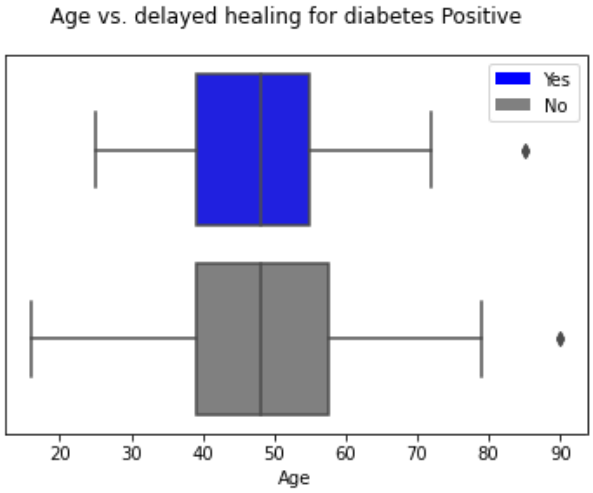

Age vs. Alopecia for diabetes Positive
Yes
No
20
30
40
50
60
70
80
90
Age

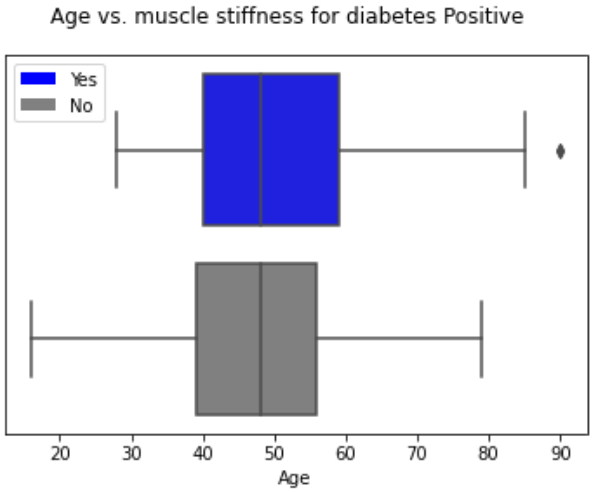
b. Different Variance in Same Age Group per Symptoms in Diabetes Positive group

Within the Diabetes Positive group, the below symptoms show different sizes of variances within similar age groups.

