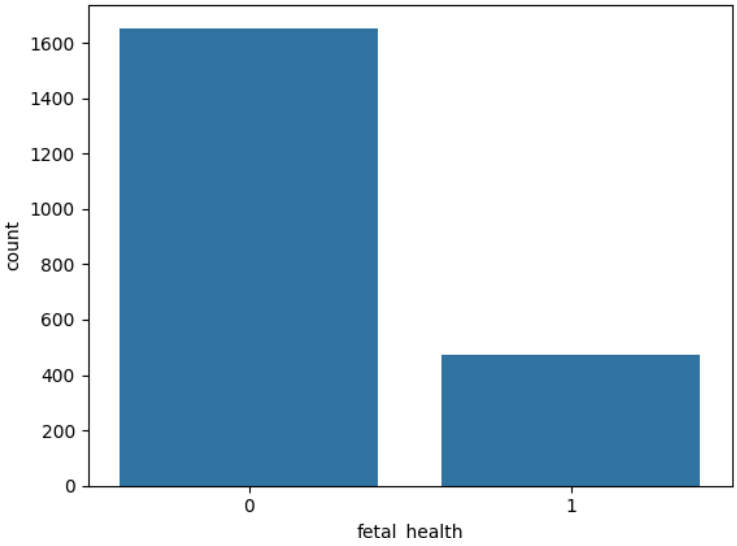
Comparing Classifiers

Yiming Liu 400419748

In this project, I have used 5 classifiers to identify the fetal health is normal or abnormal.
I get data from Kaggle, its about the fetal health classification (Reference 1). The dataset is in csv format. There are 20 features in the dataset exclude the fetal health feature. There are 2126 data points in the dataset. There are no null data points. (Picture 1)
But the fetal health feature has three classifications, I change the classification from 1 means Normal, 2 means Suspect and 3 means Pathologic, to 0 means Normal and 1 means Abnormal. (Picture 2)

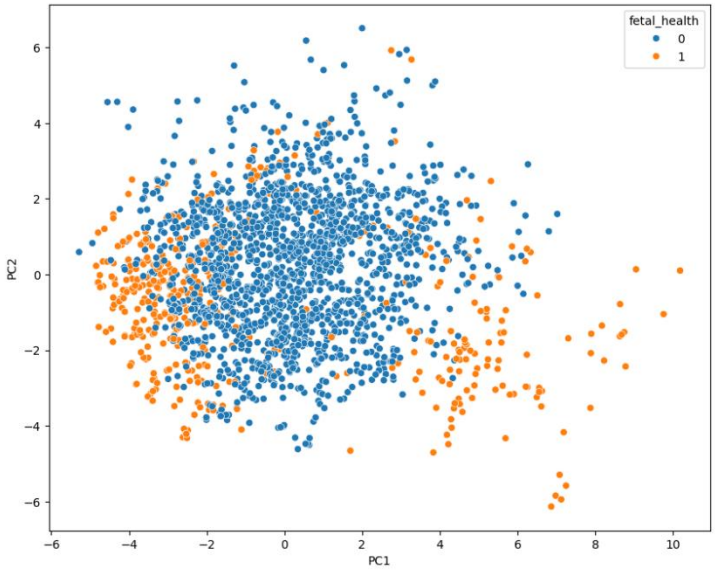
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2126 entries, 0 to 2125
Data columns (total 22 columns):
 #   Column                                                  Non-Null Count  Dtype
---  ------                                                  --------------  -----
 0   baseline value                                          2126 non-null   float64
 1   accelerations                                           2126 non-null   float64
 2   fetal_movement                                          2126 non-null   float64
 3   uterine_contractions                                    2126 non-null   float64
 4   light_decelerations                                     2126 non-null   float64
 5   severe_decelerations                                    2126 non-null   float64
 6   prolongued_decelerations                                2126 non-null   float64
 7   abnormal_short_term_variability                         2126 non-null   float64
