HACETTEPE UNIVERSITY

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

ELE489-FUNDAMENTALS OF MACHINE LEARNING

HOMEWORK II

BURAK ÇAYIRLI 2200357009

Spring 2024-2025

Contents

1 QUESTION 1 **5**

1.1 General Outline 5

1.2 Calculations of Gini Indexes 5

1.2.1 i)Gini Index for Entire Dataset 5

1.2.2 ii)Weighted Gini Index for Splitting on Weather 6

1.2.3 iii)Weighted Gini Index for Splitting on Wind 7

1.2.4 iv)Lowest Weighted Gini Index 7

2 QUESTION 2 **8**

2.1 1) Image's Variance, Skewness, Kurtosis and Entropy 8

2.1.1 Image's Variance 8

2.1.2 Skewness 8

2.1.3 Kurtosis 8

2.1.4 Entropy 9

2.2 2) 9

2.2.1 Visualization of Features 9

2.2.2 Is Decision Tree Good Algorithm for this Features ? . . 10

2.3 3) 10

2.3.1 Splitting the Data and Working on the Algorithm . . . 10

2.3.2 Classification Report 11

2.3.3 Confusion Matrix 12

2.4 4)Visualization of the Tree 13

2.5 5)Feature Importance 14

2.6 6)Result and Conclusion 14

3 My Github Link **15**

List of Figures

1 Dataset for Playing Outside 5

2 Feature Pairs Plot 9

3 Confusion Matrix 12

4 Decision Tree 13

5 Feature Importance Plot 14

1 QUESTION 1

1.1 General Outline

In this homework, a small dataset is available where we predict whether a person will play outside based on the weather and the wind conditions. So the Gini index calculations are done by hand. The data set is in the following.

ID	Weather	Wind	Play Outside?
1	Sunny	Weak	No
2	Sunny	Strong	No
3	Overcast	Weak	Yes
4	Rainy	Weak	Yes
5	Rainy	Strong	No
6	Overcast	Strong	Yes

Figure 1: Dataset for Playing Outside

1.2 Calculations of Gini Indexes

1.2.1 i)Gini Index for Entire Dataset

Gini Index for Entire Dataset

3 instance Yes

3 instance No

$$Gini(Dataset) = 1 - P^2(yes) - P^2(No) = 1 - \left(\frac{3}{6}\right)^2 - \left(\frac{3}{6}\right)^2 = \frac{1}{2}$$

1.2.2 ii)Weighted Gini Index for Splitting on Weather

Weighted Gini Index for Splitting on Weather

	Sunny	Overcast	Rainy
Yes	0	2	1
No	2	0	1
Total	2	2	2

$$\text{Gini (Sunny)} = 1 - \left(\frac{0}{2}\right)^2 - \left(\frac{2}{2}\right) = 0$$

$$\text{Gini (Overcast)} = 1 - \left(\frac{2}{2}\right)^2 - \left(\frac{0}{2}\right)^2 = 0$$

$$\text{Gini (Rainy)} = 1 - \left(\frac{1}{2}\right)^2 - \left(\frac{1}{2}\right)^2 = \frac{1}{2}$$

$$\text{Gini (Weather)} = 0 \cdot \frac{2}{6} + 0 \cdot \frac{2}{6} + \frac{1}{2} \cdot \frac{2}{6} = \frac{1}{6} = 0,167$$

1.2.3 iii)Weighted Gini Index for Splitting on Wind

Weighted Gini Index for Splitting on Wind

	Strong	Weak
Yes	1	2
No	2	1
Total	3	3

$$\text{Gini (Strong)} = 1 - \left(\frac{1}{3}\right)^2 - \left(\frac{2}{3}\right)^2 = 0{,}444$$

$$\text{Gini (Weak)} = 1 - \left(\frac{2}{3}\right)^2 - \left(\frac{1}{3}\right)^2 = 0{,}444$$

$$\text{Gini (Wind)} = 0{,}444 \cdot \frac{3}{6} + 0{,}444 \cdot \frac{3}{6} = 0{,}444$$

1.2.4 iv)Lowest Weighted Gini Index

As can be seen from the calculations, the root node is weather due to the lowest Gini index.

2 QUESTION 2

2.1 1) Image's Variance, Skewness, Kurtosis and Entropy

2.1.1 Image's Variance

Variance tells us how different the pixel values are from the average value.

