In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

In [2]:
```
idata = pd.read_csv(r"C:\Users\ASUS\Downloads\Predictive Crime Analytics-20240318T141844Z-001\Predictive Crime A
```

```
C:\Users\ASUS\AppData\Local\Temp\ipykernel_13472\1486523831.py:1: DtypeWarning: Columns (19,20) have mixed types
. Specify dtype option on import or set low_memory=False.
  idata = pd.read_csv(r"C:\Users\ASUS\Downloads\Predictive Crime Analytics-20240318T141844Z-001\Predictive Crime
Analytics\FIR_Details_Data.csv")
```

In [3]:
```
idata.shape
```

Out[3]: (1694191, 41)

In [4]:
```
idata.head()
```

Out[4]:

	District_Name	UnitName	FIRNo	RI	Year	Month	Offence_From_Date	Offence_To_Date	FIR_Reg_DateTime	FIR_Date	...
0	Bagalkot	Amengad PS	0001/2016	1	2016	1	2015-12-27 12:00:00.000	2015-12-27 12:05:00.000	2016-01-05 11:00:00.000	05/01/2016	...
1	Bagalkot	Amengad PS	0002/2016	1	2016	1	2016-01-12 17:30:00.000	2016-01-12 17:35:00.000	2016-01-12 19:00:00.000	12/01/2016	...
2	Bagalkot	Amengad PS	0003/2016	1	2016	1	2016-01-12 17:45:00.000	2016-01-12 17:50:00.000	2016-01-12 19:30:00.000	12/01/2016	...
3	Bagalkot	Amengad PS	0004/2016	1	2016	1	2016-01-14 21:30:00.000	2016-01-14 21:35:00.000	2016-01-15 14:00:00.000	15/01/2016	...
4	Bagalkot	Amengad PS	0005/2016	1	2016	1	2016-01-18 15:00:00.000	2016-01-18 15:05:00.000	2016-01-18 19:30:00.000	18/01/2016	...

5 rows × 41 columns

In [5]:
```
idata.tail()
```

Out[5]:

	District_Name	UnitName	FIRNo	RI	Year	Month	Offence_From_Date	Offence_To_Date	FIR_Reg_DateTime	FIR_Da
1694186	Yadgir	Yadgiri Women PS	0002/2024	1	2024	1	2024-01-10 05:30:00.000	2024-01-10 21:30:00.000	2024-01-11 21:30:00.000	11/01/20
1694187	Yadgir	Yadgiri Women PS	0003/2024	1	2024	1	2024-01-19 22:30:00.000	2024-01-19 22:35:00.000	2024-01-29 19:30:00.000	29/01/20
1694188	Yadgir	Yadgiri Women PS	0004/2024	1	2024	2	2024-02-06 15:42:00.000	2024-02-06 15:45:00.000	2024-02-07 17:45:00.000	07/02/20
1694189	Yadgir	Yadgiri Women PS	0005/2024	1	2024	2	2024-02-19 12:10:00.000	2024-02-19 12:15:00.000	2024-02-24 17:00:00.000	24/02/20
1694190	Yadgir	Yadgiri Women PS	0006/2024	1	2024	2	2024-02-26 22:30:00.000	2024-02-26 22:45:00.000	2024-02-28 18:30:00.000	28/02/20

5 rows × 41 columns

In [7]:
```
idata.isnull().sum()
```

Out[7]:
```
District_Name                      0
UnitName                           0
FIRNo                              0
RI                                 0
Year                               0
Month                              0
Offence_From_Date                  0
Offence_To_Date                    1
FIR_Reg_DateTime                   0
FIR_Date                           0
FIR Type                           2
FIR_Stage                          0
Complaint_Mode                 18430
CrimeGroup_Name                    0
CrimeHead_Name                     0
Latitude                           5
Longitude                          5
ActSection                        42
IOName                           150
KGID                             150
IOAssigned_Date                1694182
Internal_IO                        0
Place of Offence                   0
Distance from PS                 567
Beat_Name                        197
Village_Area_Name                138
Male                               0
Female                             0
Boy                                0
Girl                               0
Age 0                              0
VICTIM COUNT                       0
Accused Count                      0
Arrested Male                      0
Arrested Female                    0
Arrested Count\tNo.                0
Accused_ChargeSheeted Count        0
Conviction Count                   0
FIR_ID                             0
Unit_ID                            0
Crime_No                           0
dtype: int64
```

In [8]:
```
idata.sample(10)
```

Out[8]:

	District_Name	UnitName	FIRNo	RI	Year	Month	Offence_From_Date	Offence_To_Date	FIR_Reg_DateTime	FIR
334898	Bengaluru City	K.R. Puram PS	0225/2021	1	2021	8	2021-08-06 08:00:00.000	2021-08-06 08:30:00.000	2021-08-09 18:00:00.000	09/08
1386584	Ramanagara	Channapatna Town PS	0079/2021	1	2021	11	2021-10-22 19:40:00.000	2021-10-22 20:30:00.000	2021-11-01 18:30:00.000	01/11
841459	Davanagere	Basavapatna PS	0013/2021	1	2021	1	2021-01-19 18:00:00.000	2021-01-19 19:00:00.000	2021-01-19 19:15:00.000	19/01
1000685	Hubballi Dharwad City	APMC Navanagar	0139/2016	1	2016	11	2016-11-08 16:30:00.000	2016-11-08 16:31:00.000	2016-11-08 21:00:00.000	08/11
1272658	Mysuru City	Metagalli PS	0023/2023	1	2023	2	2023-02-09 11:30:00.000	2023-02-09 12:00:00.000	2023-02-13 14:15:00.000	13/02
790682	Chitradurga	Chitrahalli Gate PS	0131/2018	1	2018	8	2018-08-18 19:30:00.000	2018-08-18 19:32:00.000	2018-08-19 16:30:00.000	19/08
1122470	Kolar	Kolar Rural PS	0320/2022	1	2022	6	2022-06-26 11:00:00.000	2022-06-26 11:05:00.000	2022-06-26 20:30:00.000	26/06
1103453	Kodagu	Madikeri Rural PS	0334/2016	1	2016	11	2016-11-25 16:30:00.000	2016-11-25 16:31:00.000	2016-11-25 22:30:00.000	25/11
77162	Belagavi City	Tilakwadi PS	0068/2023	1	2023	6	2023-06-01 10:39:00.000	2023-06-01 10:40:00.000	2023-06-12 20:45:00.000	12/06
345609	Bengaluru City	Kadugondana Halli PS	0405/2020	1	2020	12	2020-12-22 20:00:00.000	2020-12-22 20:15:00.000	2020-12-22 22:15:00.000	22/12

10 rows × 41 columns

In [9]:
```
idata.iloc[0:3,0:10]
```

Out[9]:

	District_Name	UnitName	FIRNo	RI	Year	Month	Offence_From_Date	Offence_To_Date	FIR_Reg_DateTime	FIR_Date
0	Bagalkot	Amengad PS	0001/2016	1	2016	1	2015-12-27 12:00:00.000	2015-12-27 12:05:00.000	2016-01-05 11:00:00.000	05/01/2016
1	Bagalkot	Amengad PS	0002/2016	1	2016	1	2016-01-12 17:30:00.000	2016-01-12 17:35:00.000	2016-01-12 19:00:00.000	12/01/2016
2	Bagalkot	Amengad PS	0003/2016	1	2016	1	2016-01-12 17:45:00.000	2016-01-12 17:50:00.000	2016-01-12 19:30:00.000	12/01/2016

In [10]:
```
idata.iloc[0:3,10:20]
```

Out[10]:

	FIR Type	FIR_Stage	Complaint_Mode	CrimeGroup_Name	CrimeHead_Name	Latitude	Longitude	ActSection	IOName	K
0	Non Heinous	Dis/Acq	Written	POCSO	Others	0.0	0.0	PROTECTION OF CHILDREN FROM SEXUAL OFFENCES AC...	R S BIRADAR (PI)	189
1	Non Heinous	Convicted	Sue-moto by Police	KARNATAKA POLICE ACT 1963	Street Gambling (87)	0.0	0.0	KARNATAKA POLICE ACT, 1963 U/s: 87	G.H.KUPPI (PSI)	133
2	Non Heinous	Convicted	Sue-moto by Police	KARNATAKA POLICE ACT 1963	Gambling - Matka (78 Class C)	0.0	0.0	KARNATAKA POLICE ACT, 1963 U/s: 78(3)	S G HELVAR (ASI)	127

In [11]:
```
idata.iloc[0:3,20:30]
```

Out[11]:

	IOAssigned_Date	Internal_IO	Place of Offence	Distance from PS	Beat_Name	Village_Area_Name	Male	Female	Boy	Girl
0	NaN	42900007	KAMATAGI BUS STAND, KAMATAGI BUS STAND	WEST FROM PS 12 KM	RURAL BEAT NO 13	KAMATAGI	0	0	0	1
1	NaN	123600010	RAKKASAGI VILLEGE IN FRONT MOUNESHWAR TEMPEL, ...	WEST FROM PS 2 KM	RURAL BEAT NO 6	SULEBAVI	0	0	0	0
2	NaN	124000015	RAKKASAGI VILLAEG BUS STAND, RAKKASAGI VILLAEG...	EAST FROM PS 3 KM	RURAL BEAT NO 6	SULEBAVI	0	0	0	0

In [12]:
```
idata.iloc[0:3,30:42]
```

Out[12]:

	Age 0	VICTIM COUNT	Accused Count	Arrested Male	Arrested Female	Arrested Count\tNo.	Accused_ChargeSheeted Count	Conviction Count	FIR_ID	Unit_ID	Cri
0	0	0	1	1	0	1	1	0	2016000002	1245	10470124520
1	0	0	5	5	0	5	5	2	2016000003	1245	10470124520
2	0	0	1	1	0	1	1	0	2016000004	1245	10470124520

In [13]:
```
idata.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1694191 entries, 0 to 1694190
Data columns (total 41 columns):
 #   Column                       Dtype
---  ------                       -----
 0   District_Name                object
 1   UnitName                     object
 2   FIRNo                        object
 3   RI                           int64
 4   Year                         int64
 5   Month                        int64
 6   Offence_From_Date            object
 7   Offence_To_Date              object
 8   FIR_Reg_DateTime             object
 9   FIR_Date                     object
 10  FIR Type                     object
 11  FIR_Stage                    object
 12  Complaint_Mode               object
 13  CrimeGroup_Name              object
 14  CrimeHead_Name               object
 15  Latitude                     float64
 16  Longitude                    float64
 17  ActSection                   object
 18  IOName                       object
 19  KGID                         object
 20  IOAssigned_Date              object
 21  Internal_IO                  int64
 22  Place of Offence             object
 23  Distance from PS             object
 24  Beat_Name                    object
 25  Village_Area_Name            object
 26  Male                         int64
 27  Female                       int64
 28  Boy                          int64
 29  Girl                         int64
 30  Age 0                        int64
 31  VICTIM COUNT                 int64
 32  Accused Count                int64
 33  Arrested Male                int64
 34  Arrested Female              int64
 35  Arrested Count     No.           int64
 36  Accused_ChargeSheeted Count  int64
 37  Conviction Count             int64
 38  FIR_ID                       int64
 39  Unit_ID                      int64
 40  Crime_No                     int64
dtypes: float64(2), int64(19), object(20)
memory usage: 530.0+ MB
```

In [14]:
```
idata.columns
```

Out[14]:
```
Index(['District_Name', 'UnitName', 'FIRNo', 'RI', 'Year', 'Month',
       'Offence_From_Date', 'Offence_To_Date', 'FIR_Reg_DateTime', 'FIR_Date',
       'FIR Type', 'FIR_Stage', 'Complaint_Mode', 'CrimeGroup_Name',
       'CrimeHead_Name', 'Latitude', 'Longitude', 'ActSection', 'IOName',
       'KGID', 'IOAssigned_Date', 'Internal_IO', 'Place of Offence',
       'Distance from PS', 'Beat_Name', 'Village_Area_Name', 'Male', 'Female',
       'Boy', 'Girl', 'Age 0', 'VICTIM COUNT', 'Accused Count',
       'Arrested Male', 'Arrested Female', 'Arrested Count\tNo.',
       'Accused_ChargeSheeted Count', 'Conviction Count', 'FIR_ID', 'Unit_ID',
       'Crime_No'],
      dtype='object')
```

In [15]:
```
df = idata[['District_Name','FIR_Reg_DateTime']]
```

In [16]:
```
df.head()
```

Out[16]:

	District_Name	FIR_Reg_DateTime
0	Bagalkot	2016-01-05 11:00:00.000
1	Bagalkot	2016-01-12 19:00:00.000
2	Bagalkot	2016-01-12 19:30:00.000
3	Bagalkot	2016-01-15 14:00:00.000
4	Bagalkot	2016-01-18 19:30:00.000

In [17]:
```
df.tail()
```

Out[17]:

	District_Name	FIR_Reg_DateTime
1694186	Yadgir	2024-01-11 21:30:00.000
1694187	Yadgir	2024-01-29 19:30:00.000
1694188	Yadgir	2024-02-07 17:45:00.000
1694189	Yadgir	2024-02-24 17:00:00.000
1694190	Yadgir	2024-02-28 18:30:00.000

In [18]:
```
df.isnull().sum()
```

Out[18]:
```
District_Name       0
FIR_Reg_DateTime    0
dtype: int64
```

In [19]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1694191 entries, 0 to 1694190
Data columns (total 2 columns):
 #   Column            Dtype 
---  ------            ----- 
 0   District_Name     object
 1   FIR_Reg_DateTime  object
dtypes: object(2)
memory usage: 25.9+ MB
```

In [20]:
```
df.shape
```

Out[20]:
```
(1694191, 2)
```

In [21]:
```
df.FIR_Reg_DateTime = pd.to_datetime(df.FIR_Reg_DateTime)
```

```
C:\Users\ASUS\AppData\Local\Temp\ipykernel_13472\3727758170.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df.FIR_Reg_DateTime = pd.to_datetime(df.FIR_Reg_DateTime)
```

In [22]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1694191 entries, 0 to 1694190
Data columns (total 2 columns):
 #   Column            Dtype         
---  ------            -----         
 0   District_Name     object        
 1   FIR_Reg_DateTime  datetime64[ns]
dtypes: datetime64[ns](1), object(1)
memory usage: 25.9+ MB
```

In [23]:
```
df.head()
```

Out[23]:

	District_Name	FIR_Reg_DateTime
0	Bagalkot	2016-01-05 11:00:00
1	Bagalkot	2016-01-12 19:00:00
2	Bagalkot	2016-01-12 19:30:00
3	Bagalkot	2016-01-15 14:00:00
4	Bagalkot	2016-01-18 19:30:00

In [24]:
```
df.tail()
```

Out[24]:

	District_Name	FIR_Reg_DateTime
1694186	Yadgir	2024-01-11 21:30:00
1694187	Yadgir	2024-01-29 19:30:00
1694188	Yadgir	2024-02-07 17:45:00
1694189	Yadgir	2024-02-24 17:00:00
1694190	Yadgir	2024-02-28 18:30:00

In [25]:
```
df.shape
```

Out[25]: (1694191, 2)

In [26]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1694191 entries, 0 to 1694190
Data columns (total 2 columns):
 #   Column            Dtype
---  ------            -----
 0   District_Name     object
 1   FIR_Reg_DateTime  datetime64[ns]
dtypes: datetime64[ns](1), object(1)
memory usage: 25.9+ MB
```

In [27]:
```
df.columns
```

Out[27]: Index(['District_Name', 'FIR_Reg_DateTime'], dtype='object')

In [28]:
```
df.District_Name.value_counts()
```

Out[28]:
```
District_Name
Bengaluru City             430754
Bengaluru Dist              65032
Tumakuru                    62520
Shivamogga                  62047
Mandya                      60219
Belagavi Dist               59979
Hassan                      58790
Mysuru Dist                 51258
Chitradurga                 48581
Ramanagara                  44857
Vijayapur                   39878
Davanagere                  39349
Bidar                       38247
Chickballapura              36973
Raichur                     36814
Chikkamagaluru              35777
Uttara Kannada              32978
Mangaluru City              32090
Kalaburagi                  31703
Mysuru City                 30897
Haveri                      30159
Ballari                     28720
Udupi                       28625
Bagalkot                    28387
Vijayanagara                26086
Koppal                      25204
Hubballi Dharwad City       24296
Kolar                       23997
Chamarajanagar              23365
Dakshina Kannada            21430
Belagavi City               20980
Kalaburagi City             20493
Yadgir                      19708
Kodagu                      19489
Gadag                       18589
Dharwad                     14983
K.G.F                       10352
Karnataka Railways           9981
CID                           241
Coastal Security Police       214
ISD Bengaluru                 149
Name: count, dtype: int64
```

In [30]:
```
df_BengaluruCity = df[df["District_Name"]=="Bengaluru City"]
```

In [31]:
```
df_BengaluruCity.head()
```

Out[31]:

	District_Name	FIR_Reg_DateTime
138066	Bengaluru City	2016-01-01 01:30:00
138067	Bengaluru City	2016-01-02 17:40:00
138068	Bengaluru City	2016-01-07 12:30:00
138069	Bengaluru City	2016-01-07 21:45:00
138070	Bengaluru City	2016-01-08 17:50:00

In [32]:
```
df_BengaluruCity.tail()
```

Out[32]:

	District_Name	FIR_Reg_DateTime
568815	Bengaluru City	2024-02-26 14:00:00
568816	Bengaluru City	2024-02-26 21:30:00
568817	Bengaluru City	2024-03-02 20:15:00
568818	Bengaluru City	2024-03-04 21:30:00
568819	Bengaluru City	2024-03-09 18:00:00

In [33]:
```
df_BengaluruCity.sample(10)
```

Out[33]:

	District_Name	FIR_Reg_DateTime
472887	Bengaluru City	2016-04-13 20:55:00
540071	Bengaluru City	2020-03-06 11:30:00
228895	Bengaluru City	2023-07-10 11:00:00
223449	Bengaluru City	2016-07-20 10:40:00
333236	Bengaluru City	2018-06-21 11:00:00
146963	Bengaluru City	2022-01-19 16:30:00
236309	Bengaluru City	2018-08-02 16:00:00
295386	Bengaluru City	2016-12-05 15:25:00
553351	Bengaluru City	2017-12-13 12:45:00
503613	Bengaluru City	2021-04-24 18:00:00

In [34]:
```
df_BengaluruCity.shape
```

Out[34]:
```
(430754, 2)
```

In [35]:
```
df_BengaluruCity.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 430754 entries, 138066 to 568819
Data columns (total 2 columns):
 #   Column            Non-Null Count   Dtype         
---  ------            --------------   -----         
 0   District_Name     430754 non-null  object        
 1   FIR_Reg_DateTime  430754 non-null  datetime64[ns]
dtypes: datetime64[ns](1), object(1)
memory usage: 9.9+ MB
```

In [36]:
```
df_BengaluruCity.isnull().sum()
```

Out[36]:
```
District_Name       0
FIR_Reg_DateTime    0
dtype: int64
```

In [37]:
```
df_BengaluruCity.columns
```

Out[37]:
```
Index(['District_Name', 'FIR_Reg_DateTime'], dtype='object')
```

In [39]:
```
df_BengaluruCity.index = pd.DatetimeIndex(df_BengaluruCity.FIR_Reg_DateTime)
```

In [40]:
```
df_BengaluruCity.head()
```

Out[40]:

FIR_Reg_DateTime	District_Name	FIR_Reg_DateTime
2016-01-01 01:30:00	Bengaluru City	2016-01-01 01:30:00
2016-01-02 17:40:00	Bengaluru City	2016-01-02 17:40:00
2016-01-07 12:30:00	Bengaluru City	2016-01-07 12:30:00
2016-01-07 21:45:00	Bengaluru City	2016-01-07 21:45:00
2016-01-08 17:50:00	Bengaluru City	2016-01-08 17:50:00

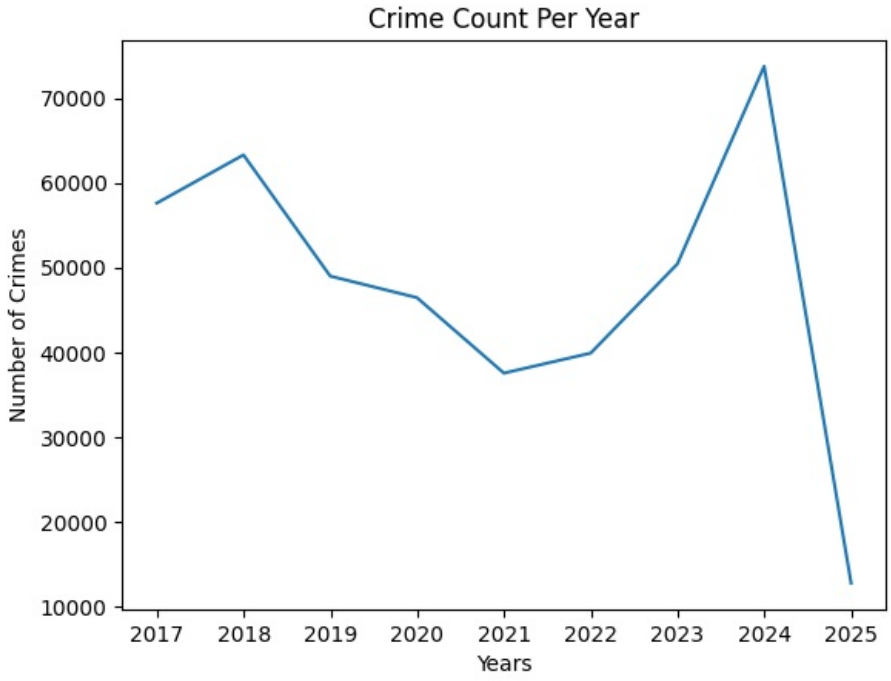
In [41]:
```
df_BengaluruCity.tail()
```

Out[41]:

FIR_Reg_DateTime	District_Name	FIR_Reg_DateTime
2024-02-26 14:00:00	Bengaluru City	2024-02-26 14:00:00
2024-02-26 21:30:00	Bengaluru City	2024-02-26 21:30:00
2024-03-02 20:15:00	Bengaluru City	2024-03-02 20:15:00
2024-03-04 21:30:00	Bengaluru City	2024-03-04 21:30:00
2024-03-09 18:00:00	Bengaluru City	2024-03-09 18:00:00

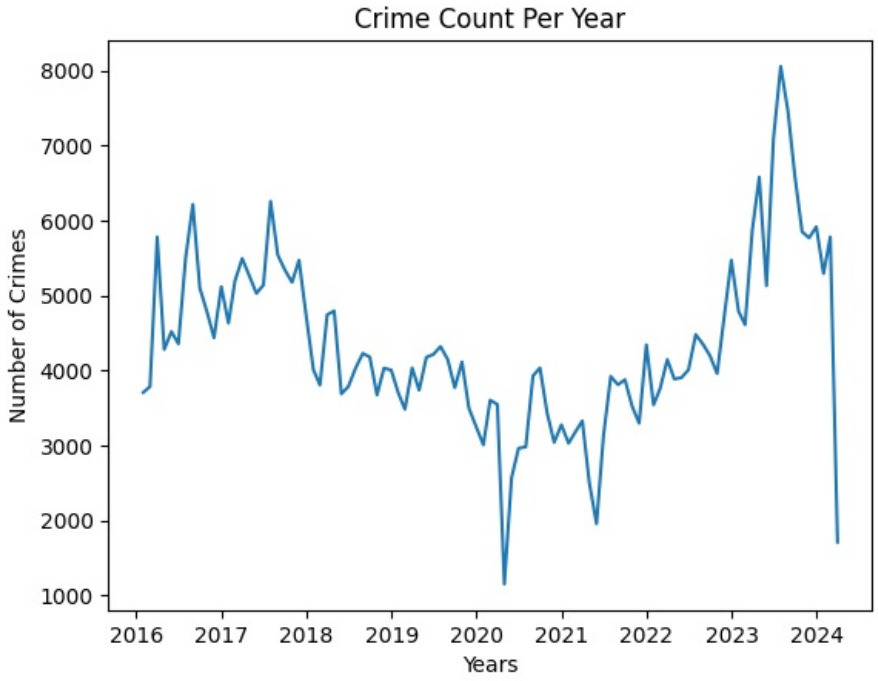
In [42]:
```
plt.plot(df_BengaluruCity.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[42]:
```
Text(0, 0.5, 'Number of Crimes')
```



In [43]:
```
plt.plot(df_BengaluruCity.resample('M').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[43]:
```
Text(0, 0.5, 'Number of Crimes')
```



In [44]:
```
df_BengaluruCity_prophet = pd.DataFrame(df_BengaluruCity.resample('M').size().reset_index())
```

In [45]:
```
df_BengaluruCity_prophet.columns =['Date','Crime Count']
```

In [46]:
```
df_BengaluruCity_prophet.head()
```

Out[46]:

	Date	Crime Count
0	2016-01-31	3709
1	2016-02-29	3787
2	2016-03-31	5783
3	2016-04-30	4281
4	2016-05-31	4521

In [47]:
```
df_BengaluruCity_prophet.shape
```

Out[47]:
```
(99, 2)
```

In [48]:
```
df_BengaluruCity_prophet=df_BengaluruCity_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [49]:
```
df_BengaluruCity_prophet.head()
```

Out[49]:

	ds	y
0	2016-01-31	3709
1	2016-02-29	3787
2	2016-03-31	5783
3	2016-04-30	4281
4	2016-05-31	4521

In [50]:
```
from prophet import Prophet
```

In [57]:
```
m_BengaluruCity = Prophet()
m_BengaluruCity.fit(df_BengaluruCity_prophet)
```

```
19:49:08 - cmdstanpy - INFO - Chain [1] start processing
19:49:09 - cmdstanpy - INFO - Chain [1] done processing
```

Out[57]:
```
<prophet.forecaster.Prophet at 0x1df7ea2c9d0>
```

In [58]:
```
pred=m_BengaluruCity.make_future_dataframe(periods=24,freq='M')
forcast = m_BengaluruCity.predict(pred)
```

In [59]:
```
forcast
```

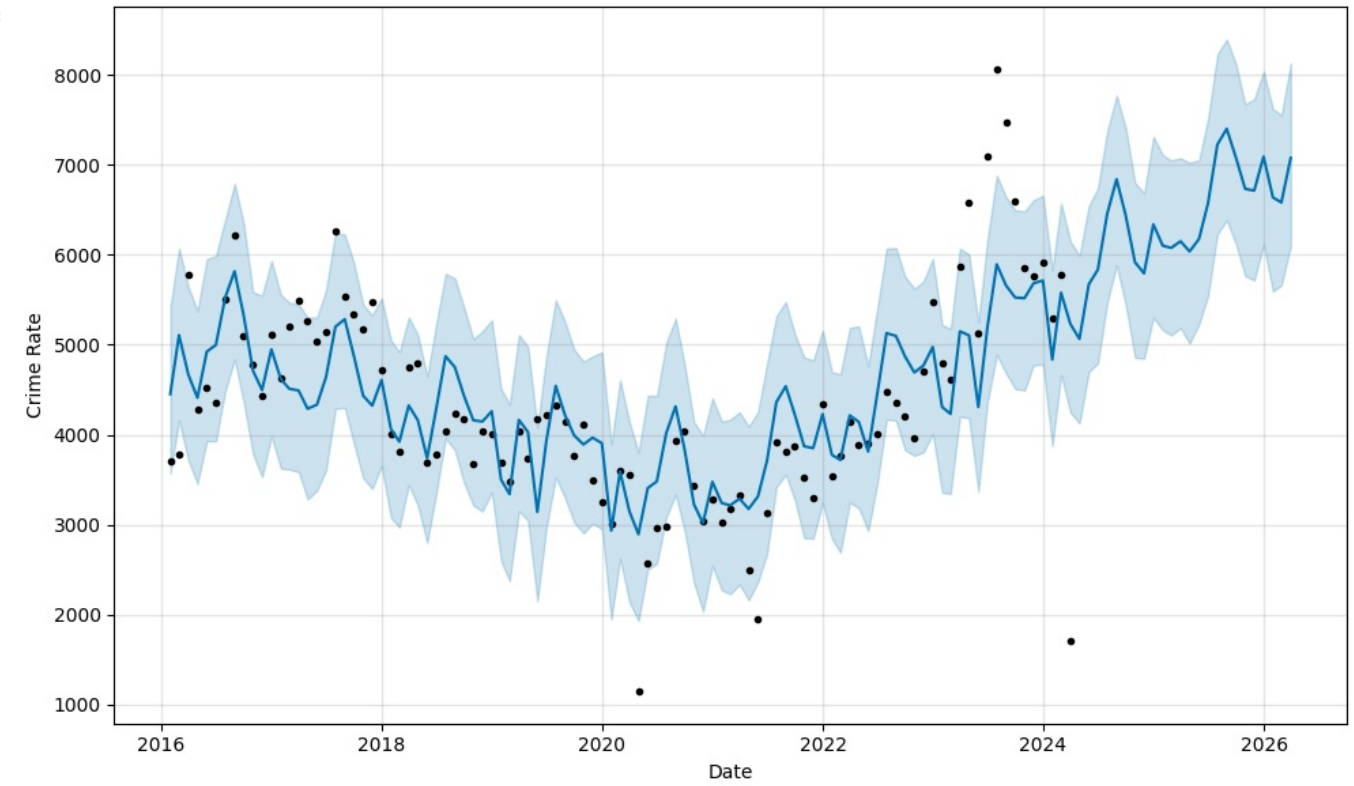
Out[59]:

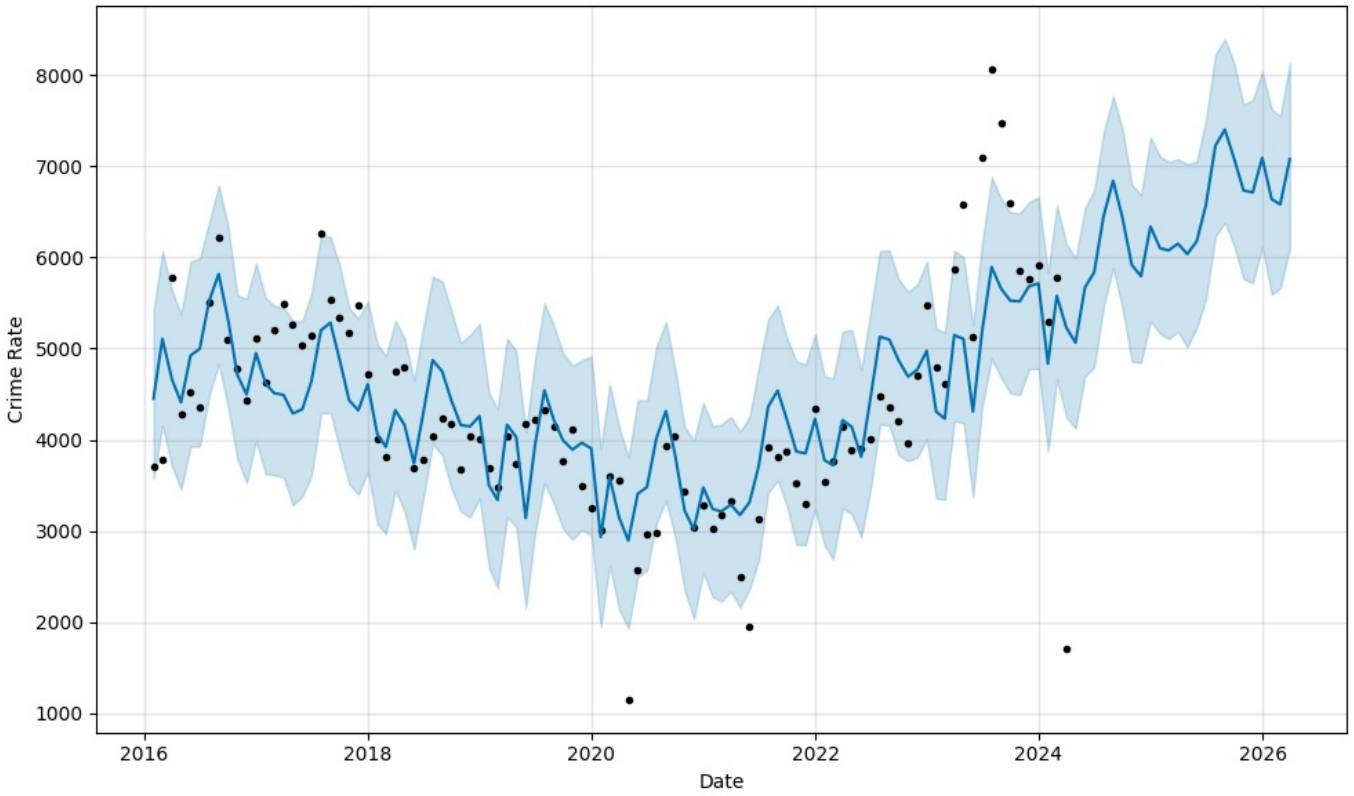
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_
0	2016-01-31	5089.477988	3571.518706	5422.756132	5089.477988	5089.477988	-637.602937	-637.602937	-637.(
1	2016-02-29	5059.371462	4159.473030	6072.625261	5059.371462	5059.371462	48.420819	48.420819	48.4
2	2016-03-31	5027.188624	3712.866532	5639.696047	5027.188624	5027.188624	-360.178234	-360.178234	-360.1
3	2016-04-30	4996.043942	3455.945296	5377.674726	4996.043942	4996.043942	-584.951401	-584.951401	-584.9
4	2016-05-31	4963.861104	3930.112789	5953.606652	4963.861104	4963.861104	-39.650936	-39.650936	-39.(
...	...	...	...	...	...	...	...	...	...
118	2025-11-30	6783.960789	5717.470696	7729.117754	6627.535729	6935.401911	-71.107189	-71.107189	-71.1
119	2025-12-31	6844.669541	6113.637461	8037.880345	6679.358095	7009.913103	246.057561	246.057561	246.(
120	2026-01-31	6905.378293	5593.250566	7623.102930	6729.216093	7082.964242	-267.652632	-267.652632	-267.(
121	2026-02-28	6960.212004	5662.554858	7551.440862	6775.539781	7151.201834	-378.783112	-378.783112	-378.7
122	2026-03-31	7020.920756	6096.947486	8128.772658	6821.859275	7227.589559	55.874131	55.874131	55.8

123 rows × 16 columns

In [60]:
```
m_BengaluruCity.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[60]:





In [61]:
```
import json
```

In [63]:
```
import json
from prophet.serialize import model_to_json, model_from_json
with open('prophet_model.json', 'w') as fout:
    json.dump(model_to_json(m_BengaluruCity), fout) # Save model
```

In [65]:
```
with open('prophet_model.json', 'r') as fin:
    m_BengaluruCity = model_from_json(json.load(fin)) # Load model
```

In [66]:
```
x = int(input("Enter Numbers Months to forecast"))
```

In [67]:
```
pred = m_BengaluruCity.make_future_dataframe(periods=x ,freq = "M")
forcast = m_BengaluruCity.predict(pred)
```

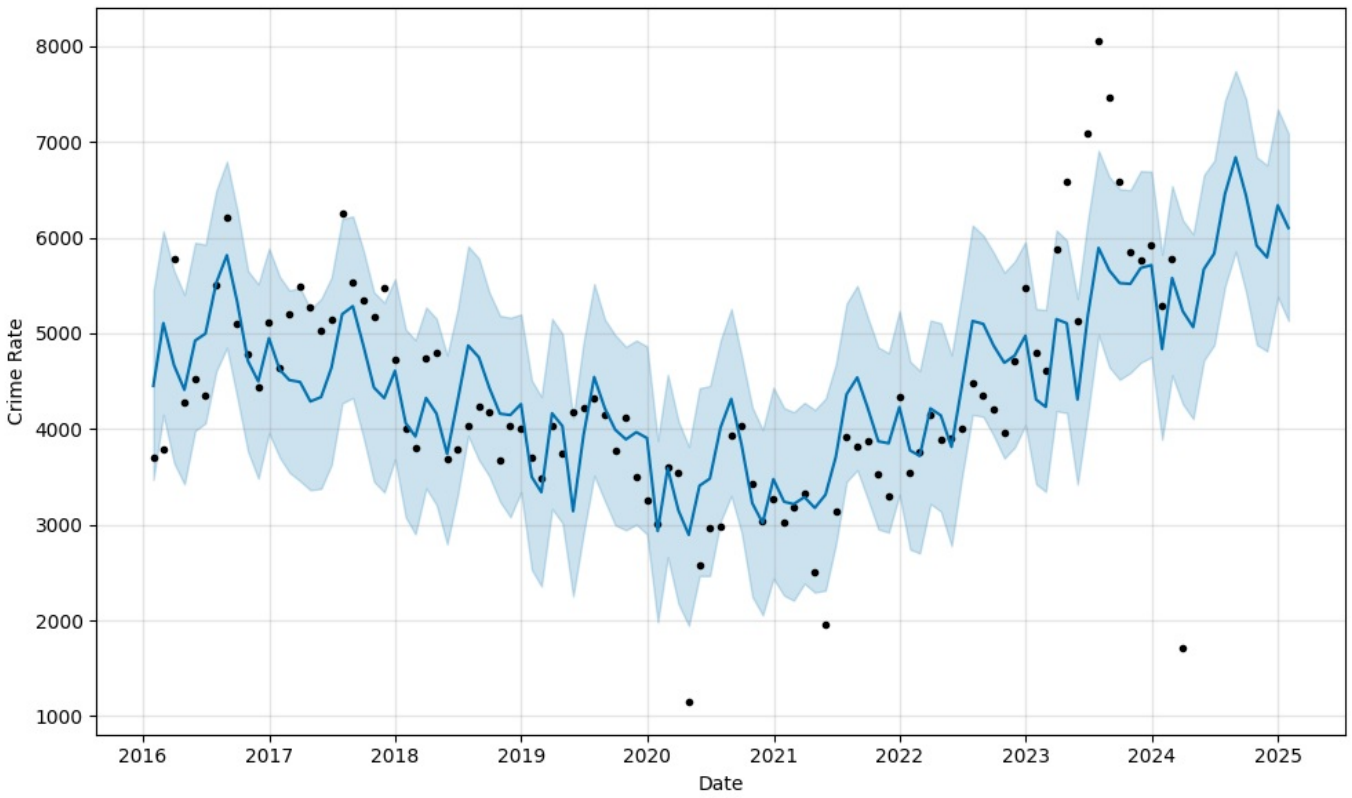
In [68]:
```
forcast
```

Out[68]:

	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_
0	2016-01-31	5089.477988	3471.386498	5445.980887	5089.477988	5089.477988	-637.602937	-637.602937	-637.(
1	2016-02-29	5059.371462	4149.768580	6070.180994	5059.371462	5059.371462	48.420819	48.420819	48.4
2	2016-03-31	5027.188624	3645.264648	5653.590453	5027.188624	5027.188624	-360.178234	-360.178234	-360.1
3	2016-04-30	4996.043942	3423.228873	5405.018149	4996.043942	4996.043942	-584.951401	-584.951401	-584.9
4	2016-05-31	4963.861104	3982.452710	5948.214812	4963.861104	4963.861104	-39.650936	-39.650936	-39.(
...	...	...	...	...	...	...	...	...	
104	2024-09-30	5949.705039	5430.034483	7448.364074	5926.953338	5969.957442	492.038600	492.038600	492.(
105	2024-10-31	6010.413791	4881.631640	6842.539143	5981.698242	6037.869146	-92.769991	-92.769991	-92.1
106	2024-11-30	6069.164196	4813.973293	6757.723939	6034.169415	6104.923125	-275.290559	-275.290559	-275.2
107	2024-12-31	6129.872948	5381.776803	7342.760672	6087.691661	6172.511459	207.534892	207.534892	207.5
108	2025-01-31	6190.581700	5131.567871	7090.101512	6142.476610	6240.569713	-89.054285	-89.054285	-89.(

109 rows × 16 columns

In [69]:
```
figure = m_BengaluruCity.plot(forcast, xlabel='Date', ylabel='Crime Rate')
```



In [70]:
```
df_BengaluruDist = df[df["District_Name"]=="Bengaluru Dist"]
```

In [71]:
```
df_BengaluruDist.head()
```

Out[71]:

	District_Name	FIR_Reg_DateTime
568820	Bengaluru Dist	2016-01-01 12:30:00
568821	Bengaluru Dist	2016-01-04 09:30:00
568822	Bengaluru Dist	2016-01-04 19:00:00
568823	Bengaluru Dist	2016-01-05 14:00:00
568824	Bengaluru Dist	2016-01-06 11:00:00

In [72]:
```
df_BengaluruDist.tail()
```

Out[72]:

	District_Name	FIR_Reg_DateTime
633847	Bengaluru Dist	2024-02-28 23:00:00
633848	Bengaluru Dist	2024-03-01 16:00:00
633849	Bengaluru Dist	2024-03-06 15:15:00
633850	Bengaluru Dist	2024-03-09 19:00:00
633851	Bengaluru Dist	2024-03-10 13:00:00

In [73]:
```
df_BengaluruDist.sample(10)
```

Out[73]:

	District_Name	FIR_Reg_DateTime
575870	Bengaluru Dist	2024-01-02 14:00:00
620722	Bengaluru Dist	2022-03-04 14:10:00
597706	Bengaluru Dist	2017-07-27 10:30:00
611795	Bengaluru Dist	2019-03-02 06:30:00
625039	Bengaluru Dist	2016-08-29 11:30:00
583713	Bengaluru Dist	2016-03-13 13:00:00
610944	Bengaluru Dist	2016-07-06 12:45:00
577090	Bengaluru Dist	2018-08-26 21:30:00
625484	Bengaluru Dist	2017-09-16 10:20:00
590531	Bengaluru Dist	2023-10-07 15:45:00

In [74]:
```
df_BengaluruDist.shape
```

Out[74]:
```
(65032, 2)
```

In [75]:
```
df_BengaluruDist.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 65032 entries, 568820 to 633851
Data columns (total 2 columns):
 #   Column            Non-Null Count  Dtype         
---  ------            --------------  -----         
 0   District_Name     65032 non-null  object        
 1   FIR_Reg_DateTime  65032 non-null  datetime64[ns]
dtypes: datetime64[ns](1), object(1)
memory usage: 1.5+ MB
```

In [76]:
```
df_BengaluruDist.isnull().sum()
```

Out[76]:
```
District_Name       0
FIR_Reg_DateTime    0
dtype: int64
```

In [77]:
```
df_BengaluruDist.columns
```

Out[77]:
```
Index(['District_Name', 'FIR_Reg_DateTime'], dtype='object')
```

In [78]:
```
df_BengaluruDist.index = pd.DatetimeIndex(df_BengaluruDist.FIR_Reg_DateTime)
```

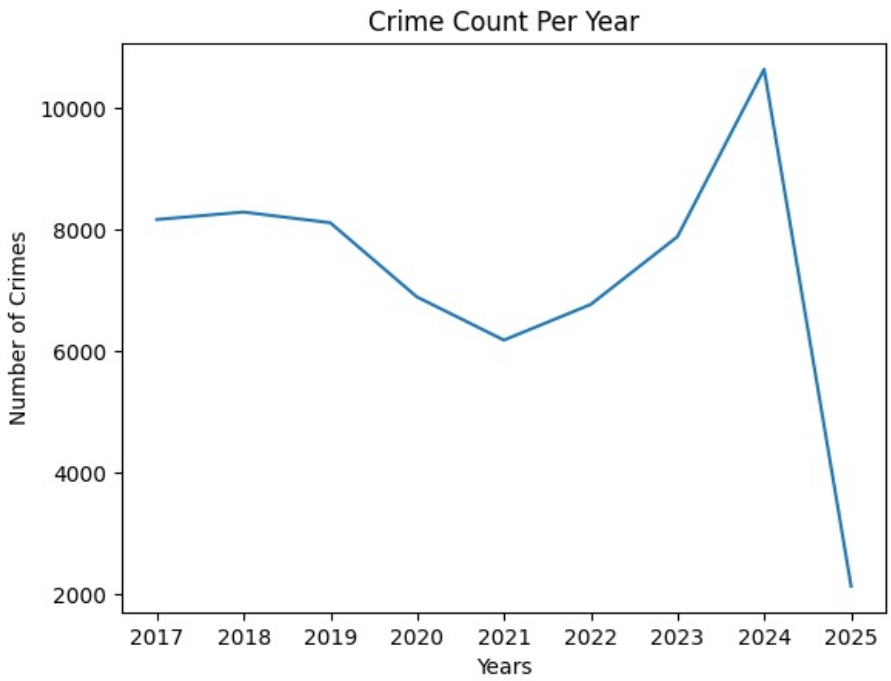
In [79]:
```
df_BengaluruDist.head()
```

Out[79]:

	District_Name	FIR_Reg_DateTime
FIR_Reg_DateTime		
2016-01-01 12:30:00	Bengaluru Dist	2016-01-01 12:30:00
2016-01-04 09:30:00	Bengaluru Dist	2016-01-04 09:30:00
2016-01-04 19:00:00	Bengaluru Dist	2016-01-04 19:00:00
2016-01-05 14:00:00	Bengaluru Dist	2016-01-05 14:00:00
2016-01-06 11:00:00	Bengaluru Dist	2016-01-06 11:00:00

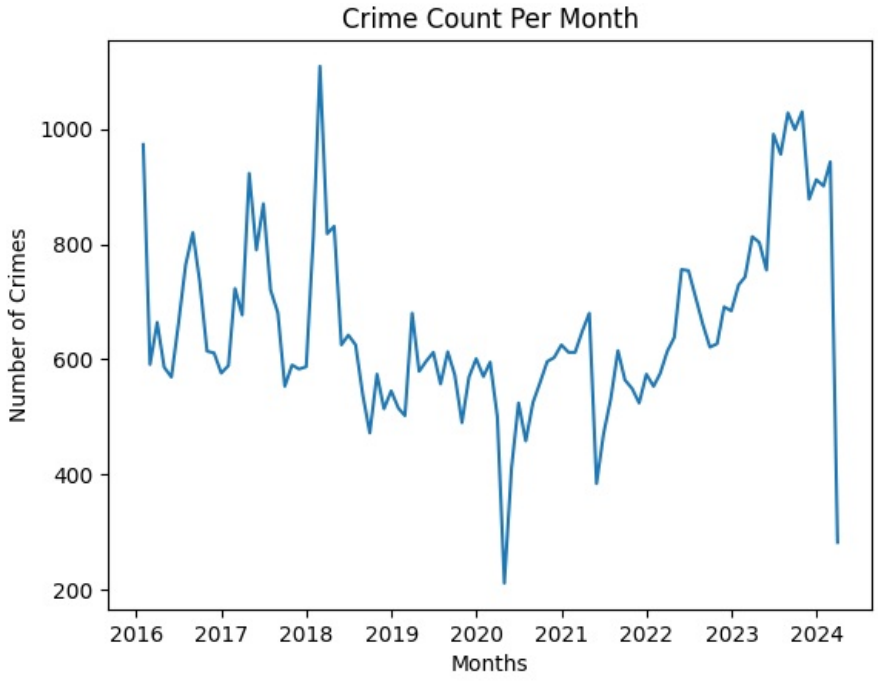
In [80]:
```
plt.plot(df_BengaluruDist.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[80]:
```
Text(0, 0.5, 'Number of Crimes')
```



In [81]:
```
plt.plot(df_BengaluruDist.resample('M').size())
plt.title("Crime Count Per Month")
plt.xlabel("Months")
plt.ylabel("Number of Crimes")
```

Out[81]:
```
Text(0, 0.5, 'Number of Crimes')
```



In [82]:
```
df_BengaluruDist_prophet = pd.DataFrame(df_BengaluruDist.resample('M').size().reset_index())
```

In [83]:
```
df_BengaluruDist_prophet.columns =['Date','Crime Count']
```

In [84]:
```
df_BengaluruDist_prophet.head()
```

Out[84]:

	Date	Crime Count
0	2016-01-31	973
1	2016-02-29	591
2	2016-03-31	664
3	2016-04-30	586
4	2016-05-31	569

In [85]:
```
df_BengaluruDist_prophet.shape
```

Out[85]: (99, 2)

In [86]:
```
df_BengaluruDist_prophet=df_BengaluruDist_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [87]:
```
df_BengaluruDist_prophet.head()
```

Out[87]:

	ds	y
0	2016-01-31	973
1	2016-02-29	591
2	2016-03-31	664
3	2016-04-30	586
4	2016-05-31	569

In [88]:
```
from prophet import Prophet
```

In [89]:
```
m_BengaluruDist = Prophet()
m_BengaluruDist.fit(df_BengaluruDist_prophet)
```

```
20:34:45 - cmdstanpy - INFO - Chain [1] start processing
20:34:45 - cmdstanpy - INFO - Chain [1] done processing
```

Out[89]: <prophet.forecaster.Prophet at 0x1df7ec161d0>

In [90]:
```
pred=m_BengaluruDist.make_future_dataframe(periods=24,freq='M')
forcast = m_BengaluruDist.predict(pred)
```

In [91]:
```
forcast
```

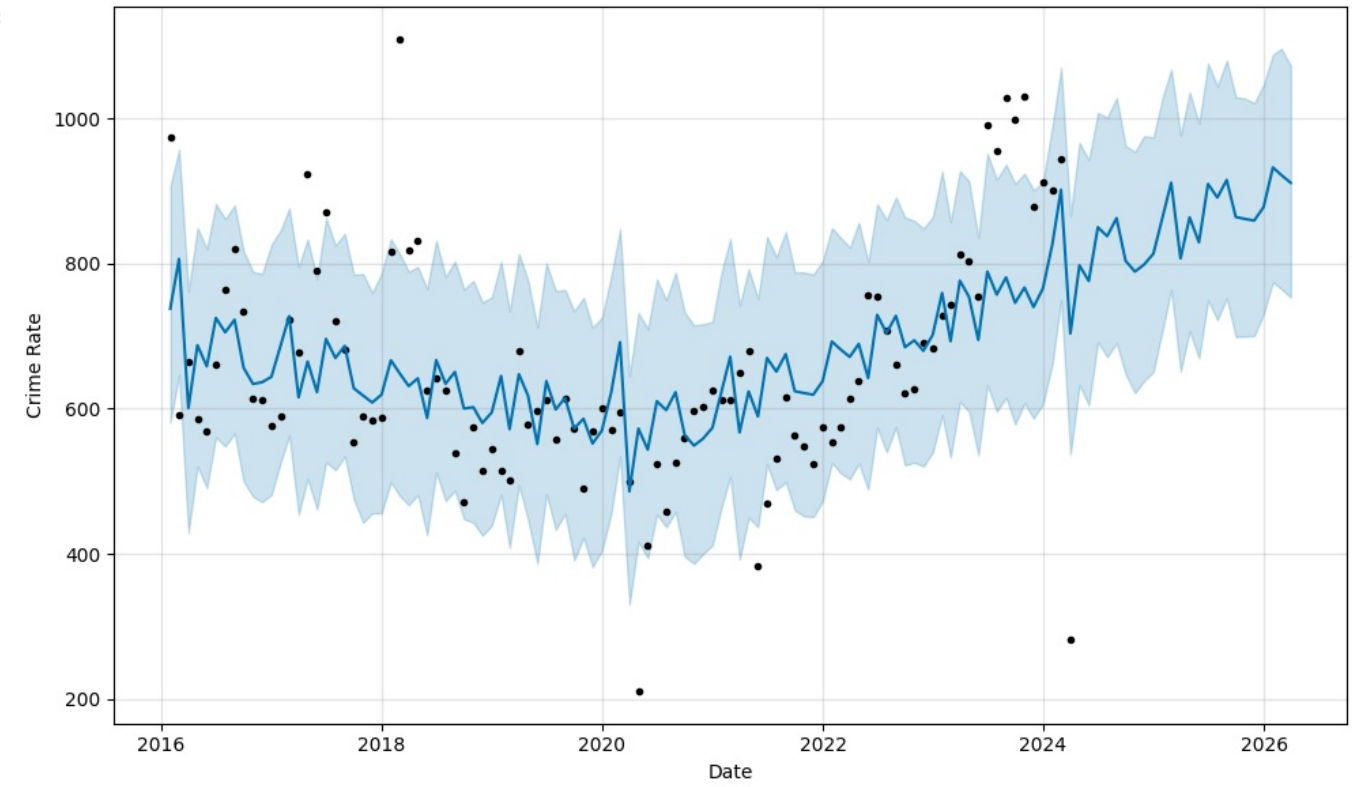
Out[91]:

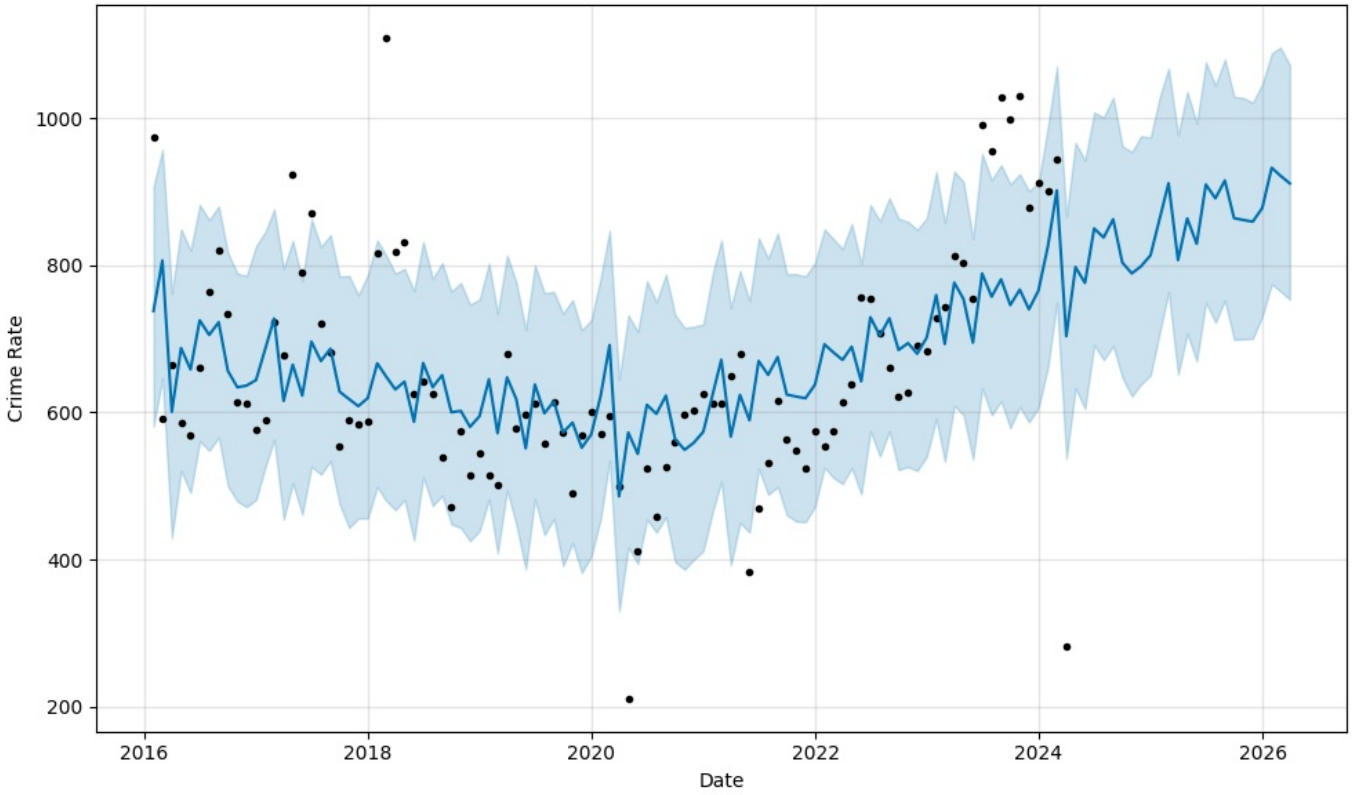
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	710.368841	580.759570	906.974998	710.368841	710.368841	27.274002	27.274002	27.274
1	2016-02-29	708.088203	646.336410	956.846384	708.088203	708.088203	98.142166	98.142166	98.142
2	2016-03-31	705.650280	429.023805	761.621205	705.650280	705.650280	-104.947896	-104.947896	-104.947
3	2016-04-30	703.291000	520.374315	848.949863	703.291000	703.291000	-15.968936	-15.968936	-15.968
4	2016-05-31	700.853077	490.589229	820.112294	700.853077	700.853077	-42.607391	-42.607391	-42.607
...	...	...	...	...	...	...	...	...	
118	2025-11-30	908.484291	700.129888	1021.130920	894.835519	921.057560	-49.419961	-49.419961	-49.419
119	2025-12-31	913.575847	730.142573	1046.201159	898.714397	927.075091	-36.031712	-36.031712	-36.031
120	2026-01-31	918.667402	774.184304	1087.876694	902.594344	932.915004	13.657400	13.657400	13.657
121	2026-02-28	923.266227	764.630840	1096.092998	905.967493	938.551703	-1.506033	-1.506033	-1.506
122	2026-03-31	928.357783	753.258987	1072.175606	910.569653	944.498472	-17.218365	-17.218365	-17.218

123 rows × 16 columns

In [92]:
```
m_BengaluruDist.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[92]:





In [93]:
```
import json
```

In [94]:
```
import json
from prophet.serialize import model_to_json, model_from_json
with open('prophet_model.json', 'w') as fout:
    json.dump(model_to_json(m_BengaluruDist), fout) # Save model
```

In [95]:
```
with open('prophet_model.json', 'r') as fin:
    m_BengaluruDist = model_from_json(json.load(fin)) # Load model
```

In [97]:
```
x = int(input("Enter Numbers Months to forecast"))
```

In [98]:
```
pred = m_BengaluruDist.make_future_dataframe(periods=x ,freq = "M")
forcast = m_BengaluruDist.predict(pred)
```

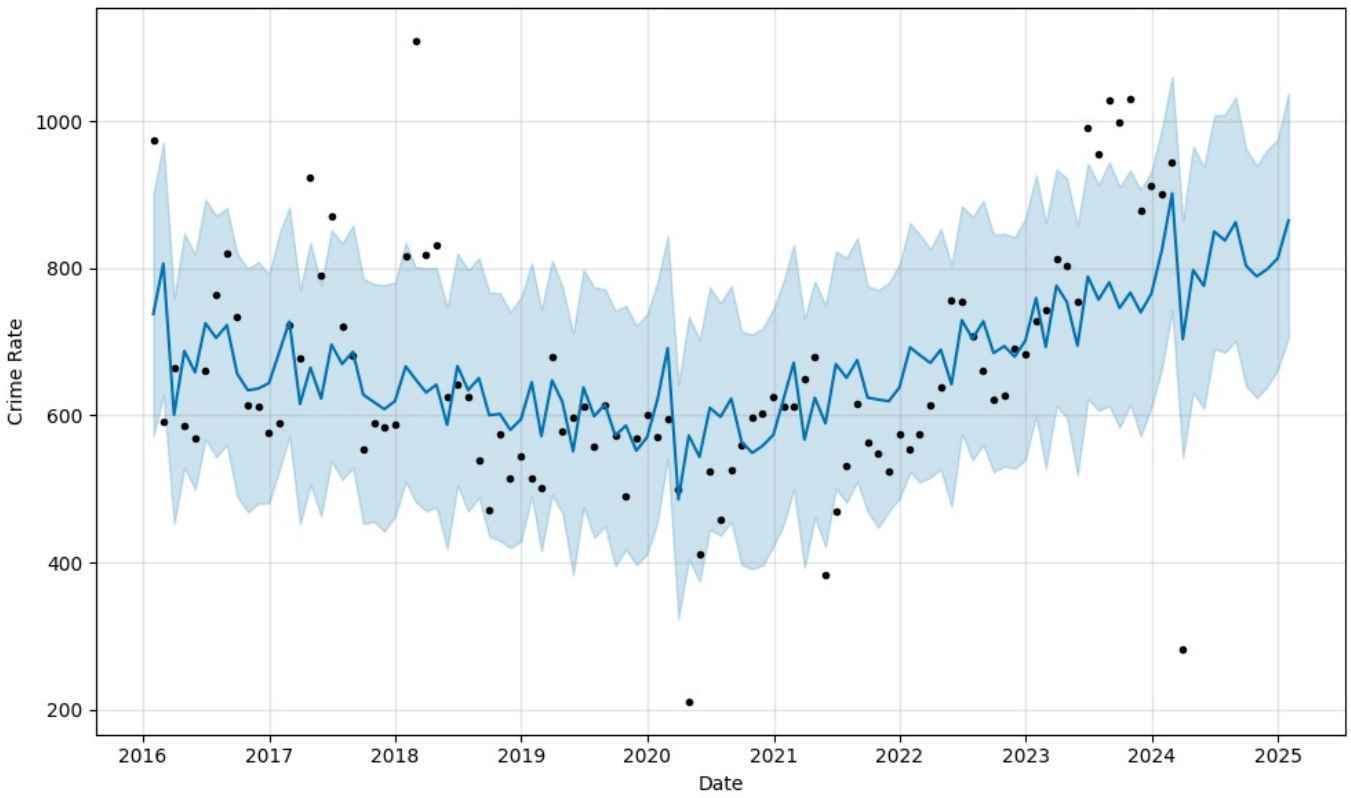
In [99]:
```
forcast
```

Out[99]:

	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	710.368841	570.899148	900.584384	710.368841	710.368841	27.274002	27.274002	27.274
1	2016-02-29	708.088203	627.907622	970.956300	708.088203	708.088203	98.142166	98.142166	98.142
2	2016-03-31	705.650280	452.113592	759.791845	705.650280	705.650280	-104.947896	-104.947896	-104.947
3	2016-04-30	703.291000	529.216190	847.182362	703.291000	703.291000	-15.968936	-15.968936	-15.968
4	2016-05-31	700.853077	499.101039	819.258729	700.853077	700.853077	-42.607391	-42.607391	-42.607
...	...	...	...	...	...	...	...	...	
104	2024-09-30	838.516459	639.684303	961.693073	836.518320	840.502334	-34.997817	-34.997817	-34.997
105	2024-10-31	843.608014	623.697895	939.562914	841.096978	846.138178	-54.824364	-54.824364	-54.824
106	2024-11-30	848.535326	638.686294	960.456804	845.458665	851.643364	-50.140092	-50.140092	-50.140
107	2024-12-31	853.626882	661.831299	975.193701	849.720048	857.478270	-40.267843	-40.267843	-40.267
108	2025-01-31	858.718438	705.879922	1037.751787	854.181142	863.400396	5.903776	5.903776	5.903

109 rows × 16 columns

In [100…
```
figure = m_BengaluruDist.plot(forcast, xlabel='Date', ylabel='Crime Rate')
```



In [101…
```
df_Tumakuru = df[df["District_Name"]=="Tumakuru"]
```

In [102…
```
df_Tumakuru.head()
```

Out[102…

	District_Name	FIR_Reg_DateTime
1484396	Tumakuru	2016-01-07 16:30:00
1484397	Tumakuru	2016-01-08 14:30:00
1484398	Tumakuru	2016-01-09 08:30:00
1484399	Tumakuru	2016-01-09 18:15:00
1484400	Tumakuru	2016-01-12 17:15:00

In [103…
```
df_Tumakuru.tail()
```

Out[103…

	District_Name	FIR_Reg_DateTime
1546911	Tumakuru	2024-02-26 13:30:00
1546912	Tumakuru	2024-03-01 15:30:00
1546913	Tumakuru	2024-03-01 18:30:00
1546914	Tumakuru	2024-03-04 17:30:00
1546915	Tumakuru	2024-03-08 12:45:00

In [104…
```
df_Tumakuru.sample(10)
```

Out[104…

	District_Name	FIR_Reg_DateTime
1504076	Tumakuru	2018-03-13 11:30:00
1546027	Tumakuru	2024-01-09 15:00:00
1545126	Tumakuru	2020-12-24 20:30:00
1539540	Tumakuru	2023-04-06 20:00:00
1530259	Tumakuru	2023-10-14 18:40:00
1533269	Tumakuru	2017-09-16 10:30:00
1503282	Tumakuru	2023-06-19 14:15:00
1487617	Tumakuru	2024-02-03 17:00:00
1500921	Tumakuru	2017-11-11 14:30:00
1494203	Tumakuru	2020-05-21 18:00:00

In [105…
```
df_Tumakuru.shape
```

Out[105… (62520, 2)

In [106…
```
df_Tumakuru.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 62520 entries, 1484396 to 1546915
Data columns (total 2 columns):
 #   Column            Non-Null Count  Dtype         
---  ------            --------------  -----         
 0   District_Name     62520 non-null  object        
 1   FIR_Reg_DateTime  62520 non-null  datetime64[ns]
dtypes: datetime64[ns](1), object(1)
memory usage: 1.4+ MB
```

In [107…
```
df_Tumakuru.isnull().sum()
```

Out[107…
```
District_Name       0
FIR_Reg_DateTime    0
dtype: int64
```

In [108…
```
df_Tumakuru.columns
```

Out[108… `Index(['District_Name', 'FIR_Reg_DateTime'], dtype='object')`

In [109…
```
df_Tumakuru.index = pd.DatetimeIndex(df_Tumakuru.FIR_Reg_DateTime)
```

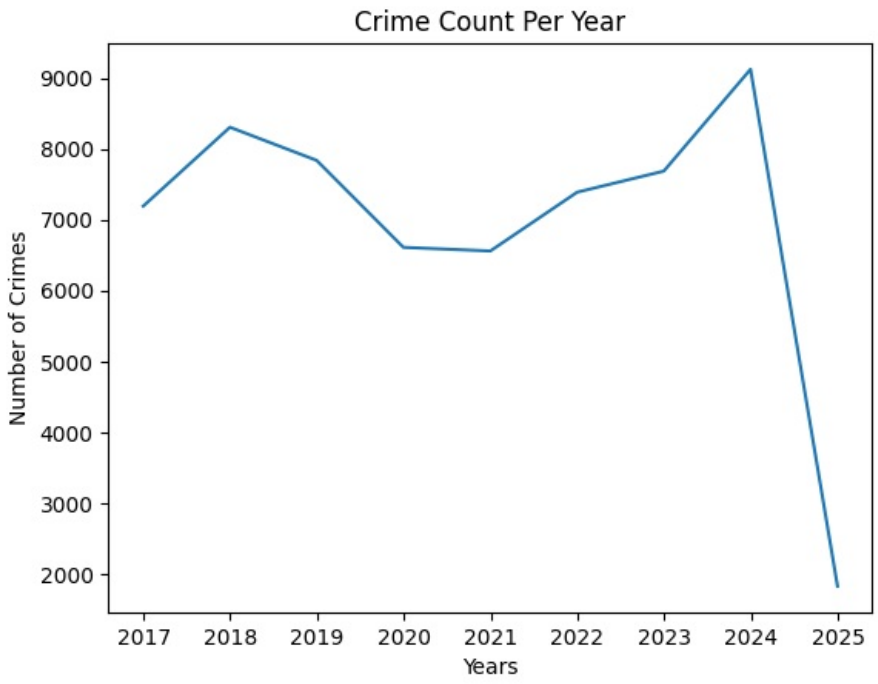
In [112…
```
df_Tumakuru.head()
```

Out[112…

	District_Name	FIR_Reg_DateTime
FIR_Reg_DateTime		
2016-01-07 16:30:00	Tumakuru	2016-01-07 16:30:00
2016-01-08 14:30:00	Tumakuru	2016-01-08 14:30:00
2016-01-09 08:30:00	Tumakuru	2016-01-09 08:30:00
2016-01-09 18:15:00	Tumakuru	2016-01-09 18:15:00
2016-01-12 17:15:00	Tumakuru	2016-01-12 17:15:00

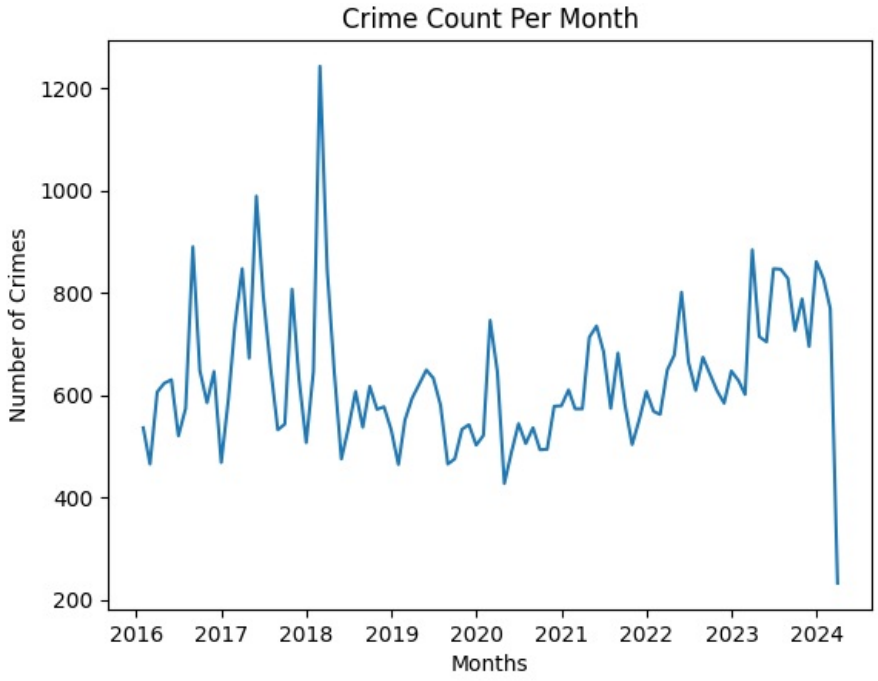
In [111…
```
plt.plot(df_Tumakuru.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[111… `Text(0, 0.5, 'Number of Crimes')`



In [113…
```
plt.plot(df_Tumakuru.resample('M').size())
plt.title("Crime Count Per Month")
plt.xlabel("Months")
plt.ylabel("Number of Crimes")
```

Out[113… `Text(0, 0.5, 'Number of Crimes')`



In [114…
```
df_Tumakuru_prophet = pd.DataFrame(df_Tumakuru.resample('M').size().reset_index())
```

In [115…
```
df_Tumakuru_prophet.columns =['Date','Crime Count']
```

In [116…
```
df_Tumakuru_prophet.head()
```

Out[116…

	Date	Crime Count
0	2016-01-31	536
1	2016-02-29	465
2	2016-03-31	606
3	2016-04-30	623
4	2016-05-31	630

In [117…
```
df_Tumakuru_prophet.shape
```

Out[117… (99, 2)

In [118…
```
df_Tumakuru_prophet=df_Tumakuru_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [119…
```
df_Tumakuru_prophet.head()
```

Out[119…

	ds	y
0	2016-01-31	536
1	2016-02-29	465
2	2016-03-31	606
3	2016-04-30	623
4	2016-05-31	630

In [120…
```
from prophet import Prophet
```

In [121…
```
m_Tumakuru = Prophet()
m_Tumakuru.fit(df_Tumakuru_prophet)
```

```
20:42:11 - cmdstanpy - INFO - Chain [1] start processing
20:42:11 - cmdstanpy - INFO - Chain [1] done processing
```

Out[121… <prophet.forecaster.Prophet at 0x1df7f3b68d0>

In [122…
```
pred=m_Tumakuru.make_future_dataframe(periods=24,freq='M')
forcast = m_Tumakuru.predict(pred)
```

In [123…
```
forcast
```

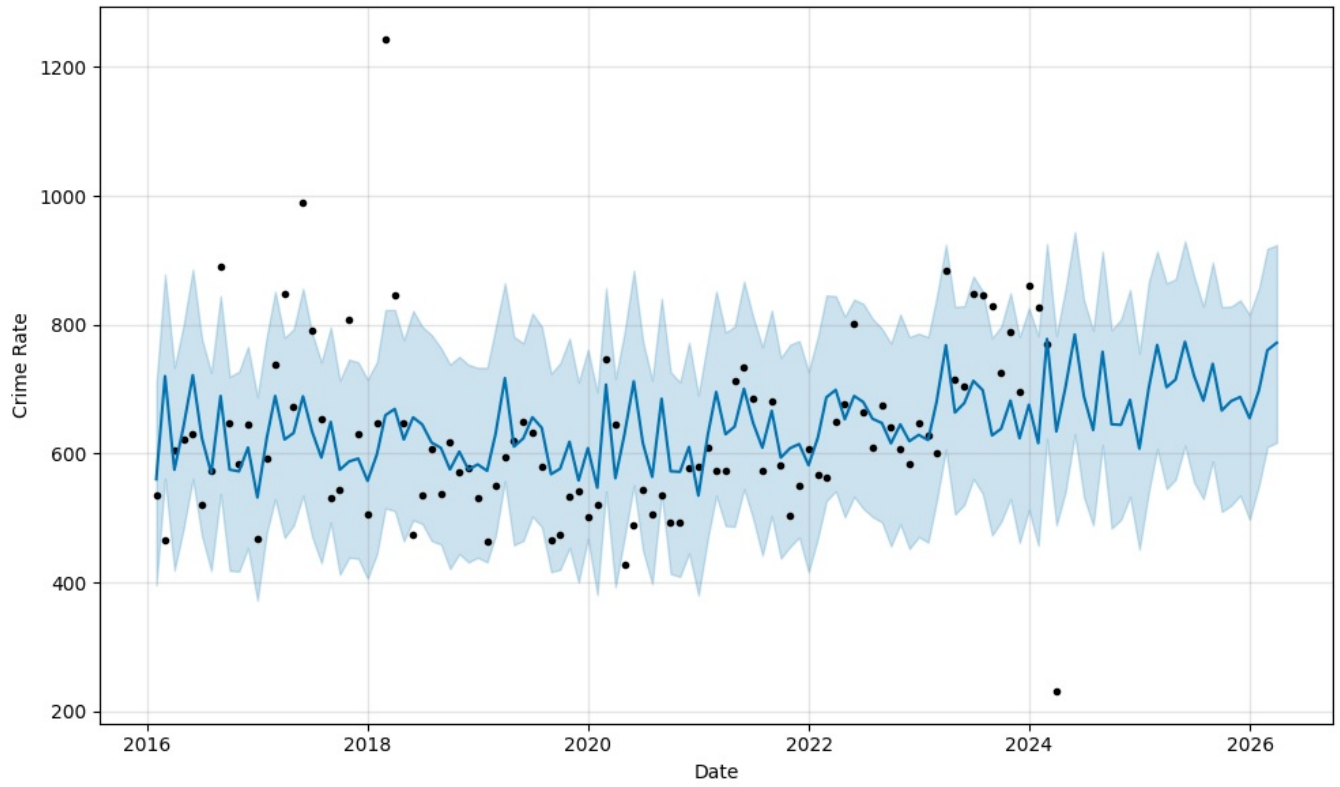
Out[123…

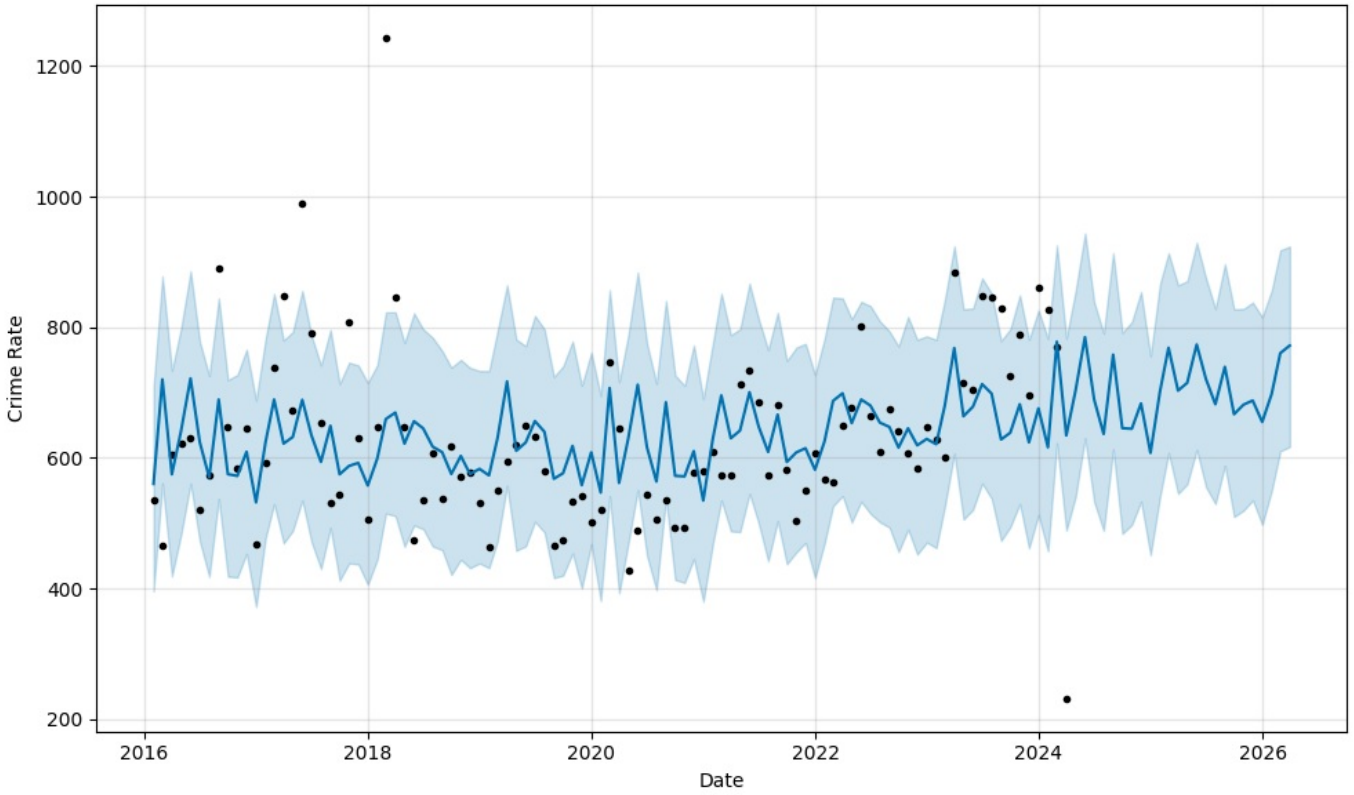
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	620.309206	396.054591	708.856740	620.309206	620.309206	-59.915847	-59.915847	-59.915
1	2016-02-29	620.045539	561.585026	878.125802	620.045539	620.045539	100.248174	100.248174	100.248
2	2016-03-31	619.763689	419.085132	733.485629	619.763689	619.763689	-44.705585	-44.705585	-44.705
3	2016-04-30	619.490931	482.970116	798.989191	619.490931	619.490931	21.871305	21.871305	21.871
4	2016-05-31	619.209080	560.830801	885.658851	619.209080	619.209080	102.578981	102.578981	102.578
...	...	...	...	...	...	...	...	...	
118	2025-11-30	709.717870	535.322350	838.098399	706.757869	712.680333	-21.773832	-21.773832	-21.773
119	2025-12-31	711.267775	497.933201	815.504587	708.100491	714.459969	-56.115103	-56.115103	-56.115
120	2026-01-31	712.817681	550.435134	856.796098	709.399025	716.305827	-14.575173	-14.575173	-14.575
121	2026-02-28	714.217595	610.392269	917.989646	710.612051	717.981487	46.197968	46.197968	46.197
122	2026-03-31	715.767501	617.151736	923.708800	711.945240	719.871547	56.311041	56.311041	56.311

