Lab - 4

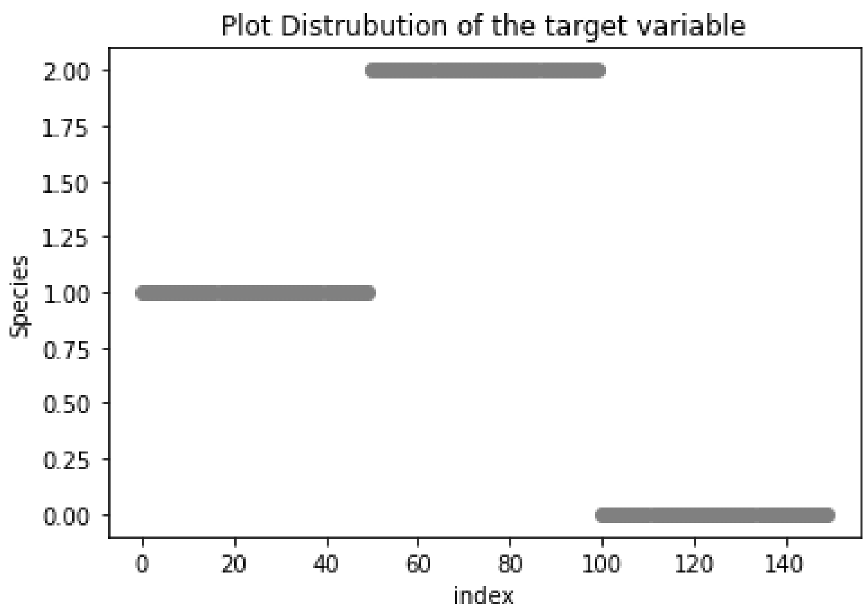
PRML
AY 2020-21 Trimester - III

March 30, 2021

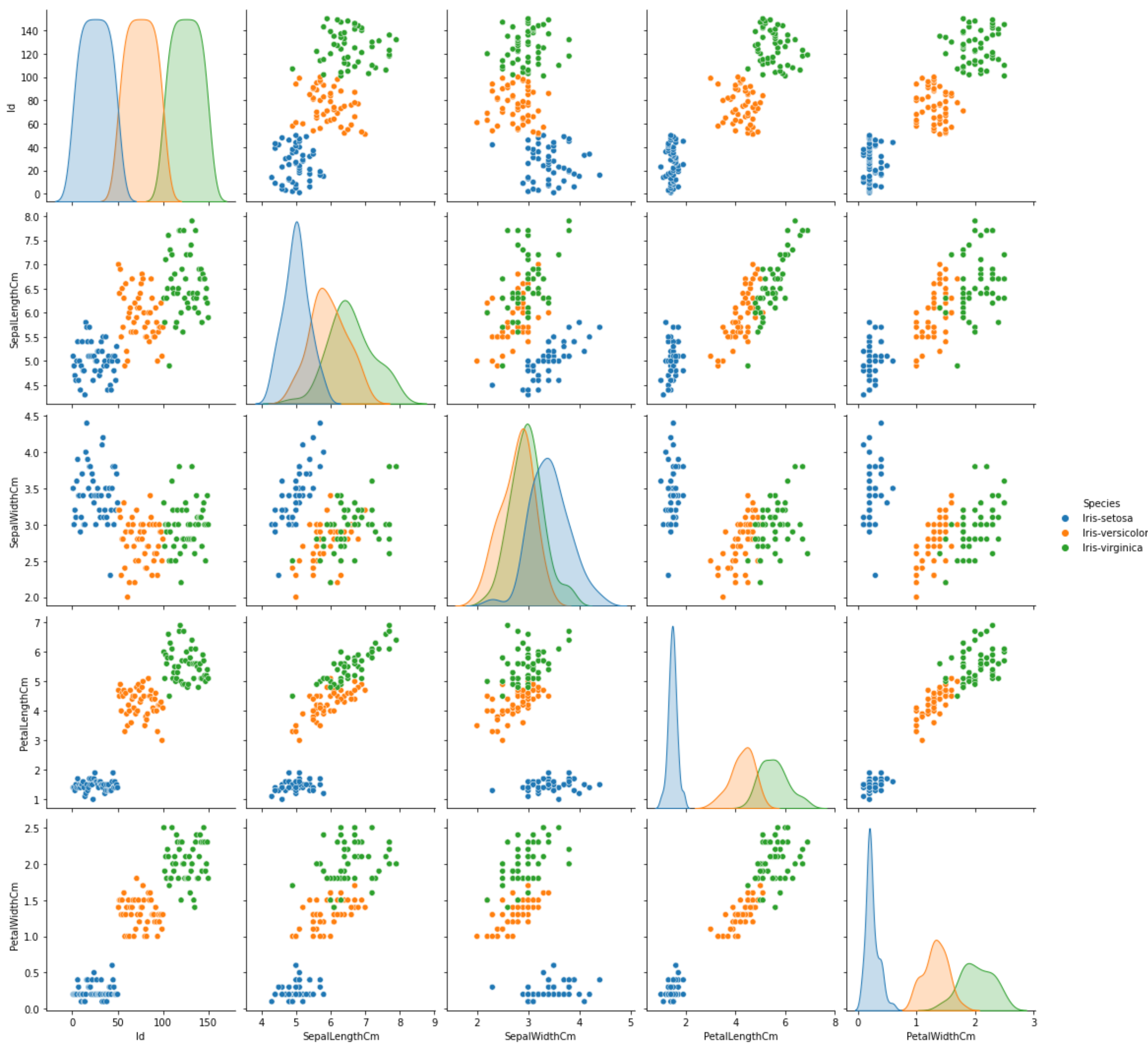
Boosting and Bayes Classification J Jahnavi(B19CSE109)