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2$$

We can compute the variance with this formula

2.1.2 Skewness

Skewness shows if the pixel values are more on the dark side or the bright side.If skewness is positive then we can say more dark pixels or if skewness is negative more bright pixels.

$$\text{Skewness} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x_i - \mu}{\sigma}\right)^3$$

We can compute the skewness with this formula

2.1.3 Kurtosis

Kurtosis gives information about how sharp or flat the image's histogram. High kurtosis means image has sharp contrast and low kurtosis means image flatter and look more equal.

$$\text{Kurtosis} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x_i - \mu}{\sigma}\right)^4$$

We can compute the kurtosis with this formula

2.1.4 Entropy

Entropy shows how much information or detail is in the image.

$$H = -\sum_{i=0}^{c-1} p_i \log_2(p_i)$$

We can compute the entropy with this formula

2.2 2)

2.2.1 Visualization of Features

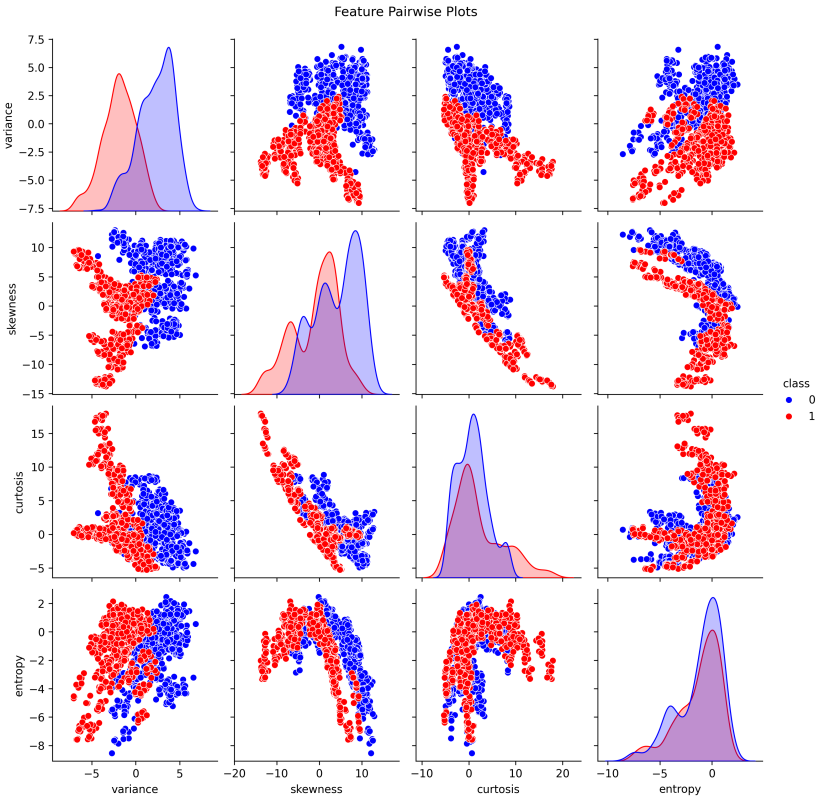

Figure 2: Feature Pairs Plot

2.2.2 Is Decision Tree Good Algorithm for this Features ?

There is an evident differences between class 1 and class 0. Especially for skewness and variance pairs is visible how distinct the classes are from the Figure 2. So I can say that decision tree algorithm is good for these features.

2.3 3)

2.3.1 Splitting the Data and Working on the Algorithm

I split the data using `train_test_split()` from `sklearn.model_selection`.

```
#Features and Class
X=DataFrame.drop('class',axis=1)
Y=DataFrame['class']
```

```
#Splitting the Data
X_train ,X_test,Y_train,Y_test= train_test_split(X,Y,test_size=0.2,random_state=0)
```

To avoid using loops, I used 'GridSearchCV' to find the best parameters automatically with a small piece of code. 'GridSearchCV' tried the all combinations that i gave and found the best parameters.

```
GridSearchCV_parameters = {
    'max_depth': [2,3, 4, 5,6,7],
    'min_samples_split': [2, 5, 10, 15,20,25,50],
    'criterion': ['gini', 'entropy']
}

model = DecisionTreeClassifier(random_state=42)

Grid = GridSearchCV(
    estimator=model,
    param_grid=GridSearchCV_parameters,
    cv=5,
    scoring='accuracy',
    n_jobs=-1,
    verbose=1
)

Grid.fit(X_train, Y_train)
```

2.3.2 Classification Report

Classification Report for the best parameters.

```
Best Parameters: {'criterion': 'entropy', 'max_depth': 7, 'min_samples_split': 2}

Classification Report:
              precision    recall  f1-score   support

           0       0.99      0.98      0.99       157
           1       0.97      0.99      0.98       118

    accuracy                           0.99       275
   macro avg       0.98      0.99      0.99       275
weighted avg       0.99      0.99      0.99       275
```

2.3.3 Confusion Matrix

Confusion Matrix for the best parameters.

