In [196…

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")
```

In [197…

```python
df=pd.read_csv(r"C:\Users\himan\Downloads\customer_support_tickets.csv")
```

In [198…

```python
df.columns
```

Out[198…

```
Index(['Ticket ID', 'Customer Name', 'Customer Email', 'Customer Age',
       'Customer Gender', 'Product Purchased', 'Date of Purchase',
       'Ticket Type', 'Ticket Subject', 'Ticket Description', 'Ticket Status',
       'Resolution', 'Ticket Priority', 'Ticket Channel',
       'First Response Time', 'Time to Resolution',
       'Customer Satisfaction Rating'],
      dtype='object')
```

In [199…

```python
df.head()
```

Out[199…

	Ticket ID	Customer Name	Customer Email	Customer Age	Customer Gender	Product Purchased	D Pur
0	1	Marisa Obrien	carrollallison@example.com	32	Other	GoPro Hero	202
1	2	Jessica Rios	clarkeashley@example.com	42	Female	LG Smart TV	202
2	3	Christopher Robbins	gonzalestracy@example.com	48	Other	Dell XPS	202
3	4	Christina Dillon	bradleyolson@example.org	27	Female	Microsoft Office	202
4	5	Alexander Carroll	bradleymark@example.com	67	Female	Autodesk AutoCAD	202

In [200…

```python
df.shape
```

Out[200…

```
(8469, 17)
```

In [201…

```
df.isna().sum()
```

Out[201…

```
Ticket ID                          0
Customer Name                      0
Customer Email                     0
Customer Age                       0
Customer Gender                    0
Product Purchased                  0
Date of Purchase                   0
Ticket Type                        0
Ticket Subject                     0
Ticket Description                 0
Ticket Status                      0
Resolution                      5700
Ticket Priority                    0
Ticket Channel                     0
First Response Time             2819
Time to Resolution              5700
Customer Satisfaction Rating    5700
dtype: int64
```

In [202…

```
df.dtypes
```

Out[202…

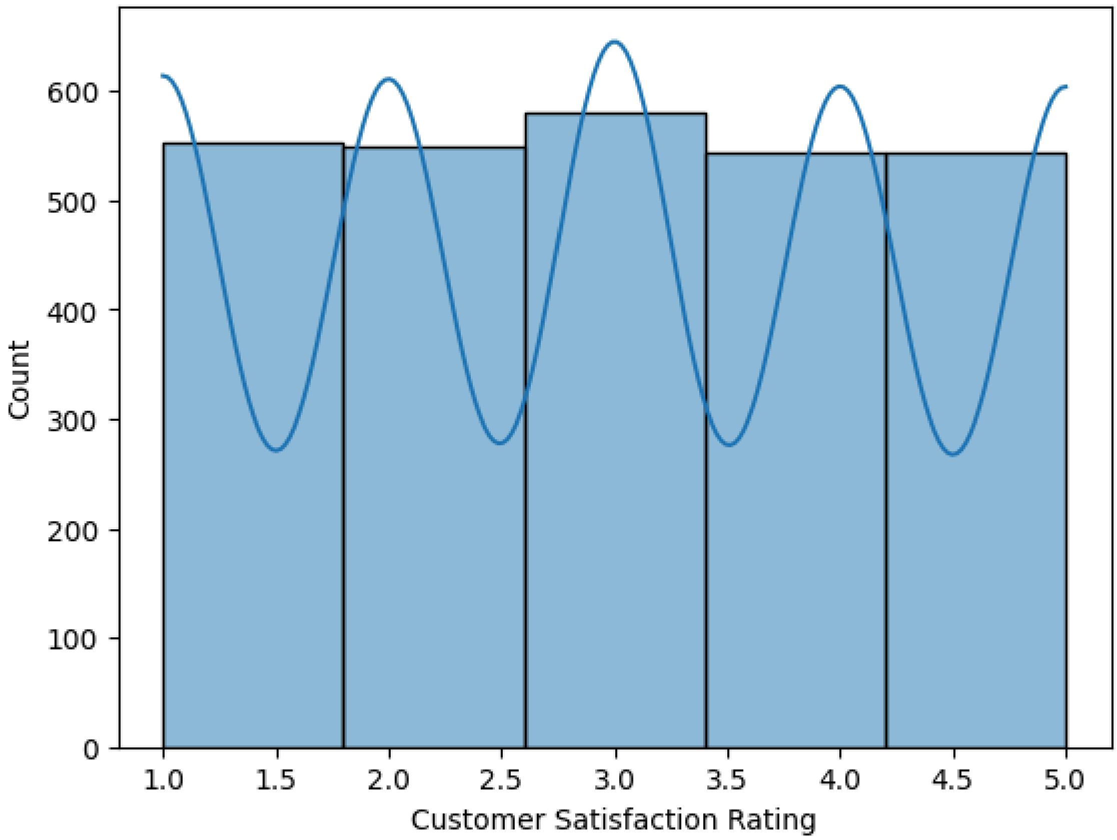
```
Ticket ID                         int64
Customer Name                    object
Customer Email                   object
Customer Age                      int64
Customer Gender                  object
Product Purchased                object
Date of Purchase                 object
Ticket Type                      object
Ticket Subject                   object
Ticket Description               object
Ticket Status                    object
Resolution                       object
Ticket Priority                  object
Ticket Channel                   object
First Response Time              object
Time to Resolution               object
Customer Satisfaction Rating    float64
dtype: object
```

In [203…