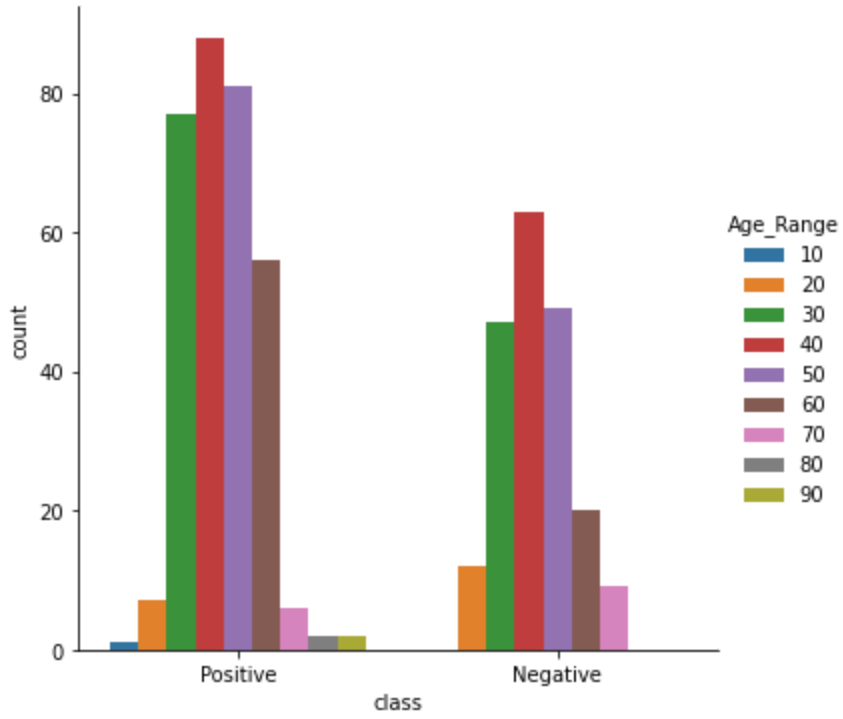








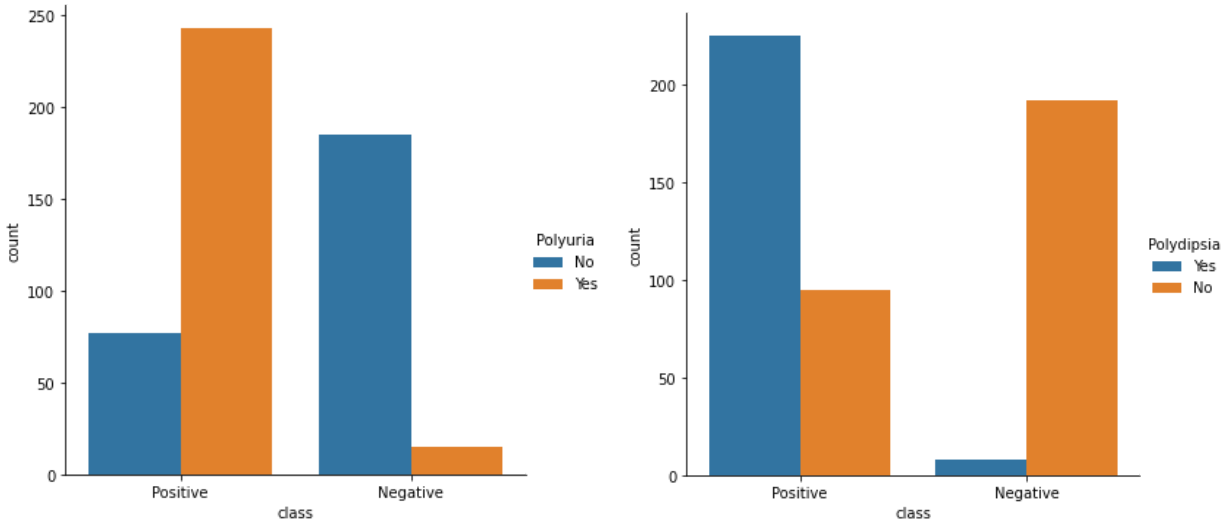
3. Difference in Age Group Distribution

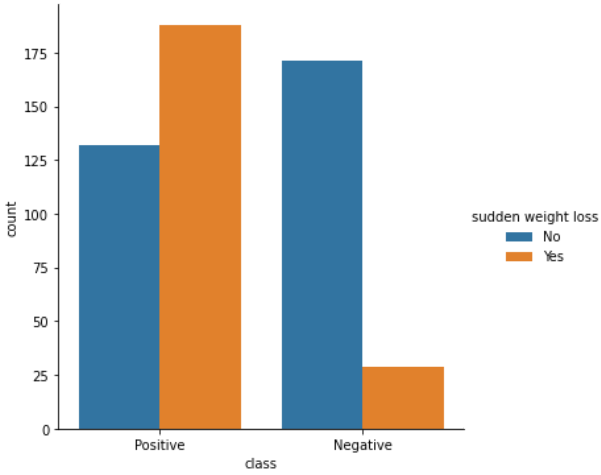


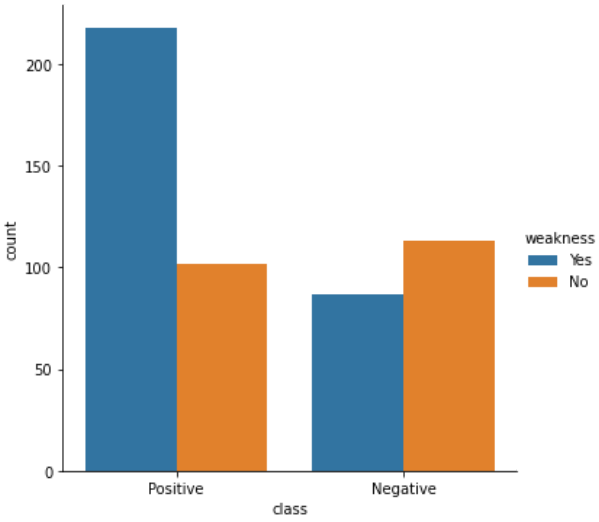
Although each distribution size is different, their age brackets distribute similarly.

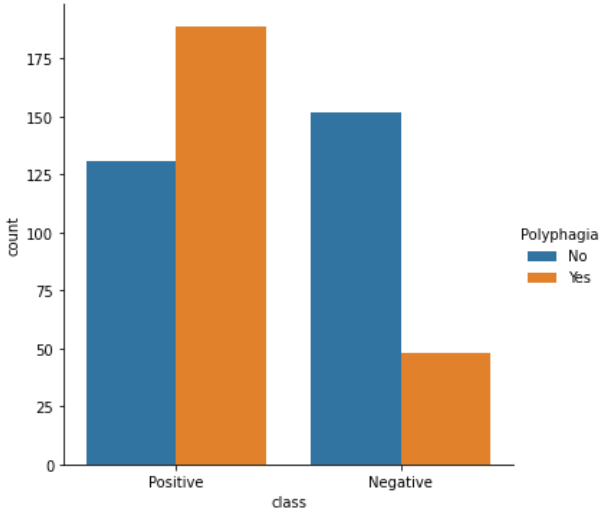
4. Catplots

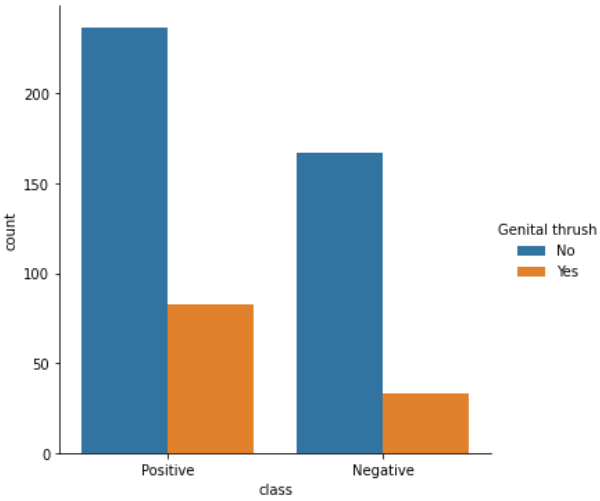
The below Catplots show distinct differences per symptoms whether diabetes is diagnosed as positive or negative.

