COEN242 Final Project Report

Team 25

Haoning Xia

Tianyi Liu

Menghao Huo

Introduction:

In recent years, big data and machine learning have been widely used in various fields of human's life. Such fields as manufacturing, service, financial, medical have also become more prosperous due to the addition of machine learning and big data technology. At present, most of the achievements in the development of machine learning are closely related to big data. Through data collection, processing, and analysis, we can obtain valuable insights from the massive data of various industries and provide predictions of those difficult-to-predict future events. Breast cancer is a cancer disease that develops from breast tissue. Signs of breast cancer may include a lump in the breast, a change in breast shape, dimpling of the skin, fluid coming from the nipple, a newly-inverted nipple, or a red or scaly patch of skin. This big data analytics project can be used to study various patterns that affect the recurrence of breast cancer such as: age, menopause, tumor-size, inv-nodes, node-caps, degree of malignancy, breast, breast quadrant, irradiate. We use Breast Cancer UCI dataset as our initial data set for this project. Later on, we also applied our algorithm to the Chess (King-Rook vs. King) UCI dataset to predict the optimal depth-of-win for White through King and Rook location, which human-machine battle is also a very popular field of machine learning nowadays.

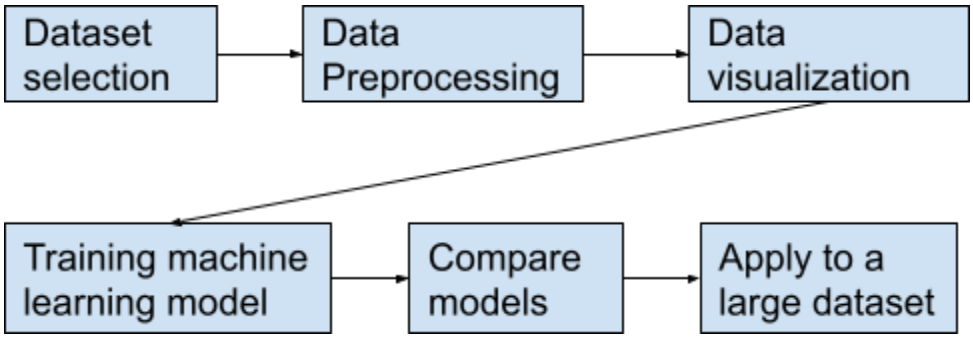
Motivation(problem design):

1.We want to find which factor has a more important impact on cancer. So we try to get answers from visualizing the dataset into bar graphs and box graphs. And analyze the impact on recurrence events by controlling different variables.

2.We want to use those most important attributes of breast cancer to predict the recurrence event using binary classification algorithms such as SVM, Naive Bayes, Decision Tree and Random Forest. We will introduce those algorithms in the following sections.

3.We will make comparisons between each machine learning algorithm to find which machine learning model can give the best prediction and analyze why this algorithm is better than the other.

4. We want to apply the same process and the same algorithm to the Chess dataset, which Is a large data set with sufficient data.We want to find out whether more sufficient data will improve our prediction results



The figure above is our rough process.

Dataset Overview:

We use UCI's breast cancer dataset which have 286 rows of patients, and 10 fields: Class, age, menopause, tumor.size, inv.nodes, node.caps, deg.malig, breast, breast.quad, irradiated

	class	age	menopause	tumor-size	inv-nodes	node-caps	deg-malig	breast	breast-quad	irradiat
256	recurrence-events	40-49	premeno	30-34	0-2	no	1	left	left_low	yes
257	recurrence-events	40-49	premeno	20-24	3-5	yes	2	left	left_low	yes
258	recurrence-events	50-59	ge40	30-34	6-8	yes	2	left	right_low	yes
259	recurrence-events	50-59	ge40	30-34	3-5	no	3	right	left_up	no
260	recurrence-events	60-69	ge40	25-29	3-5	no	2	right	right_up	no
261	recurrence-events	40-49	ge40	25-29	12-14	yes	3	left	right_low	yes
262	recurrence-events	60-69	ge40	25-29	0-2	no	3	left	left_up	no
263	recurrence-events	50-59	lt40	20-24	0-2	?	1	left	left_up	no
264	recurrence-events	50-59	lt40	20-24	0-2	?	1	left	left_low	no
265	recurrence-events	30-39	premeno	35-39	9-11	yes	3	left	left_low	no

The figure shows that there are some missing values in the dataset which are marked with question mark “?”.

