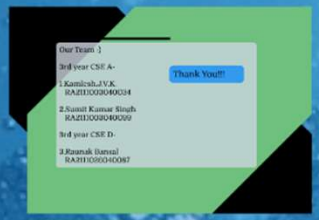
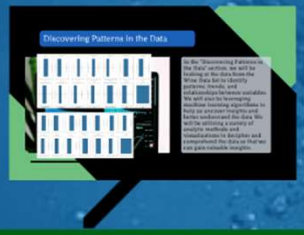
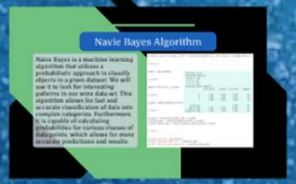
Industry Seminar 2023-2024

Exploring and Interpretating with Wine Data Sets







Prezi

Introducing the Wine Data Set

```
import pandas as pd
import numpy as np
import scipy as sp
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
```

```
df = pd.read_csv("wine.data",names=["Class","Alcohol","Malic acid","Ash","Alcalinity of ash","Magnesium","Total phenols","Flavanoids","Nonflavanoid phenols","Proanthocyanins","Color intensity","Hue","OD280/OD315 of diluted wines","Proline"])
```

The Wine Data Set comprises records of different types of wine produced at different wineries in different countries around the world. The data set used covers features such as various chemical composition of the wine, including pH levels, alcohol content, sugar content, as well as others. Furthermore, the data set also includes what type of wine the bottles are and the brand of the wine. With such an immense and comprehensive data set, it is possible to make insightful interpretations of the different kinds of wine available in the market.

Exploratory Wine Data Analysis

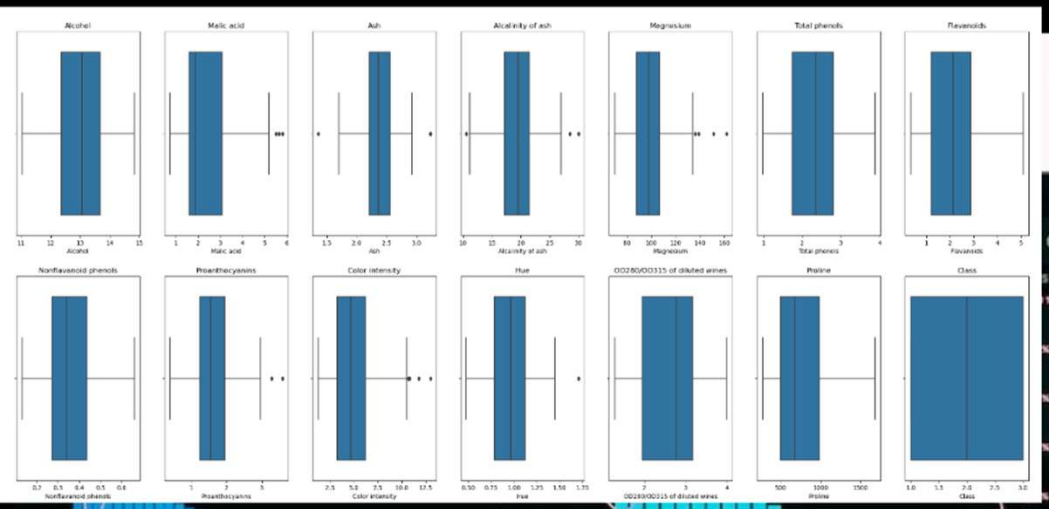
Exploratory Wine Data Analysis is a powerful tool for examining and understanding the underlying structure and patterns of a wine dataset. It provides the ability to identify important features, create visualizations for discovering trends, make inferences, and determine correlations. With this, we can get a glimpse into the data and understand how it is composed and what it can tell us.

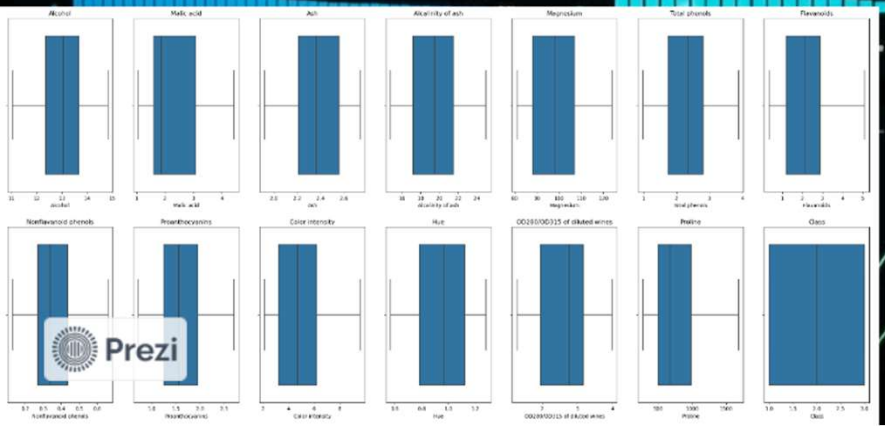
`df.head()` `df.shape` `df.info()` `df.isnull().sum()` `df.describe()`

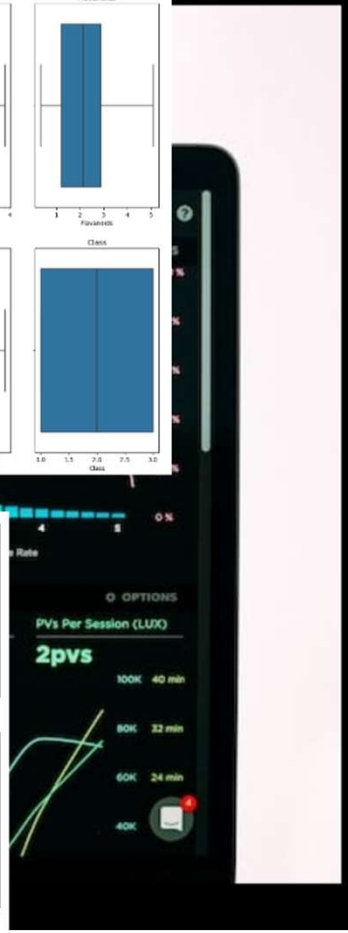
	Class	Alcohol	Malic acid	Ash	Alcalinity of ash	Magnesium	Total phenols	Flavanoids	Nonflavanoid phenols	Proanthocyanins	Color intensity
count	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000
mean	1.938202	13.000618	2.336348	2.366517	19.494944	99.741573	2.295112	2.029270	0.361854	1.590899	5.058090
std	0.775035	0.811827	1.117146	0.274344	3.339564	14.282484	0.625851	0.998859	0.124453	0.572359	2.318286
min	1.000000	11.030000	0.740000	1.360000	10.600000	70.000000	0.980000	0.340000	0.130000	0.410000	1.280000
25%	1.000000	12.362500	1.602500	2.210000	17.200000	88.000000	1.742500	1.205000	0.270000	1.250000	3.220000
50%	2.000000	13.050000	1.865000	2.360000	19.500000	98.000000	2.355000	2.135000	0.340000	1.555000	4.690000
75%	3.000000	13.677500	3.082500	2.557500	21.500000	107.000000	2.800000	2.875000	0.437500	1.950000	6.200000
max	3.000000	14.830000	5.800000	3.230000	30.000000	162.000000	3.880000	5.080000	0.660000	3.580000	13.000000

	Class	Alcohol	Malic acid	Ash	Alcalinity of ash	Magnesium	Total phenols	Flavanoids	Nonflavanoid phenols	Proanthocyanins	Color intensity	Hue	OD280/OD315 of diluted wines	Proline
0	1	14.23	1.71	2.43	15.6	127	2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065
1	1	13.20	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050
2	1	13.16	2.36	2.67	18.6	101	2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185
3	1	14.37	1.95	2.50	16.8	113	3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480
4	1	13.24	2.59	2.87	21.0	118	2.80	2.69	0.39	1.82	4.32	1.04	2.93	735

Discovering Patterns in the Data







In the "Discovering Patterns in the Data" section, we will be looking at the data from the Wine Data Set to identify patterns, trends, and relationships between variables. We will also be leveraging machine learning algorithms to help us uncover insights and better understand the data. We will be utilizing a variety of analytic methods and visualizations to decipher and comprehend the data so that we can gain valuable insights.

Navie Bayes Algorithm

Naive Bayes is a machine learning algorithm that utilizes a probabilistic approach to classify objects in a given dataset. We will use it to look for interesting patterns in our wine data set. This algorithm allows for fast and accurate classification of data into complex categories. Furthermore, it is capable of calculating probabilities for various classes of data points, which allows for more accurate predictions and results.

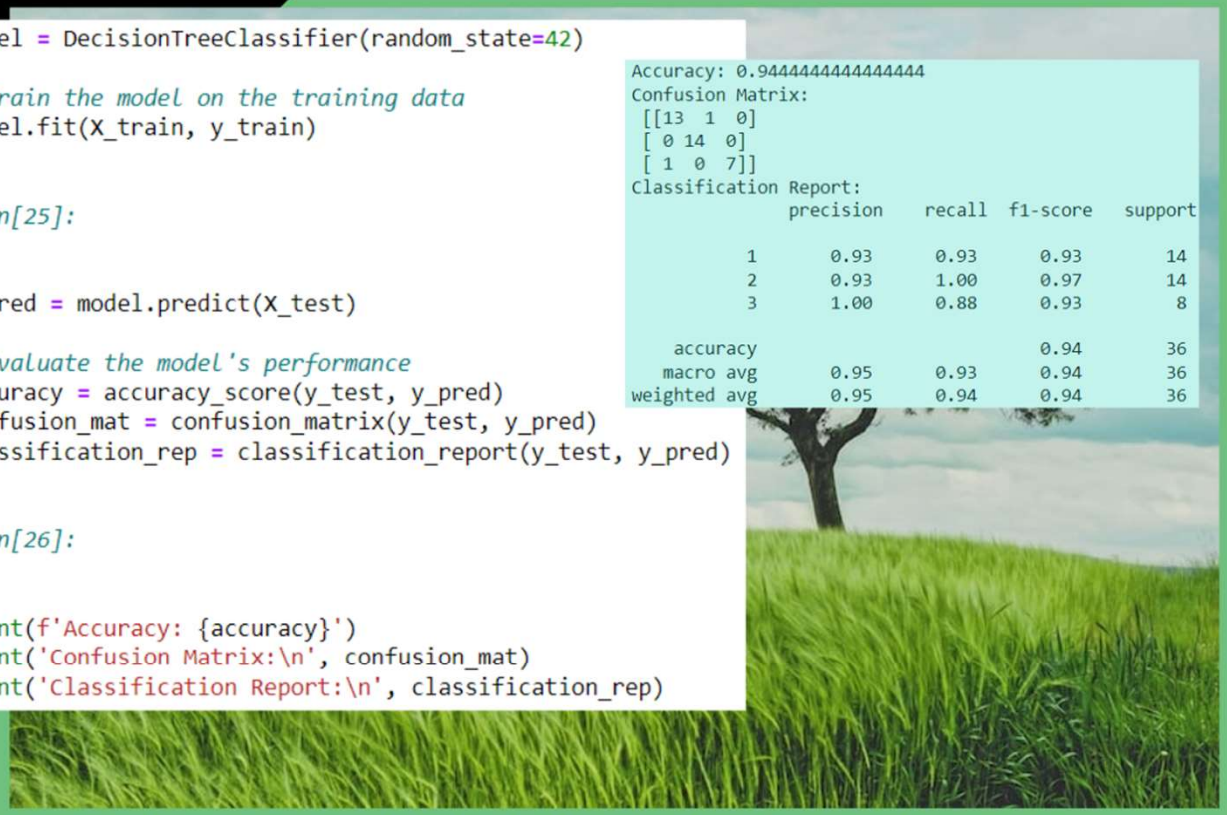
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# In[19]:

model = GaussianNB()

# In[20]:

model.fit(X_train, y_train)

# In[21]:

y_pred = model.predict(X_test)

# In[22]:

accuracy = accuracy_score(y_test, y_pred)
confusion_mat = confusion_matrix(y_test, y_pred)
classification_rep = classification_report(y_test, y_pred)

# In[23]:

print(f'Accuracy: {accuracy}')
print('Confusion Matrix:\n', confusion_mat)
print('Classification Report:\n', classification_rep)
```

```
Accuracy: 1.0
Confusion Matrix:
 [[14  0  0]
 [ 0 14  0]
 [ 0  0  8]]
Classification Report:
              precision    recall  f1-score   support

           1       1.00      1.00      1.00        14
           2       1.00      1.00      1.00        14
           3       1.00      1.00      1.00         8

    accuracy                           1.00        36
   macro avg       1.00      1.00      1.00        36
weighted avg       1.00      1.00      1.00        36
```

Decision Tree Algorithm

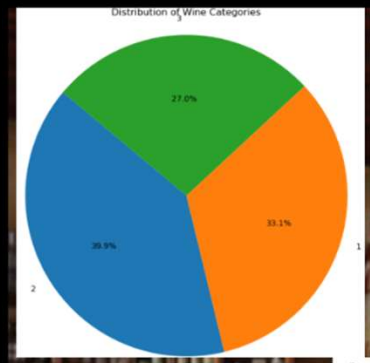
```
model = DecisionTreeClassifier(random_state=42)

# Train the model on the training data
model.fit(X_train, y_train)


# In[25]:


y_pred = model.predict(X_test)

# Evaluate the model's performance
accuracy = accuracy_score(y_test, y_pred)
confusion_mat = confusion_matrix(y_test, y_pred)
classification_rep = classification_report(y_test, y_pred)


# In[26]:


print(f'Accuracy: {accuracy}')
print('Confusion Matrix:\n', confusion_mat)
print('Classification Report:\n', classification_rep)
```

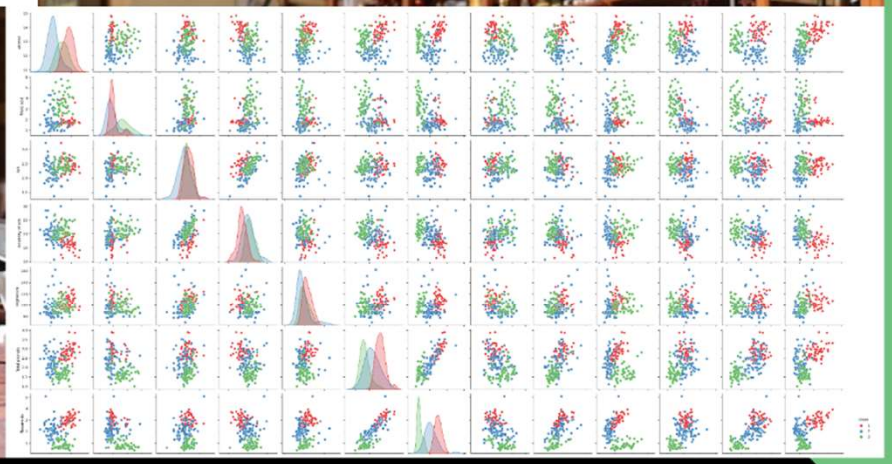
```
Accuracy: 0.9444444444444444
Confusion Matrix:
 [[13  1  0]
 [ 0 14  0]
 [ 1  0  7]]
Classification Report:
              precision    recall  f1-score   support

           1       0.93      0.93      0.93        14
           2       0.93      1.00      0.97        14
           3       1.00      0.88      0.93         8

    accuracy                           0.94        36
   macro avg       0.95      0.93      0.94        36
weighted avg       0.95      0.94      0.94        36
```



Decision trees are a powerful tool that allow for an extensive evaluation of the possible options that the data set is able to provide. It uses a graphical visualization of the possible options in a tree-like structure, which allows us to make better decisions that could help us maximize the output obtained from the data. We can use decision tree models to forecast the different possibilities of the data set and make more informed decisions.

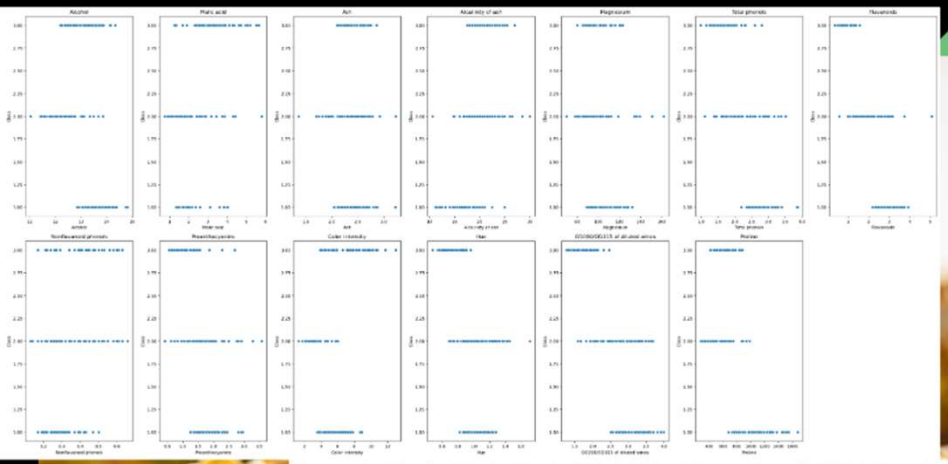
Conclusion and Future Implications

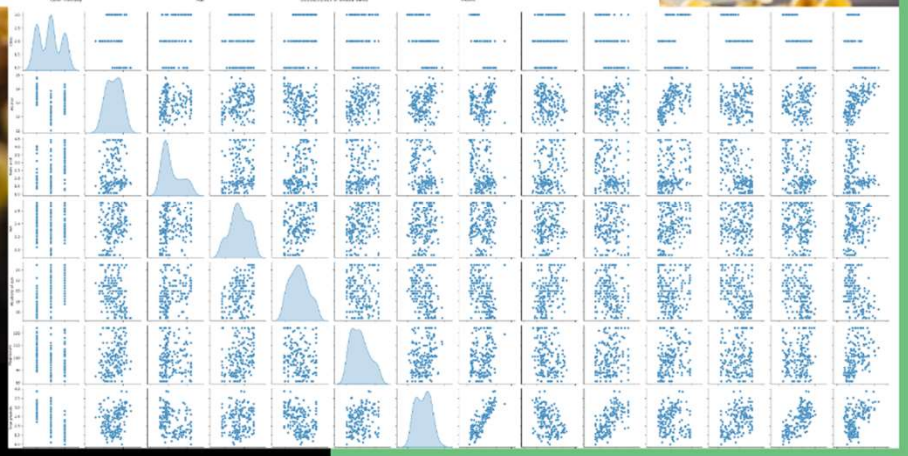




In conclusion, the analysis of the wine data set reveals potential opportunities for further exploration that will help advance our understanding of the relationships between certain factors. We have seen that the Navie Bayes Algorithm and Decision Tree techniques were the most useful in interpretation. As such, these results can be utilized as a guide in future research projects.

Utilizing Results for Further Research





The results of the exploration of interpreatation options with the Wine Data Set can be utilized for further research. For instance, other algorithms such as Linear Discriminant Analysis and Support Vector Machines can be employed to identify correlations between various elements of the data set. In addition, additional elements such as descriptive data of the wines can be included for data analysis. Visualizing the results of the analysis in various graphs can further provide deeper insight into the behavior of the data set. Utilizing the insights gained, further experiments and research can be done to understand the data set better.

Our Team :)

3rd year CSE A-

1.Kamlesh.J.V.K.
RA2111003040034

2.Sumit Kumar Singh
RA2111003040099

3rd year CSE D-

3.Raunak Bansal
RA2111026040087

Thank You!!!