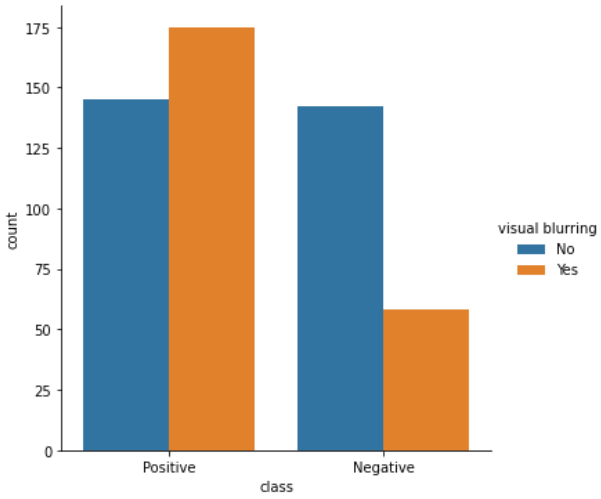


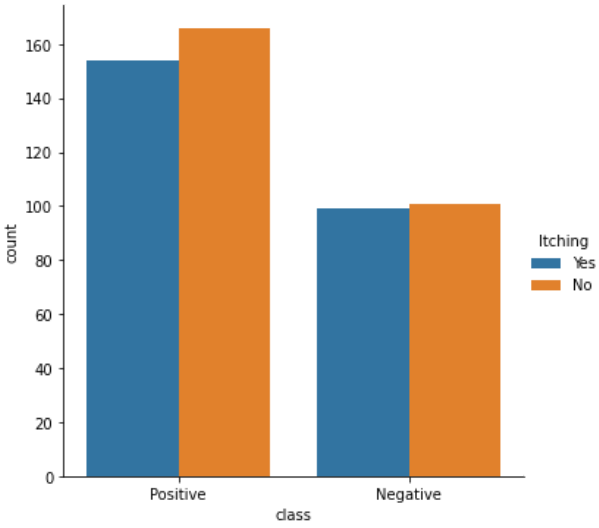

count
class
Positive
Negative
sudden weight loss
No
Yes

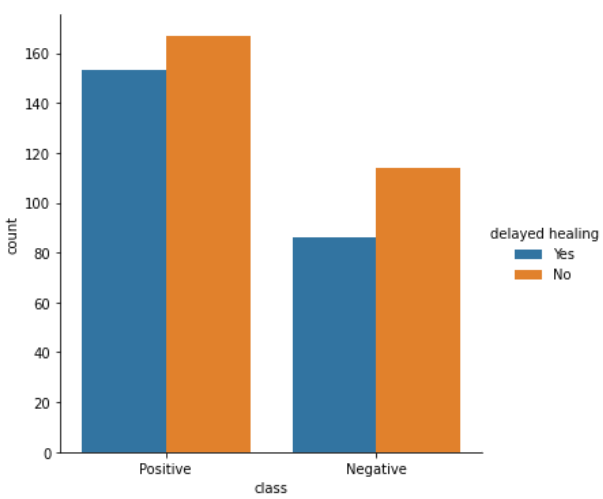

count
class
Positive
Negative
weakness
Yes
No

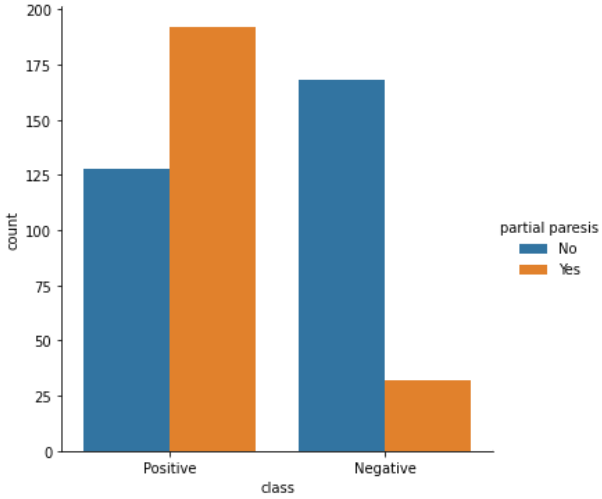

count
class
Positive
Negative
Polyphagia
No
Yes

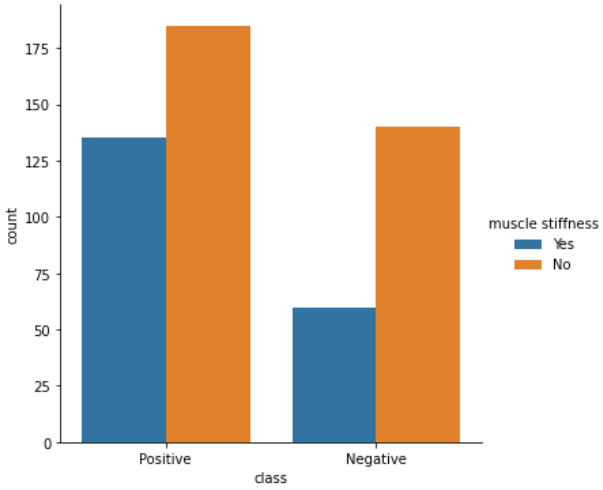

count
class
Positive
Negative
Genital thrush
No
Yes

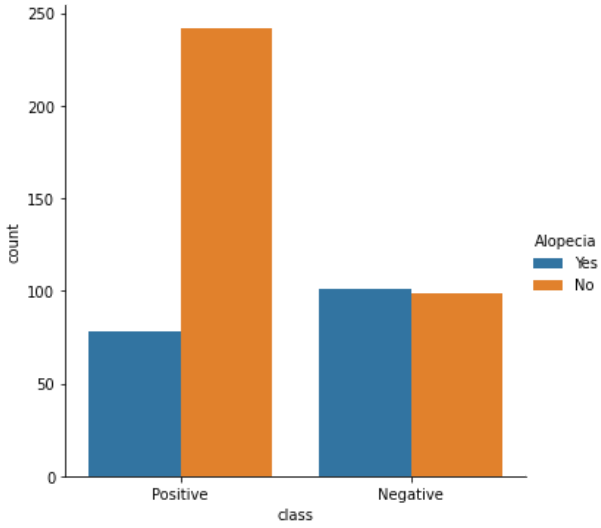

count
class
Positive
Negative
visual blurring
No
Yes

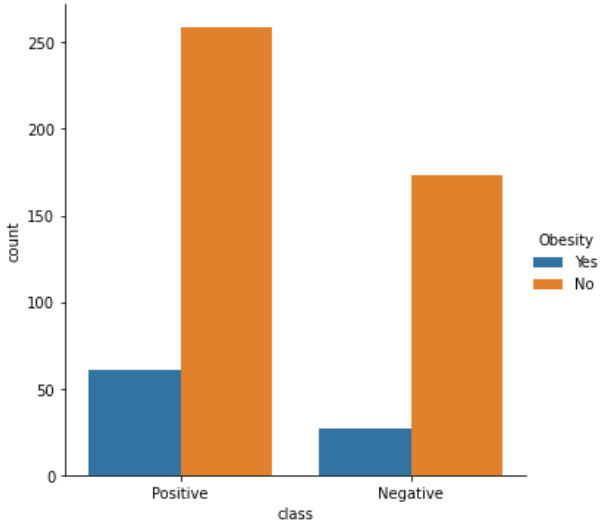