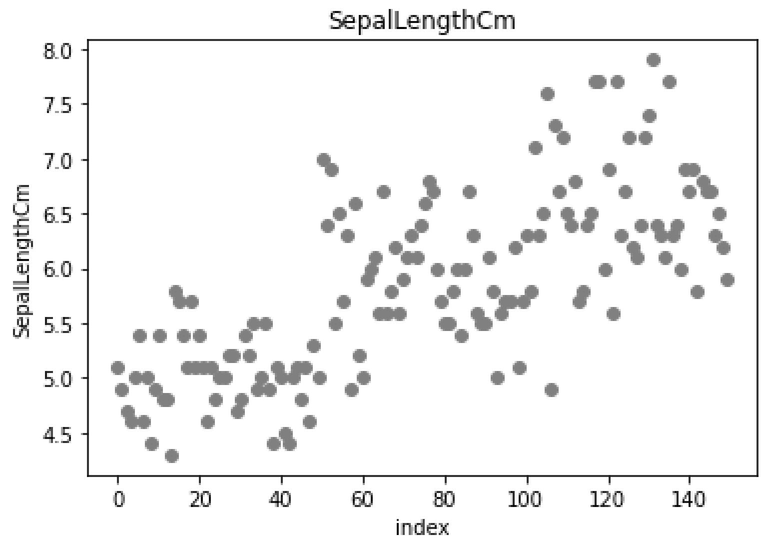
Boosting

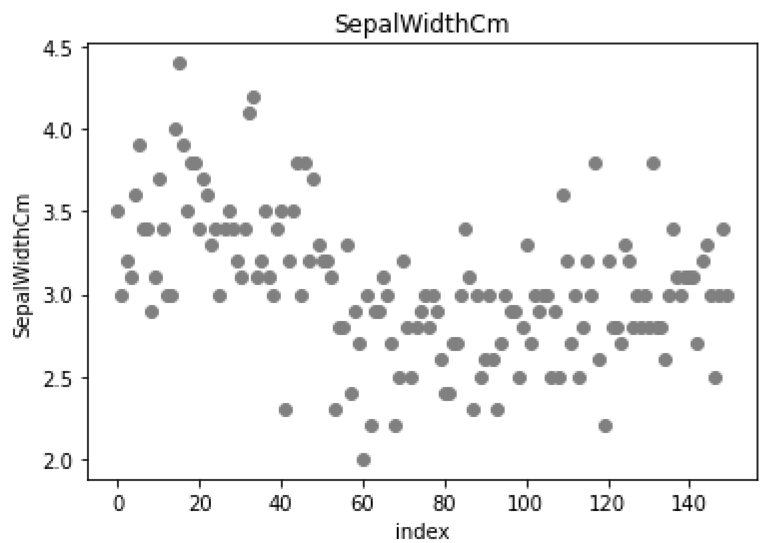
1. Preprocessing the data. (5 Marks)
    a. Plot the distribution of the target variable

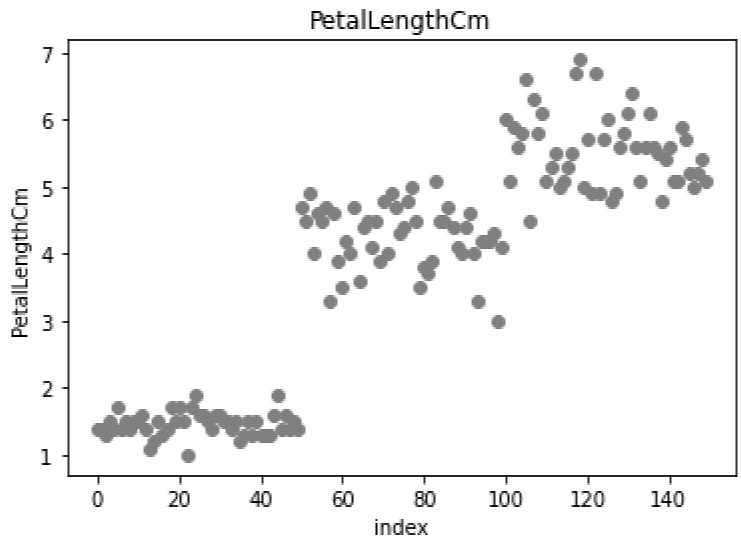


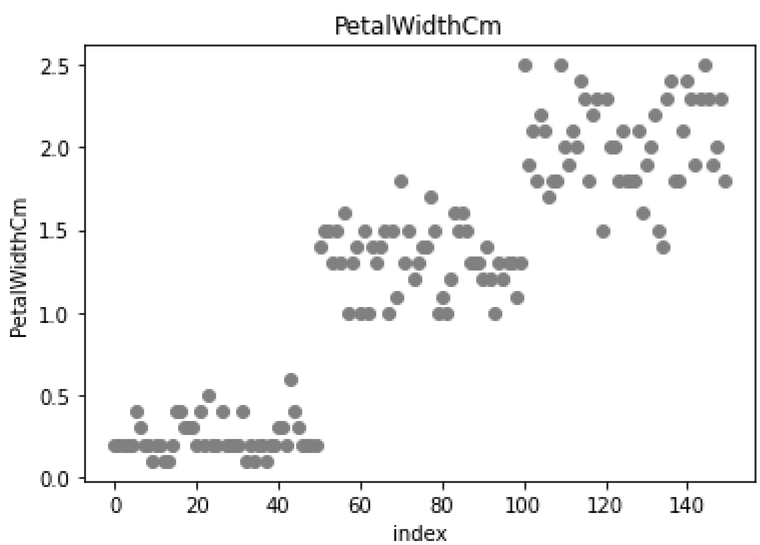
b.Visualize the distribution of data for every feature




SepalLengthCm
SepalLengthCm
index
8.0
7.5
7.0
6.5
6.0
5.5
5.0
4.5
0
20
40
60
80
100
120
140


SepalWidthCm
SepalWidthCm
index
4.5
4.0
3.5
3.0
2.5
2.0
0
20
40
60
80
100
120
140





2.Performed boosting-based classification using Decision Tree as the base classifier