There are 8 missing values in the node-capes column and 1 missing value in the breast-quad column in total.

There are two methods to do the data preprocessing:

1.Drop all rolls which have “?” in them

2.Replace the “?” with the most occurrence value of that column.

In this project we choose the first one.

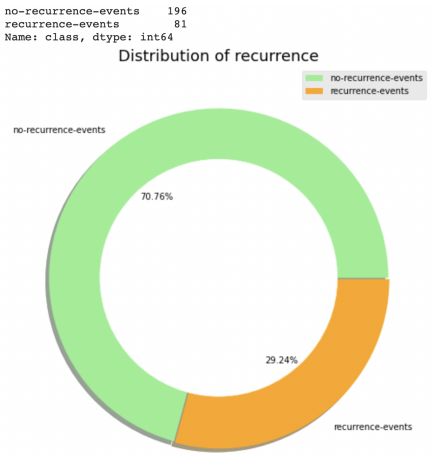
The Chess dataset got 28056 number of instances in total.

Data Visualization:

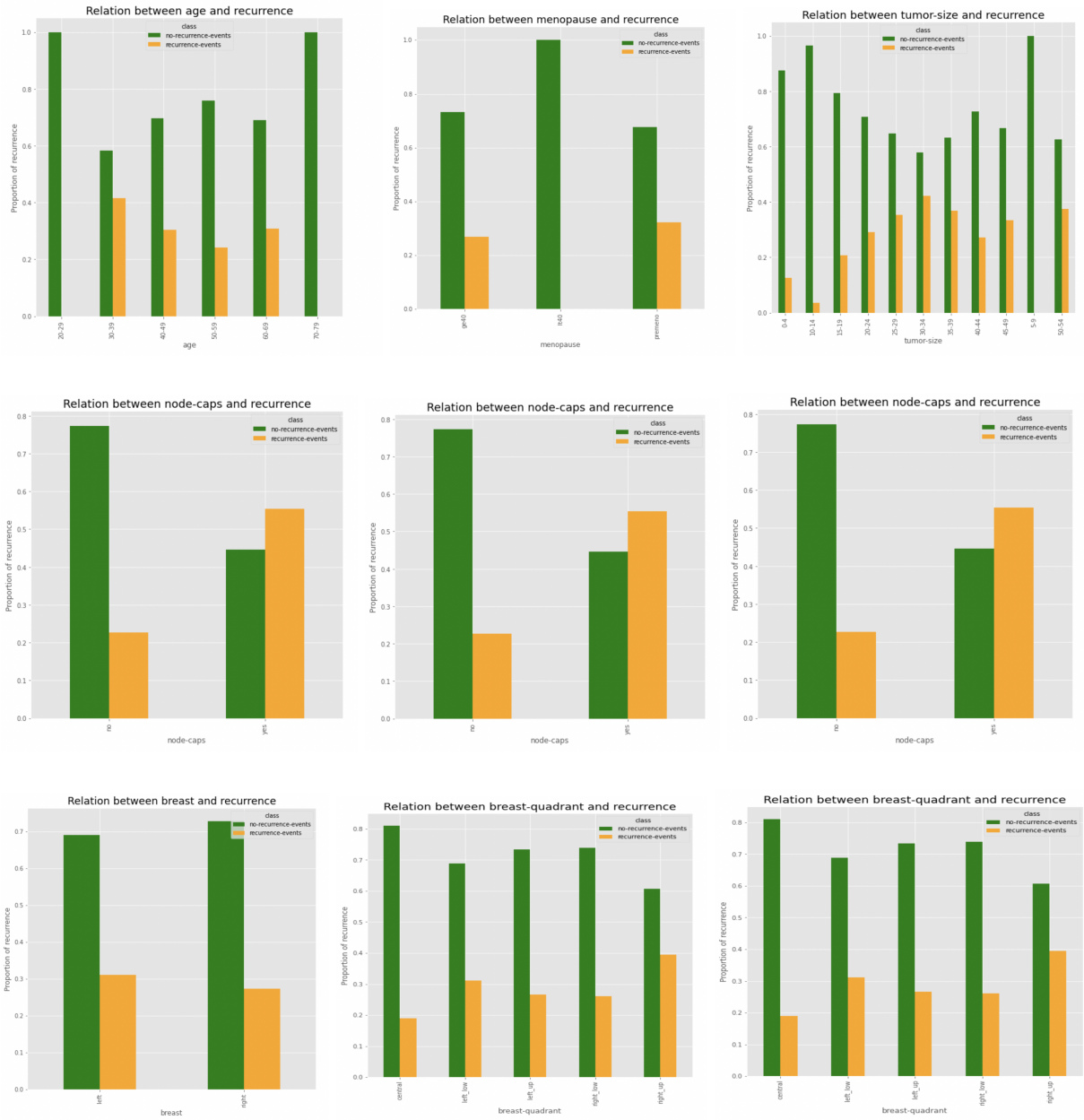
For big data, Data visualization has become an important tool to represent millions of data by making it easier to understand and highlighting only important trends and patterns.