123 rows × 16 columns

In [124…
```
m_Tumakuru.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[124…





In [125…
```
import json
```

In [126…
```
import json
from prophet.serialize import model_to_json, model_from_json
with open('prophet_model.json', 'w') as fout:
    json.dump(model_to_json(m_Tumakuru), fout) # Save model
```

In [127…
```
with open('prophet_model.json', 'r') as fin:
    m_Tumakuru = model_from_json(json.load(fin)) # Load model
```

In [128…
```
x = int(input("Enter Numbers Months to forecast"))
```

In [129…
```
pred = m_Tumakuru.make_future_dataframe(periods=x ,freq = "M")
forcast = m_Tumakuru.predict(pred)
```

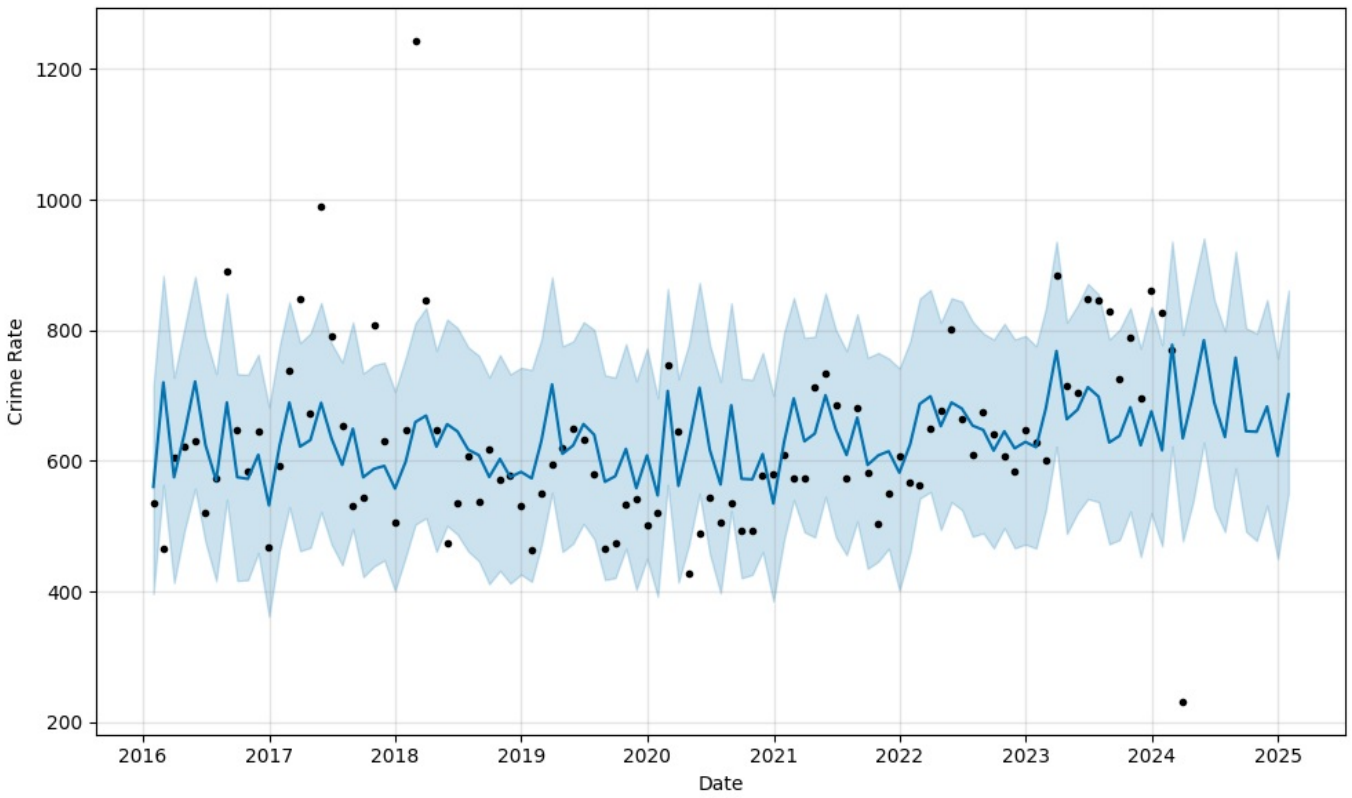
In [130…
```
forcast
```

Out[130…

	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	620.309206	396.591197	714.860845	620.309206	620.309206	-59.915847	-59.915847	-59.915
1	2016-02-29	620.045539	563.526929	883.375279	620.045539	620.045539	100.248174	100.248174	100.248
2	2016-03-31	619.763689	413.484983	727.654906	619.763689	619.763689	-44.705585	-44.705585	-44.705
3	2016-04-30	619.490931	493.207353	803.464088	619.490931	619.490931	21.871305	21.871305	21.871
4	2016-05-31	619.209080	557.812952	882.145766	619.209080	619.209080	102.578981	102.578981	102.578
...	...	...	...	...	...	...	...	...	
104	2024-09-30	688.419170	492.011453	803.534114	687.980874	688.893032	-42.847620	-42.847620	-42.847
105	2024-10-31	689.969075	477.885925	795.015780	689.404806	690.583588	-45.216667	-45.216667	-45.216
106	2024-11-30	691.468984	531.941563	846.571148	690.761755	692.226575	-7.996100	-7.996100	-7.996
107	2024-12-31	693.018889	450.109246	757.803673	692.134592	693.962227	-85.238646	-85.238646	-85.238
108	2025-01-31	694.568795	550.453970	860.963128	693.538993	695.650405	7.157297	7.157297	7.157

109 rows × 16 columns

In [131…
```
figure = m_Tumakuru.plot(forcast, xlabel='Date', ylabel='Crime Rate')
```

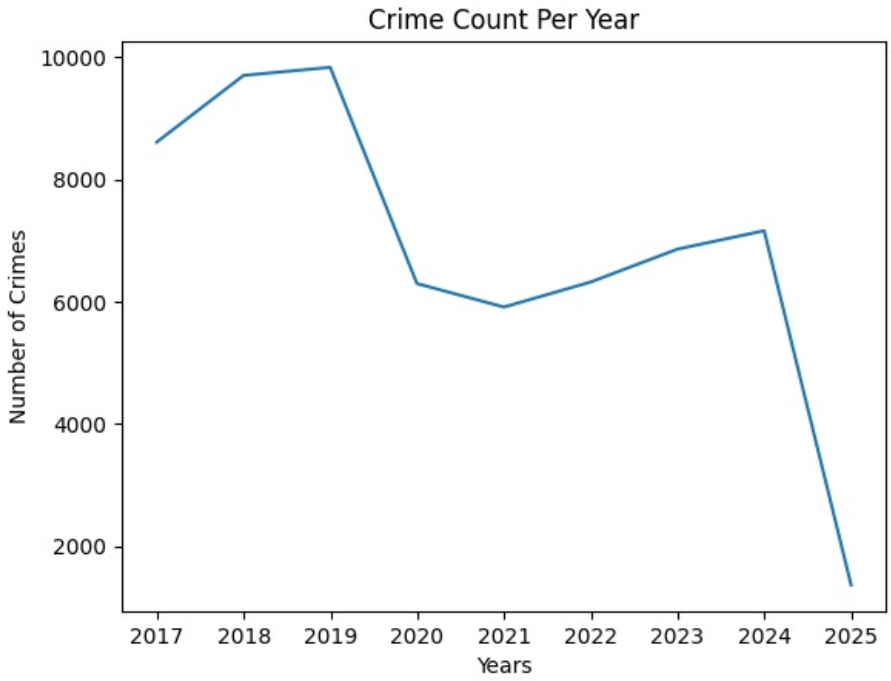


In [132…
```
df_Shivamogga = df[df["District_Name"]=="Shivamogga"]
```

In [133…
```
df_Shivamogga.index = pd.DatetimeIndex(df_Shivamogga.FIR_Reg_DateTime)
```

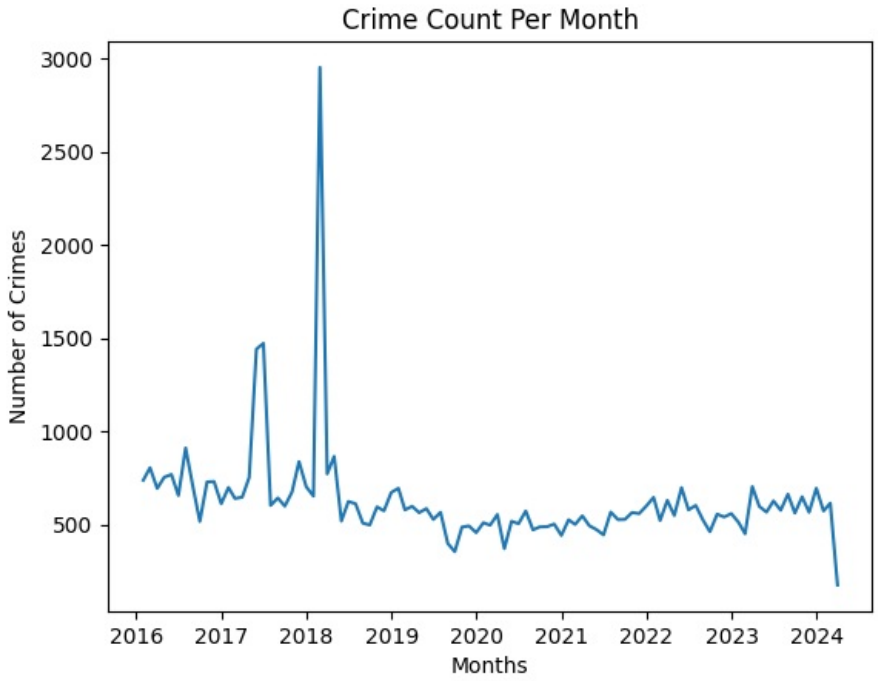
In [134…
```
plt.plot(df_Shivamogga.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[134… `Text(0, 0.5, 'Number of Crimes')`



In [135…
```
plt.plot(df_Shivamogga.resample('M').size())
plt.title("Crime Count Per Month")
plt.xlabel("Months")
plt.ylabel("Number of Crimes")
```

Out[135… Text(0, 0.5, 'Number of Crimes')



In [136…
```
df_Shivamogga_prophet = pd.DataFrame(df_Shivamogga.resample('M').size().reset_index())
```

In [137…
```
df_Shivamogga_prophet.columns =['Date','Crime Count']
```

In [138…
```
df_Shivamogga_prophet=df_Shivamogga_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [139…
```
m_Shivamogga = Prophet()
m_Shivamogga.fit(df_Shivamogga_prophet)
```

```
21:15:02 - cmdstanpy - INFO - Chain [1] start processing
21:15:02 - cmdstanpy - INFO - Chain [1] done processing
```

Out[139… <prophet.forecaster.Prophet at 0x1df7e9dd510>

In [140…
```
pred=m_Shivamogga.make_future_dataframe(periods=24,freq='M')
```

```
forcast = m_Shivamogga.predict(pred)
```

In [141…
```
forcast
```

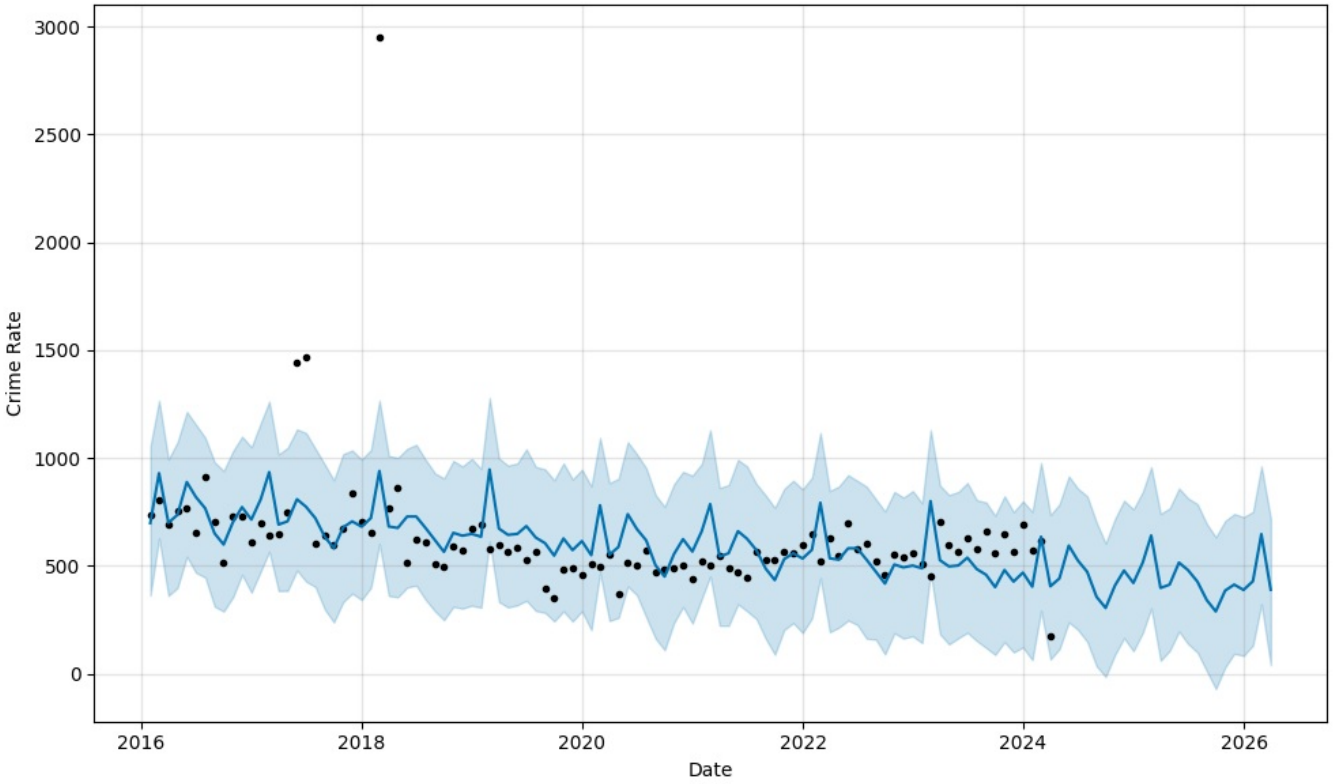
Out[141…

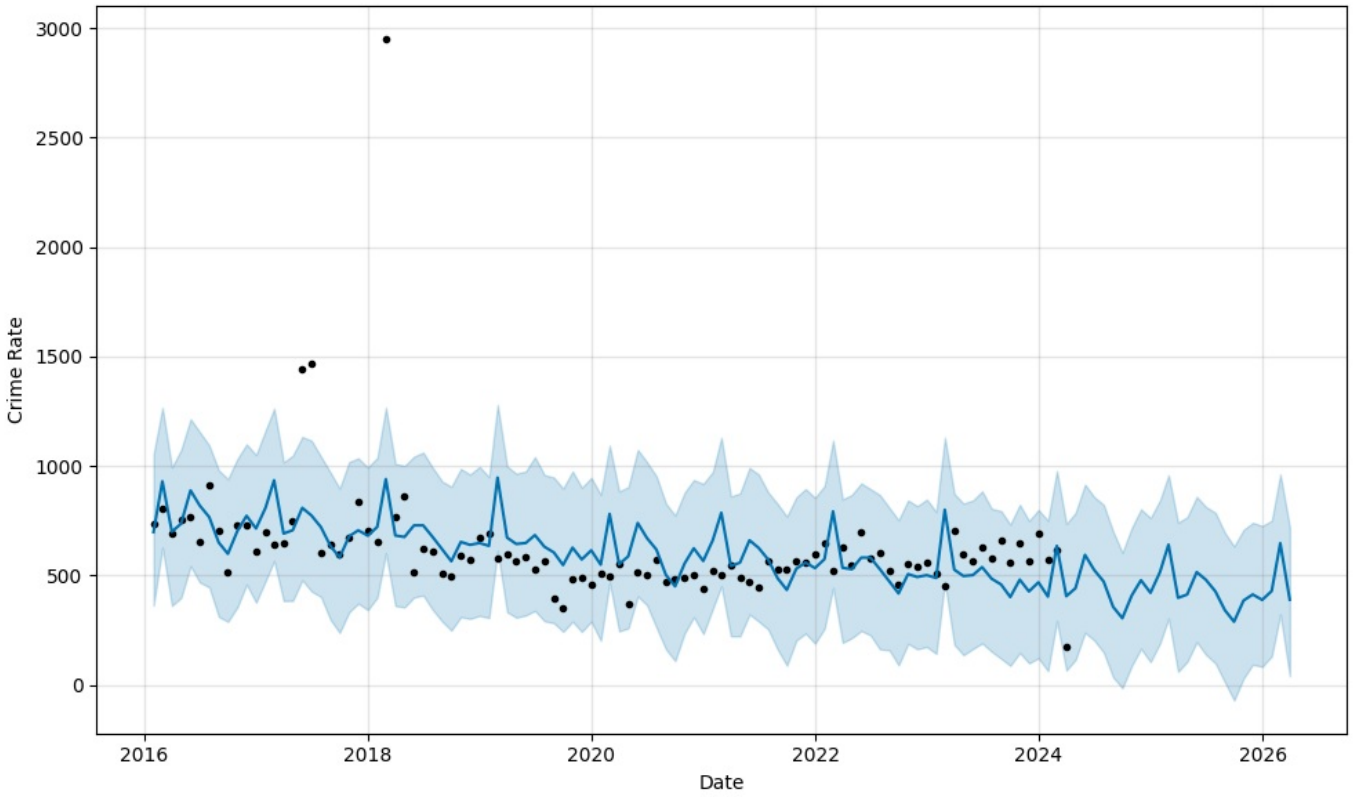
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	750.126494	362.916414	1054.487573	750.126494	750.126494	-51.457291	-51.457291	-51.457
1	2016-02-29	747.171177	628.146987	1266.524167	747.171177	747.171177	183.211371	183.211371	183.211
2	2016-03-31	744.012046	361.908612	994.218680	744.012046	744.012046	-43.617379	-43.617379	-43.617
3	2016-04-30	740.954822	398.439025	1074.132503	740.954822	740.954822	-4.182750	-4.182750	-4.182
4	2016-05-31	737.795690	543.316395	1214.689115	737.795690	737.795690	151.083218	151.083218	151.083
...	...	...	...	...	...	...	...	...	
118	2025-11-30	388.863172	93.632012	741.667926	388.734296	388.981935	24.891893	24.891893	24.891
119	2025-12-31	385.777107	83.844387	726.812826	385.638157	385.905820	3.236552	3.236552	3.236
120	2026-01-31	382.691042	131.850010	751.217374	382.539268	382.831703	46.703383	46.703383	46.703
121	2026-02-28	379.903629	323.579965	962.357803	379.741932	380.056179	268.025795	268.025795	268.025
122	2026-03-31	376.817564	40.465916	720.251230	376.642711	376.979730	13.278633	13.278633	13.278

123 rows × 16 columns

In [142…
```
m_Shivamogga.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[142…





In [143…
```
import json
from prophet.serialize import model_to_json, model_from_json
with open('prophet_model.json', 'w') as fout:
    json.dump(model_to_json(m_Shivamogga), fout) # Save model
```

In [144…
```
with open('prophet_model.json', 'r') as fin:
    m_Shivamogga = model_from_json(json.load(fin)) # Load model
```

In [146…
```
x = int(input("Enter Numbers Months to forecast"))
```

In [147…
```
pred = m_Shivamogga.make_future_dataframe(periods=x ,freq = "M")
forcast = m_Shivamogga.predict(pred)
```

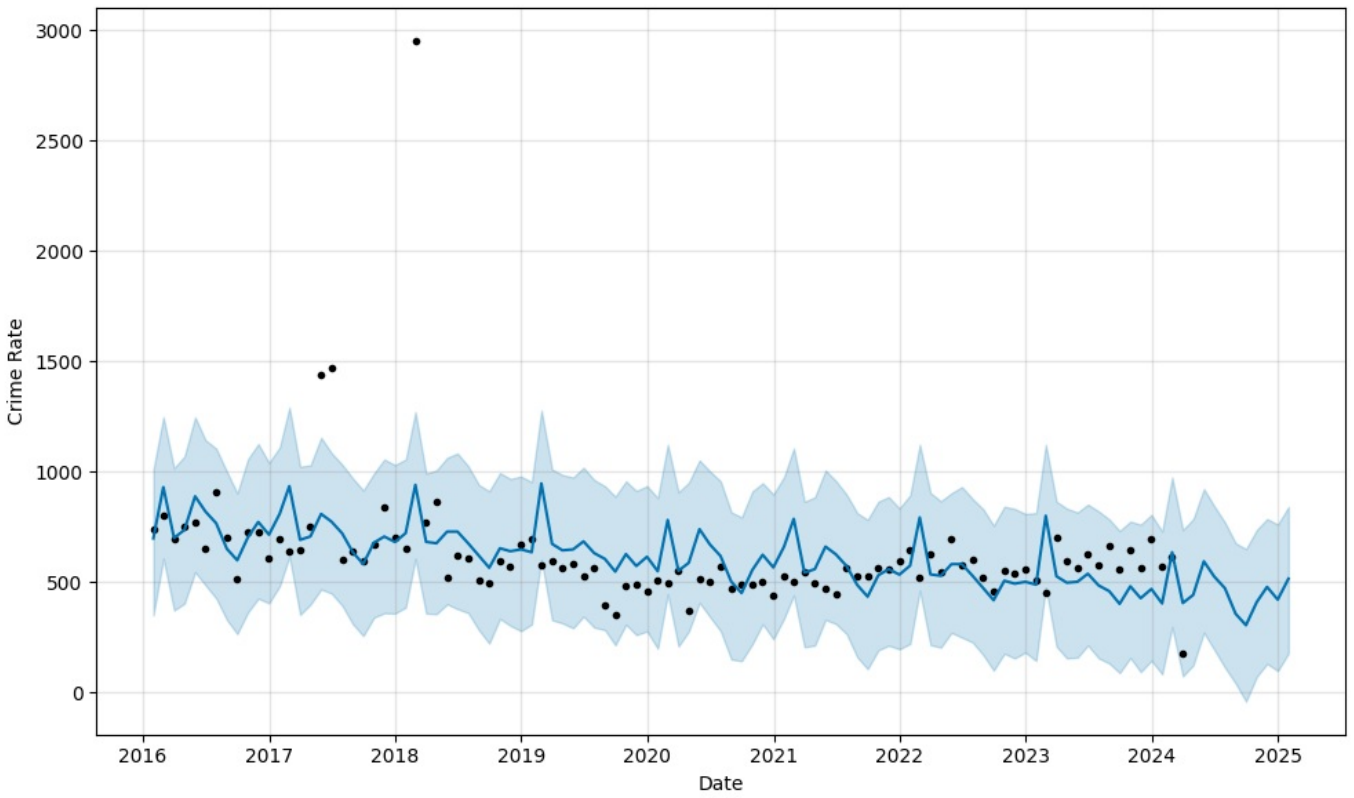
In [148…
```
forcast
```

Out[148…

	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	750.126494	350.362696	1010.576052	750.126494	750.126494	-51.457291	-51.457291	-51.457
1	2016-02-29	747.171177	606.877836	1247.395903	747.171177	747.171177	183.211371	183.211371	183.211
2	2016-03-31	744.012046	372.613531	1016.600469	744.012046	744.012046	-43.617379	-43.617379	-43.617
3	2016-04-30	740.954822	403.640525	1070.063678	740.954822	740.954822	-4.182750	-4.182750	-4.182
4	2016-05-31	737.795690	546.168118	1246.660294	737.795690	737.795690	151.083218	151.083218	151.083
...	...	...	...	...	...	...	...	...	
104	2024-09-30	431.271676	-39.784074	650.668130	431.254130	431.290921	-125.859683	-125.859683	-125.859
105	2024-10-31	428.185611	72.020236	737.266586	428.163204	428.211536	-18.599321	-18.599321	-18.599
106	2024-11-30	425.199096	130.873674	786.631126	425.170118	425.230894	53.703715	53.703715	53.703
107	2024-12-31	422.113032	98.583738	761.653036	422.077075	422.149551	-0.769285	-0.769285	-0.769
108	2025-01-31	419.026967	177.878305	841.080084	418.984857	419.070014	96.818154	96.818154	96.818

109 rows × 16 columns

In [149…
```
figure = m_Shivamogga.plot(forcast, xlabel='Date', ylabel='Crime Rate')
```

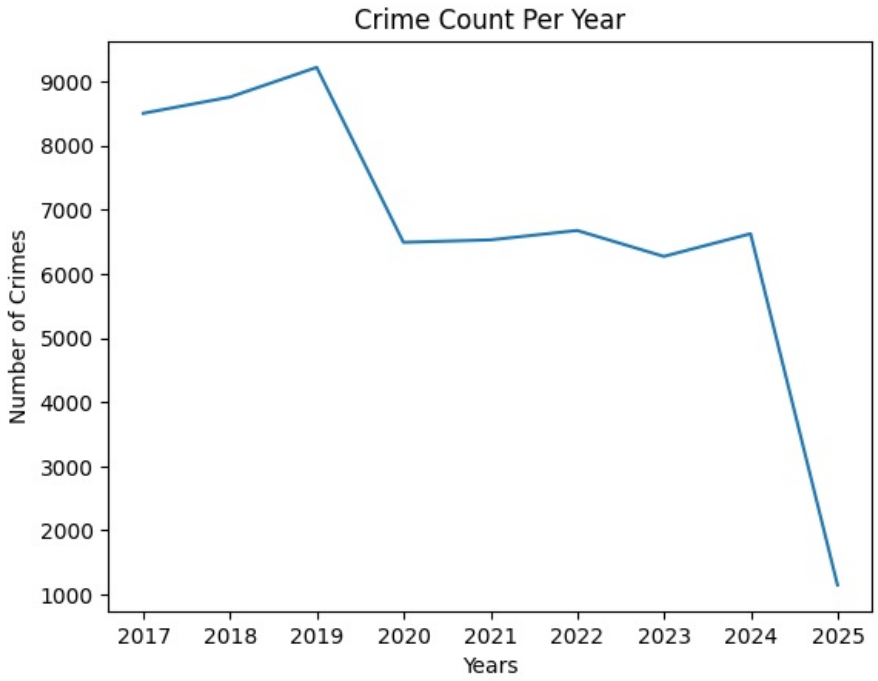


In [150…
```
df_Mandya = df[df["District_Name"]=="Mandya"]
```

In [151…
```
df_Mandya.index = pd.DatetimeIndex(df_Mandya.FIR_Reg_DateTime)
```

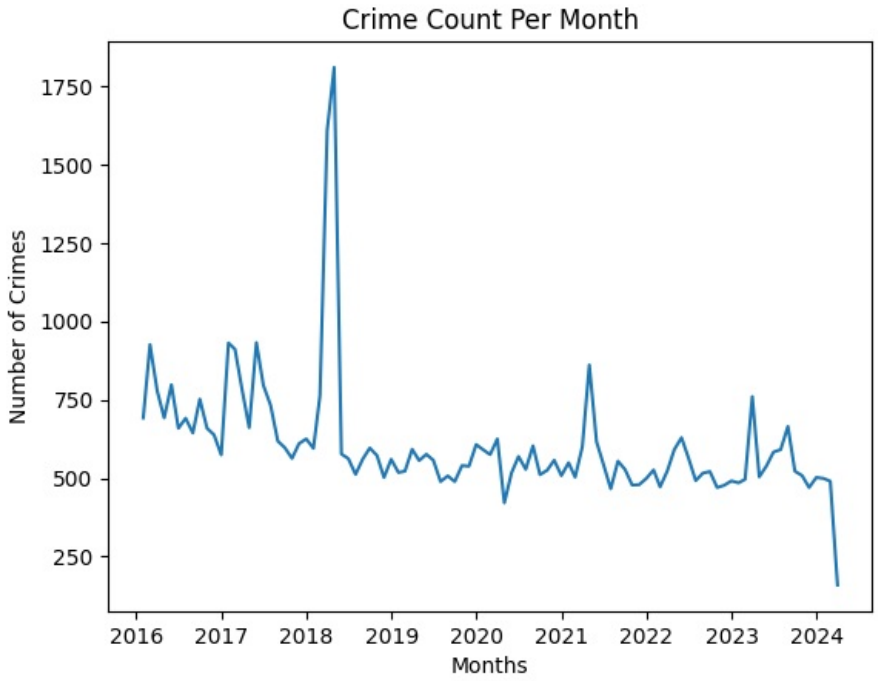
In [152…
```
plt.plot(df_Mandya.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[152… `Text(0, 0.5, 'Number of Crimes')`



In [153…
```
plt.plot(df_Mandya.resample('M').size())
plt.title("Crime Count Per Month")
plt.xlabel("Months")
plt.ylabel("Number of Crimes")
```

Out[153… Text(0, 0.5, 'Number of Crimes')



In [154…
```
df_Mandya_prophet = pd.DataFrame(df_Mandya.resample('M').size().reset_index())
```

In [155…
```
df_Mandya_prophet.columns =['Date','Crime Count']
```

In [156…
```
df_Mandya_prophet=df_Mandya_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [157…
```
m_Mandya = Prophet()
m_Mandya.fit(df_Mandya_prophet)
```

```
21:19:19 - cmdstanpy - INFO - Chain [1] start processing
21:19:20 - cmdstanpy - INFO - Chain [1] done processing
```

Out[157… <prophet.forecaster.Prophet at 0x1df816a9290>

In [158…
```
pred=m_Mandya.make_future_dataframe(periods=24,freq='M')
```

```
forcast = m_Mandya.predict(pred)
```