 8   mean_value_of_short_term_variability                    2126 non-null   float64
 9   percentage_of_time_with_abnormal_long_term_variability  2126 non-null   float64
 10  mean_value_of_long_term_variability                     2126 non-null   float64
 11  histogram_width                                         2126 non-null   float64
 12  histogram_min                                           2126 non-null   float64
 13  histogram_max                                           2126 non-null   float64
 14  histogram_number_of_peaks                               2126 non-null   float64
 15  histogram_number_of_zeroes                              2126 non-null   float64
 16  histogram_mode                                          2126 non-null   float64
 17  histogram_mean                                          2126 non-null   float64
 18  histogram_median                                        2126 non-null   float64
 19  histogram_variance                                      2126 non-null   float64
 20  histogram_tendency                                      2126 non-null   float64
 21  fetal_health                                            2126 non-null   float64
dtypes: float64(22)
memory usage: 365.5 KB
```

Picture 1

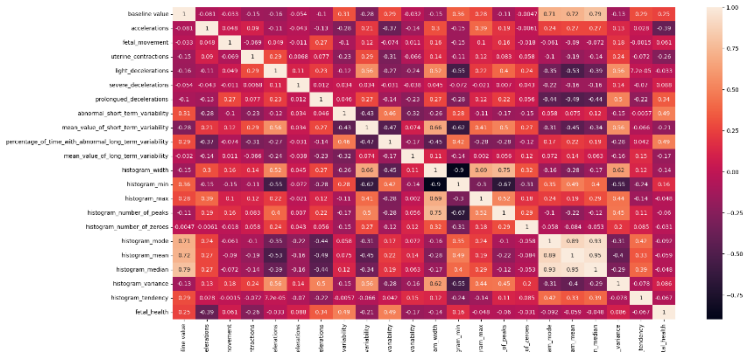


Picture 2

After reframing the data frame with fetal_health feature in 2 classifications. I want to see the data distributed on a graph. There are 20 features in the data, so I use scaler to scale the data in unit variance and remove the mean. Then I use Principal component analysis to make the data to be visualized (Picture 3).