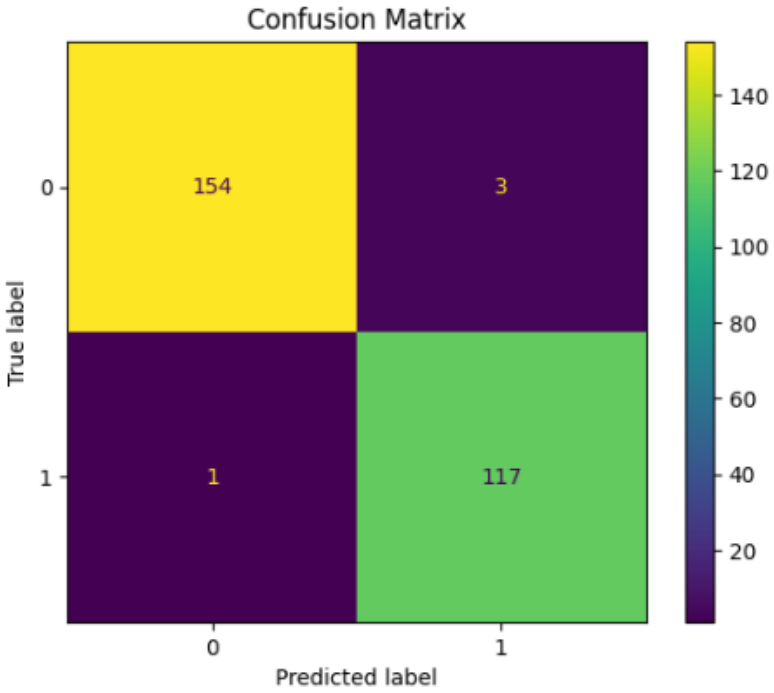

Figure 3: Confusion Matrix

2.4 4)Visualization of the Tree

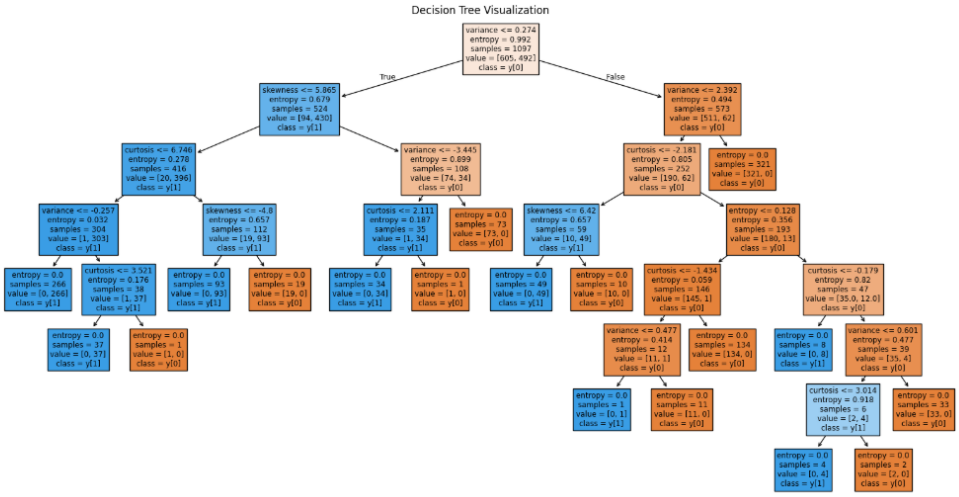

Figure 4: Decision Tree

Making the decision tree deeper helps the model learn more about the data. But, it also makes the tree harder to understand because there are more steps and rules. A tree that is not very deep is easier for people to read and explain.

2.5 5)Feature Importance

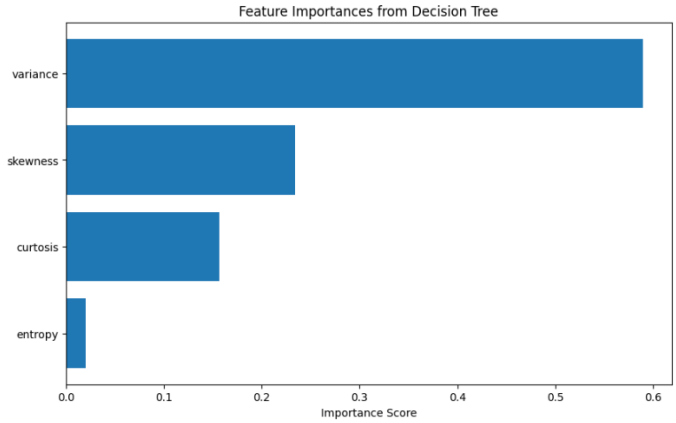

Figure 5: Feature Importance Plot

2.6 6)Result and Conclusion

From this question,I have learned how to train and tune a decision tree by using different parameters like `max_depth`,`min_samples_split` and `criteron`. I observed how different parameters affect the model's performance. I also learned 'GridSearchCV' to find the best combinations of the parameters. By using the Confusion matrix and the classification report, I reached out the accuracy and effectiveness of the model. So I still think decision tree is good model for this dataset. Because it is able to classify the samples accurately. It also showed the feature importances.

3 My Github Link

You can find my Github repository link below.

`https://github.com/burakcayirli/ELE-489_Homework_2`