In [159…
```
forcast
```

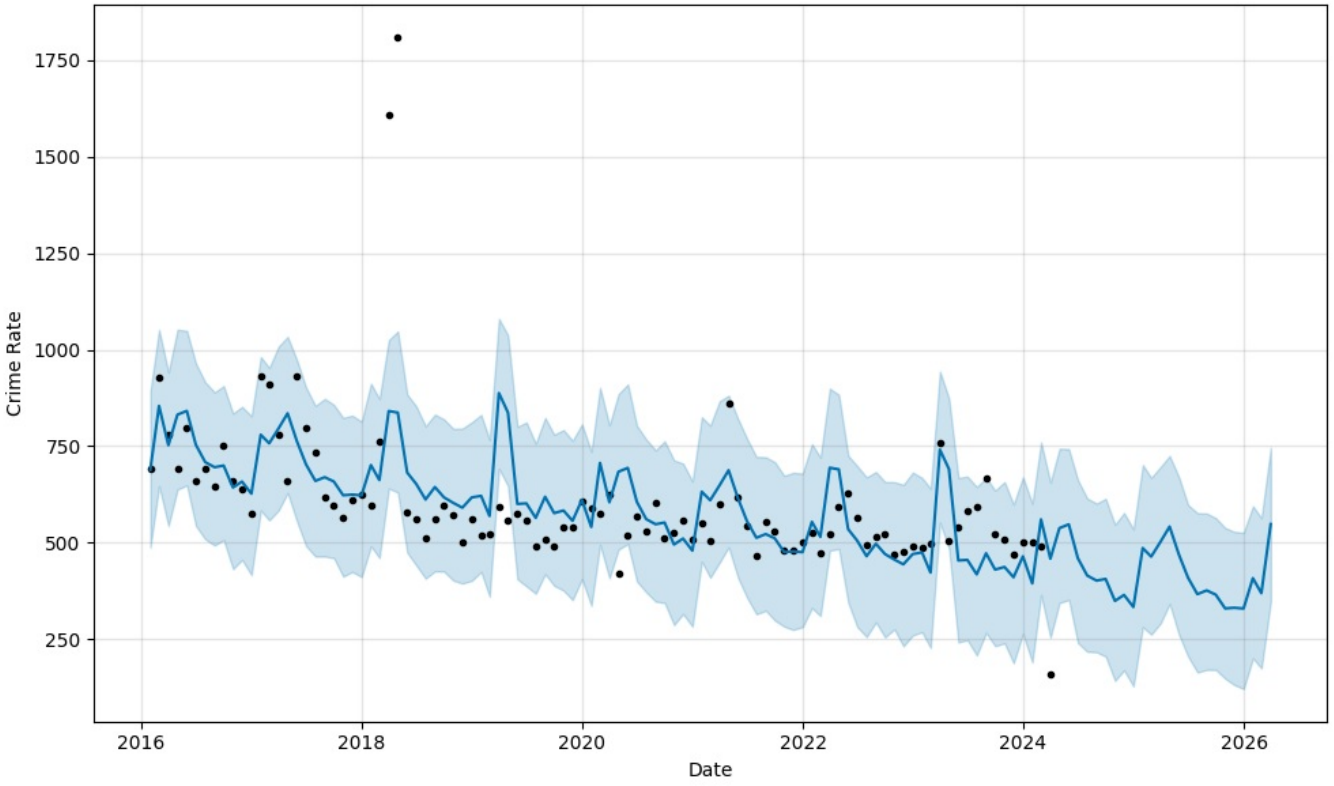
Out[159…

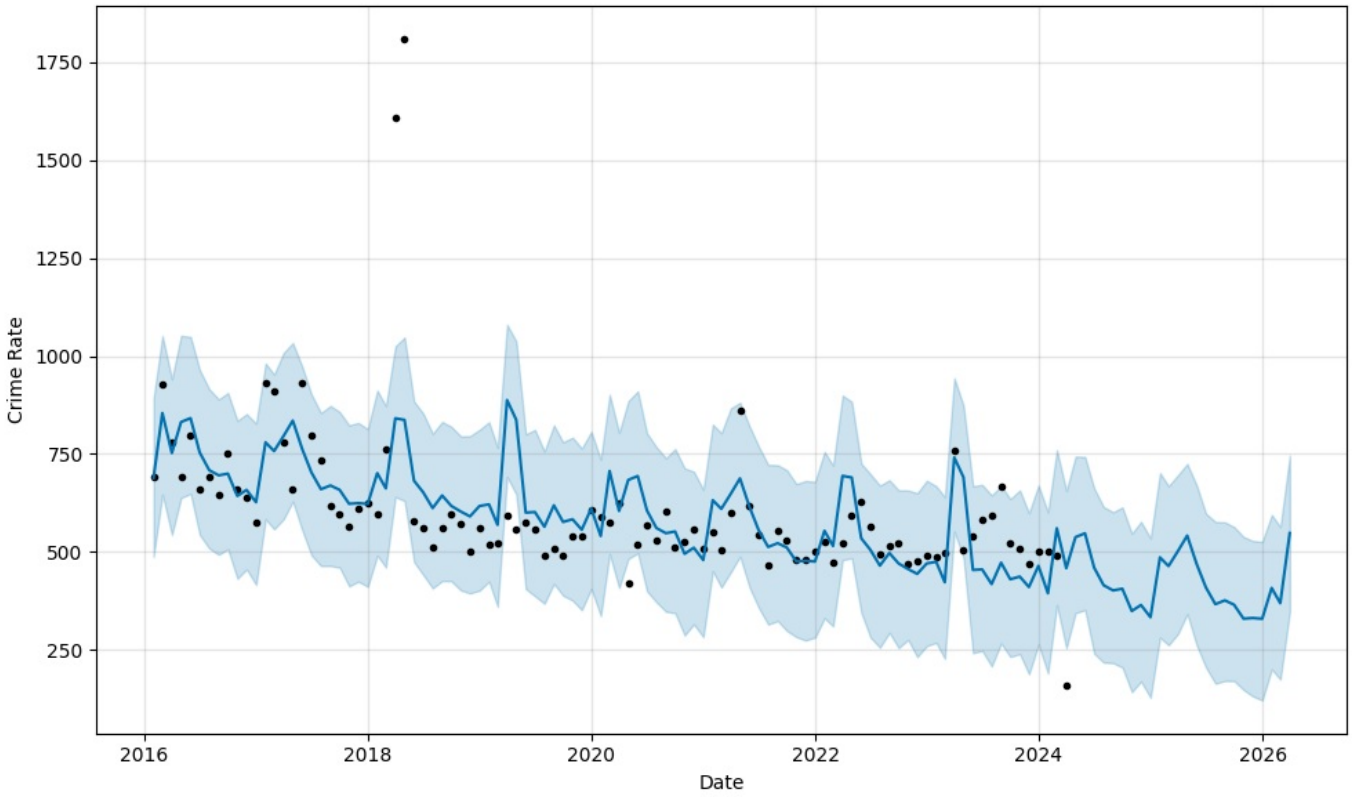
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	756.475020	487.529748	892.265596	756.475020	756.475020	-68.372168	-68.372168	-68.372
1	2016-02-29	753.537041	646.258317	1051.539502	753.537041	753.537041	100.703122	100.703122	100.703
2	2016-03-31	750.396443	543.385550	941.140295	750.396443	750.396443	1.762391	1.762391	1.762
3	2016-04-30	747.357154	637.481174	1051.928140	747.357154	747.357154	84.073130	84.073130	84.073
4	2016-05-31	744.216556	648.281583	1048.905706	744.216556	744.216556	97.007625	97.007625	97.007
...	...	...	...	...	...	...	...	...	
118	2025-11-30	395.303845	130.946098	528.056949	395.233406	395.370642	-64.519501	-64.519501	-64.519
119	2025-12-31	392.203517	120.471254	525.350291	392.127291	392.276644	-63.365871	-63.365871	-63.365
120	2026-01-31	389.103190	199.814935	594.474815	389.020900	389.182884	18.355649	18.355649	18.355
121	2026-02-28	386.302893	173.858380	563.342189	386.214675	386.388859	-17.689345	-17.689345	-17.689
122	2026-03-31	383.202566	347.182177	746.057656	383.107822	383.293524	164.437048	164.437048	164.437

123 rows × 16 columns

In [160…
```
m_Mandya.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[160…



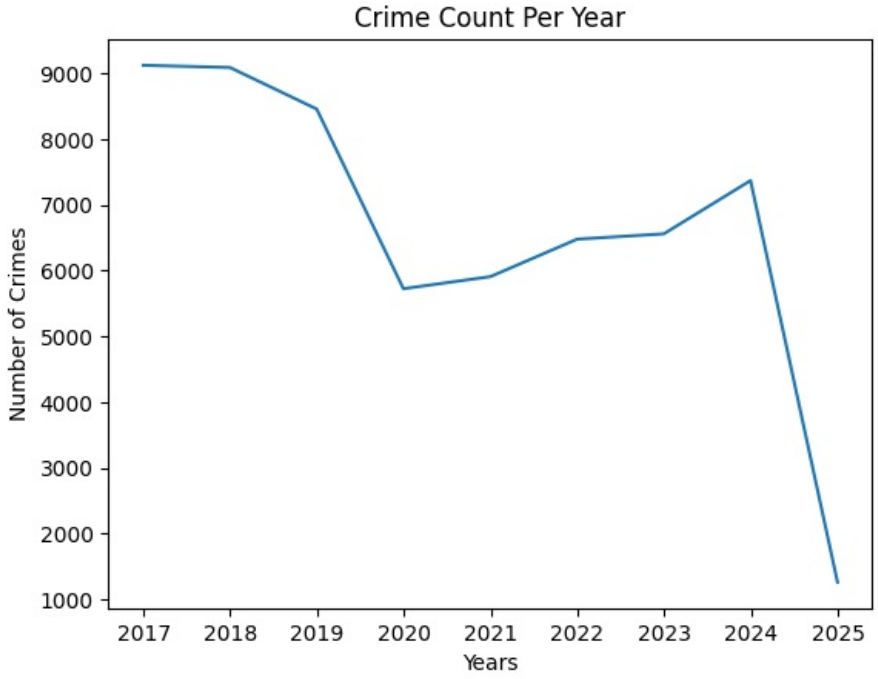


In [161…
```
df_BelagaviDist = df[df["District_Name"]=="Belagavi Dist"]
```

In [162…
```
df_BelagaviDist.index = pd.DatetimeIndex(df_BelagaviDist.FIR_Reg_DateTime)
```

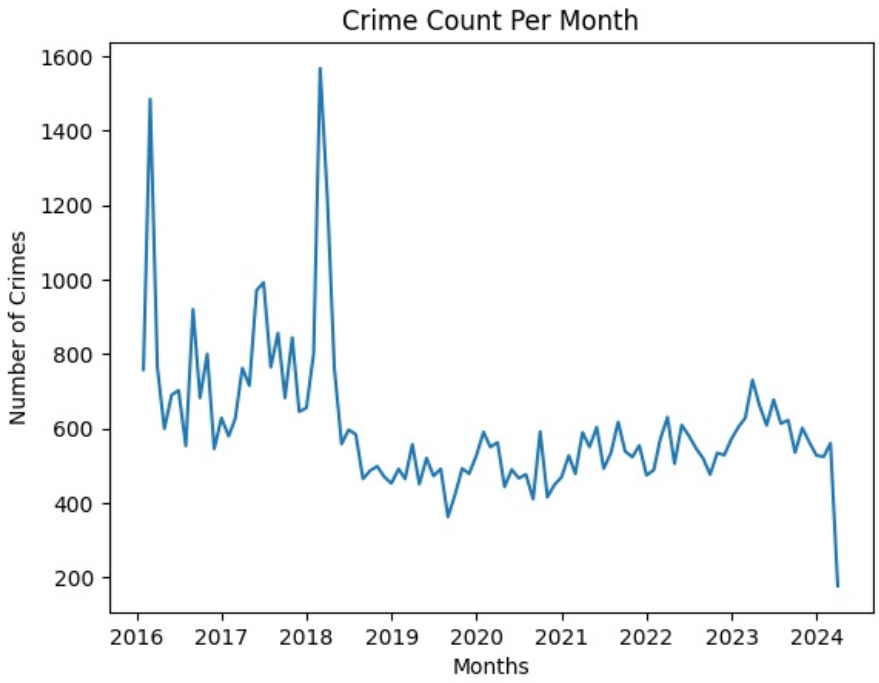
In [163…
```
plt.plot(df_BelagaviDist.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[163…
```
Text(0, 0.5, 'Number of Crimes')
```



In [164…
```
plt.plot(df_BelagaviDist.resample('M').size())
plt.title("Crime Count Per Month")
plt.xlabel("Months")
plt.ylabel("Number of Crimes")
```

Out[164… Text(0, 0.5, 'Number of Crimes')



In [165…
```
df_BelagaviDist_prophet = pd.DataFrame(df_BelagaviDist.resample('M').size().reset_index())
```

In [166…
```
df_BelagaviDist_prophet.columns =['Date','Crime Count']
```

In [167…
```
df_BelagaviDist_prophet=df_BelagaviDist_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [168…
```
m_BelagaviDist = Prophet()
m_BelagaviDist.fit(df_BelagaviDist_prophet)
```

```
21:29:08 - cmdstanpy - INFO - Chain [1] start processing
21:29:08 - cmdstanpy - INFO - Chain [1] done processing
```

Out[168… `<prophet.forecaster.Prophet at 0x1df8155a910>`

In [169…
```
pred=m_BelagaviDist.make_future_dataframe(periods=24,freq='M')
forcast = m_BelagaviDist.predict(pred)
```

In [170…
```
forcast
```

Out[170…

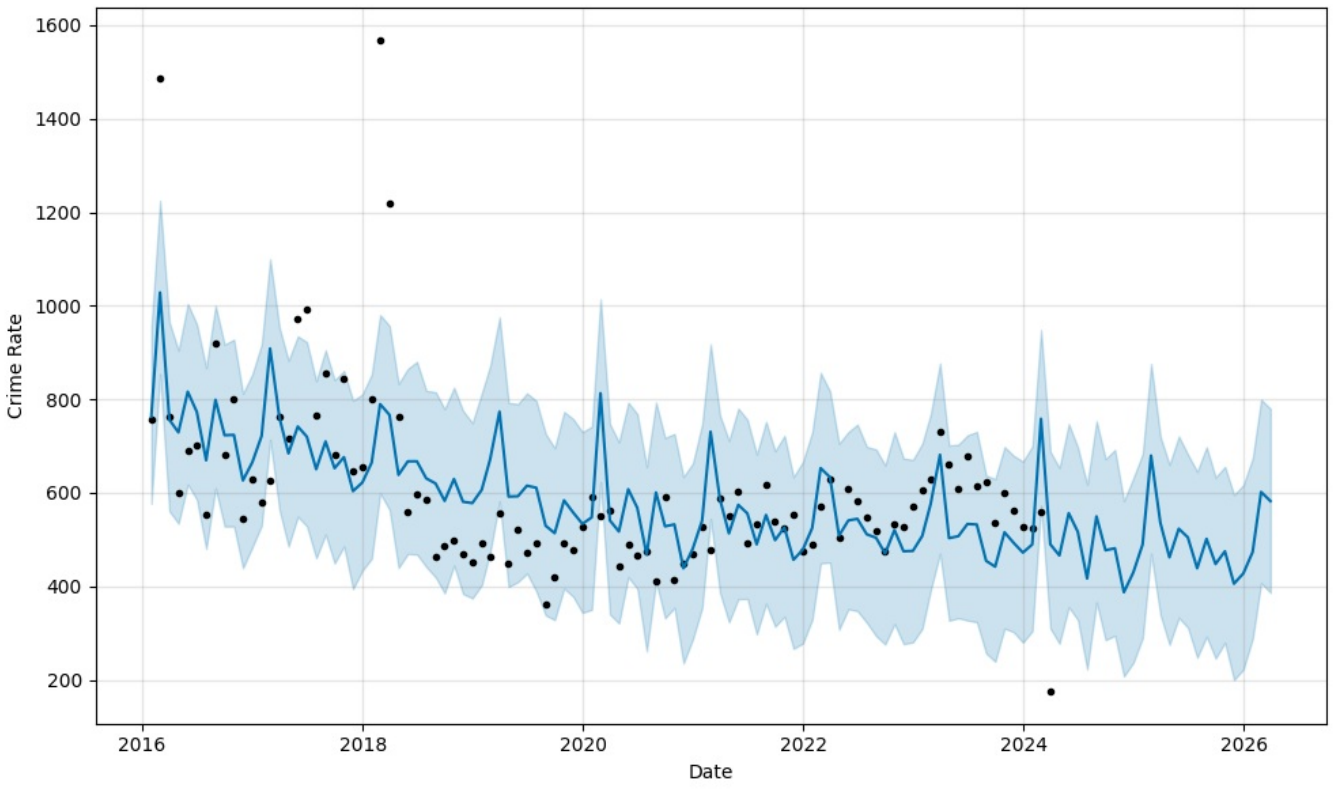
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	790.720448	576.623406	953.701192	790.720448	790.720448	-27.845394	-27.845394	-27.845
1	2016-02-29	786.439096	854.076033	1224.912584	786.439096	786.439096	241.863392	241.863392	241.863
2	2016-03-31	781.862478	560.485959	964.881477	781.862478	781.862478	-25.520727	-25.520727	-25.520
3	2016-04-30	777.433494	534.175287	904.338296	777.433494	777.433494	-48.273247	-48.273247	-48.273
4	2016-05-31	772.856876	617.928863	1003.944984	772.856876	772.856876	43.193556	43.193556	43.193
...	...	...	...	...	...	...	...	...	
118	2025-11-30	493.929897	199.686588	595.781601	488.131089	500.002116	-88.408396	-88.408396	-88.408
119	2025-12-31	492.845409	222.277533	616.530287	486.533584	499.428627	-65.500318	-65.500318	-65.500
120	2026-01-31	491.760920	288.592473	674.287602	484.844592	498.748858	-18.354288	-18.354288	-18.354
121	2026-02-28	490.781382	407.122304	799.328241	483.395126	498.224329	110.936432	110.936432	110.936
122	2026-03-31	489.696893	386.283542	780.021889	481.915374	497.518129	92.540126	92.540126	92.540

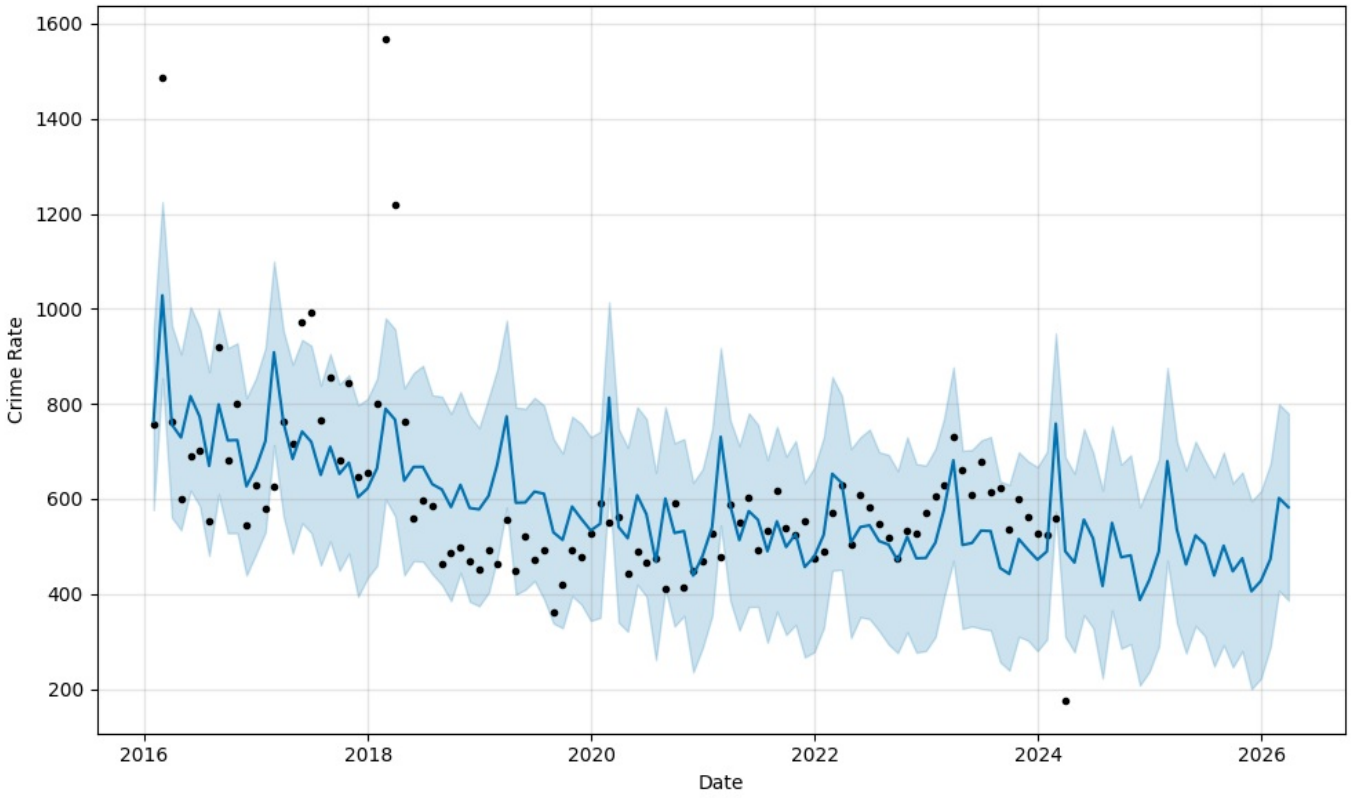
123 rows × 16 columns

In [171…

```
m_BelagaviDist.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[171…



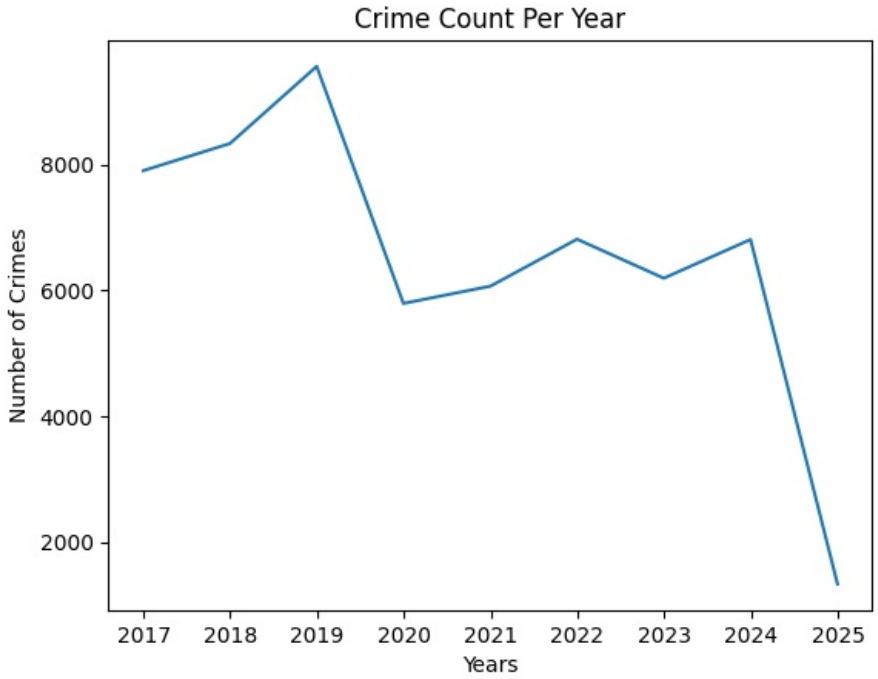


In [172…
```
df_Hassan = df[df["District_Name"]=="Hassan"]
```

In [173…
```
df_Hassan.index = pd.DatetimeIndex(df_Hassan.FIR_Reg_DateTime)
```

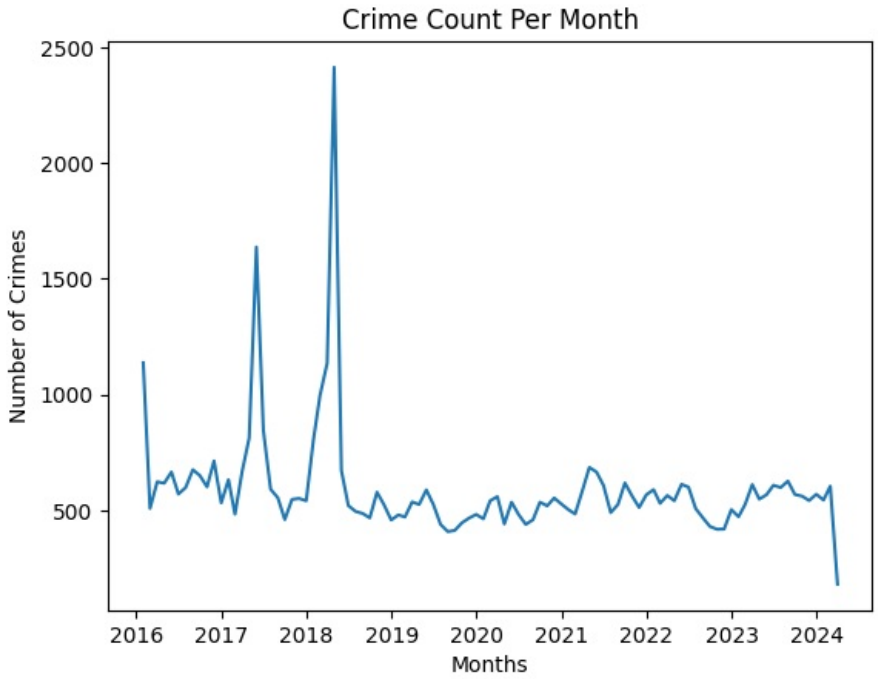
In [174…
```
plt.plot(df_Hassan.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[174… Text(0, 0.5, 'Number of Crimes')



In [175…
```
plt.plot(df_Hassan.resample('M').size())
plt.title("Crime Count Per Month")
plt.xlabel("Months")
plt.ylabel("Number of Crimes")
```

Out[175… Text(0, 0.5, 'Number of Crimes')



In [176…
```
df_Hassan_prophet = pd.DataFrame(df_Hassan.resample('M').size().reset_index())
```

In [177…
```
df_Hassan_prophet.columns =['Date','Crime Count']
```

In [178…
```
df_Hassan_prophet=df_Hassan_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [179…
```
m_Hassan = Prophet()
m_Hassan.fit(df_Hassan_prophet)
```

```
21:33:56 - cmdstanpy - INFO - Chain [1] start processing
21:33:56 - cmdstanpy - INFO - Chain [1] done processing
```

Out[179… `<prophet.forecaster.Prophet at 0x1df81e628d0>`

In [180…
```
pred=m_Hassan.make_future_dataframe(periods=24,freq='M')
forcast = m_Hassan.predict(pred)
```

In [181…
```
forcast
```

Out[181…

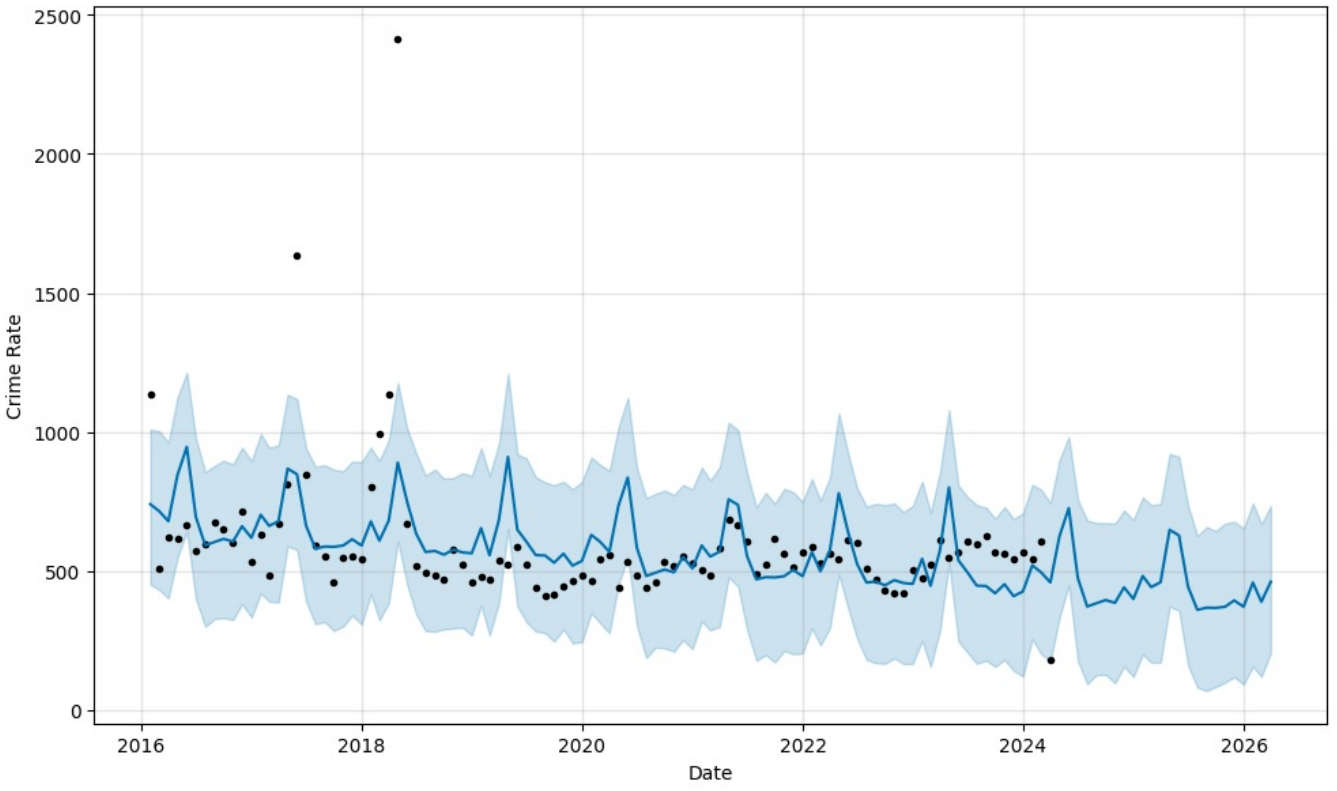
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	707.245474	452.692647	1009.599350	707.245474	707.245474	33.521937	33.521937	33.521
1	2016-02-29	705.055996	433.394505	1003.638956	705.055996	705.055996	11.300610	11.300610	11.300
2	2016-03-31	702.715520	402.830859	963.109798	702.715520	702.715520	-22.987356	-22.987356	-22.987
3	2016-04-30	700.450544	549.340961	1126.261468	700.450544	700.450544	144.799526	144.799526	144.799
4	2016-05-31	698.110068	641.392598	1213.752613	698.110068	698.110068	248.685925	248.685925	248.685
...	...	...	...	...	...	...	...	...	
118	2025-11-30	436.927177	119.396612	678.237848	436.904293	436.948386	-42.023424	-42.023424	-42.023
119	2025-12-31	434.599032	93.052730	654.464760	434.574349	434.621750	-62.241491	-62.241491	-62.241
120	2026-01-31	432.270886	157.775208	744.216883	432.244609	432.295397	26.652738	26.652738	26.652
121	2026-02-28	430.168046	120.572205	669.786928	430.140048	430.194397	-40.222343	-40.222343	-40.222
122	2026-03-31	427.839900	206.016074	735.316599	427.809980	427.867863	34.064985	34.064985	34.064