Picture 3



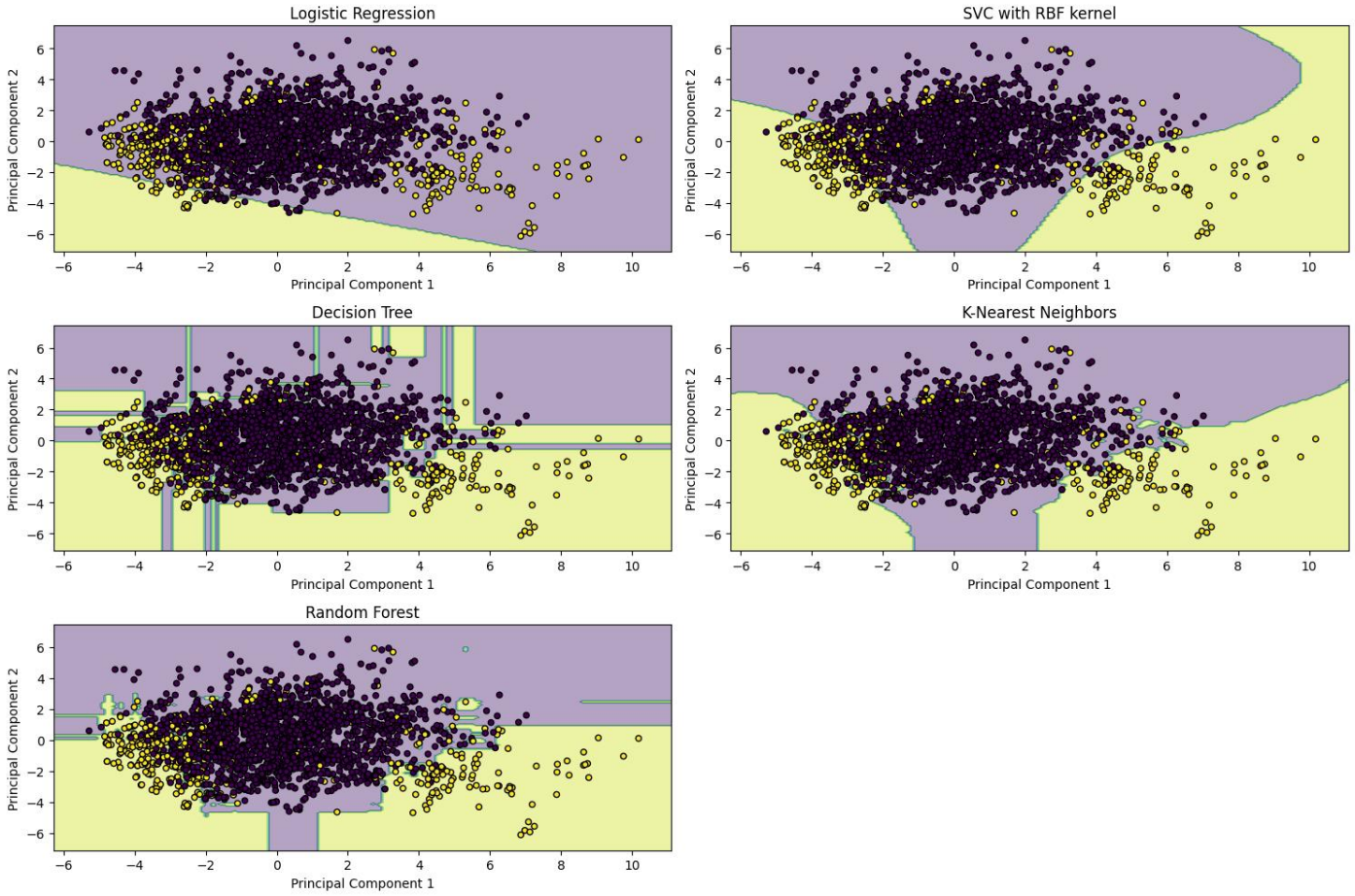
Picture 4

Next step is to choose the significant features from 20 features. I use the heat map to present the correlation matrix to visualize the relationship between each feature (Picture 4). The fetal health feature is the result we want to get. So, I sort the values in the correlation matrix between fetal health and all other features. I made a list of them, and I choose the features which absolute value is larger than 0.1. After sorting and choosing, there are 10 features left in the dataset.

For training the datasets, I scaled the new reframed dataset and separate them in to training data and test data in a test size of 25%.
For model choosing, I searched on google and looks around some websites (Reference 2 & Reference 3). I choose to use Logistic Regression, Support Vector Machine, Decision Tree, K Neighbors and Random Forest models to test the model. I found that these five models are interesting and want to give them a try.
I use the classifications from sklearn and use a model selection function called cross_val_score to calculate the cross-value score for each algorithm.

- Logistic Regression Accuracy: 0.90 (+/- 0.04)
- SVC Accuracy: 0.91 (+/- 0.05)
- Decision Tree Accuracy: 0.91 (+/- 0.05)
- KNN Accuracy: 0.92 (+/- 0.05)
- Random Forest Accuracy: 0.95 (+/- 0.04)

The Random Forest get the highest accuracy 95% with a 4% standard deviation. All other four methods are similar. After that I am curious about the how the decision boundary of each algorithm, I use PCA again to make it into a graph with good visualization (Picture 5).

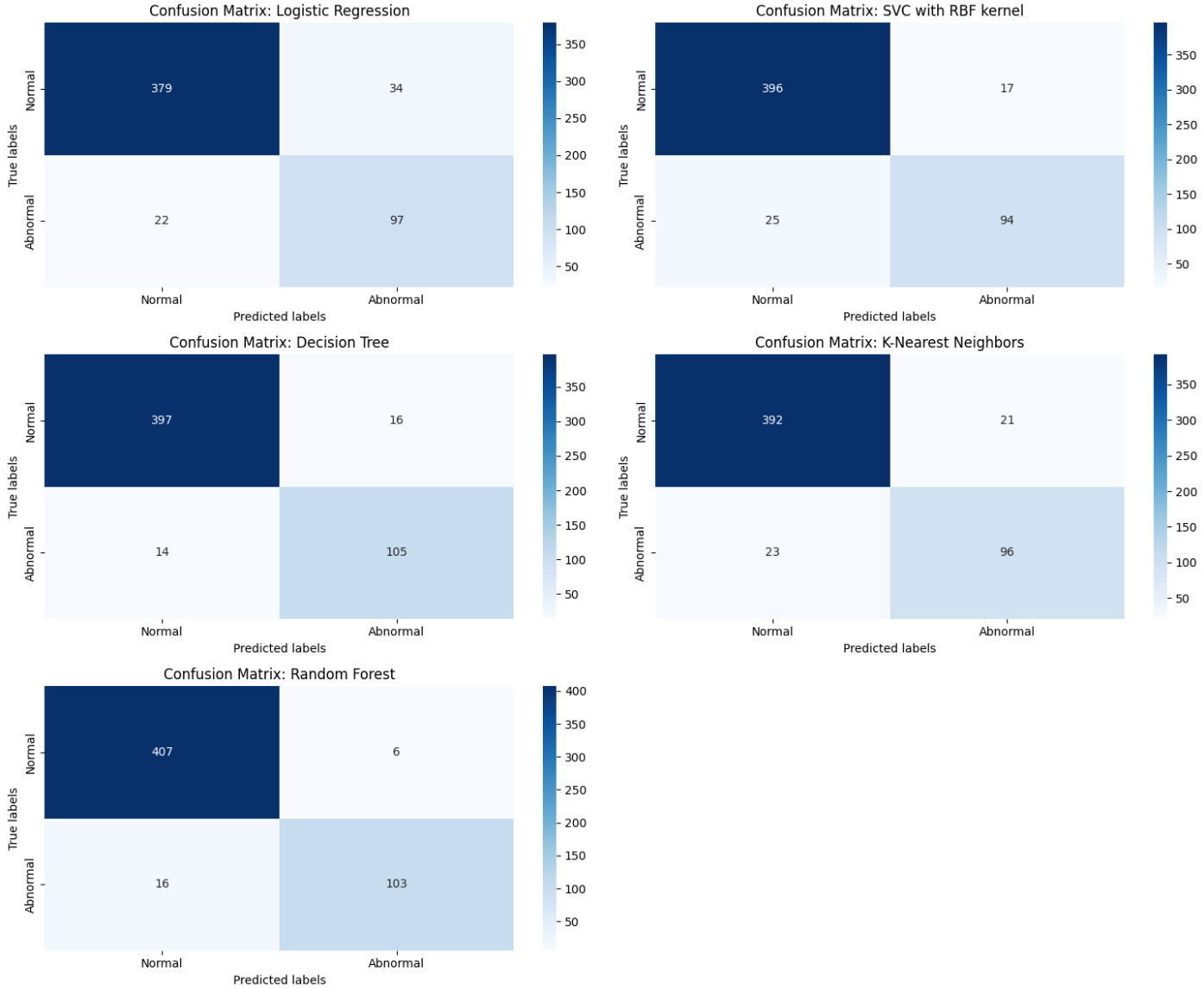


Picture 5

From the graph, I can imagine that logistic regression can have a 90% of accuracy.
I record each classifier's training time and testing time for this dataset.

- Logistic Regression takes 9ms in training and 1ms in testing.
- SVC with RBF kernel takes 101ms in training and 14ms in testing.
- Decision Tree takes 6ms in training and 0.0ms in testing.
- K-Nearest Neighbors takes 3ms in training and 20ms in testing.
- Random Forest takes 190.2ms in training and 8ms in testing.

Finally, I make a confusion matrix for each algorithm to see the correct and false predictions using confusion_matrix in sklearn.metrics (Picture 6). From the matrix, I see that the decision tree has the lowest error in the False Negative part and the Random Forest has the second lowest in False negative. I think the False Negative part low in this dataset is the most important because this part is which the fetal health is not normal, but the algorithm predict it as normal. This will delay the abnormal situation of fetal to be discovered. I will choose Random Forest model to train for this dataset.



Picture 6

Reference

1. https://www.kaggle.com/datasets/andrewmvd/fetal-health-classification/data
2. https://towardsdatascience.com/top-10-binary-classification-algorithms-a-beginners-guide-feeacbd7a3e2
3. https://www.mathworks.com/campaigns/offers/next/choosing-the-best-machine-learning-classification-model-and-avoiding-overfitting.html