	class	age	menopause	tumor-size	inv-nodes	node-caps	deg-malig	breast	breast-quad	irradiat
count	286	286	286	286	286	286	286	286	286	286
unique	2	6	3	11	7	3	3	2	6	2
top	no-recurrence-events	50-59	premeno	30-34	0-2	no	2	left	left_low	no
freq	201	96	150	60	213	222	130	152	110	218

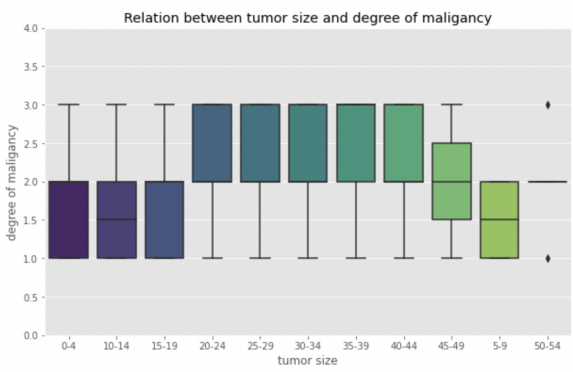
```
no-recurrence-events    196
recurrence-events        81
Name: class, dtype: int64
```

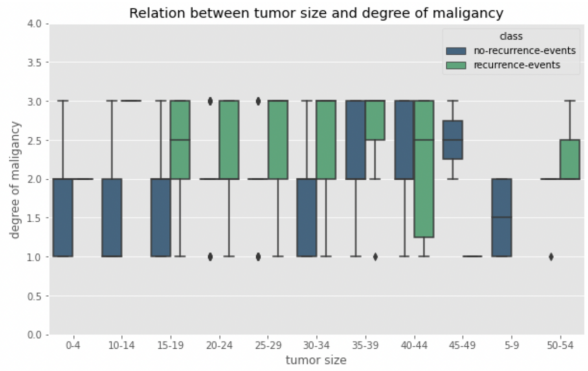


The figure above on the left is a general statistic of the dataset. And the figure above on the right gives people a more intuitive way to visualize the ratio of recurrence and non-recurrence patients in this dataset.Before exploring other aspects of the data set and building a model, we generate a bar graph for each attribute in the plot, where the class variables are highlighted in color to see if there are any interesting interactions between the covariates and class variables. Generate a chart for each of the 9 attributes.



From the graphs above, we can add some comments: 1.Age group and breast quadrant seem to follow normal distributions. 2.the higher degree of malignancy is, the more recurrence-events increase 3. It nearly has the same chance for the recurrence no matter if the tumor is on the left or right of the breast. Thus the tumor location on which side Will not play an important role when we are training the model.





From the box graph above on the left we can see that Tumor size within 20-44 is more likely to have a high degree of malignancy. Then we can combine more factors to observe the effect of recurrence events. From the box graph above on the right we can see that Tumor size within 20-44 and the degree of malignancy is high are more likely to recur.

Data preprocessing:

Loading the data set as RDD type. RDD's paralleliztion feature so we can process and execute data parallelly. The missing value will lead to the incorrect result. In the models of this report, Drop all rolls which have "?" in them. Then, transforming the element in RDD from string to float. Adding the index to all the string type attributes. It will enhance the efficiency of model prediction and make data processing convenient. The dataSet1 transforming detail shows as below:

```
>>> pprint(linesDel.take(5))
[[u'no-recurrence-events',
  u'30-39',
  u'premeno',
  u'30-34',
  u'0-2',
  u'no',
  u'3',
  u'left',
  u'left_low',
  u'no'],
 [u'no-recurrence-events',
  u'40-49',
  u'premeno',
  u'20-24',
  u'0-2',
  u'no',
  u'2',
  u'right',
  u'right_up',
  u'no'],
```

Before

```
>>> pprint(lineTransF.take(5))
[[0.0, 3.0, 2.0, 30.0, 0.0, 0.0, 3.0, 0.0, 22.0, 0.0],
 [0.0, 4.0, 2.0, 20.0, 0.0, 0.0, 2.0, 1.0, 22.0, 0.0],
 [0.0, 4.0, 2.0, 20.0, 0.0, 0.0, 2.0, 0.0, 22.0, 0.0],
 [0.0, 6.0, 1.0, 15.0, 0.0, 0.0, 2.0, 1.0, 22.0, 0.0],
 [0.0, 4.0, 2.0, 0.0, 0.0, 0.0, 2.0, 1.0, 22.0, 0.0]]
>>>
```

After

In the model of this report, the main goal is to predict a target using other features. Thus all of the models use label point RDD. The label means target data and the vector means data used for training and prediction. The label point RDD shows as below:

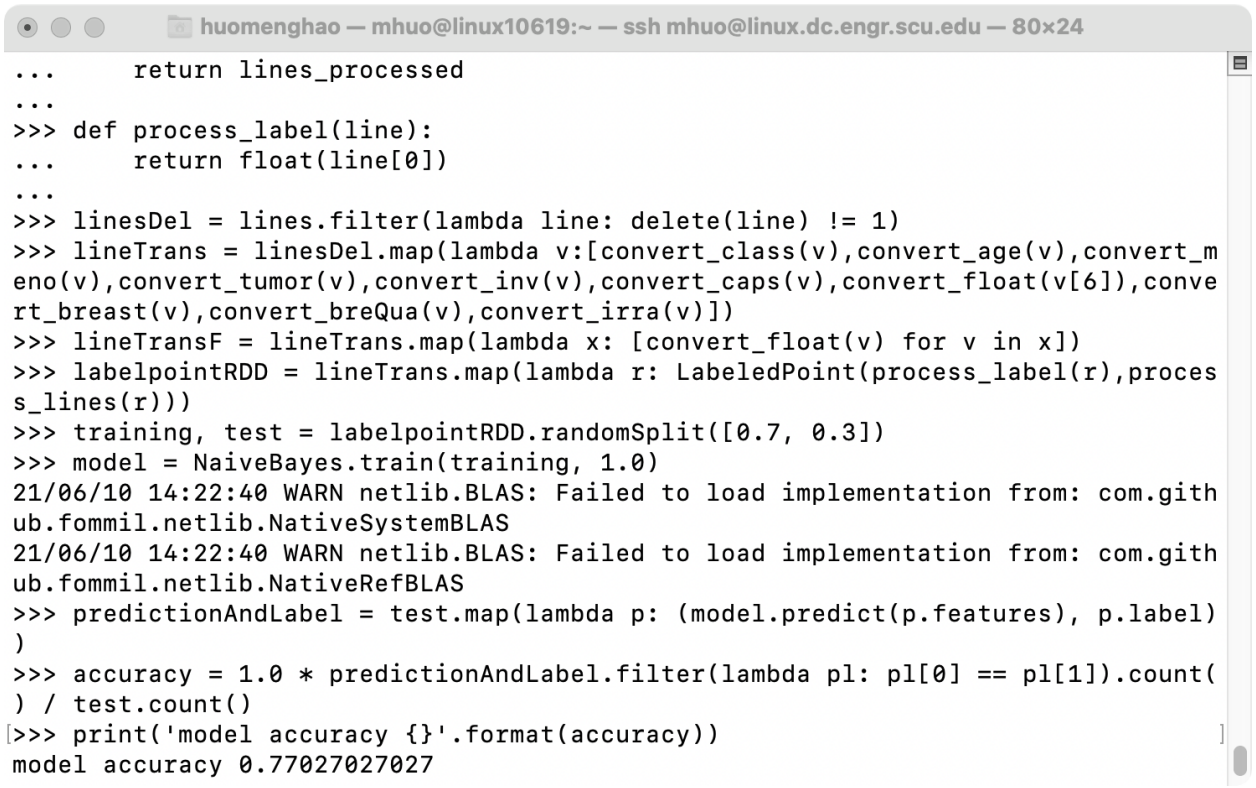
```
>>> pprint(labelpointRDD.take(5))
[Stage 74:>

[LabeledPoint(0.0, [3.0,2.0,30.0,0.0,0.0,3.0,0.0,22.0,0.0]),
 LabeledPoint(0.0, [4.0,2.0,20.0,0.0,0.0,2.0,1.0,22.0,0.0]),
 LabeledPoint(0.0, [4.0,2.0,20.0,0.0,0.0,2.0,0.0,22.0,0.0]),
 LabeledPoint(0.0, [6.0,1.0,15.0,0.0,0.0,2.0,1.0,22.0,0.0]),
 LabeledPoint(0.0, [4.0,2.0,0.0,0.0,0.0,2.0,1.0,22.0,0.0])]
>>>
```