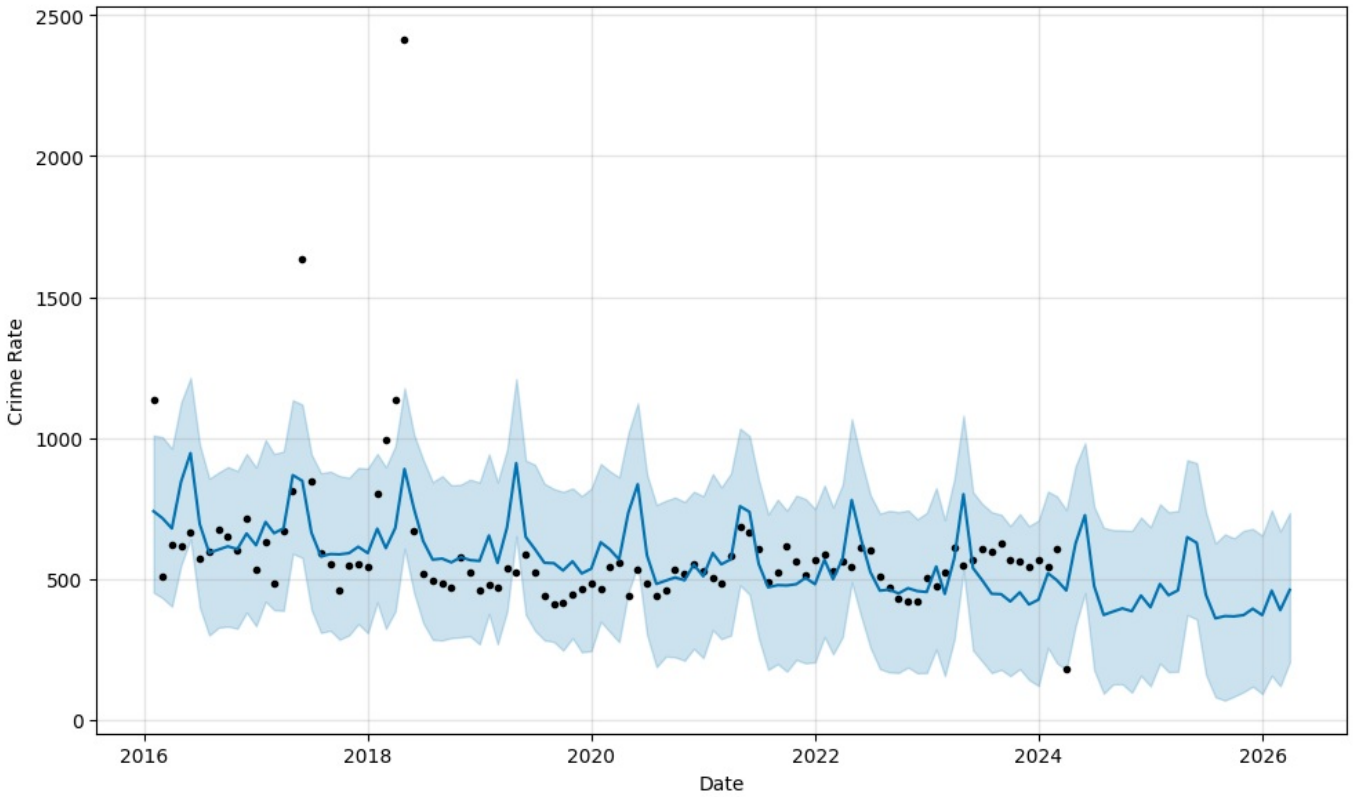
123 rows × 16 columns

In [182…

```
m_Hassan.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[182…



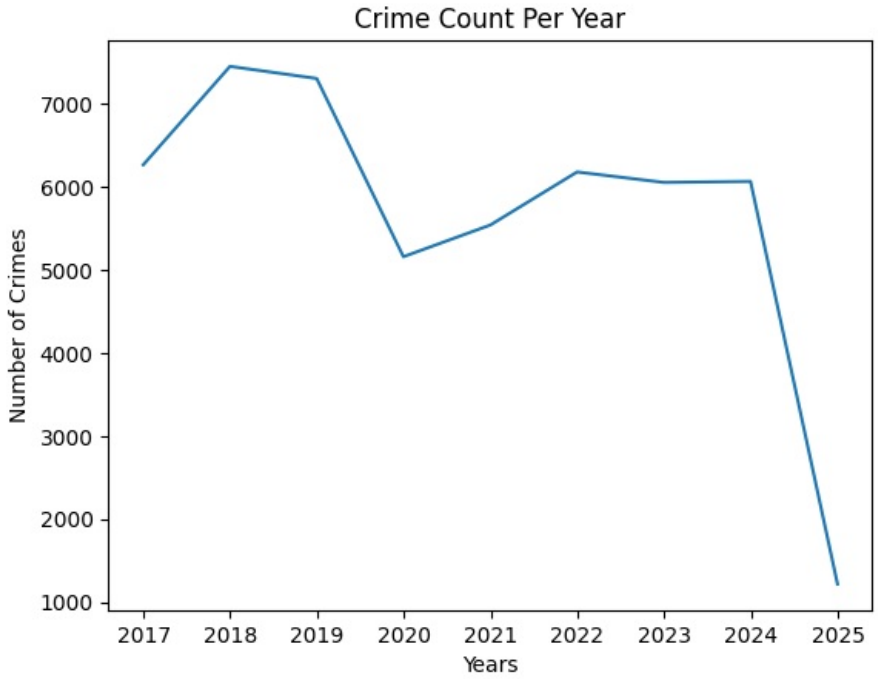


In [186…
```
df_MysuruDist = df[df["District_Name"]=="Mysuru Dist"]
```

In [187…
```
df_MysuruDist.index = pd.DatetimeIndex(df_MysuruDist.FIR_Reg_DateTime)
```

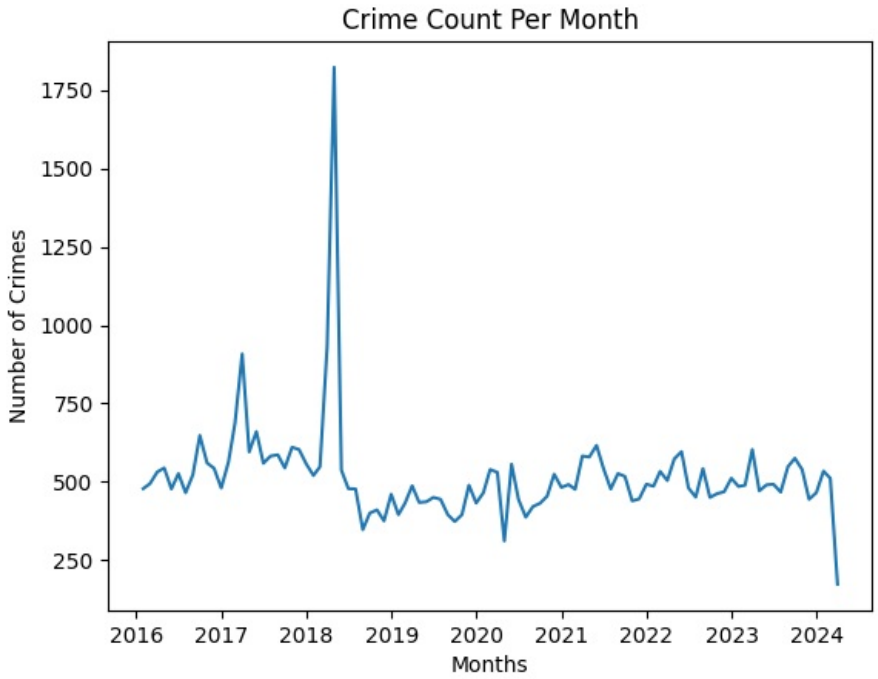
In [188…
```
plt.plot(df_MysuruDist.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[188… Text(0, 0.5, 'Number of Crimes')



In [189…
```
plt.plot(df_MysuruDist.resample('M').size())
plt.title("Crime Count Per Month")
plt.xlabel("Months")
plt.ylabel("Number of Crimes")
```

Out[189…
```
Text(0, 0.5, 'Number of Crimes')
```



In [190…
```
df_MysuruDist_prophet = pd.DataFrame(df_MysuruDist.resample('M').size().reset_index())
```

In [191…
```
df_MysuruDist_prophet.columns =['Date','Crime Count']
```

In [192…
```
df_MysuruDist_prophet=df_MysuruDist_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [193…
```
m_MysuruDist = Prophet()
m_MysuruDist.fit(df_MysuruDist_prophet)
```

```
21:41:20 - cmdstanpy - INFO - Chain [1] start processing
21:41:20 - cmdstanpy - INFO - Chain [1] done processing
```

Out[193… `<prophet.forecaster.Prophet at 0x1df7f38f6d0>`

In [194…
```
pred=m_MysuruDist.make_future_dataframe(periods=24,freq='M')
forcast = m_MysuruDist.predict(pred)
```

In [195…
```
forcast
```

Out[195…

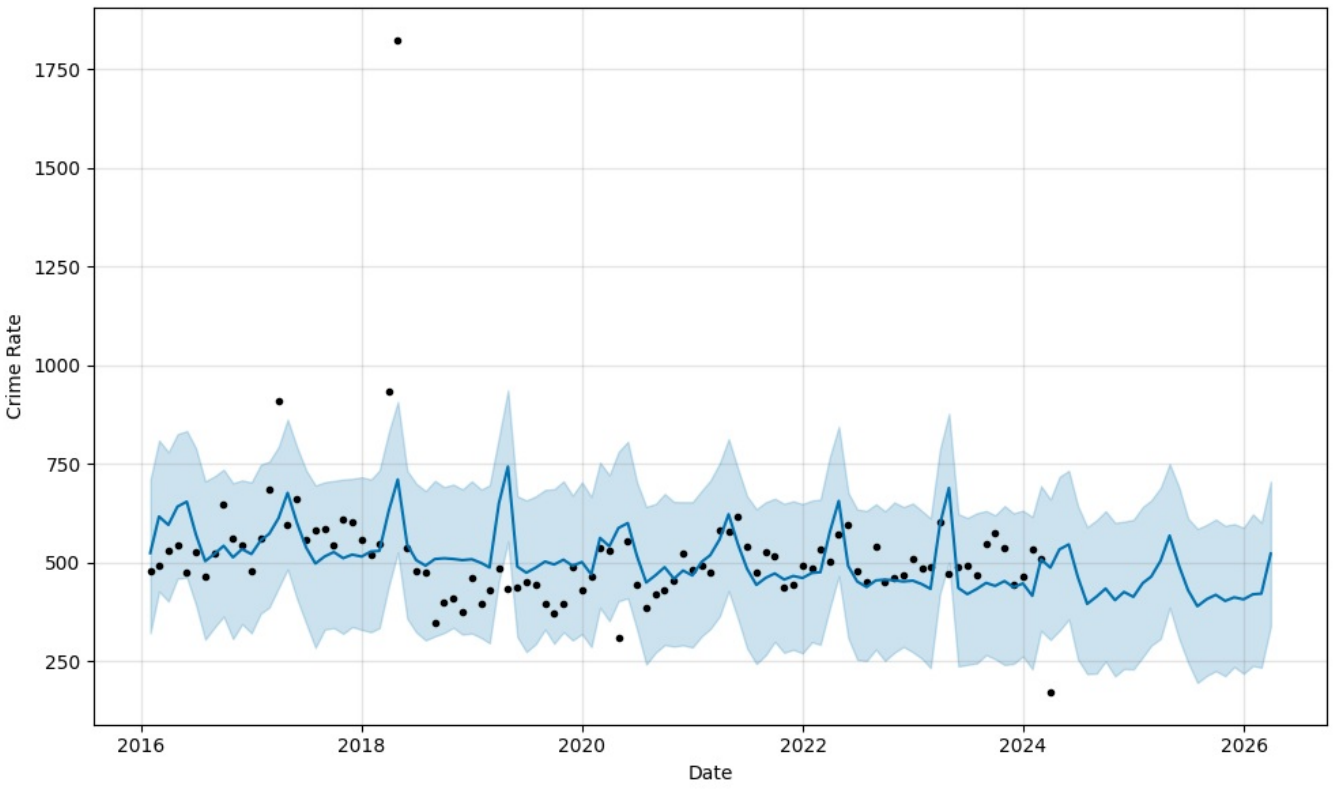
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	570.752132	321.931953	710.323209	570.752132	570.752132	-46.159515	-46.159515	-46.159
1	2016-02-29	569.676751	427.393149	809.884690	569.676751	569.676751	47.350691	47.350691	47.350
2	2016-03-31	568.527206	402.314151	780.024727	568.527206	568.527206	27.241503	27.241503	27.241
3	2016-04-30	567.414744	459.885329	825.922431	567.414744	567.414744	74.650569	74.650569	74.650
4	2016-05-31	566.265199	462.700452	833.816956	566.265199	566.265199	88.417674	88.417674	88.417
...	...	...	...	...	...	...	...	...	
118	2025-11-30	437.681913	236.271674	598.166409	437.679551	437.684220	-25.326641	-25.326641	-25.326
119	2025-12-31	436.533690	219.355477	588.787451	436.531146	436.536236	-29.197345	-29.197345	-29.197
120	2026-01-31	435.385468	239.031577	622.741745	435.382732	435.388195	-15.280819	-15.280819	-15.280
121	2026-02-28	434.348363	234.644428	601.710802	434.345405	434.351333	-12.532386	-12.532386	-12.532
122	2026-03-31	433.200141	339.214519	706.022619	433.197023	433.203188	89.742064	89.742064	89.742

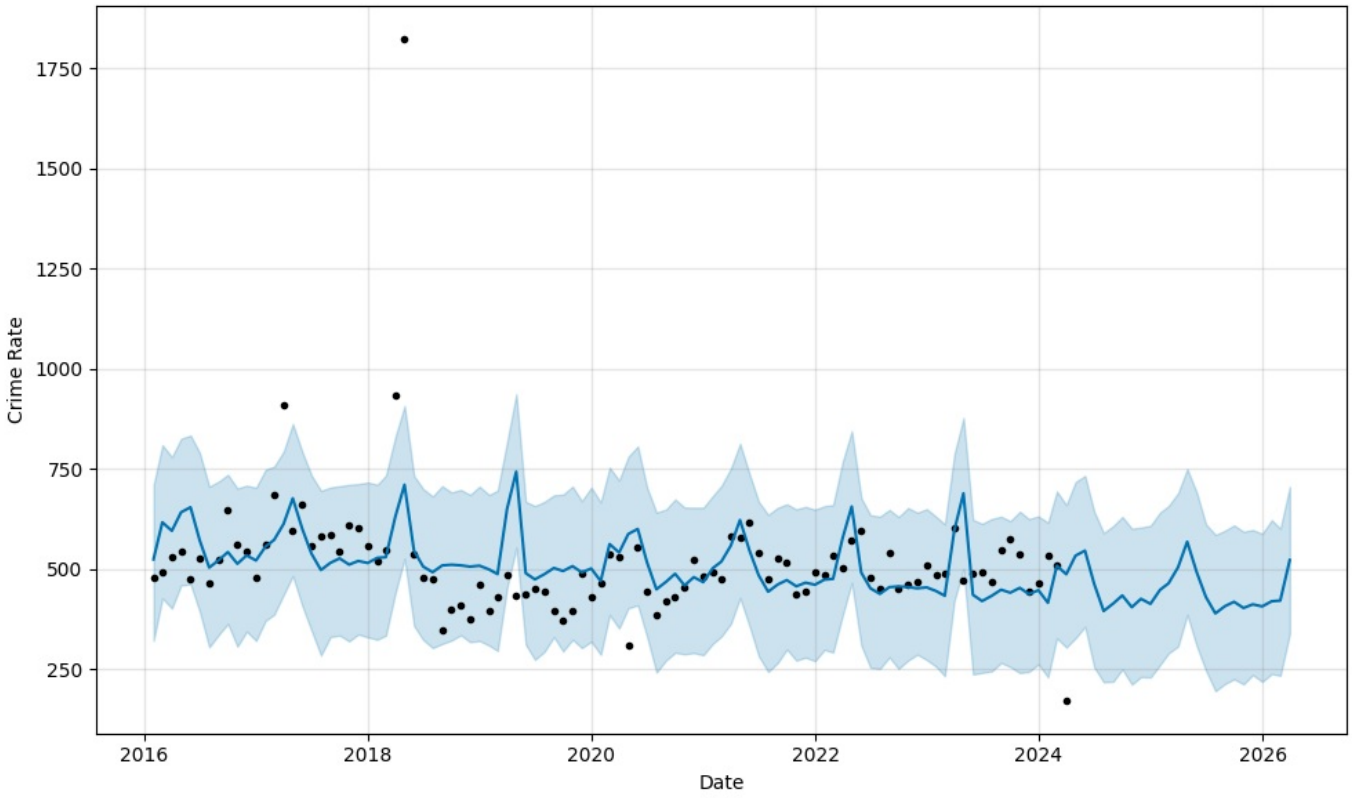
123 rows × 16 columns

In [196…

```
m_MysuruDist.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[196…



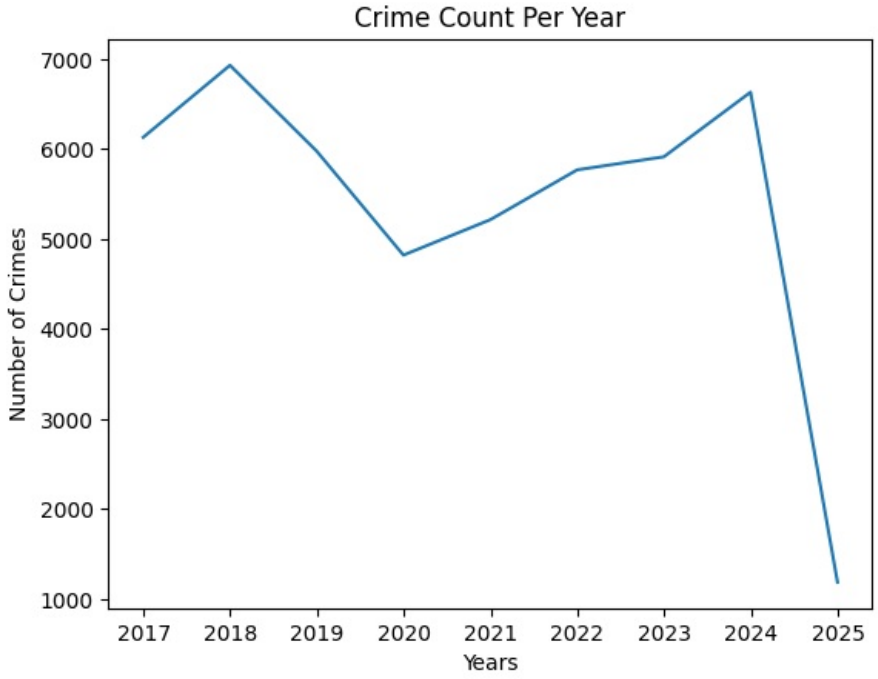


In [197…
```
df_Chitradurga = df[df["District_Name"]=="Chitradurga"]
```

In [198…
```
df_Chitradurga.index = pd.DatetimeIndex(df_Chitradurga.FIR_Reg_DateTime)
```

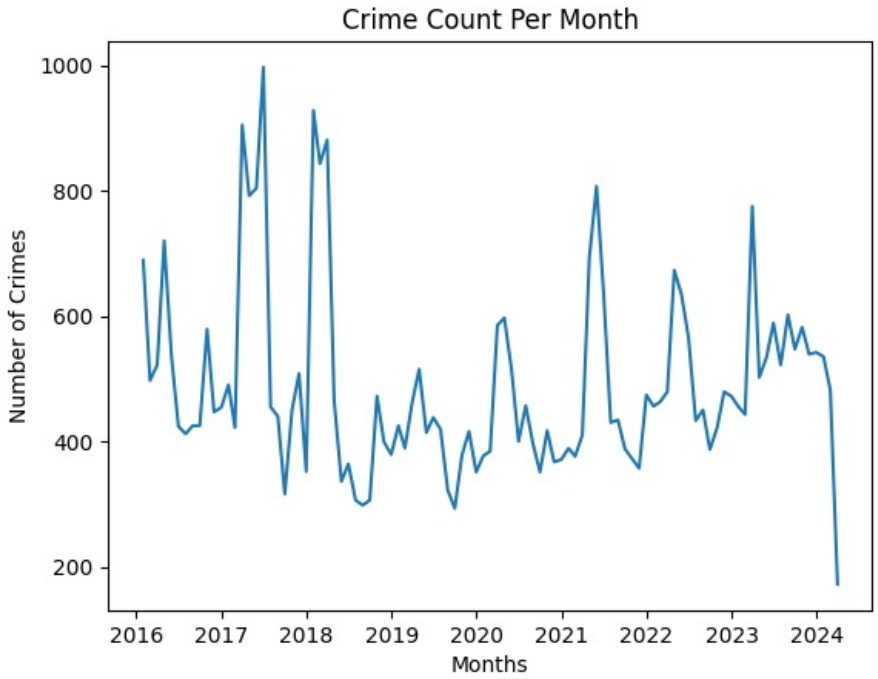
In [199…
```
plt.plot(df_Chitradurga.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[199…
```
Text(0, 0.5, 'Number of Crimes')
```



In [200…
```
plt.plot(df_Chitradurga.resample('M').size())
plt.title("Crime Count Per Month")
plt.xlabel("Months")
plt.ylabel("Number of Crimes")
```

Out[200… Text(0, 0.5, 'Number of Crimes')



In [201…
```
df_Chitradurga_prophet = pd.DataFrame(df_Chitradurga.resample('M').size().reset_index())
```

In [202…
```
df_Chitradurga_prophet.columns =['Date','Crime Count']
```

In [203…
```
df_Chitradurga_prophet=df_Chitradurga_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [204…
```
m_Chitradurga = Prophet()
m_Chitradurga.fit(df_Chitradurga_prophet)
```

```
21:45:45 - cmdstanpy - INFO - Chain [1] start processing
21:45:45 - cmdstanpy - INFO - Chain [1] done processing
```

Out[204… `<prophet.forecaster.Prophet at 0x1df7ea9d3d0>`

In [205…
```
pred=m_Chitradurga.make_future_dataframe(periods=24,freq='M')
forcast = m_Chitradurga.predict(pred)
```

In [206…
```
forcast
```

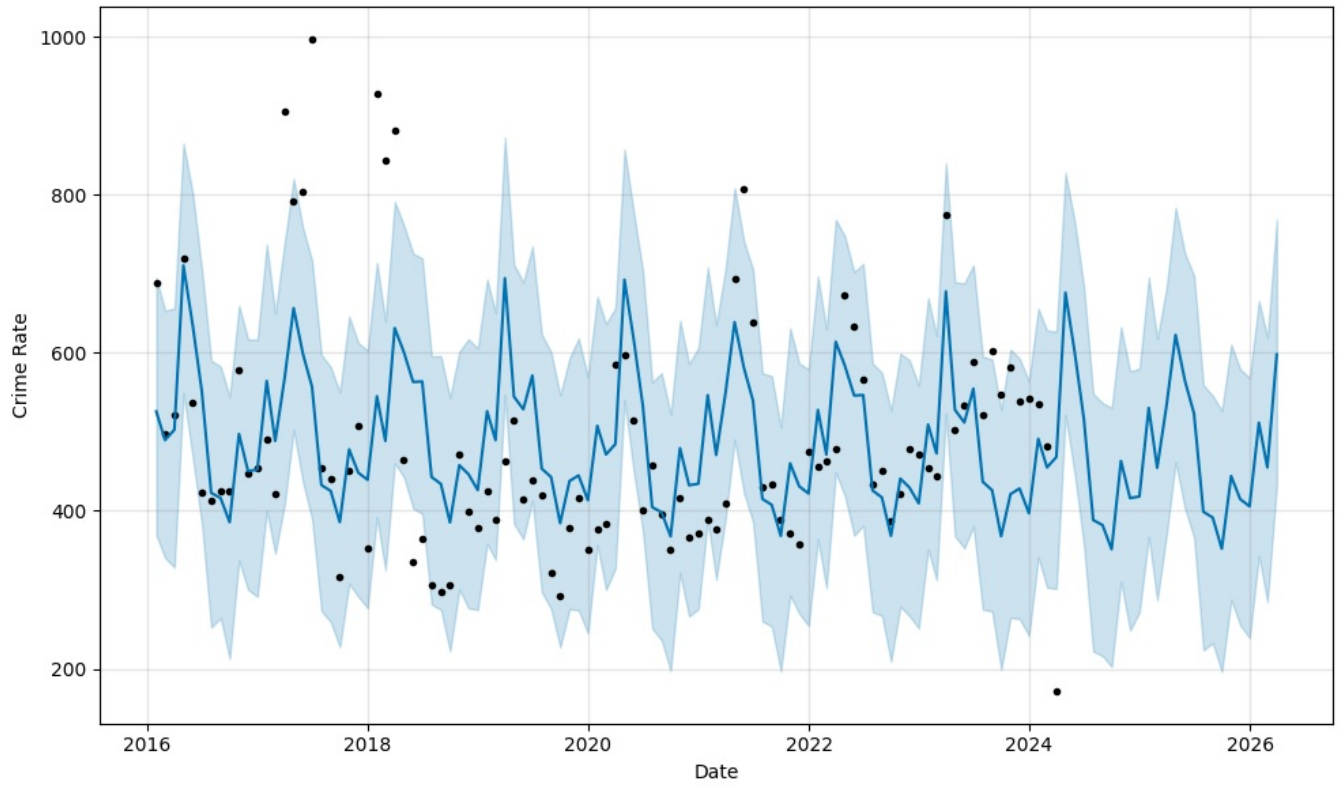
Out[206…

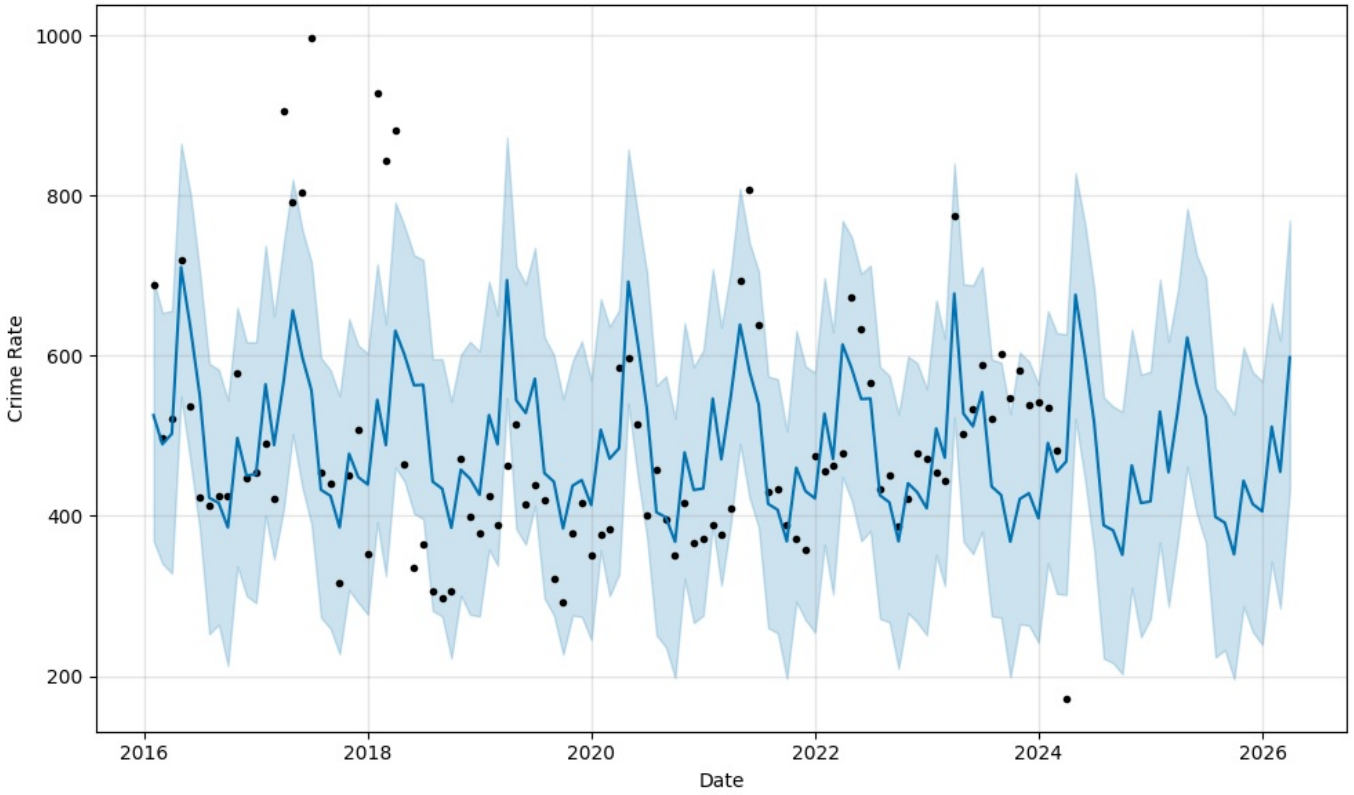
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	503.690382	368.966773	694.829897	503.690382	503.690382	22.065042	22.065042	22.065(
1	2016-02-29	503.326347	340.327483	653.535527	503.326347	503.326347	-13.836052	-13.836052	-13.836(
2	2016-03-31	502.937206	328.410364	656.160210	502.937206	502.937206	-0.359051	-0.359051	-0.359(
3	2016-04-30	502.560618	549.077748	864.938624	502.560618	502.560618	208.007484	208.007484	208.007
4	2016-05-31	502.171476	466.859576	801.404114	502.171476	502.171476	132.115324	132.115324	132.115
...	...	...	...	...	...	...	...	...	
118	2025-11-30	461.673263	255.160210	579.442466	461.593257	461.754916	-46.923864	-46.923864	-46.923
119	2025-12-31	461.329734	239.157030	568.181849	461.242887	461.418340	-55.650553	-55.650553	-55.650
120	2026-01-31	460.986204	342.917957	665.687993	460.891898	461.082090	50.425253	50.425253	50.425
121	2026-02-28	460.675919	284.414291	619.116899	460.574966	460.777307	-5.690688	-5.690688	-5.690
122	2026-03-31	460.332390	432.976712	768.708873	460.223571	460.440574	137.484702	137.484702	137.484