The score for this weak learner is as follows:

```
0.9333333333333333
```

3.Performed cross-validation over the data and calculated accuracy for a weak learner which was selected to be a Decision Tree classifier: a weak learner with max_depth=2

The following are the cross-validation scores:

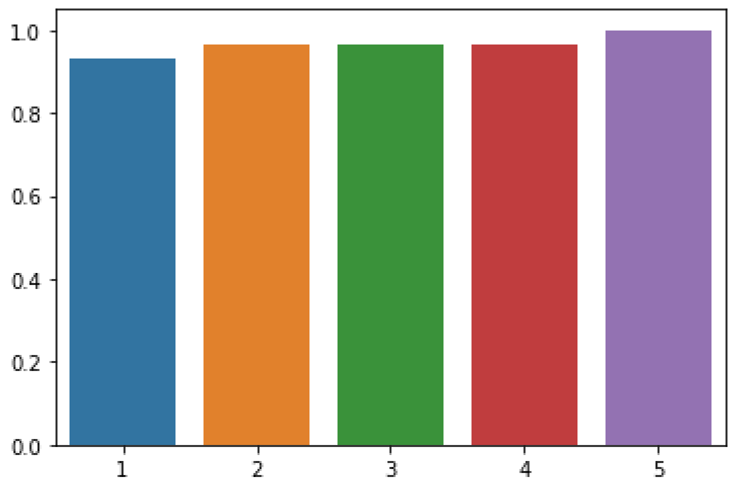
```
[0.95833333, 1. , 0.91666667, 0.95833333, 1. ]
```

4.Built the AdaBoost models using the weak learner by increasing the number of trees from 1 to 5 with a step of 1.

Comparing the scores for the 5 Adaboost models: On x-axis - no. of trees used and

On y-axis - following scores:

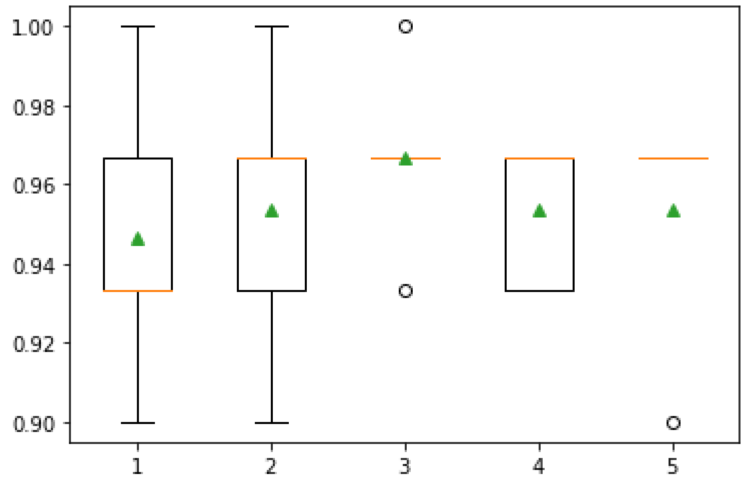
```
[0.9333333333333333, 0.9666666666666667, 0.9666666666666667, 0.9666666666666667, 1.0]
```



Model Performances:

No. of Trees	Mean	Standard Deviation
1	0.9466666666666667	0.03399346342395189
2	0.9533333333333334	0.03399346342395189
3	0.9666666666666668	0.02108185106778919
4	0.9533333333333334	0.016329931618554516
5	0.9533333333333334	0.02666666666666666

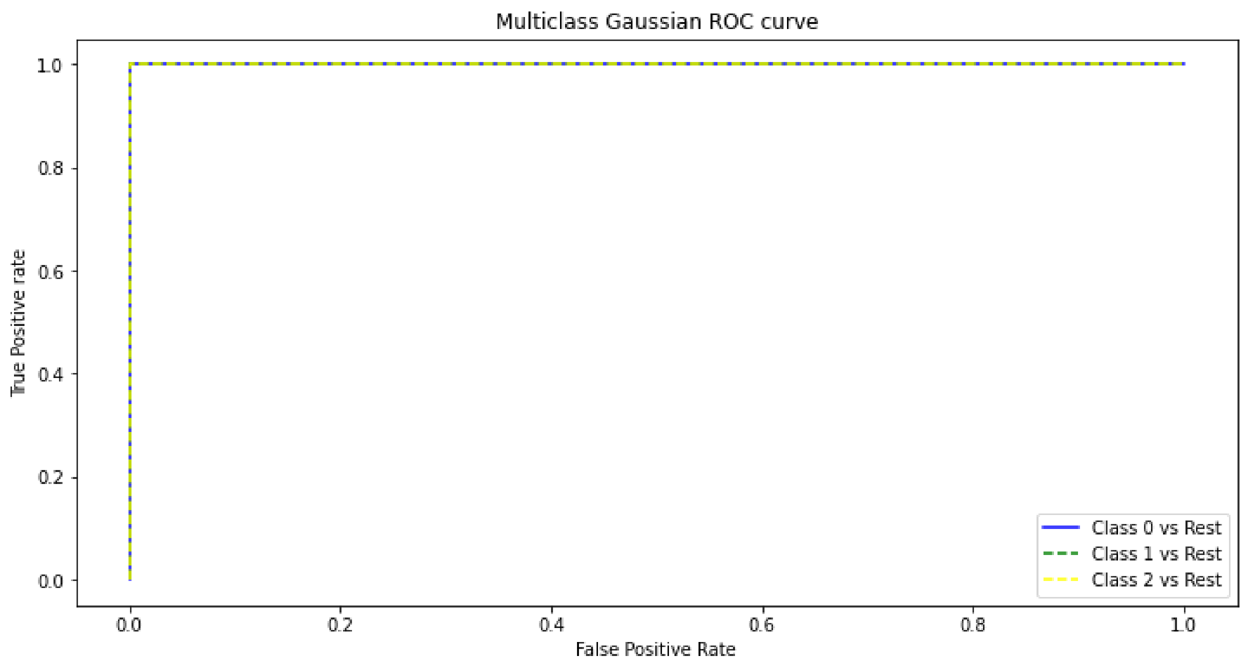
Corresponding Box Plots:



Bayes classification

1. Estimate the accuracy of the Naive Bayes algorithm using 5-fold cross-validation on the data set

Type of Bayes Classification	Accuracy	Standard Deviation	Variance
Gaussian Naive Bayes	0.9533333333333334	0.02666666666666666	0.02666666666666666
Multinomial Naive Bayes	0.9533333333333334	0.04521553322083511	0.0020444444444444433
Complement Naive Bayes	0.6666666666666666	0.0	0.0
Bernoulli Naive Bayes	0.3333333333333333	0.0	0.0



2.The accuracy using logarithmic discriminant function is: `0.9666666666666667`

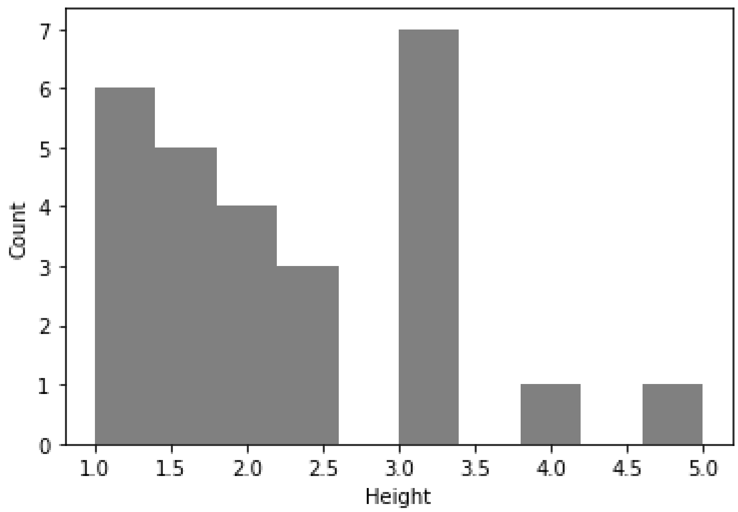
3.The Bayes risks:

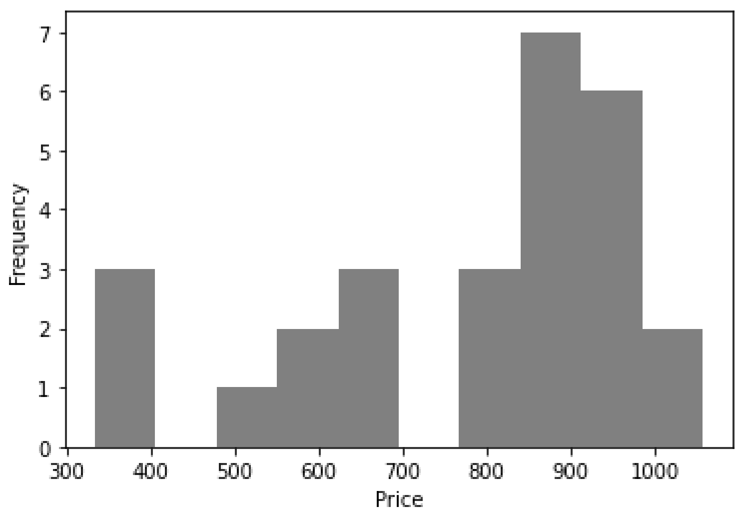
Model	Bayes Risk
Gaussian	[56.60614017 146.60614017 236.60614017]
Multinomial	[61.98350245 151.98350245 241.98350245]
Categorical	[58.49281574 148.49281574 238.49281574]
Bernoulli	[61.36631644 151.36631644 241.36631644]
Linear Discriminant	[57.36338681 147.36338681 237.36338681]

Visualization in Bayesian Decision Theory

DATASET 1

a. Created the labels from the given data.
b. Plotted the distribution of samples using the histogram.





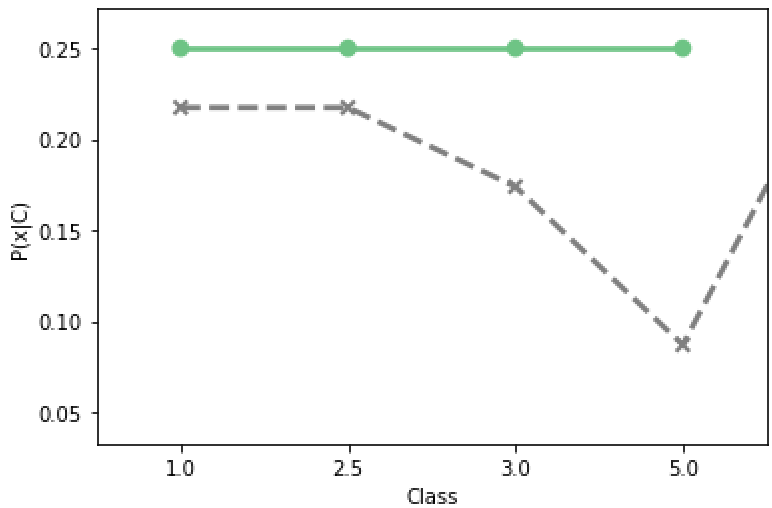
c. Determined the prior probability for both classes.