count
class
Positive
Negative
Itching
Yes
No


count
class
Positive
Negative
delayed healing
Yes
No

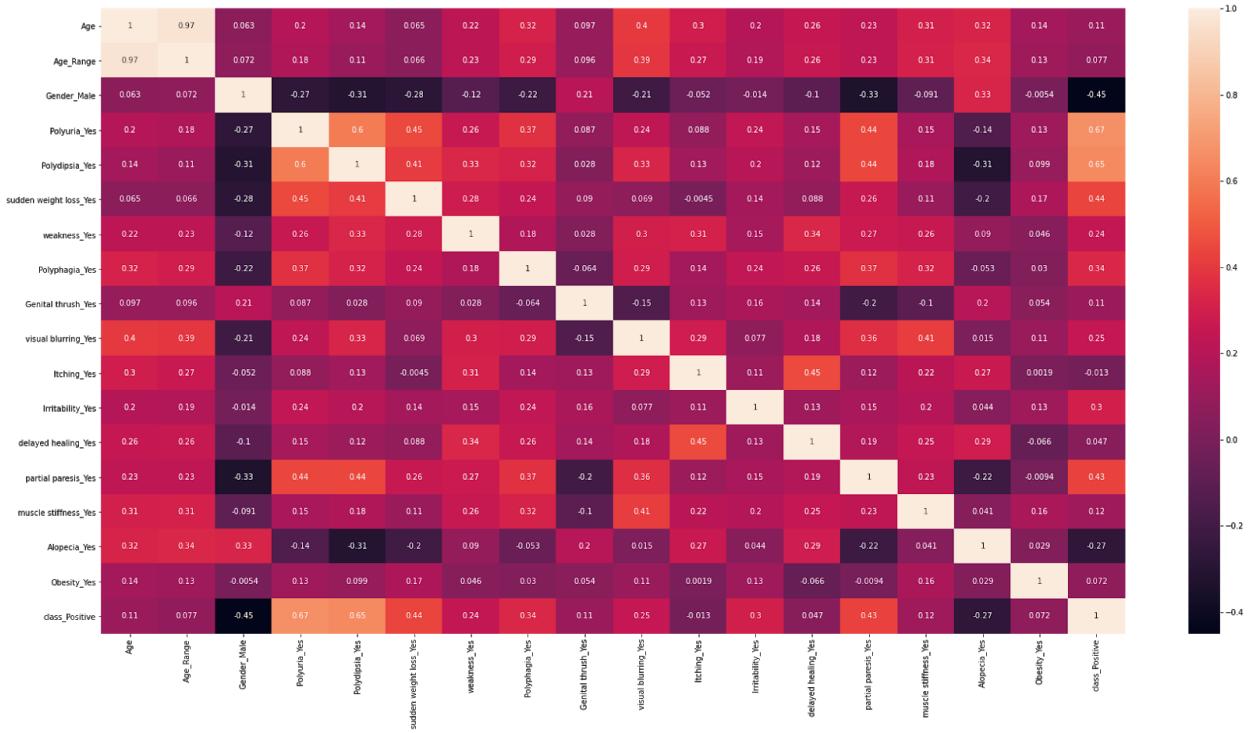

count
class
Positive
Negative
partial paresis
No
Yes


count
class
Positive
Negative
muscle stiffness
Yes
No


count
class
Positive
Negative
Alopecia
Yes
No


count
class
Positive
Negative
Obesity
Yes
No

5. Heatmap



As seen above, too many features seem correlated to each other, so it is needed to simplify their correlation by feature importance of several machine learning algorithms.