123 rows × 16 columns

In [207…
```
m_Chitradurga.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[207…



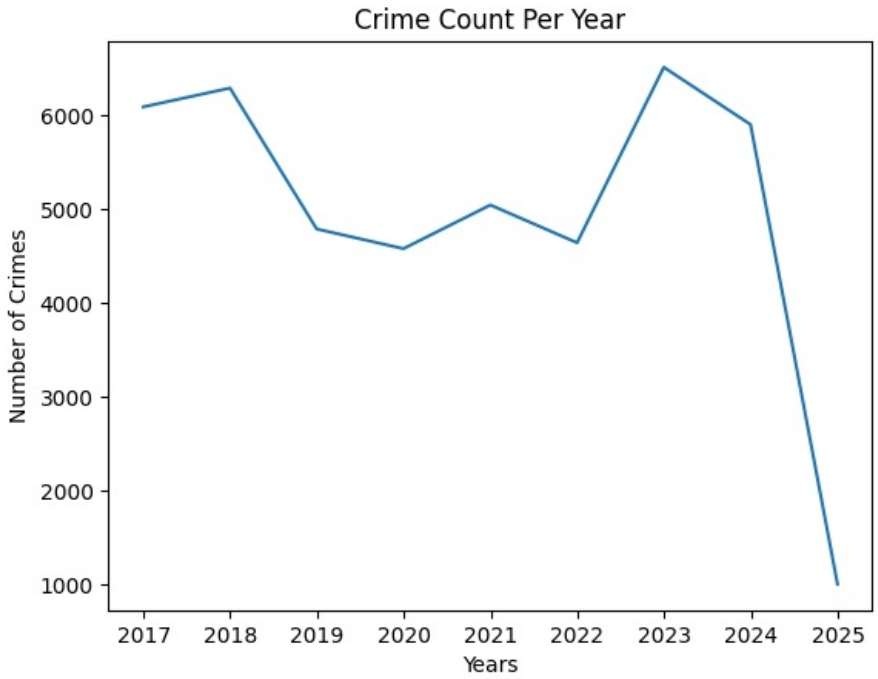


In [208…
```
df_Ramanagara = df[df["District_Name"]=="Ramanagara"]
```

In [209…
```
df_Ramanagara.index = pd.DatetimeIndex(df_Ramanagara.FIR_Reg_DateTime)
```

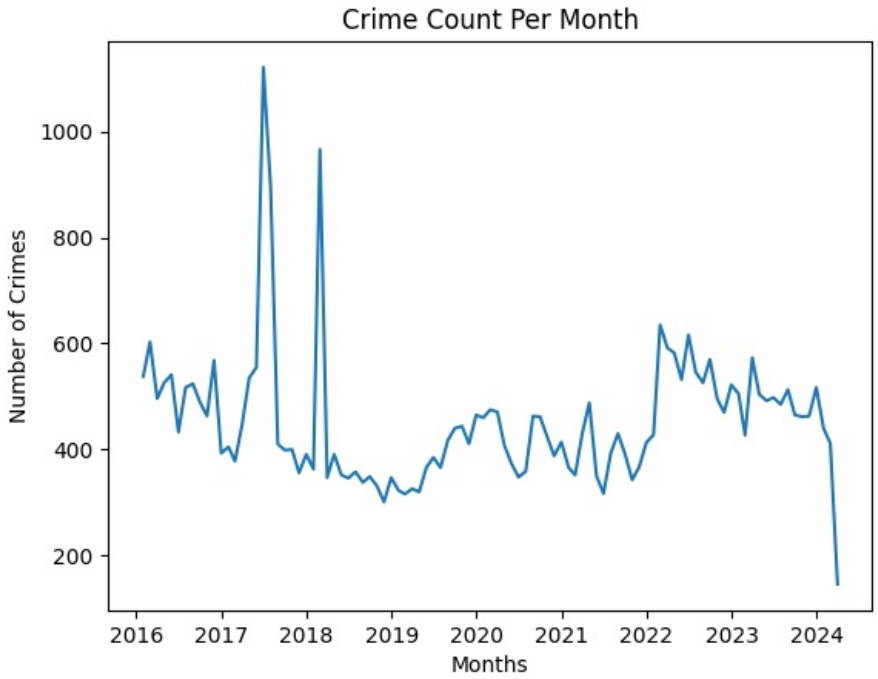
In [210…
```
plt.plot(df_Ramanagara.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[210…
```
Text(0, 0.5, 'Number of Crimes')
```



In [211…
```
plt.plot(df_Ramanagara.resample('M').size())
plt.title("Crime Count Per Month")
plt.xlabel("Months")
plt.ylabel("Number of Crimes")
```

Out[211… `Text(0, 0.5, 'Number of Crimes')`



In [212…
```
df_Ramanagara_prophet = pd.DataFrame(df_Ramanagara.resample('M').size().reset_index())
```

In [213…
```
df_Ramanagara_prophet.columns =['Date','Crime Count']
```

In [216…
```
df_Ramanagara_prophet=df_Ramanagara_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [215…
```
m_Ramanagara = Prophet()
m_Ramanagara.fit(df_Ramanagara_prophet)
```

```
21:49:19 - cmdstanpy - INFO - Chain [1] start processing
21:49:19 - cmdstanpy - INFO - Chain [1] done processing
```

Out[215… <prophet.forecaster.Prophet at 0x1df8099d110>

In [217…
```
pred=m_Ramanagara.make_future_dataframe(periods=24,freq='M')
forcast = m_Ramanagara.predict(pred)
```

In [218…
```
forcast
```

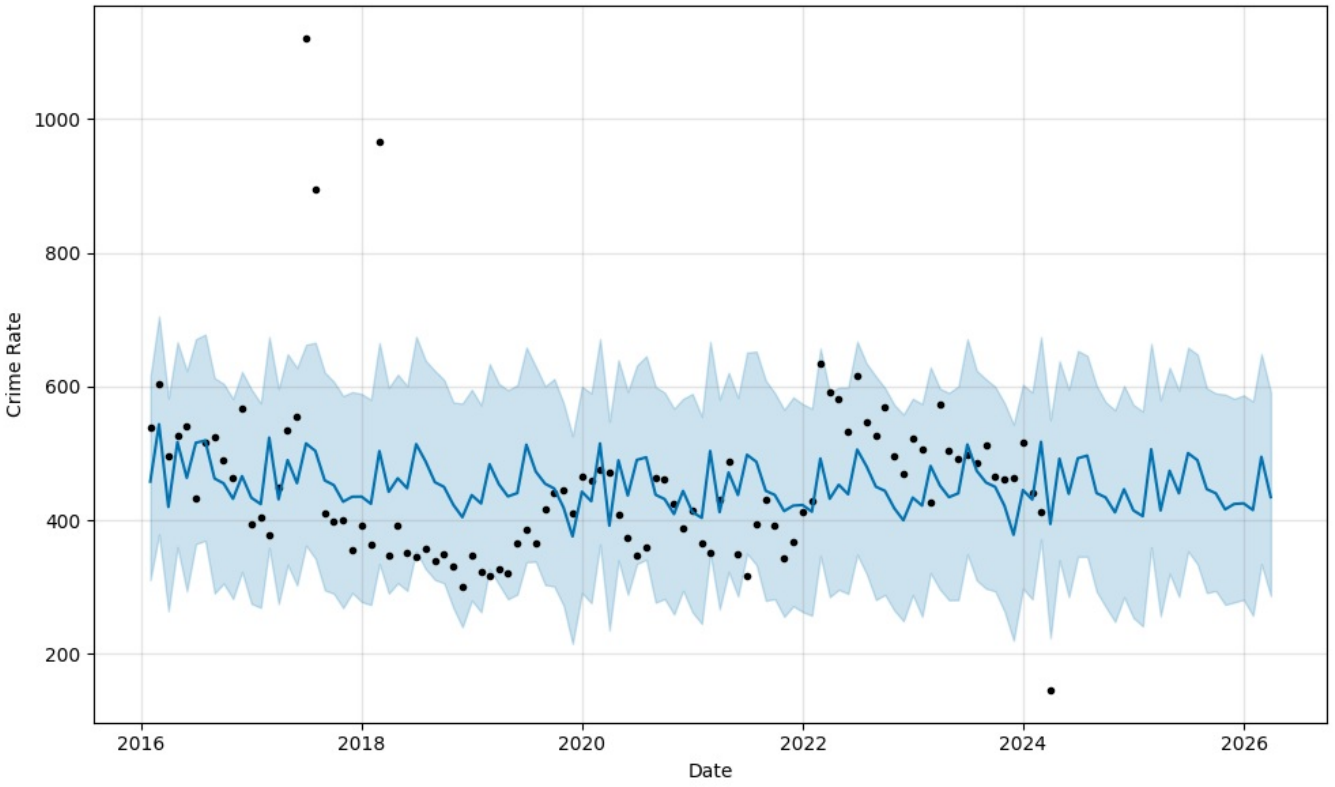
Out[218…

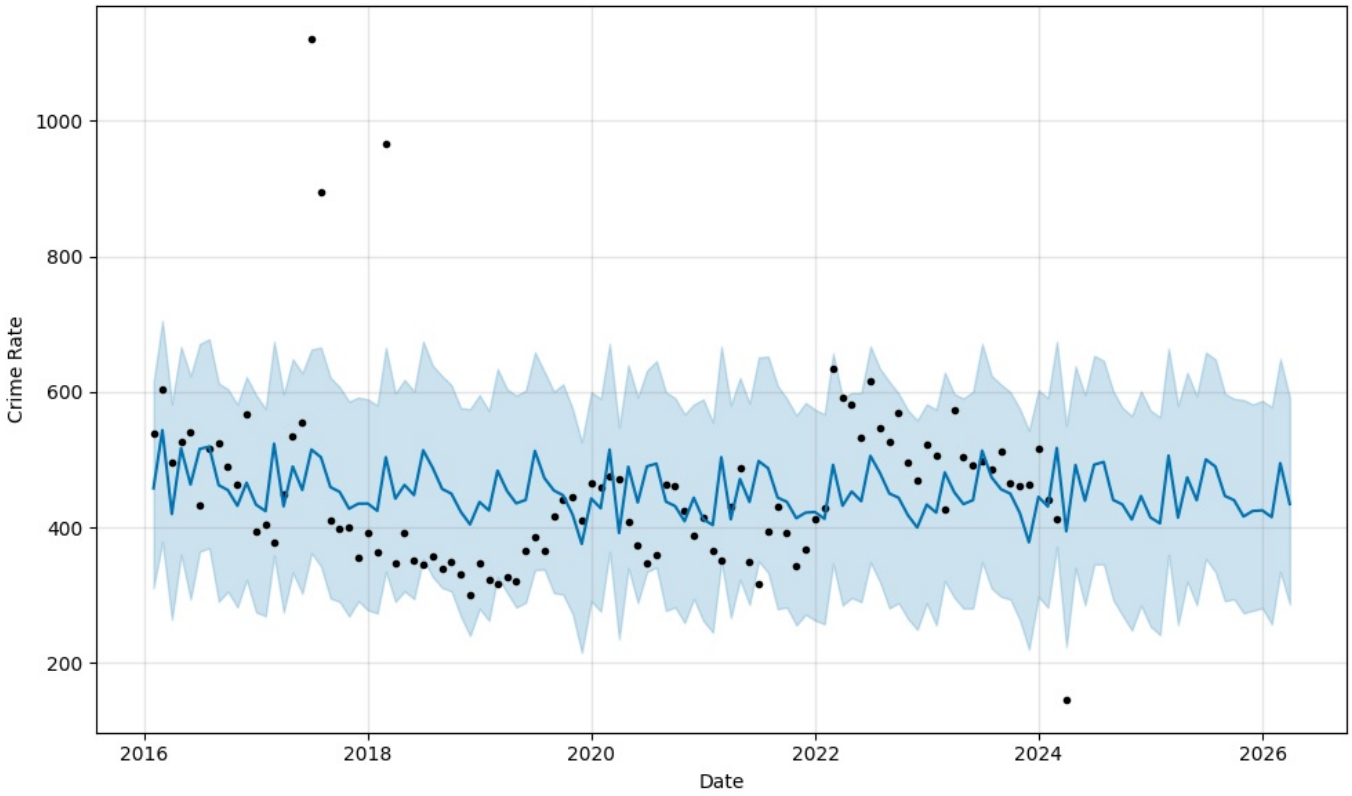
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	486.544267	310.604064	614.365684	486.544267	486.544267	-28.870994	-28.870994	-28.870
1	2016-02-29	485.895263	378.726881	704.936348	485.895263	485.895263	57.515770	57.515770	57.515
2	2016-03-31	485.201501	263.996494	582.395836	485.201501	485.201501	-65.494960	-65.494960	-65.494
3	2016-04-30	484.530118	360.000084	666.302238	484.530118	484.530118	32.215646	32.215646	32.215
4	2016-05-31	483.836355	293.862181	623.307674	483.836355	483.836355	-20.569277	-20.569277	-20.569
...	...	...	...	...	...	...	...	...	
118	2025-11-30	460.772095	277.095330	581.543418	459.457131	461.954095	-36.594700	-36.594700	-36.594
119	2025-12-31	460.825264	280.909697	586.887597	459.393041	462.109848	-35.696326	-35.696326	-35.696
120	2026-01-31	460.878433	257.233518	577.712771	459.319433	462.241347	-45.751066	-45.751066	-45.751
121	2026-02-28	460.926457	334.515882	648.654873	459.253256	462.370534	33.812114	33.812114	33.812
122	2026-03-31	460.979626	286.356191	591.539536	459.190948	462.509540	-26.529332	-26.529332	-26.529

123 rows × 16 columns

In [219…
```
m_Ramanagara.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[219…



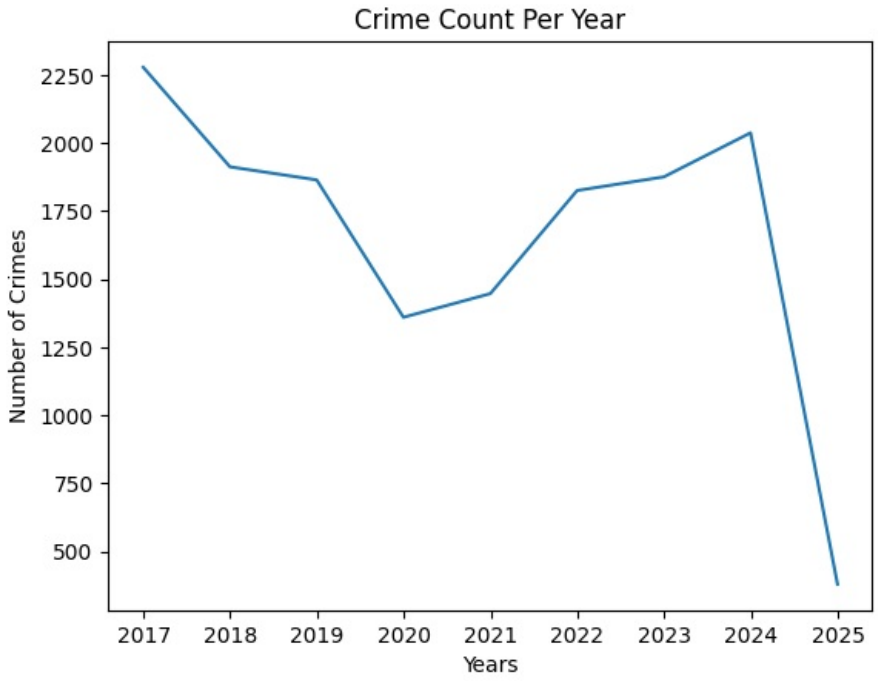


In [220…
```
df_Dharwad = df[df["District_Name"]=="Dharwad"]
```

In [221…
```
df_Dharwad.index = pd.DatetimeIndex(df_Dharwad.FIR_Reg_DateTime)
```

In [222…
```
plt.plot(df_Dharwad.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

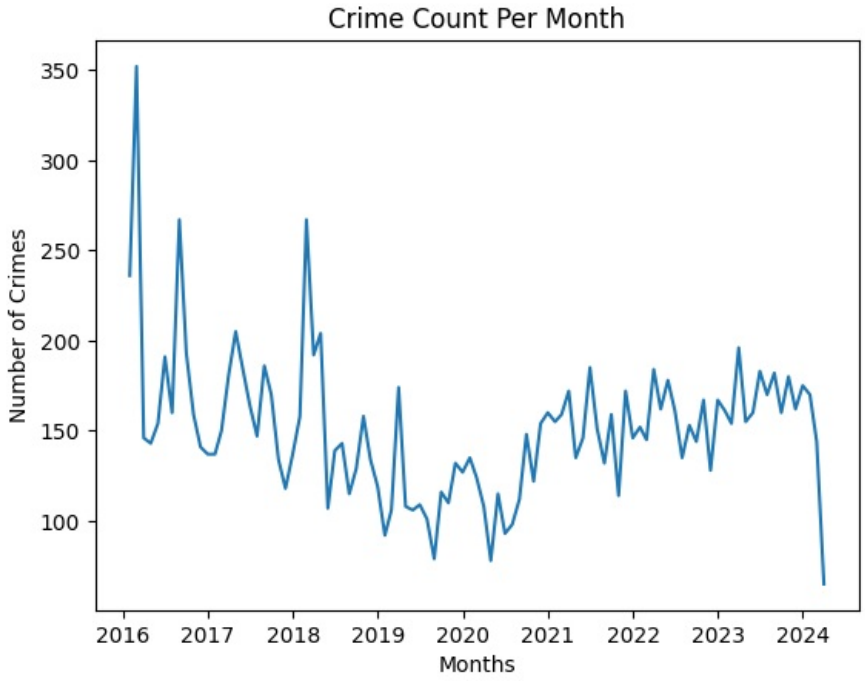
Out[222… Text(0, 0.5, 'Number of Crimes')



In [223…
```
plt.plot(df_Dharwad.resample('M').size())
plt.title("Crime Count Per Month")
```

```
plt.xlabel("Months")
plt.ylabel("Number of Crimes")
```

Out[223… `Text(0, 0.5, 'Number of Crimes')`



In [224…
```
df_Dharwad_prophet = pd.DataFrame(df_Dharwad.resample('M').size().reset_index())
```

In [225…
```
df_Dharwad_prophet.columns =['Date','Crime Count']
```

In [226…
```
df_Dharwad_prophet=df_Dharwad_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [227…
```
m_Dharwad = Prophet()
m_Dharwad.fit(df_Dharwad_prophet)
```

```
21:53:01 - cmdstanpy - INFO - Chain [1] start processing
21:53:02 - cmdstanpy - INFO - Chain [1] done processing
```

Out[227… `<prophet.forecaster.Prophet at 0x1df830551d0>`

In [228…
```
pred=m_Dharwad.make_future_dataframe(periods=24,freq='M')
forcast = m_Dharwad.predict(pred)
```

In [229…
```
forcast
```

Out[229…

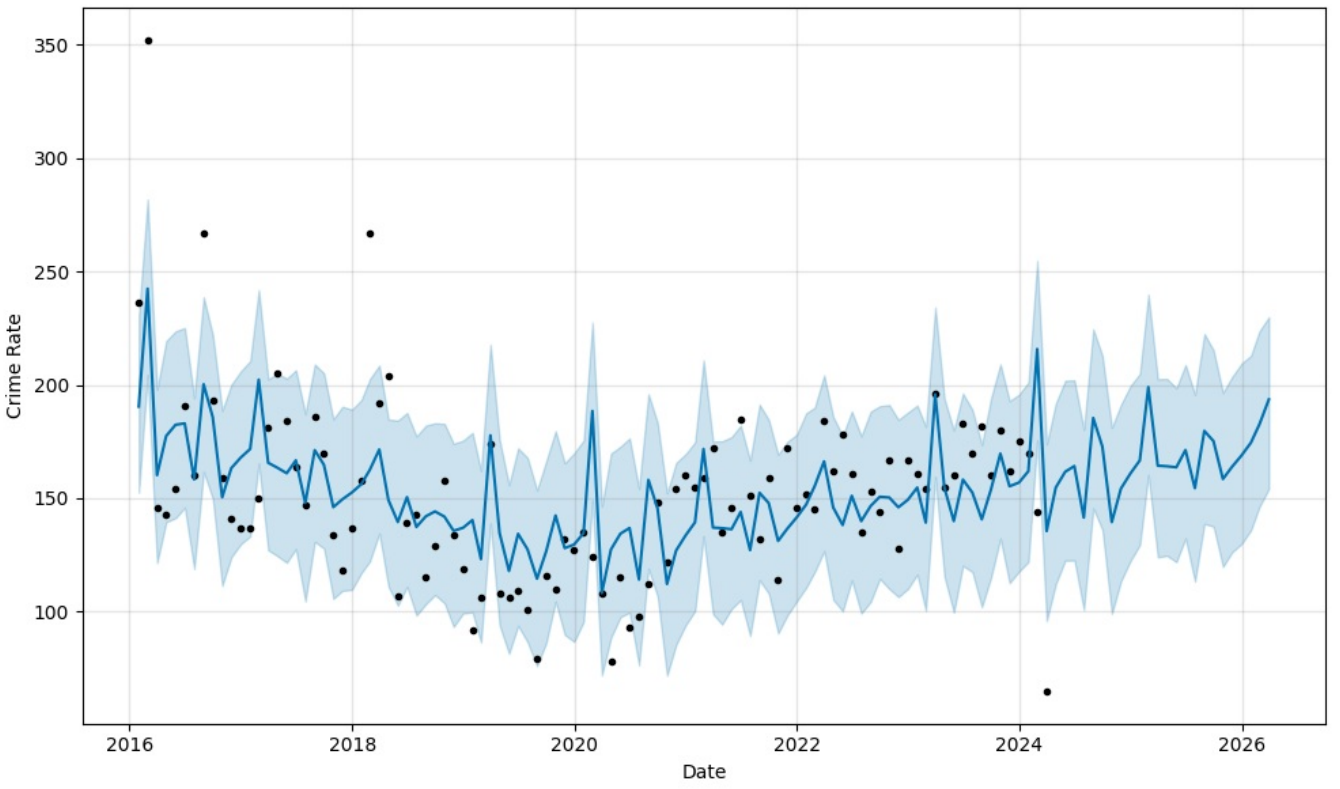
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_up
0	2016-01-31	192.678112	152.551160	230.457089	192.678112	192.678112	-2.269148	-2.269148	-2.269
1	2016-02-29	191.364618	204.375357	281.783795	191.364618	191.364618	51.037554	51.037554	51.037
2	2016-03-31	189.960537	121.520487	197.922380	189.960537	189.960537	-29.768050	-29.768050	-29.768
3	2016-04-30	188.601750	139.480260	219.378361	188.601750	188.601750	-11.153349	-11.153349	-11.153
4	2016-05-31	187.197670	141.267512	223.671886	187.197670	187.197670	-4.780299	-4.780299	-4.780
...	...	...	...	...	...	...	...	...	
118	2025-11-30	176.702039	126.207844	203.541776	173.177442	180.035837	-12.824801	-12.824801	-12.824
119	2025-12-31	177.281364	129.869990	209.405304	173.510472	180.919018	-8.432239	-8.432239	-8.432
120	2026-01-31	177.860689	135.948136	212.931952	173.826098	181.810663	-3.361566	-3.361566	-3.361
121	2026-02-28	178.383950	146.266170	223.774836	174.044105	182.599628	4.239062	4.239062	4.239
122	2026-03-31	178.963275	154.199772	229.894551	174.193104	183.433951	14.610295	14.610295	14.610