Class	Prior Probability
1	0.8518518518518519
2	0.14814814814814814

d. Determine the likelihood/class conditional probabilities for the classes. (Hint: Discretize the car heights into bins, you can use normalized histograms)

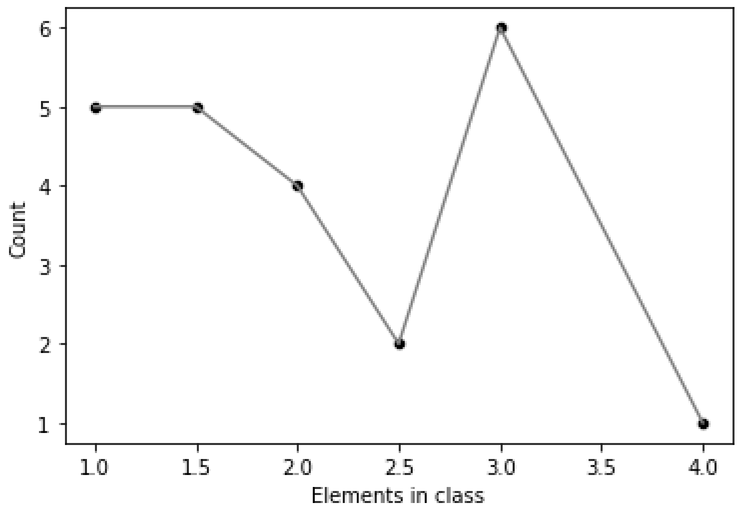
In green: Class Price

In grey: Class Height

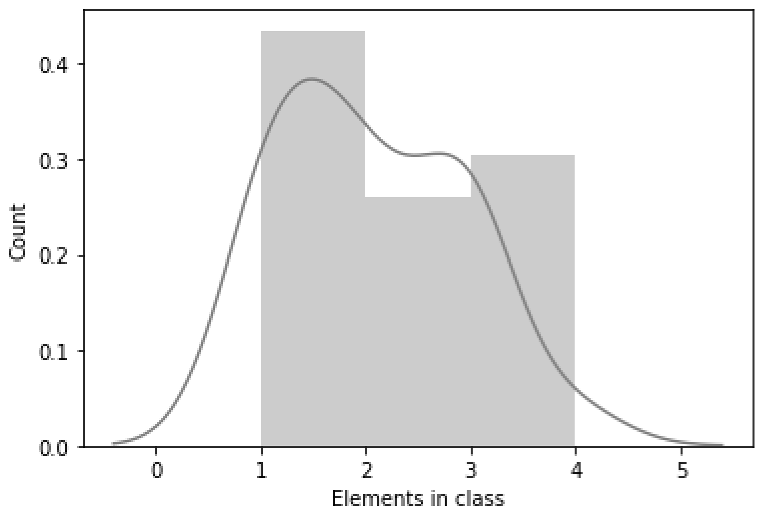


e. Plotted the count of each unique element for each class followed by the probability distribution

Class: Height

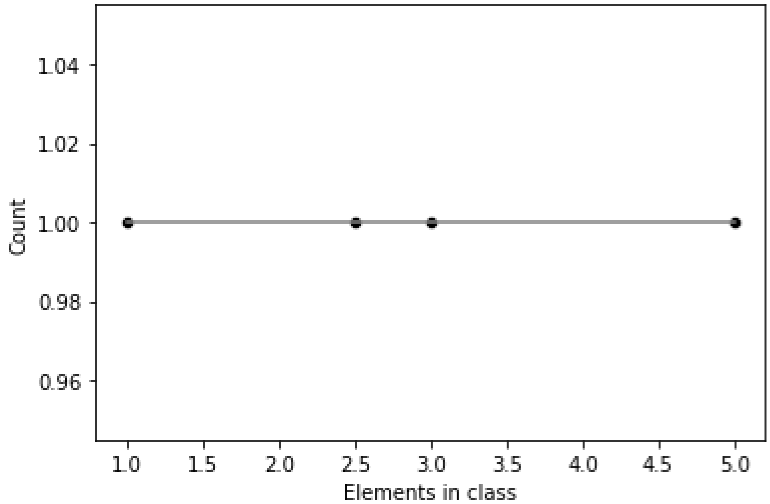


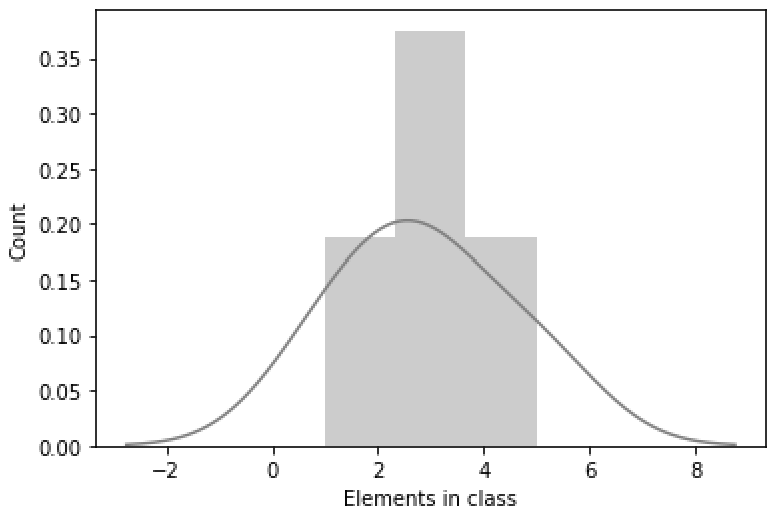
Count
Elements in class
1.0
1.5
2.0
2.5
3.0
3.5
4.0
1
2
3
4
5
6



Count
Elements in class
0
1
2
3
4
5
0.0
0.1
0.2
0.3
0.4

Class: Price

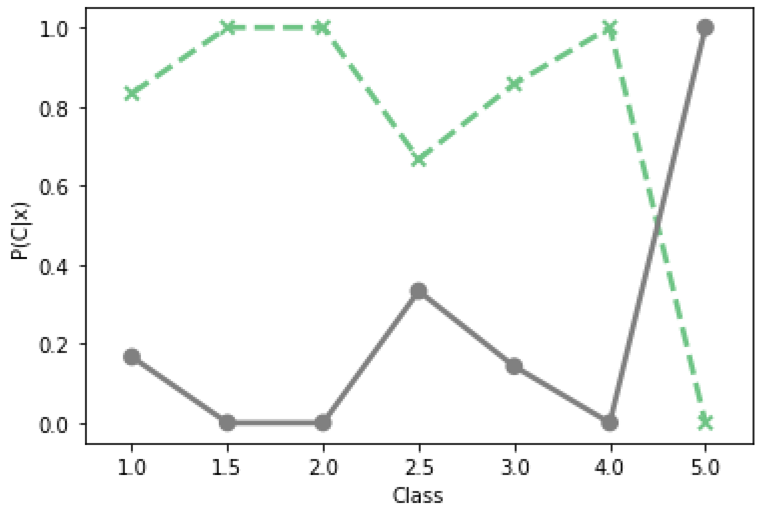




f. Calculated the P(C1|x) and P(C2|x) i.e posterior probabilities and plot them in a single graph.