```
sns.histplot(df['Customer Satisfaction Rating'], kde='True', bins=5)
```

Out[203…

```
<Axes: xlabel='Customer Satisfaction Rating', ylabel='Count'>
```



```
In [204…  df[['Ticket Status', 'Customer Satisfaction Rating']].isnull().groupby(df['Ticke
```

Out[204…

	Ticket Status	Customer Satisfaction Rating
Ticket Status		
Closed	0	0
Open	0	2819
Pending Customer Response	0	2881

```
In [205…  closed_df = df[df['Ticket Status'] == 'Closed']
```

```
In [206…  closed_df.head()
```

Out[206…

	Ticket ID	Customer Name	Customer Email	Customer Age	Customer Gender	Product Purchased	I Pu
2	3	Christopher Robbins	gonzalestracy@example.com	48	Other	Dell XPS	20
3	4	Christina Dillon	bradleyolson@example.org	27	Female	Microsoft Office	20
4	5	Alexander Carroll	bradleymark@example.com	67	Female	Autodesk AutoCAD	20
10	11	Joseph Moreno	mbrown@example.org	48	Male	Nintendo Switch	20
11	12	Brandon Arnold	davisjohn@example.net	51	Male	Microsoft Xbox Controller	20

In [207…

```
closed_df.isna().sum()
```

Out[207…

```
Ticket ID                       0
Customer Name                   0
Customer Email                  0
Customer Age                    0
Customer Gender                 0
Product Purchased               0
Date of Purchase                0
Ticket Type                     0
Ticket Subject                  0
Ticket Description              0
Ticket Status                   0
Resolution                      0
Ticket Priority                 0
Ticket Channel                  0
First Response Time             0
Time to Resolution              0
Customer Satisfaction Rating    0
dtype: int64
```

In [208…

```
ndf=closed_df
```

In [209…

```
ndf.head()
```

Out[209…

	Ticket ID	Customer Name	Customer Email	Customer Age	Customer Gender	Product Purchased	I Pu
2	3	Christopher Robbins	gonzalestracy@example.com	48	Other	Dell XPS	20
3	4	Christina Dillon	bradleyolson@example.org	27	Female	Microsoft Office	20
4	5	Alexander Carroll	bradleymark@example.com	67	Female	Autodesk AutoCAD	20
10	11	Joseph Moreno	mbrown@example.org	48	Male	Nintendo Switch	20
11	12	Brandon Arnold	davisjohn@example.net	51	Male	Microsoft Xbox Controller	20

In [210…

```
ndf.isna().sum()
```

Out[210…

```
Ticket ID                       0
Customer Name                   0
Customer Email                  0
Customer Age                    0
Customer Gender                 0
Product Purchased               0
Date of Purchase                0
Ticket Type                     0
Ticket Subject                  0
Ticket Description              0
Ticket Status                   0
Resolution                      0
Ticket Priority                 0
Ticket Channel                  0
First Response Time             0
Time to Resolution              0
Customer Satisfaction Rating    0
dtype: int64
```

In [211…

```
ndf.columns
```

```
Out[211…  Index(['Ticket ID', 'Customer Name', 'Customer Email', 'Customer Age',
                 'Customer Gender', 'Product Purchased', 'Date of Purchase',
                 'Ticket Type', 'Ticket Subject', 'Ticket Description', 'Ticket Status',
                 'Resolution', 'Ticket Priority', 'Ticket Channel',
                 'First Response Time', 'Time to Resolution',
                 'Customer Satisfaction Rating'],
                dtype='object')
```

```
In [212…  ndf.drop(['Ticket ID', 'Customer Email'], axis=1, inplace=True)
```

```
In [213…  ndf.columns
```

```
Out[213…  Index(['Customer Name', 'Customer Age', 'Customer Gender', 'Product Purchased',
                 'Date of Purchase', 'Ticket Type', 'Ticket Subject',
                 'Ticket Description', 'Ticket Status', 'Resolution', 'Ticket Priority',
                 'Ticket Channel', 'First Response Time', 'Time to Resolution',
                 'Customer Satisfaction Rating'],
                dtype='object')
```

```
In [214…  ndf.dtypes
```

```
Out[214…  Customer Name                    object
          Customer Age                      int64
          Customer Gender                  object
          Product Purchased                object
          Date of Purchase                 object
          Ticket Type                      object
          Ticket Subject                   object
          Ticket Description               object
          Ticket Status                    object
          Resolution                       object
          Ticket Priority                  object
          Ticket Channel                   object
          First Response Time              object
          Time to Resolution               object
          Customer Satisfaction Rating    float64
          dtype: object
```

```
In [215…  ndf['Ticket Status'].unique()
```

```
Out[215…  array(['Closed'], dtype=object)
```

```
In [216…  ndf['Date of Purchase'] = pd.to_datetime(ndf['Date of Purchase'])
```

```
In [217…  ndf.dtypes
```

Out[217…

```
Customer Name                         object
Customer Age                           int64
Customer Gender                       object
Product Purchased                     object
Date of Purchase              datetime64[ns]
Ticket Type                           object
Ticket Subject                        object
Ticket Description                    object
Ticket Status                         object
Resolution                            object
Ticket Priority                       object
Ticket Channel                        object
First Response Time                   object
Time to Resolution                    object
Customer Satisfaction Rating         float64
dtype: object
```

In [218…

```python
ndf.drop(['Ticket Status'],axis=1, inplace=True)
```

In [219…

```python
ndf.drop(['Resolution'], axis=1, inplace=True)
```

In [220…

```python
ndf['First Response Time']=pd.to_datetime(ndf['First Response Time'], errors='co
ndf['Time to Resolution']=pd.to_datetime(ndf['Time to Resolution'], errors='coer
ndf['Resolution Duration (mins)'] = (ndf['Time to Resolution'] - ndf['First Resp
```

In [221…

```python
ndf.head()
```

Out[221…

	Customer Name	Customer Age	Customer Gender	Product Purchased	Date of Purchase	Ticket Type	Ticket Subject	Tick
2	Christopher Robbins	48	Other	Dell XPS	2020-07-14	Technical issue	Network problem	pr {proc
3	Christina Dillon	27	Female	Microsoft Office	2020-11-13	Billing inquiry	Account access	I'm {proc
4	Alexander Carroll	67	Female	Autodesk AutoCAD	2020-02-04	Billing inquiry	Data loss	I'm {proc
10	Joseph Moreno	48	Male	Nintendo Switch	2021-01-19	Cancellation request	Data loss	I'm {proc
11	Brandon Arnold	51	Male	Microsoft Xbox Controller	2021-10-24	Product inquiry	Software bug	I'm {proc

In [222…

```python
ndf['Resolution Duration (mins)'] = (
    (ndf['Time to Resolution'] - ndf['First Response Time']).dt.total_seconds().
)
```

In [223…

```python
ndf.head()
```

Out[223…

	Customer Name	Customer Age	Customer Gender	Product Purchased	Date of Purchase	Ticket Type	Ticket Subject	Tick
2	Christopher Robbins	48	Other	Dell XPS	2020-07-14	Technical issue	Network problem	pr {proc
3	Christina Dillon	27	Female	Microsoft Office	2020-11-13	Billing inquiry	Account access	I'm {proc
4	Alexander Carroll	67	Female	Autodesk AutoCAD	2020-02-04	Billing inquiry	Data loss	I'm {proc
10	Joseph Moreno	48	Male	Nintendo Switch	2021-01-19	Cancellation request	Data loss	I'm {proc
11	Brandon Arnold	51	Male	Microsoft Xbox Controller	2021-10-24	Product inquiry	Software bug	I'm {proc

In [224…

```
ndf.dtypes
```

Out[224…

```
Customer Name                            object
Customer Age                              int64
Customer Gender                          object
Product Purchased                        object
Date of Purchase                 datetime64[ns]
Ticket Type                              object
Ticket Subject                           object
Ticket Description                       object
Ticket Priority                          object
Ticket Channel                           object
First Response Time              datetime64[ns]
Time to Resolution               datetime64[ns]
Customer Satisfaction Rating            float64
Resolution Duration (mins)              float64
dtype: object
```

In [225…

```
ndf.drop(['Customer Name'], axis=1, inplace=True)
```

In [226…

```
ndf.dtypes
```

Out[226…

```
Customer Age                              int64
Customer Gender                          object
Product Purchased                        object
Date of Purchase                 datetime64[ns]
Ticket Type                              object
Ticket Subject                           object
Ticket Description                       object
Ticket Priority                          object
Ticket Channel                           object
First Response Time              datetime64[ns]
Time to Resolution               datetime64[ns]
Customer Satisfaction Rating            float64
Resolution Duration (mins)              float64
dtype: object
```

In [227…

```
from sklearn.preprocessing import LabelEncoder

le=LabelEncoder()
ndf['Customer Gender']=le.fit_transform(ndf['Customer Gender'])
ndf['Ticket Type']=le.fit_transform(ndf['Ticket Type'])
ndf['Ticket Channel']=le.fit_transform(ndf['Ticket Channel'])
ndf['Product Purchased']=le.fit_transform(ndf['Product Purchased'])
ndf['Ticket Priority']=le.fit_transform(ndf['Ticket Priority'])
ndf.head()
```

Out[227…

	Customer Age	Customer Gender	Product Purchased	Date of Purchase	Ticket Type	Ticket Subject	Ticket Description	T Pr
2	48	2	10	2020-07-14	4	Network problem	I'm facing a problem with my {product_purchase...	
3	27	0	25	2020-11-13	0	Account access	I'm having an issue with the {product_purchase...	
4	67	0	5	2020-02-04	0	Data loss	I'm having an issue with the {product_purchase...	
10	48	1	30	2021-01-19	1	Data loss	I'm having an issue with the {product_purchase...	
11	51	1	27	2021-10-24	2	Software bug	I'm having an issue with the {product_purchase...	

In [228…

```
ndf.dtypes
```

Out[228…

```
Customer Age                                int64
Customer Gender                             int32
Product Purchased                           int32
Date of Purchase                   datetime64[ns]
Ticket Type                                 int32
Ticket Subject                             object
Ticket Description                         object
Ticket Priority                             int32
Ticket Channel                              int32
First Response Time                datetime64[ns]
Time to Resolution                 datetime64[ns]
Customer Satisfaction Rating              float64
Resolution Duration (mins)                float64
dtype: object
```

In [229…

```python
ndf['Ticket Subject'].value_counts()
```

Out[229…

```
Ticket Subject
Network problem             201
Software bug                199
Product compatibility       195
Product recommendation      186
Product setup               183
Hardware issue              183
Delivery problem            178
Refund request              178
Battery life                173
Account access              171
Installation support        158
Peripheral compatibility    158
Payment issue               156
Display issue               155
Cancellation request        148
Data loss                   147
Name: count, dtype: int64
```

In [230…

```python
from sklearn.preprocessing import LabelEncoder

le=LabelEncoder()
ndf['Ticket Subject']=le.fit_transform(ndf['Ticket Subject'])
```

In [231…

```python
ndf.head()
```

Out[231…

	Customer Age	Customer Gender	Product Purchased	Date of Purchase	Ticket Type	Ticket Subject	Ticket Description	Ti Pri
2	48	2	10	2020-07-14	4	8	I'm facing a problem with my {product_purchase…	
3	27	0	25	2020-11-13	0	0	I'm having an issue with the {product_purchase…	
4	67	0	5	2020-02-04	0	3	I'm having an issue with the {product_purchase…	
10	48	1	30	2021-01-19	1	3	I'm having an issue with the {product_purchase…	
11	51	1	27	2021-10-24	2	15	I'm having an issue with the {product_purchase…	

In [232…

```
import re
import string
from nltk.corpus import stopwords
import nltk
nltk.download('stopwords')

def clean_text(text):
    text=text.lower()
    text=re.sub(r'\{.*?\}','',text)
    text=re.sub(f"[{re.escape(string.punctuation)}]",'',text)
    text=re.sub(r'\d+','',text)
    text="".join([word for word in text.split()if word not in stopwords.words('e
    return text
ndf['Cleaned_description']=ndf['Ticket Description'].astype(str).apply(clean_tex
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\himan\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

In [283…

```
from sklearn.feature_extraction.text import TfidfVectorizer

tfidf=TfidfVectorizer(max_features=100)
tfidf_matrix=tfidf.fit_transform(ndf['Cleaned_description'])
```

In [239…

```
ndf.head()
```

Out[239…

	Customer Age	Customer Gender	Product Purchased	Date of Purchase	Ticket Type	Ticket Subject	Ticket Description	Ti Pri
2	48	2	10	2020-07-14	4	8	I'm facing a problem with my {product_purchase...	
3	27	0	25	2020-11-13	0	0	I'm having an issue with the {product_purchase...	
4	67	0	5	2020-02-04	0	3	I'm having an issue with the {product_purchase...	
10	48	1	30	2021-01-19	1	3	I'm having an issue with the {product_purchase...	
11	51	1	27	2021-10-24	2	15	I'm having an issue with the {product_purchase...	

In [243…

```
ndf['purchase_month']=ndf['Date of Purchase'].dt.month
```

In [245…

```
ndf.dtypes
```

Out[245…

```
Customer Age                            int64
Customer Gender                         int32
Product Purchased                       int32
Date of Purchase               datetime64[ns]
Ticket Type                             int32
Ticket Subject                          int32
Ticket Description                     object
Ticket Priority                         int32
Ticket Channel                          int32
First Response Time            datetime64[ns]
Time to Resolution             datetime64[ns]
Customer Satisfaction Rating          float64
Resolution Duration (mins)            float64
Cleaned_description                    object
purchase_month                          int32
dtype: object
```

In [247…

```
ndf.drop(['Time to Resolution','Date of Purchase'],axis=1,inplace=True)
```

In [285…

```
ndf.drop(['Ticket Description'],axis=1,inplace=True)
```

In [249…

```
ndf.drop(['First Response Time'],axis=1,inplace=True)
```

In [251…

```
X=ndf.drop(['Customer Satisfaction Rating'],axis=1)
y=ndf['Customer Satisfaction Rating']
```

In [289…

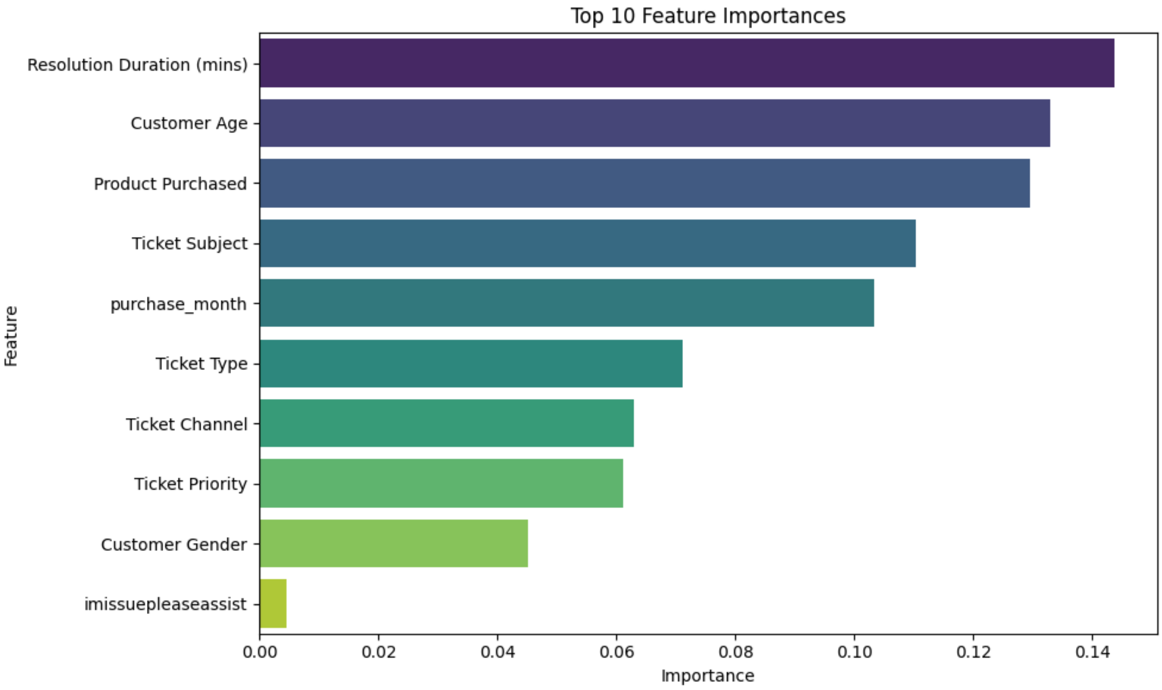
```
from sklearn.ensemble import RandomForestClassifier

X_text = tfidf_matrix.toarray()
```

```
text_feature_names = tfidf.get_feature_names_out()
X_structured = ndf.drop(columns=[
    'Cleaned_description',
    'Customer Satisfaction Rating'
])
X_structured_array = X_structured.values
structured_feature_names = X_structured.columns.tolist()
X = np.hstack((X_structured_array, X_text))
feature_names = structured_feature_names + list(text_feature_names)


rf=RandomForestClassifier()
rf.fit(X,y)

importances = rf.feature_importances_
feat_imp_df = pd.DataFrame({'Feature': feature_names , 'Importance': importances
feat_imp_df = feat_imp_df.sort_values(by='Importance', ascending=False)
print(feat_imp_df.head(10))
```

```
                    Feature  Importance
7  Resolution Duration (mins)    0.160237
0              Customer Age    0.147929
2         Product Purchased    0.146155
4            Ticket Subject    0.122778
8            purchase_month    0.114588
3               Ticket Type    0.076834
6            Ticket Channel    0.066971
5           Ticket Priority    0.065149
1           Customer Gender    0.049737
9       imissuepleaseassist    0.004302
```

In [255…

```
import matplotlib.pyplot as plt
import seaborn as sns

plt.figure(figsize=(10, 6))
sns.barplot(data=feat_imp_df.head(10), x='Importance', y='Feature', palette='vir
plt.title("Top 10 Feature Importances")
plt.tight_layout()
plt.show()
```



In [291…

```
from sklearn.model_selection import train_test_split

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

In [293…

```
from sklearn.ensemble import RandomForestClassifier

rf_model=RandomForestClassifier()
rf_model.fit(X_train,y_train)
pred=rf_model.predict(X_test)
```

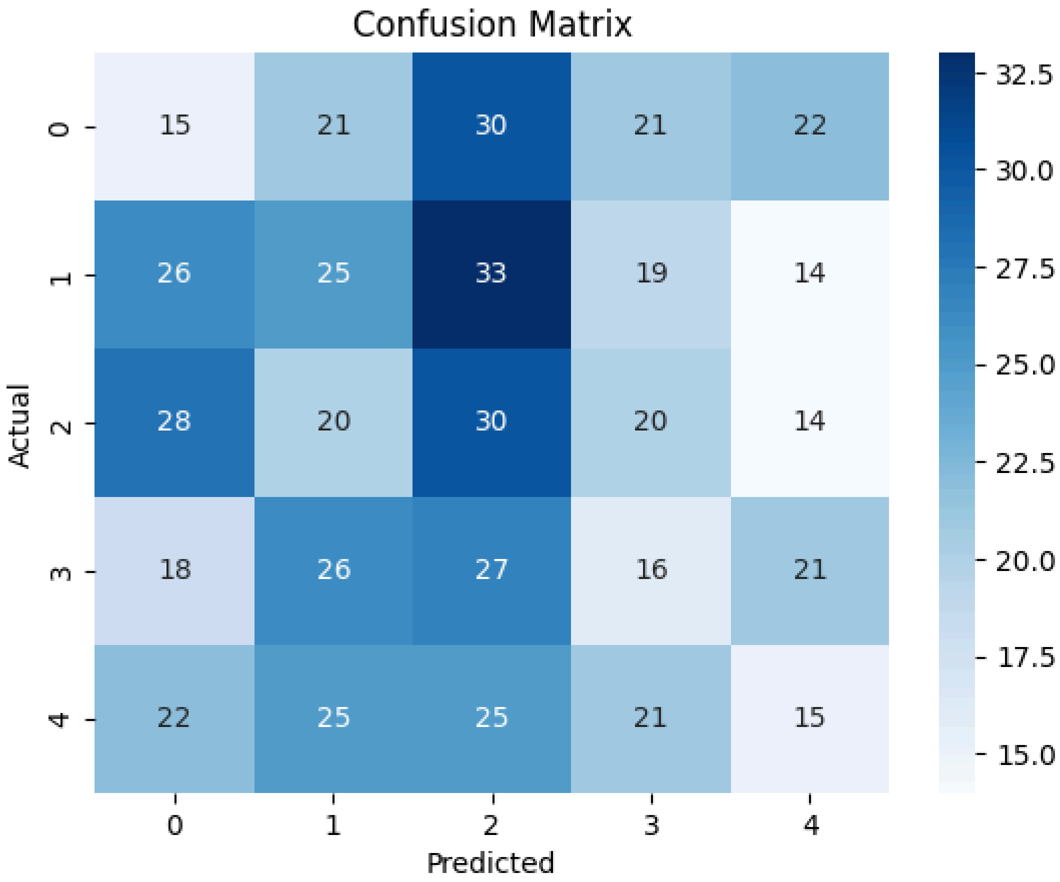
In [295…

```
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_sc

accuracy= accuracy_score(y_test,pred)
precision= precision_score(y_test,pred,average='weighted')
recal=  recall_score(y_test,pred,average='weighted')
f1_score=  f1_score(y_test,pred,average='weighted')
print(f"Accuracy: {accuracy:.2f}")
print(f"Precision: {precision:.2f}")
print(f"Recall: {recal:.2f}")
print(f"f1_score: {f1_score:.2f}")
```

```
Accuracy: 0.18
Precision: 0.18
Recall: 0.18
f1_score: 0.18
```

In [297…

```
from sklearn.metrics import confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt

cm = confusion_matrix(y_test, pred)
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues')
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.title("Confusion Matrix")
plt.show()
print(cm)
```



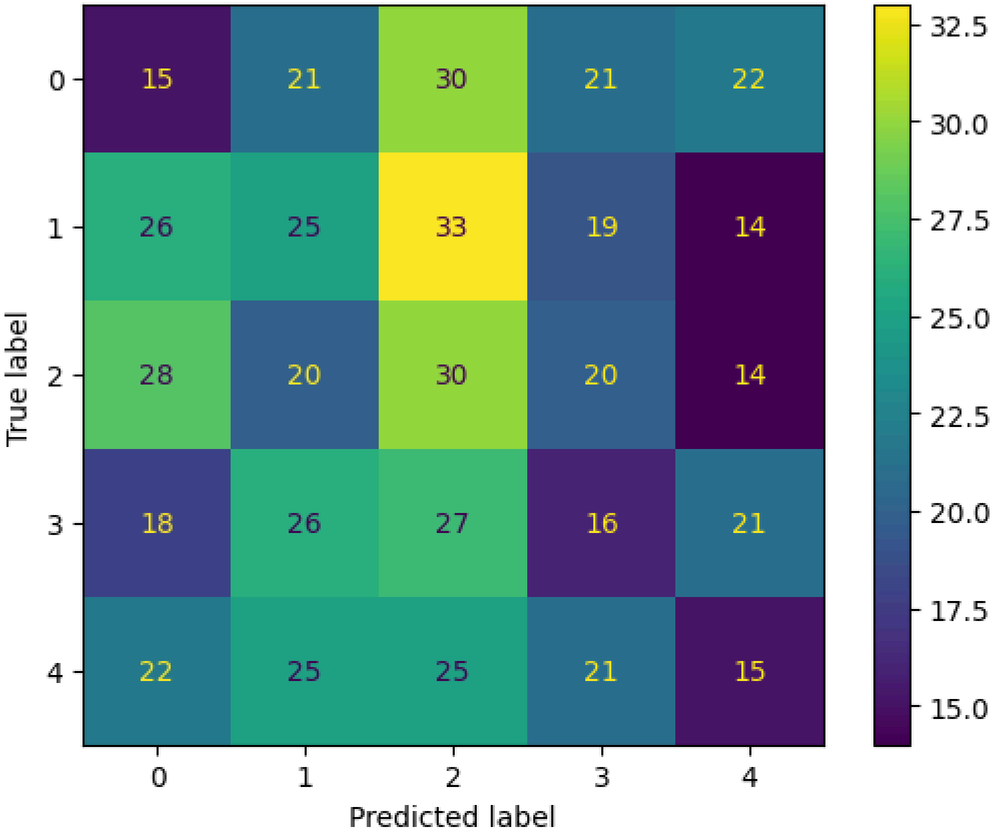
```
[[15 21 30 21 22]
 [26 25 33 19 14]
 [28 20 30 20 14]
 [18 26 27 16 21]
 [22 25 25 21 15]]
```

In [299…

```
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay

cm = confusion_matrix(y_test, pred)
disp = ConfusionMatrixDisplay(confusion_matrix=cm)
disp.plot()
```

Out[299…

```
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x178d1db2360
>
```


0
1
2
3
4
True label
15
21
30
21
22
26
25
33
19
14
28
20
30
20
14
18
26
27
16
21
22
25
25
21
15
0
1
2
3
4
Predicted label
32.5
30.0
27.5
25.0
22.5
20.0
17.5
15.0