Removing Outlier by Machine Learning Algorithm: Isolation Forest

Just for confirming the outlier detection, the Isolation Forest algorithm was applied.

```
In [94]:
%%time
from sklearn.ensemble import IsolationForest
clf=IsolationForest(n_estimators=50, max_samples=50, contamination=float(0.004),
                    max_features=1.0, bootstrap=False, n_jobs=-1,
                    random_state=42, verbose=0,behaviour="new")
clf.fit(df)
pred = clf.predict(df)
df['anomaly']=pred
outliers=df.loc[df['anomaly']==-1]
outlier_index=list(outliers.index)
print(df['anomaly'].value_counts())
```

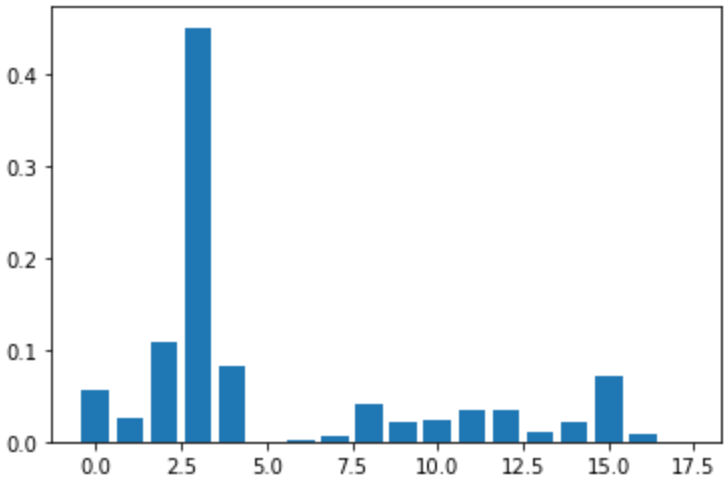
```
 1    518
-1      2
Name: anomaly, dtype: int64
Wall time: 805 ms

C:\Users\danie\Anaconda3\lib\site-packages\sklearn\ensemble\_iforest.py:265: FutureWarning: 'behaviour' is
deprecated in 0.22 and will be removed in 0.24. You should not pass or set this parameter.
  FutureWarning
```

As seen left, only two outliers (“-1”) were detected. So it is not necessary to remove the 2 outliers in that the number of outliers is too small.

Comparison of Feature Importance:

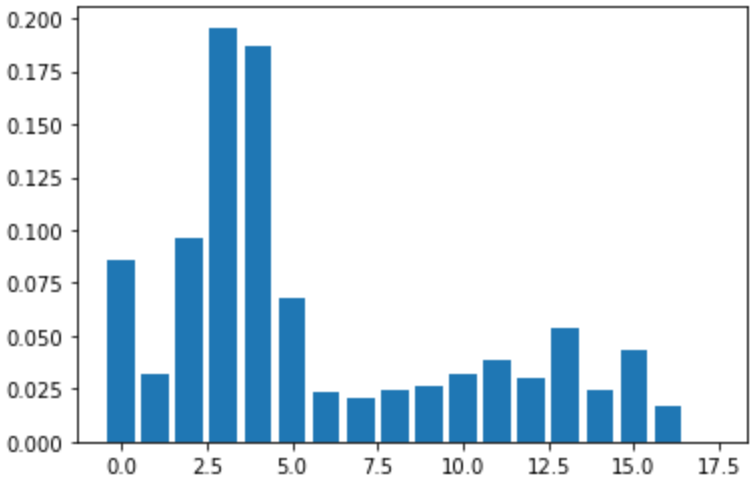
Decision Tree Classifier vs. Random Forest Classifier vs. Gradient Boost Classifier

The below compares each algorithms' feature importances whose importance has more than 5% only.



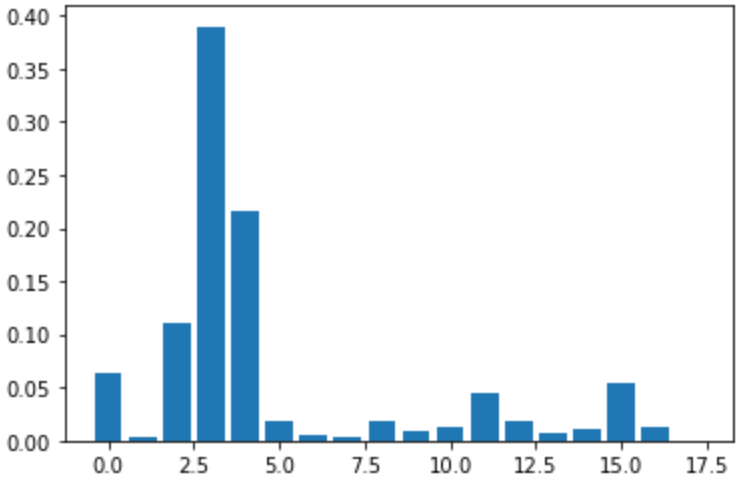
a. Decision Tree Classifier (x>=5% only)

Feature: Age, Score: 6%
Feature: Gender_Male, Score: 11%
Feature: Polyuria, Score: 45%
Feature: Polydipsia, Score: 8%
Feature: Alopecia, Score: 7%



b. Random Forest Classifier (x>=5% only)

Feature: Age, Score: 9%
Feature: Gender_Male, Score: 10%
Feature: Polyuria, Score: 20%
Feature: Polydipsia, Score: 19%
Feature: sudden weight loss, Score: 7%
Feature: partial paresis, Score: 5%



c. Gradient Boost Classifier (x>=5% only)

Feature: Age, Score: 6%
Feature: Gender_Male, Score: 11%
Feature: Polyuria, Score: 39%
Feature: Polydipsia, Score: 22%
Feature: Alopecia, Score: 5%

By comparing each feature importance from three different machine learning algorithms, ‘Polyuria’ showed the strongest feature when it comes to identifying ‘class_positive’’. So we can conclude that ‘Polyuria’ plays a key role in detecting diabetes.

Besides ‘Polyuria’, ‘Gender_Male’ is the next common strong feature to detect diabetes even if their importance rates vary among machine learning algorithms.

Last but not least, except for the above-mentioned features, ‘Polydipsia’ is the next strong feature to detect diabetes.

In a nutshell, combinations of the above-mentioned features can give you optimal diabetes procedures. For example, a test of ‘Polyuria’ and checking ‘Gender’ should be an independent variable for detecting diabetes. ‘Polydipsia’ should be dependent variable with higher weights than others. And then, the other features should be optional test variables with lower weights.

Comparison of Accuracy

Algorithms	**Decision Tree Classifier**	Random Forest Classifier	Gradient Boost Classifier
Best Score	**97% by Precision**	95% by Recall	96% by Precision
Best Parameters by Randomized Search CV	**'splitter': 'random', 'random_state': 42, 'min_samples_split' : 5, 'min_samples_leaf': 2, 'max_features': None, 'max_depth': 15, 'criterion': 'entropy', 'class_weight': 'balanced'**	'n_estimators': 1200, 'min_samples_split': 5, 'min_samples_leaf': 2, 'max_features': ‘sqrt’, 'max_depth': 15, 'criterion': 'entropy', 'class_weight': 'balanced'	'n_estimators': 1200, 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_features': 'auto', 'max_depth': 15, 'loss': 'deviance', 'criterion': 'mse'

Given the above test results, the Decision Tree Classifier showed the best score (97% by precision). Other algorithms showed quite remarkable accuracy scores, but their scores are still lower than the Decision Tree Classifier.

Recommendation & Solution

In order to identify diabetes, you may conduct either of the following testing methods on a trade-off basis for both efficiency and effectiveness.

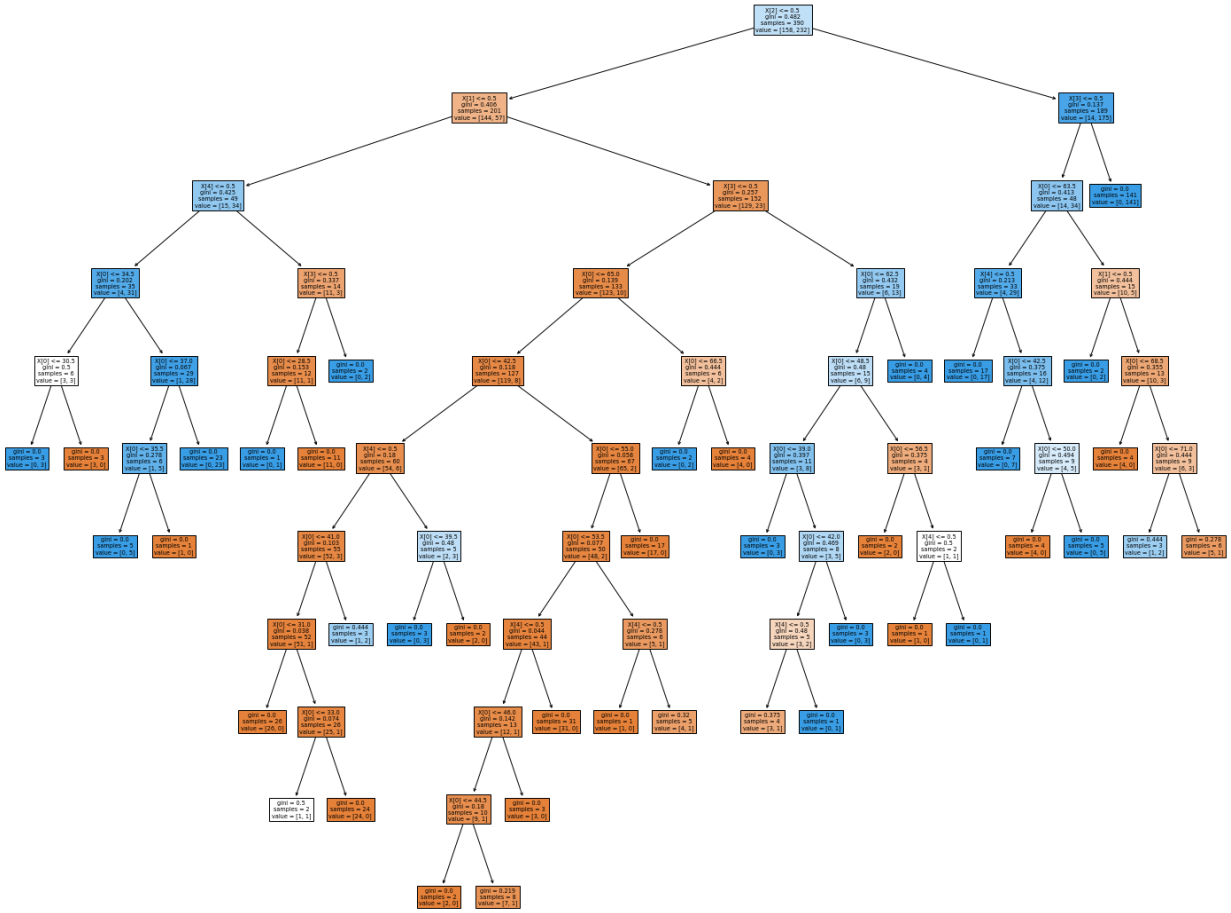
On one hand, combinations of the above-mentioned features can give you optimal diabetes procedures. For example, a test of ‘Polyuria’ and confirmation of ‘Gender’ should be an independent variable for detecting diabetes. ‘Polydipsia’ should be dependent variables with higher weights than others. And then, the other features should be optional test variables with lower weights.

On the other hand, you can just conduct a diabetes test by the Decision Tree Classifier method.

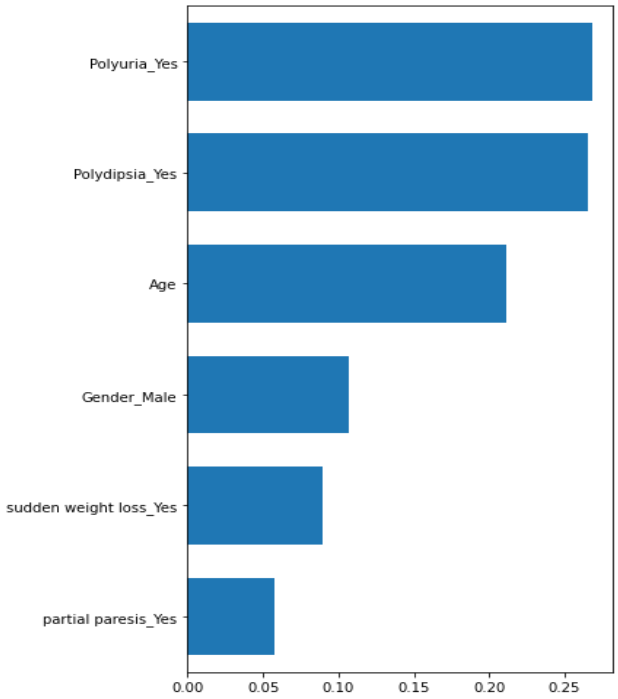
The recommended solution can be both effective and efficient for medical labs to detect diabetes by reducing not only the number of tests but also lead time to test diabetes.

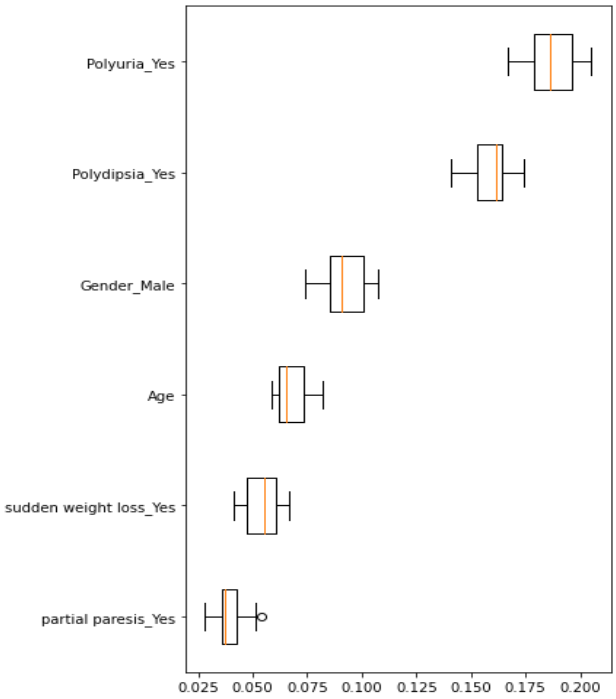
In-depth analysis using advanced and/or specialized techniques

1. Decision Tree

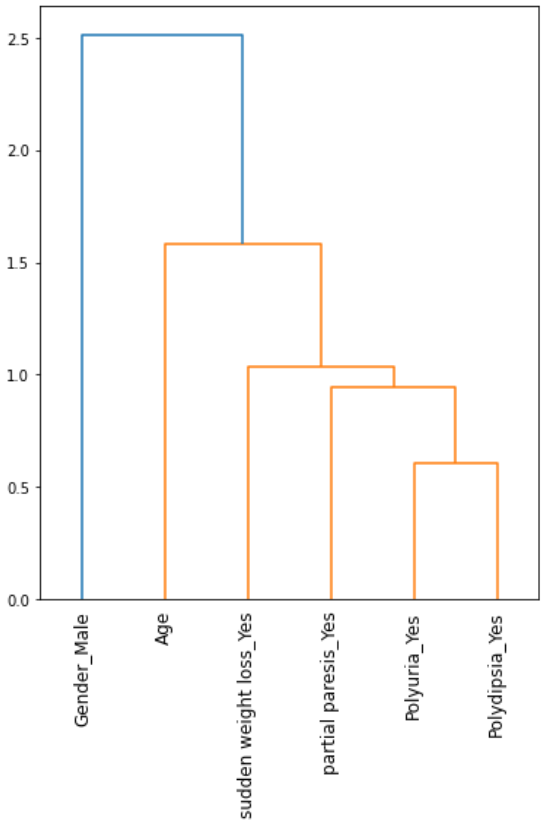


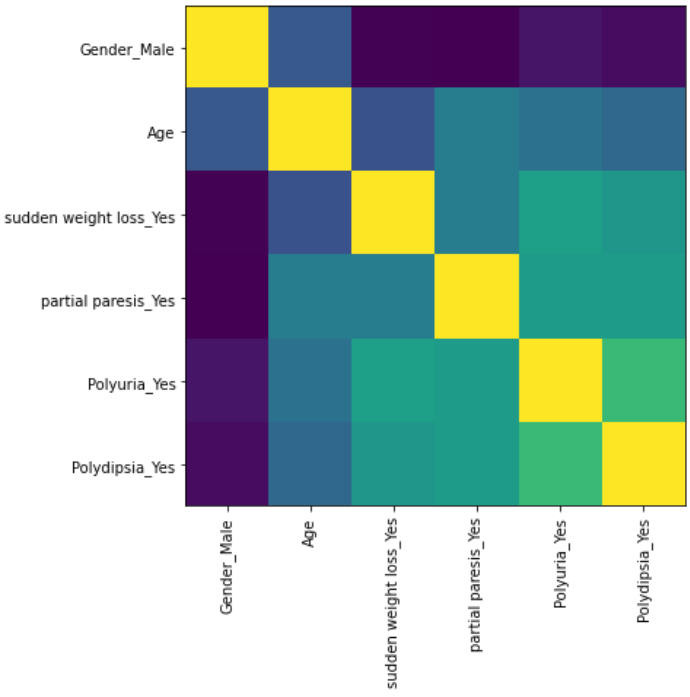
2-1. Permutated Key Importance & Boxplot of Random Forest Classifier



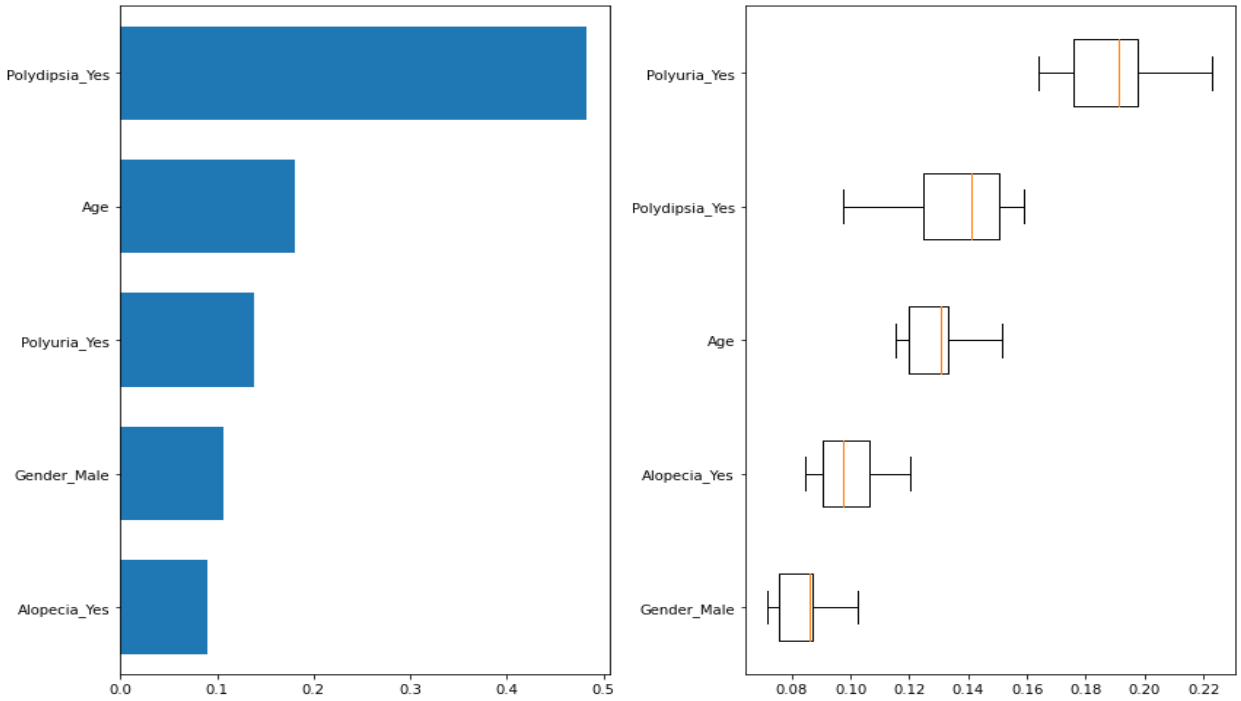


2-2. Dendrogram & Heatmap of Random Forest Classifier





3-1. Permutated Key Importance & Boxplot of Gradient Boost Classifier



3-2. Dendrogram & Heatmap of Gradient Classifier