In green: Class Height

In grey: Class Price

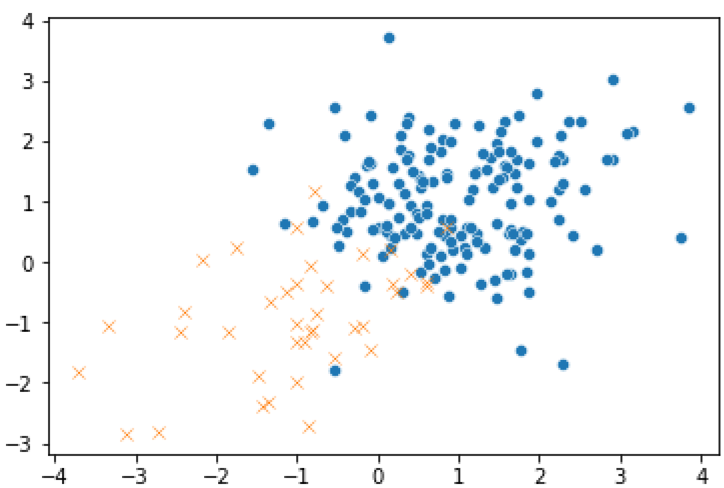


DATASET 2

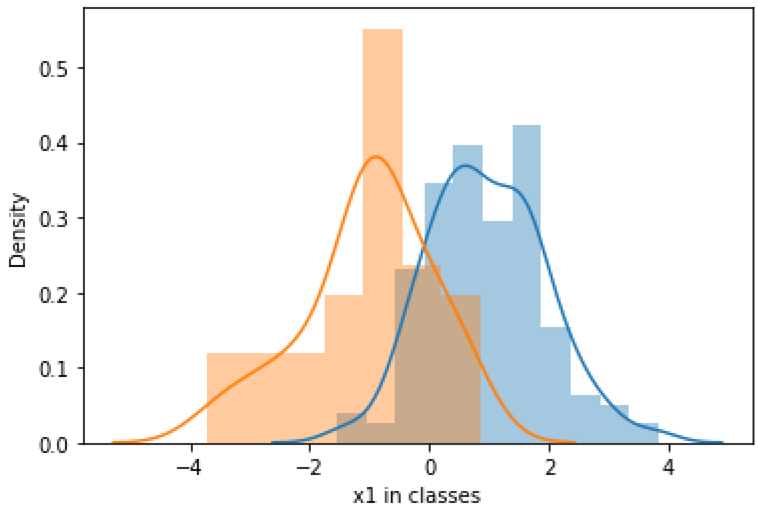
1.Plotting the distribution of samples

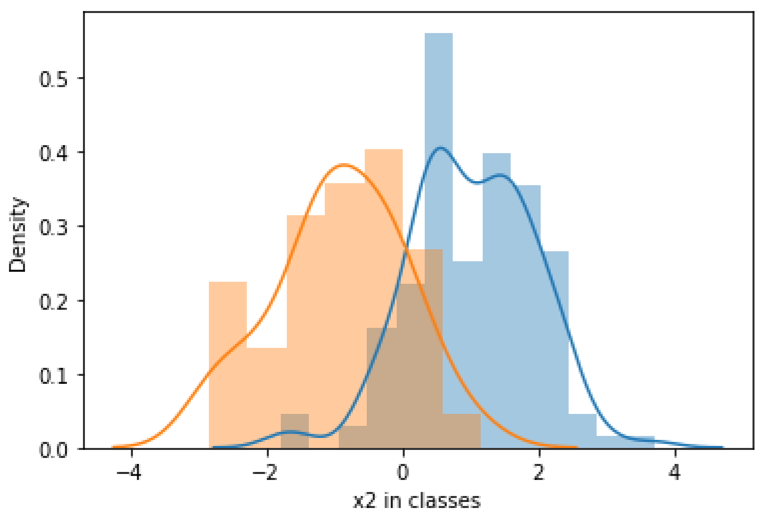
In Blue: Class 1

In Orange: Class 2

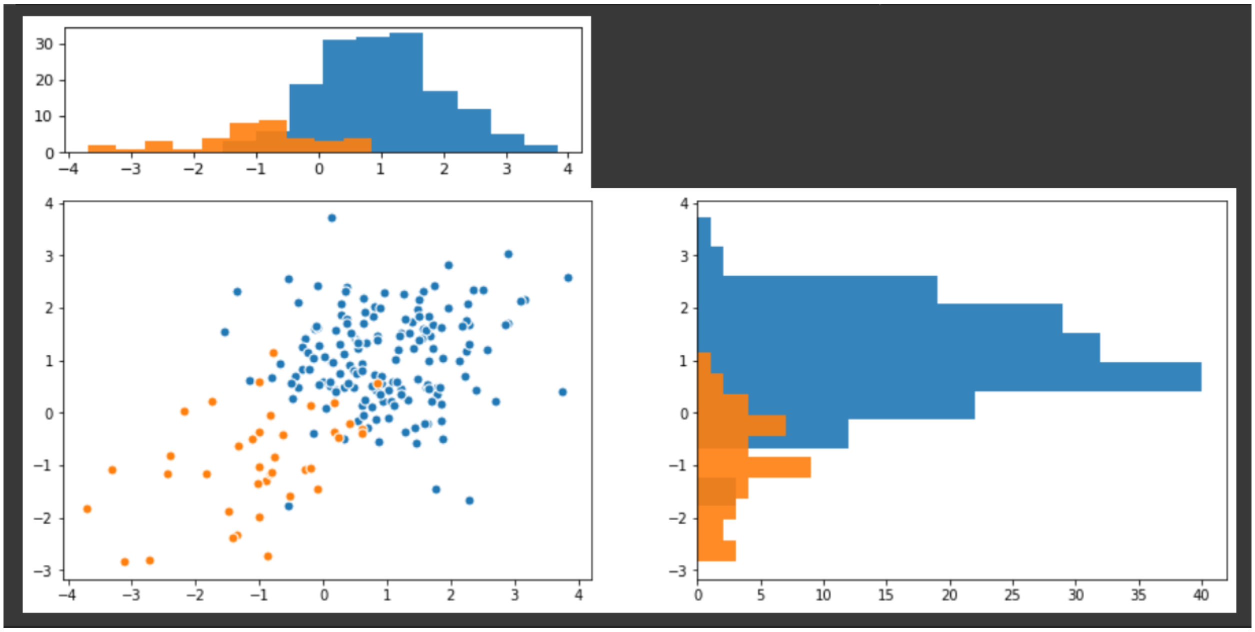


2. Density vs x


Density
x1 in classes


Density
x2 in classes

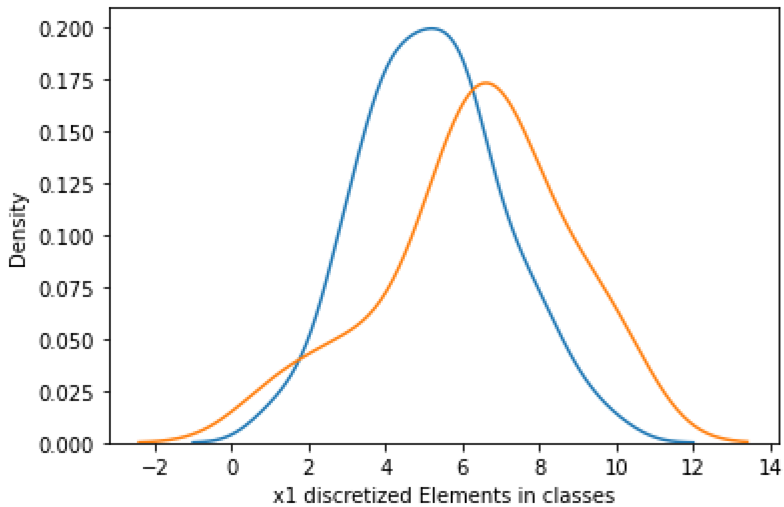
3. Plot the data distribution and the histogram of feature 1 and feature 2 in the X-axis and Y-axis respectively. The distribution of feature 1 will be along the top of the X-axis and feature 2 along the right of the Y-axis:

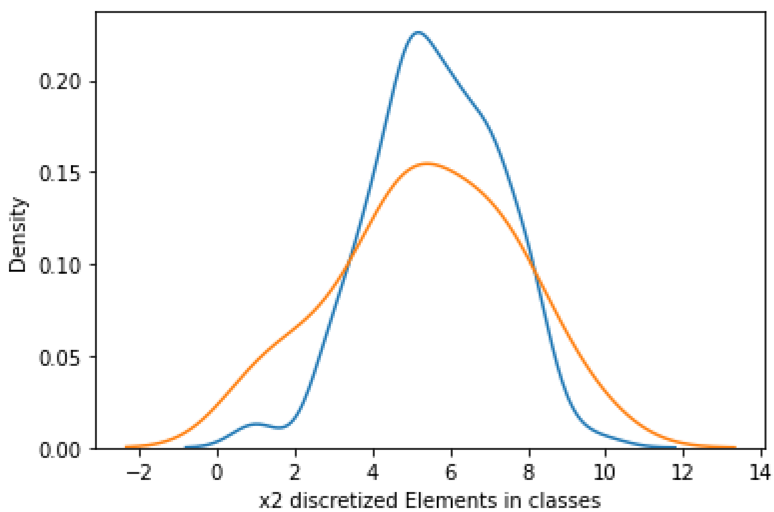


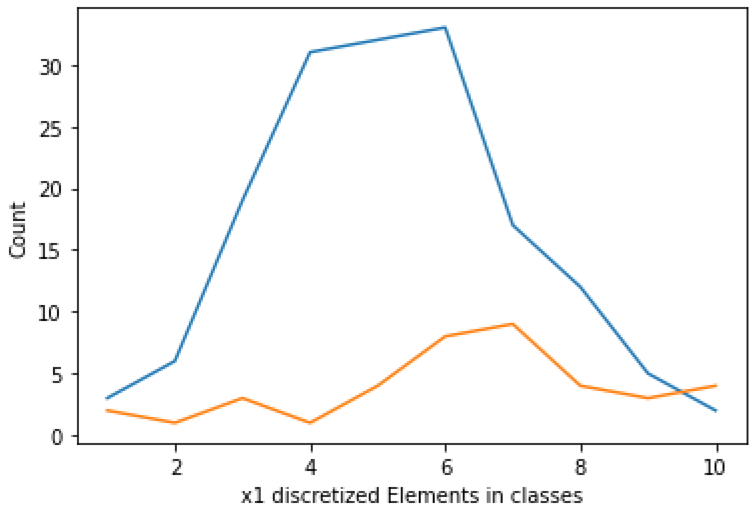
4. Discretized elements in classes

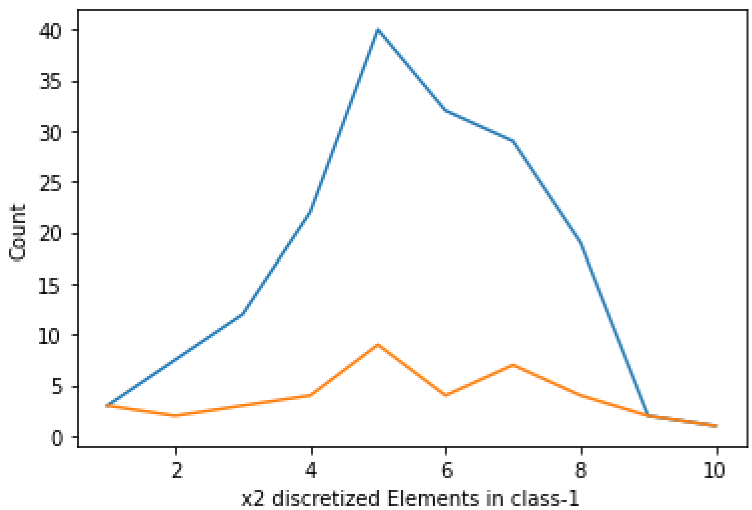
```
sns.distplot(x1_c1,hist=False)
sns.distplot(x1_c2,hist=False)
plt.xlabel("x1 discretized Elements in classes")
plt.show()
print('\n')
sns.distplot(x2_c1,hist=False)
sns.distplot(x2_c2,hist=False)
plt.xlabel("x2 discretized Elements in classes")
plt.show()
print('\n')
sns.lineplot(list(unique1_c1),count_unique1_c1)
sns.lineplot(list(unique1_c2),count_unique1_c2)
plt.xlabel("x1 discretized Elements in classes")
plt.ylabel("Count")
plt.show()
print('\n')
sns.lineplot(list(unique2_c1),count_unique2_c1)
sns.lineplot(list(unique2_c2),count_unique2_c2)
plt.xlabel("x2 discretized Elements in class-1")
plt.ylabel("Count")
plt.show()
```

Resulting plots:


Density
x1 discretized Elements in classes

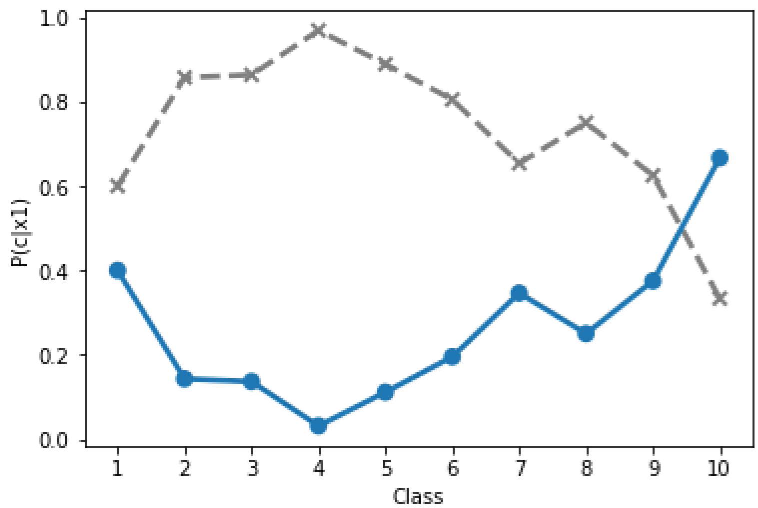

Density
x2 discretized Elements in classes


Count
0
5
10
15
20
25
30
2
4
6
8
10
x1 discretized Elements in classes

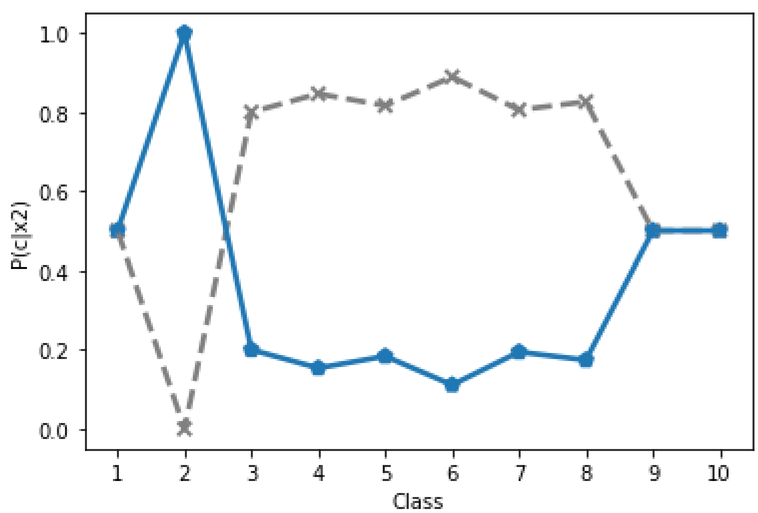

Count
0
5
10
15
20
25
30
35
40
2
4
6
8
10
x2 discretized Elements in class-1

5. Plotting posterior probabilities

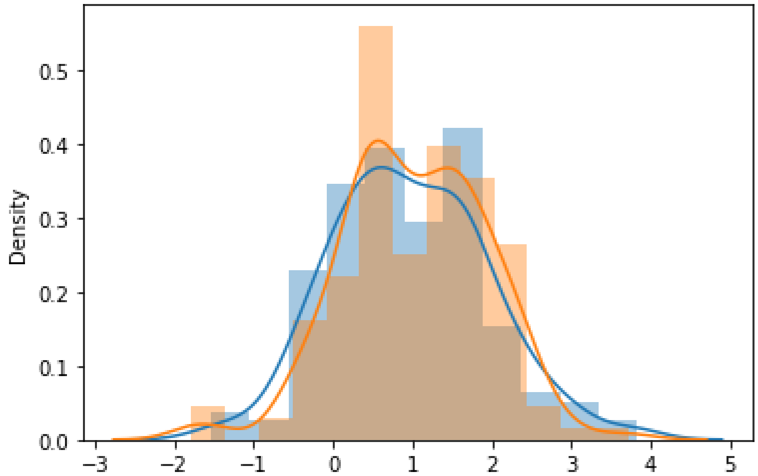
For x1:



For x2:



6. Histogram of C1



7. Histogram of C2