Label point RDD

DataSet I:

Naive Bayes

We implement the naive bayes by pyspark machine learning library. Finally, we get the result of this model:



For the given category to be classified, naive bayes solve the probability of each category under the condition of the occurrence of this category, which is the largest, the category to be classified is considered to belong to.

Random Forest

We implement the naive bayes by the pyspark machine learning library and the experimental result changes when we change the two attributes: number of trees, maximum depth.

When maximum depth is 4:

numTrees	accuracy

3	63.73%
10	74.07%
30	75.30%
50	75.60%
70	75%
90	74.11%

When number of trees is 50:

maxdepth	accuracy
4	76.54%
10	77.77%
20	81.39%
30	80.18%

Based on the result, we can easily get the conclusion that the number of trees and maximum depth will affect the accuracy. And when the maximum depth of random forests is 20 the model can get the highest accuracy.

Linear Support Vector Machines (SVM)

Based on Pyspark machine learning library implementation, We received the following results:

Number of iterations	Accuracy
1	70.76%
10	32.85%

100	70.76%
1000	73.65%
10000	71.12%
100000	74.37%
1000000	74.37%

Based on the result, we can clearly observe that the accuracy decreases when the number of iterations is 10 and 10000. This could be due to model overfitting. Overall, we can still observe an increasing trend.

Decision Tree

Based on the Pyspark machine learning library implementation, we received the following results (with Gini impurity):

Max Depth	Accuracy
5	71.28%
10	64.89%
15	63.96%
20	60.72%

Based on the result, we can clearly observe a decreasing trend and this decrease in accuracy could be due to model overfitting.

Overall result

Algorithm	Accuracy
SVM	74.37%
Decision tree	71.28%
Naive Bayes	77.02%
Random forest	81.39%

SVM is a relatively simple machine learning algorithm, and the classifier also offers good accuracy. Under the current dataset, SVM and Naive Bayes have similar results.

The reason why decision trees have such poor performance is due to overfitting (We can observe that during our experiment: when maxDepth increases, accuracy decreases).

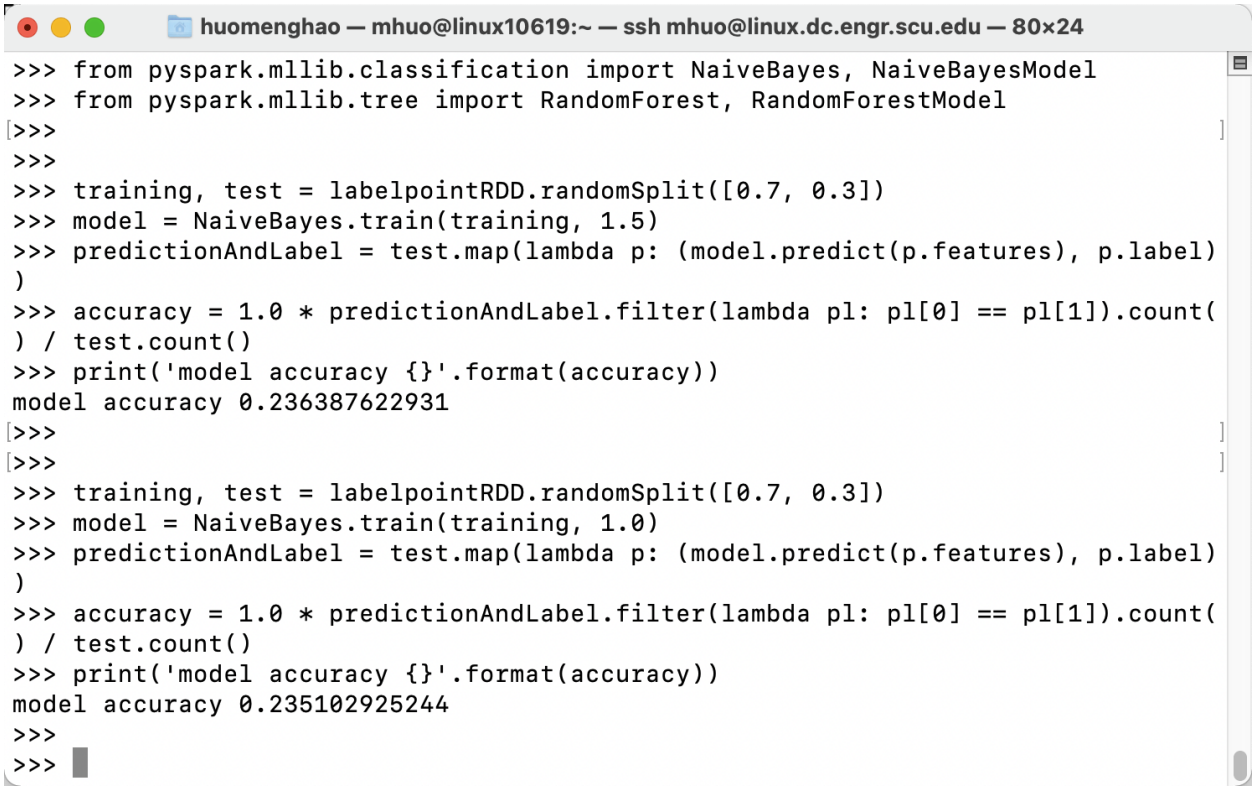
Random forests have higher accuracy and this is probably because random forest is a collection of decision trees(This result less prone to overfitting and a much better result).

Due to limited dataset size, it is hard to improve the accuracy of machine learning algorithms.This is why we also perform the similar steps on a larger dataset.

DataSet II:

Due to Linear Support Vector Machines only works with binary classification, it is impossible to perform this machine learning algorithm on this dataset.

Naive bayes



Because the dataset II has more types of target features, the naive bayes have not achieved good accuracy.

Random Forest

When maximum depth is 20:

numTrees	accuracy
10	76.13%
20	78.38%
30	78.97%
40	80.73%
50	79.85%

When number of trees is 40:

maxdepth	accuracy
4	31.68%
10	60.08%
20	80.73%
30	79.41%

Combined with results of data set I and data set II, we can conclude that when max depth is 20 the model will achieve the best accuracy. When the number of trees is between 40 and 50 the model can achieve similar results.

Decision Tree (with Gini impurity)

Max Depth	Accuracy
5	34.26%
10	57.89%
15	77.51%
20	82.10%

We can clearly notice that we have a much better accuracy this time.

Overall Result

Algorithm	Accuracy
Decision tree	82.19%

Naive Bayes	23.51%
Random Forest	80.73%

This time the decision tree and random forest have similar results. This is probably due to a small number of features (6 this time) and we also have a simpler input data format. Therefore, we have a stable decision tree (this cause random forest is just the same as the prediction of each single tree). The reason behind Naive Bayes poor performance is due to training data not independent (data is the location of three chess pieces on a chessboard and the result is optimal depth of win).

Conclusion

In this project, we have compared the efficiency and accuracy of different classification models to predict the state of breast cancer and depth of win for White through King and Rook locations. Through two data set testing, we get the conclusion that the Decision Tree and Random Forest have better accuracy(about 80%) to do prediction and classification than Naive Bayes and SVM (for our two datasets).