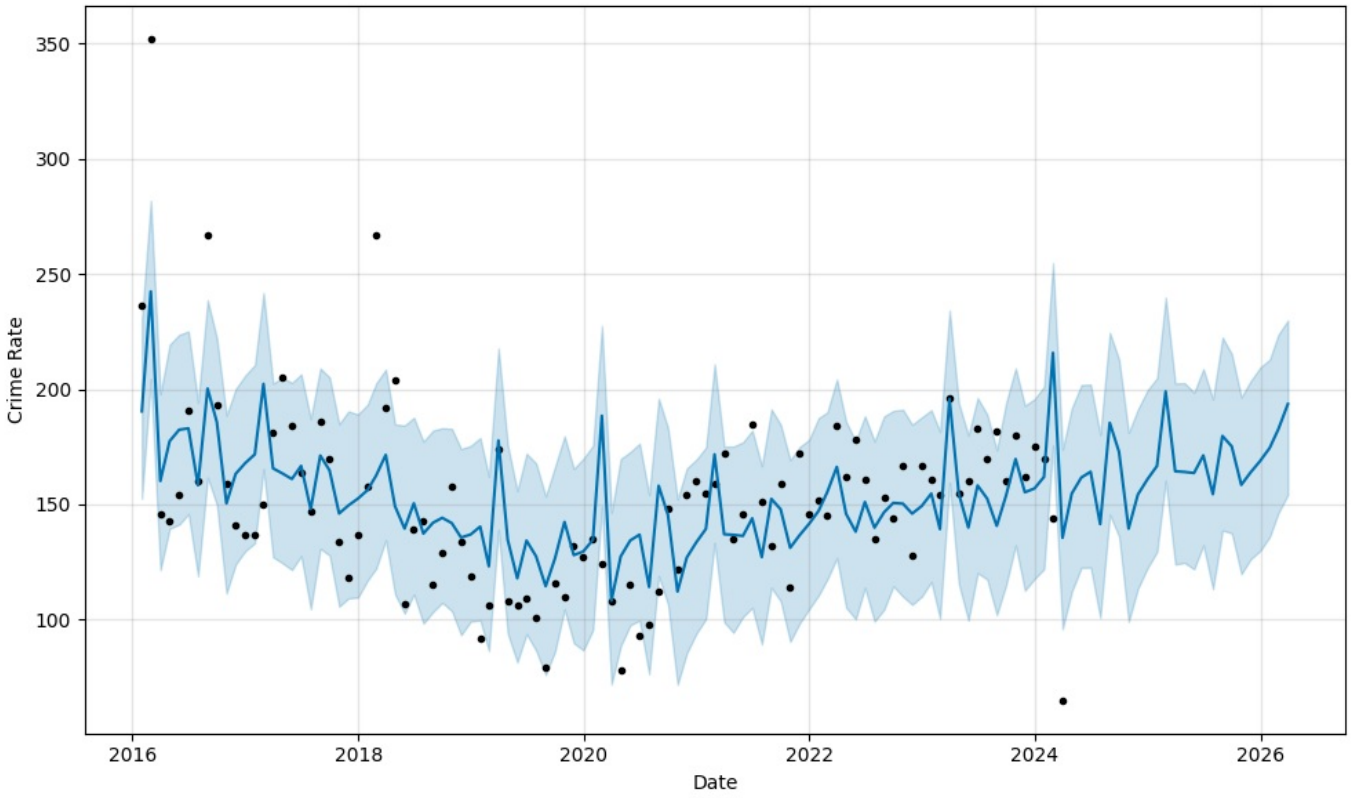
123 rows × 16 columns

In [230…

```
m_Dharwad.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[230…



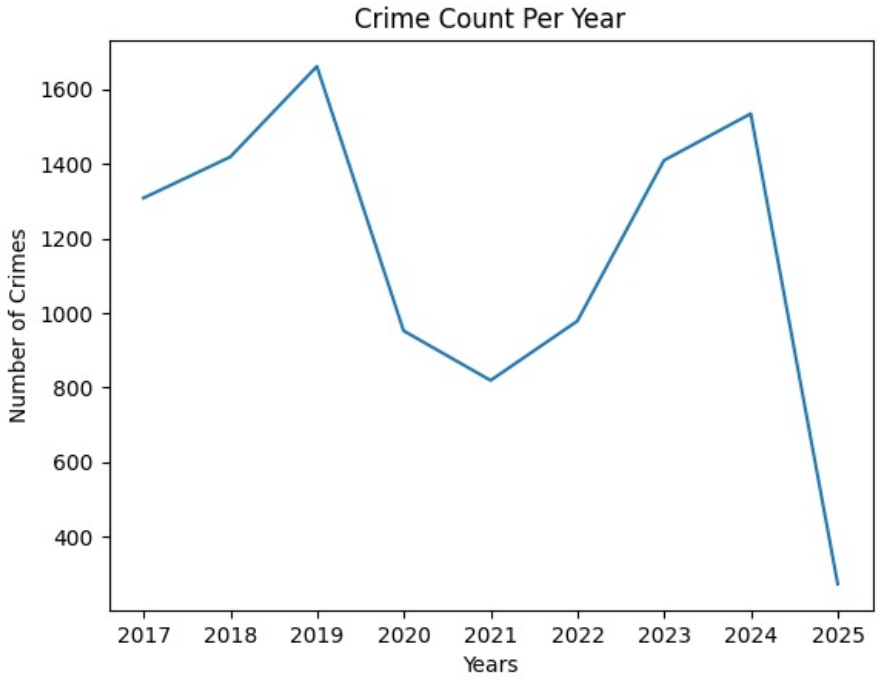


In [232…
```
df_KGF = df[df["District_Name"]=="K.G.F"]
```

In [234…
```
df_KGF.index = pd.DatetimeIndex(df_KGF.FIR_Reg_DateTime)
```

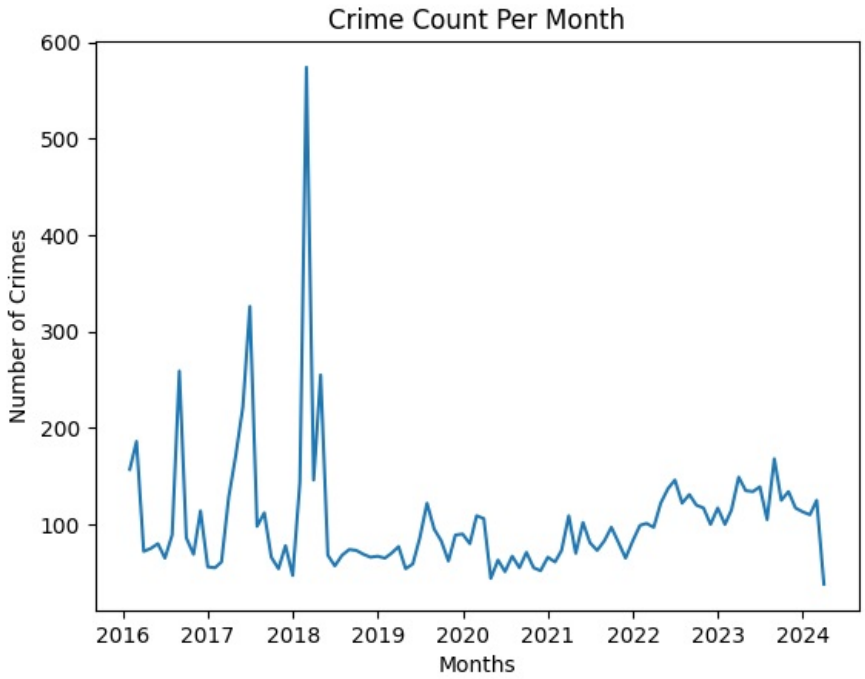
In [235…
```
plt.plot(df_KGF.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[235… Text(0, 0.5, 'Number of Crimes')



In [236…
```
plt.plot(df_KGF.resample('M').size())
plt.title("Crime Count Per Month")
plt.xlabel("Months")
plt.ylabel("Number of Crimes")
```

Out[236…
```
Text(0, 0.5, 'Number of Crimes')
```



In [237…
```
df_KGF_prophet = pd.DataFrame(df_KGF.resample('M').size().reset_index())
```

In [238…
```
df_KGF_prophet.columns =['Date','Crime Count']
```

In [239…
```
df_KGF_prophet=df_KGF_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [240…
```
m_KGF = Prophet()
m_KGF.fit(df_KGF_prophet)
```

```
21:59:06 - cmdstanpy - INFO - Chain [1] start processing
21:59:07 - cmdstanpy - INFO - Chain [1] done processing
```

Out[240… <prophet.forecaster.Prophet at 0x1df82c33cd0>

In [241…
```
pred=m_KGF.make_future_dataframe(periods=24,freq='M')
forcast = m_KGF.predict(pred)
```

In [242…
```
forcast
```

Out[242…

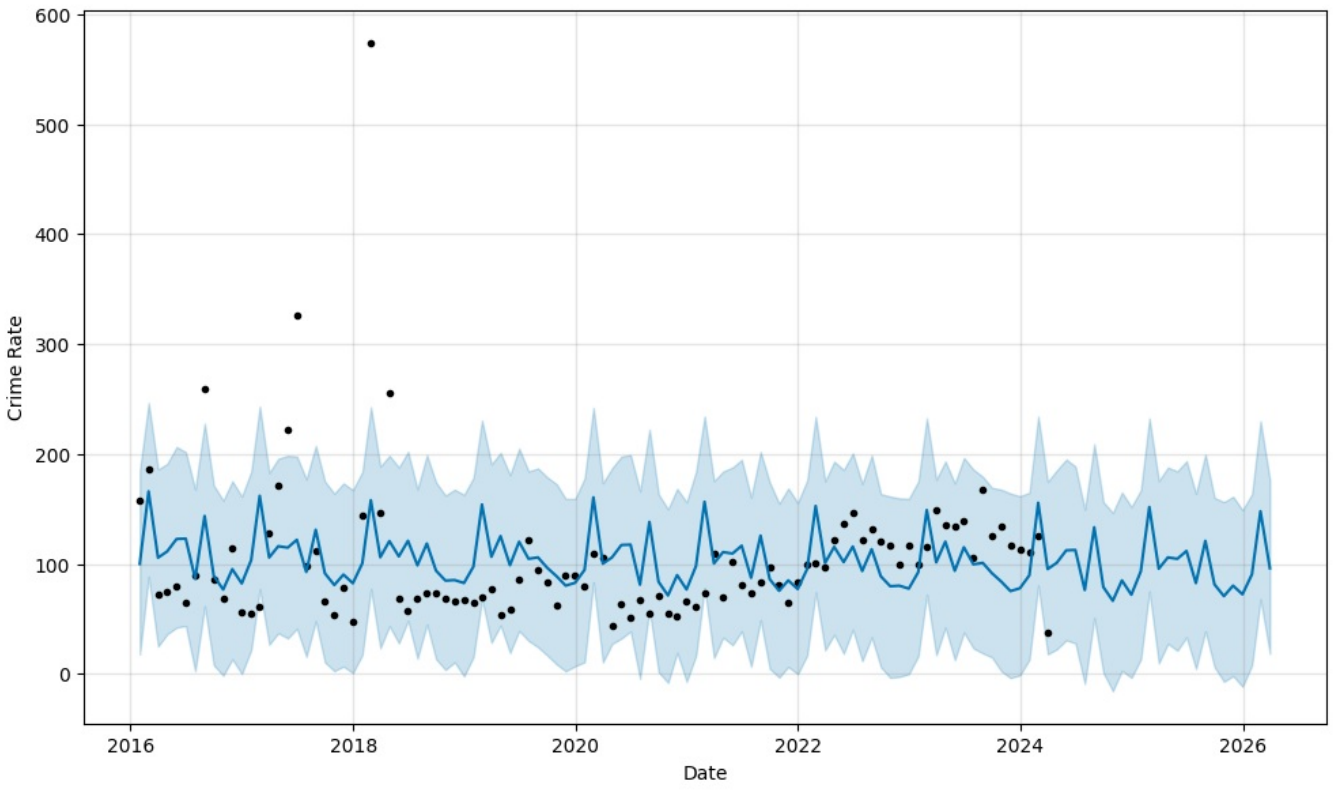
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_upp
0	2016-01-31	107.595152	17.751673	184.937005	107.595152	107.595152	-7.600033	-7.600033	-7.6000
1	2016-02-29	107.486576	88.381214	246.372967	107.486576	107.486576	58.447297	58.447297	58.4472
2	2016-03-31	107.370511	24.900345	185.939062	107.370511	107.370511	-1.759290	-1.759290	-1.7592
3	2016-04-30	107.258191	36.041007	191.136838	107.258191	107.258191	4.056789	4.056789	4.0567
4	2016-05-31	107.142126	42.099197	206.398518	107.142126	107.142126	15.486973	15.486973	15.4869
...	...	...	...	...	...	...	...	...	
118	2025-11-30	94.996862	-1.994342	161.190162	94.973343	95.021914	-14.834983	-14.834983	-14.8349
119	2025-12-31	94.894908	-11.689697	148.976107	94.869062	94.922352	-22.601542	-22.601542	-22.6015
120	2026-01-31	94.792954	7.808991	163.489989	94.764984	94.821623	-4.342374	-4.342374	-4.3423
121	2026-02-28	94.700867	67.888811	229.700890	94.670502	94.731783	53.173947	53.173947	53.1739
122	2026-03-31	94.598913	18.213145	176.110428	94.566653	94.631951	1.395451	1.395451	1.3954

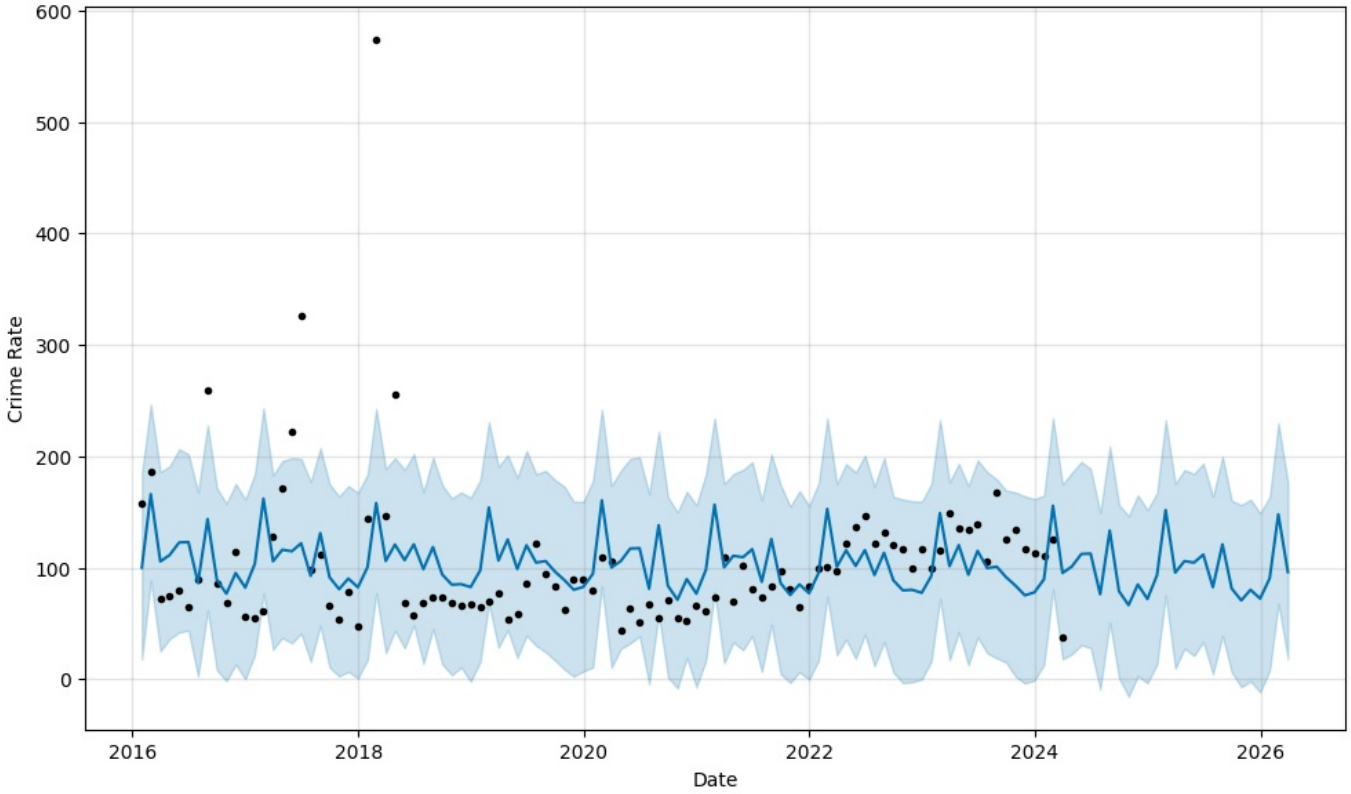
123 rows × 16 columns

In [243…

```
m_KGF.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[243…



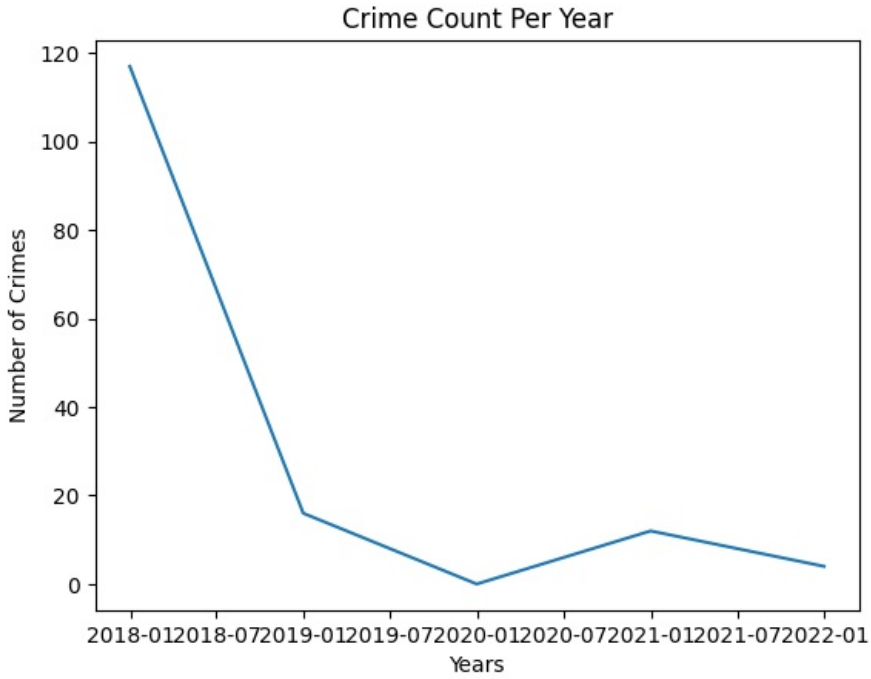


In [244…
```
df_ISDBengaluru = df[df["District_Name"]=="ISD Bengaluru"]
```

In [245…
```
df_ISDBengaluru.index = pd.DatetimeIndex(df_ISDBengaluru.FIR_Reg_DateTime)
```

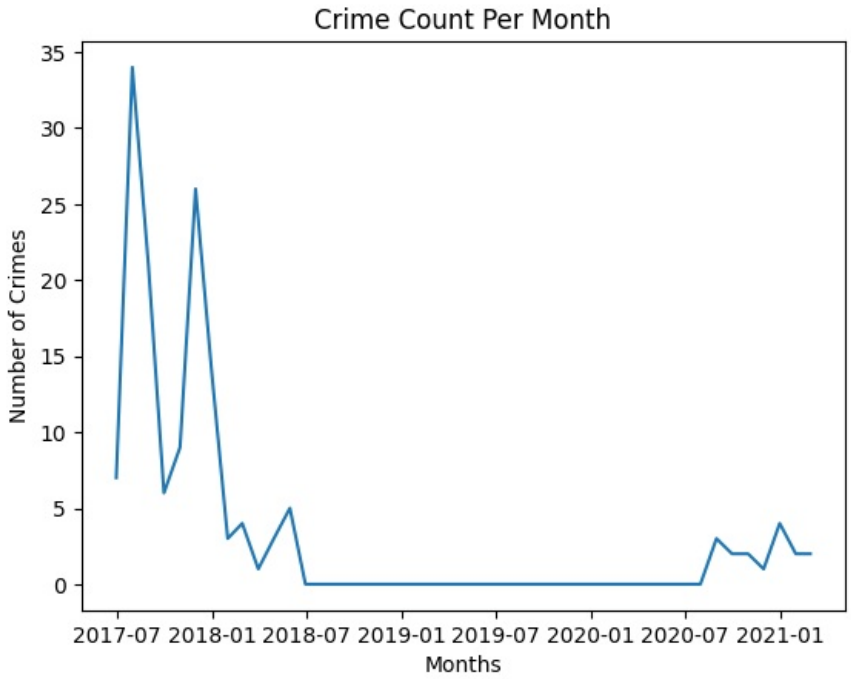
In [246…
```
plt.plot(df_ISDBengaluru.resample('Y').size())
plt.title("Crime Count Per Year")
plt.xlabel("Years")
plt.ylabel("Number of Crimes")
```

Out[246… Text(0, 0.5, 'Number of Crimes')



In [247…
```
plt.plot(df_ISDBengaluru.resample('M').size())
plt.title("Crime Count Per Month")
plt.xlabel("Months")
plt.ylabel("Number of Crimes")
```

Out[247… Text(0, 0.5, 'Number of Crimes')



In [248…
```
df_ISDBengaluru_prophet = pd.DataFrame(df_ISDBengaluru.resample('M').size().reset_index())
```

In [249…
```
df_ISDBengaluru_prophet.columns =['Date','Crime Count']
```

In [250…
```
df_ISDBengaluru_prophet=df_ISDBengaluru_prophet.rename(columns={'Date':'ds','Crime Count':'y'})
```

In [251…
```
m_ISDBengaluru = Prophet()
m_ISDBengaluru.fit(df_ISDBengaluru_prophet)
```

```
22:03:47 - cmdstanpy - INFO - Chain [1] start processing
22:03:47 - cmdstanpy - INFO - Chain [1] done processing
```

Out[251… `<prophet.forecaster.Prophet at 0x1df836d1c10>`

In [252…
```
pred=m_ISDBengaluru.make_future_dataframe(periods=24,freq='M')
forcast = m_ISDBengaluru.predict(pred)
```

In [253…
```
forcast
```

Out[253…

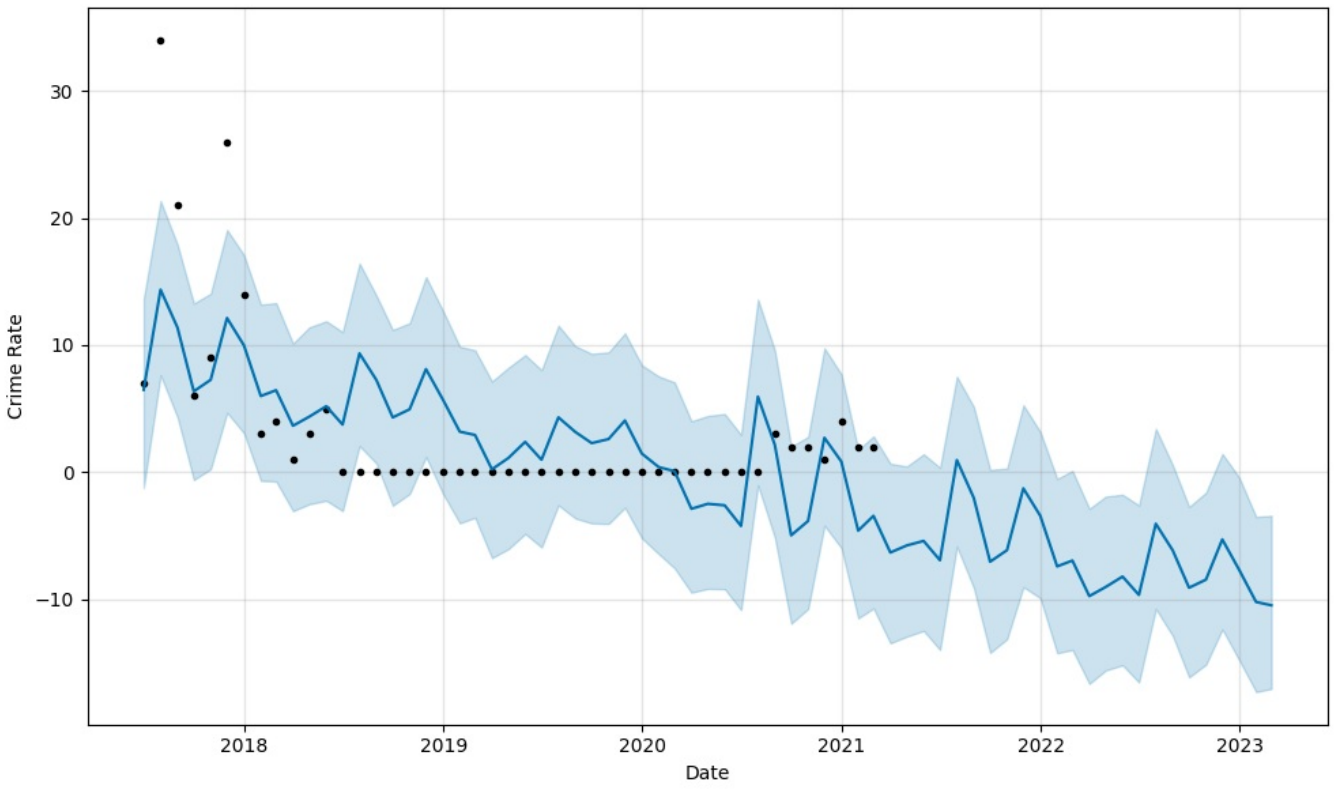
	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_upper
0	2017-06-30	9.311818	-1.220626	13.647572	9.311818	9.311818	-2.831066	-2.831066	-2.831066
1	2017-07-31	9.027093	7.632620	21.341531	9.027093	9.027093	5.337819	5.337819	5.337819
2	2017-08-31	8.742368	4.321718	17.918311	8.742368	8.742368	2.648086	2.648086	2.648086
3	2017-09-30	8.466828	-0.605668	13.283591	8.466828	8.466828	-2.104584	-2.104584	-2.104584
4	2017-10-31	8.182103	0.244362	14.062767	8.182103	8.182103	-0.911106	-0.911106	-0.911106
...	...	...	...	...	...	...	...	...	...
64	2022-10-31	-8.575228	-15.140765	-1.601860	-8.575970	-8.574426	0.119123	0.119123	0.119123
65	2022-11-30	-8.850479	-12.365845	1.452362	-8.851298	-8.849622	3.558038	3.558038	3.558038
66	2022-12-31	-9.134906	-14.798676	-0.417753	-9.135786	-9.133989	1.446831	1.446831	1.446831
67	2023-01-31	-9.419333	-17.292106	-3.513990	-9.420272	-9.418359	-0.795643	-0.795643	-0.795643
68	2023-02-28	-9.676235	-17.048154	-3.423697	-9.677264	-9.675184	-0.799095	-0.799095	-0.799095

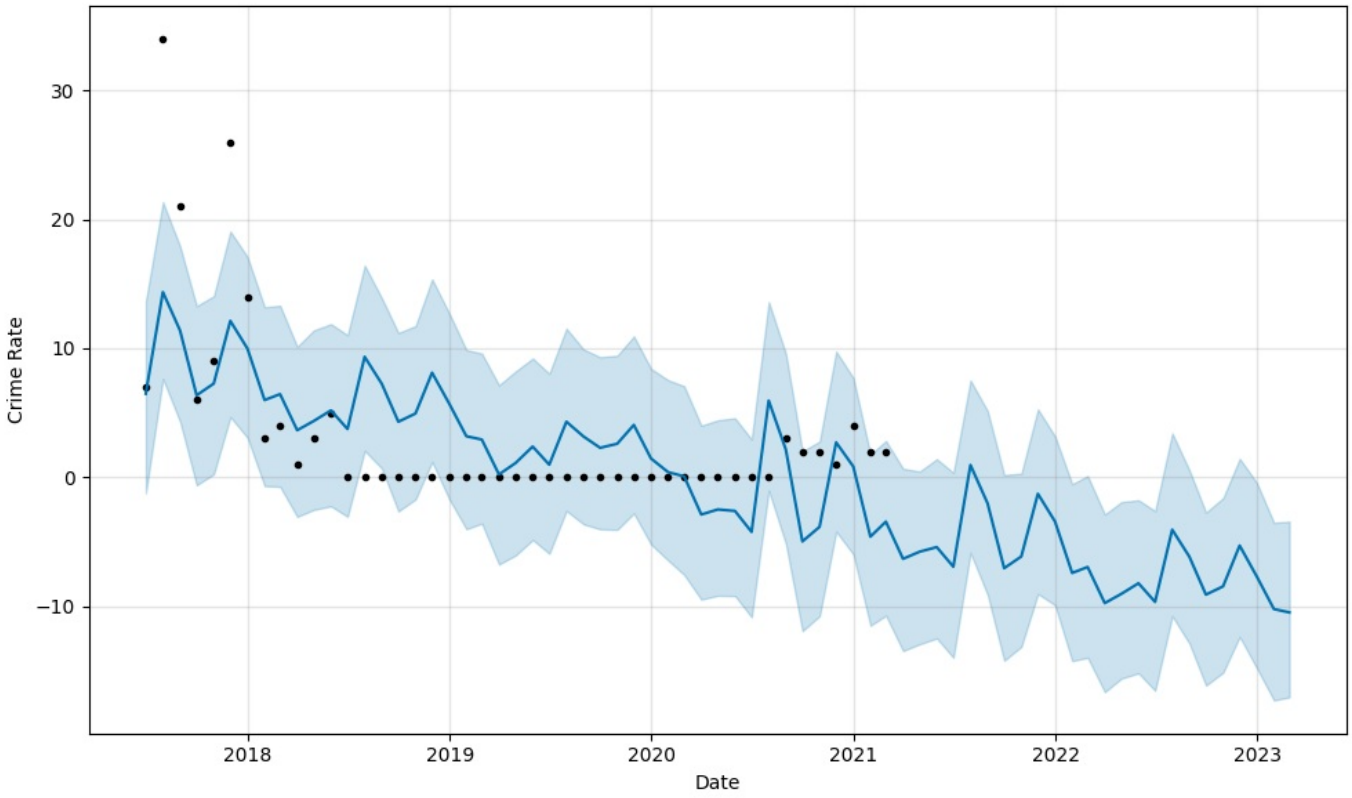
69 rows × 16 columns

In [254…

```
m_ISDBengaluru.plot(forcast,xlabel="Date",ylabel="Crime Rate")
```

Out[254…





In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: