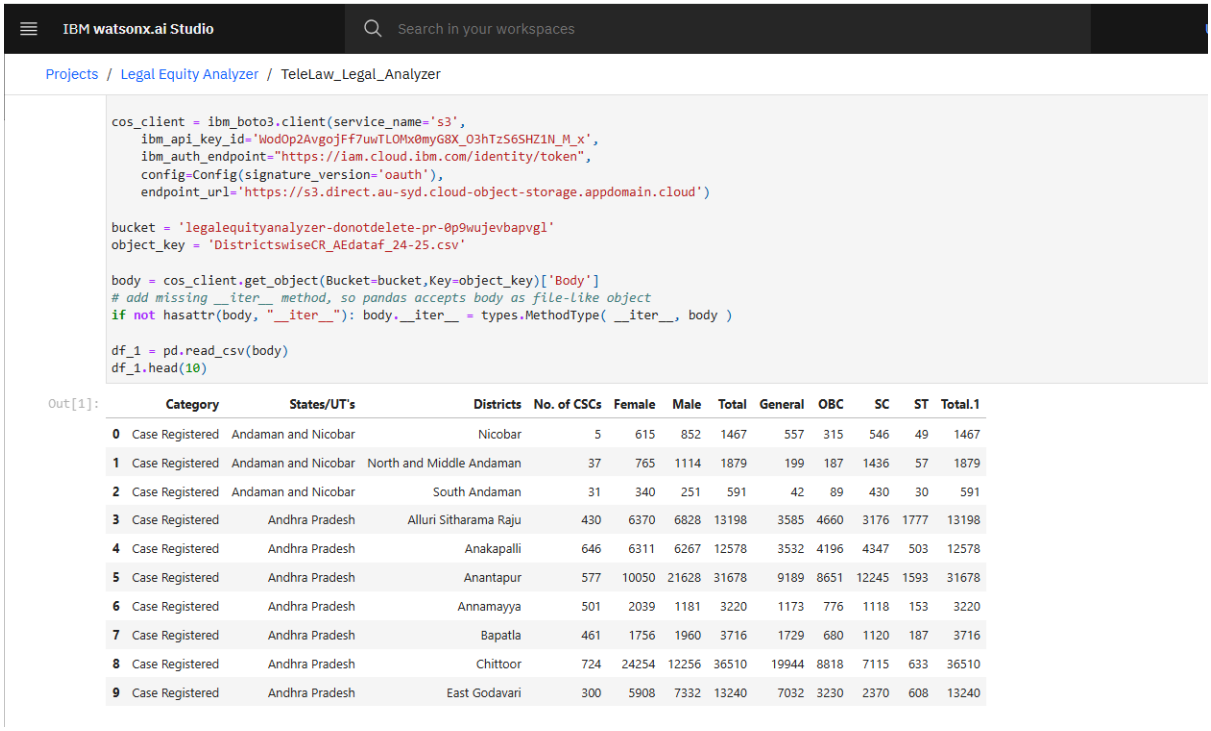
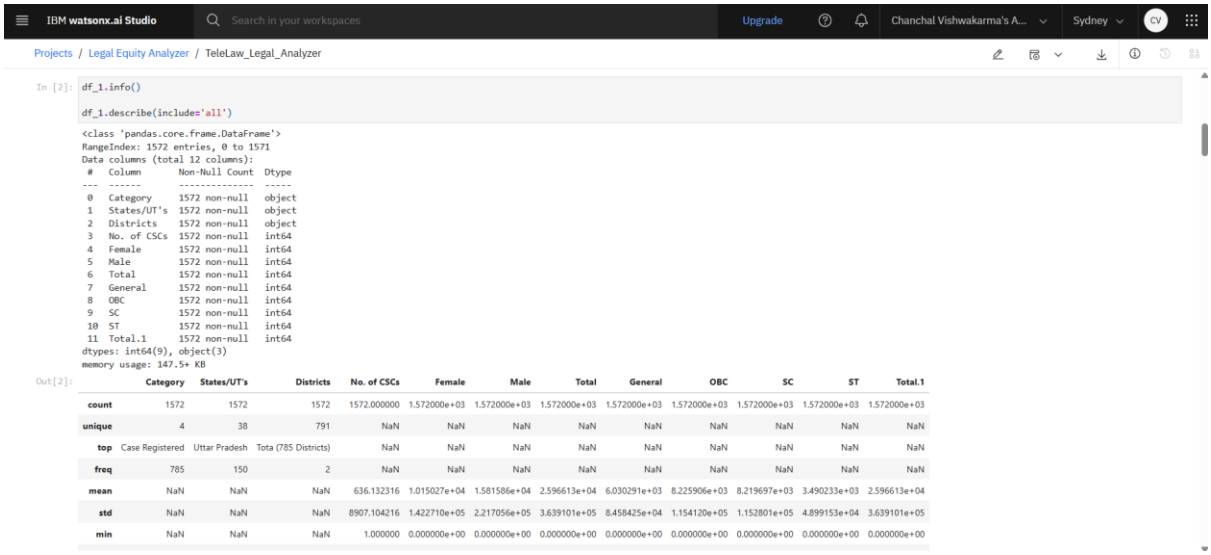
TeleLaw_Legal_Analyzer




```
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='WodOp2AvgojFf7uwTLOMx0myG8X_O3hTzS6SHZ1N_M_x',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/identity/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.direct.au-syd.cloud-object-storage.appdomain.cloud')

bucket = 'legalequityanalyzer-donotdelete-pr-0p9wujevbapvgl'
object_key = 'DistrictswiseCR_AEdataf_24-25.csv'

body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

df_1 = pd.read_csv(body)
df_1.head(10)
```

Out[1]:

	Category	States/UT's	Districts	No. of CSCs	Female	Male	Total	General	OBC	SC	ST	Total.1
0	Case Registered	Andaman and Nicobar	Nicobar	5	615	852	1467	557	315	546	49	1467
1	Case Registered	Andaman and Nicobar	North and Middle Andaman	37	765	1114	1879	199	187	1436	57	1879
2	Case Registered	Andaman and Nicobar	South Andaman	31	340	251	591	42	89	430	30	591
3	Case Registered	Andhra Pradesh	Alluri Sitharama Raju	430	6370	6828	13198	3585	4660	3176	1777	13198
4	Case Registered	Andhra Pradesh	Anakapalli	646	6311	6267	12578	3532	4196	4347	503	12578
5	Case Registered	Andhra Pradesh	Anantapur	577	10050	21628	31678	9189	8651	12245	1593	31678
6	Case Registered	Andhra Pradesh	Annamayya	501	2039	1181	3220	1173	776	1118	153	3220
7	Case Registered	Andhra Pradesh	Bapatla	461	1756	1960	3716	1729	680	1120	187	3716
8	Case Registered	Andhra Pradesh	Chittoor	724	24254	12256	36510	19944	8818	7115	633	36510
9	Case Registered	Andhra Pradesh	East Godavari	300	5908	7332	13240	7032	3230	2370	608	13240



In [2]:
```
df_1.info()

df_1.describe(include='all')
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1572 entries, 0 to 1571
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   Category     1572 non-null   object
 1   States/UT's  1572 non-null   object
 2   Districts    1572 non-null   object
 3   No. of CSCs  1572 non-null   int64
 4   Female       1572 non-null   int64
 5   Male         1572 non-null   int64
 6   Total        1572 non-null   int64
 7   General      1572 non-null   int64
 8   OBC          1572 non-null   int64
 9   SC           1572 non-null   int64
 10  ST           1572 non-null   int64
 11  Total.1      1572 non-null   int64
dtypes: int64(9), object(3)
memory usage: 147.5+ KB
```

Out[2]:

	Category	States/UT's	Districts	No. of CSCs	Female	Male	Total	General	OBC	SC	ST	Total.1
count	1572	1572	1572	1572.000000	1.572000e+03	1.572000e+03	1.572000e+03	1.572000e+03	1.572000e+03	1.572000e+03	1.572000e+03	1.572000e+03
unique	4	38	791	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
top	Case Registered	Uttar Pradesh	Tota (785 Districts)	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
freq	785	150	2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
mean	NaN	NaN	NaN	636.132316	1.015027e+04	1.581586e+04	2.596613e+04	6.030291e+03	8.225906e+03	8.219697e+03	3.490233e+03	2.596613e+04
std	NaN	NaN	NaN	8907.104216	1.422710e+05	2.217056e+05	3.639101e+05	8.458425e+04	1.154120e+05	1.152801e+05	4.899153e+04	3.639101e+05
min	NaN	NaN	NaN	1.000000	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00

IBM watsonx.ai Studio Search in your workspaces Upgra

Projects / Legal Equity Analyzer / TeleLaw_Legal_Analyzer

In [3]:
```
df_1.isnull().sum()
```

Out[3]:
```
Category       0
States/UT's    0
Districts      0
No. of CSCs    0
Female         0
Male           0
Total          0
General        0
OBC            0
SC             0
ST             0
Total.1        0
dtype: int64
```

In [4]:
```
df_1.columns = [col.strip().replace(' ', '_') for col in df_1.columns]
print(df_1.columns)
```

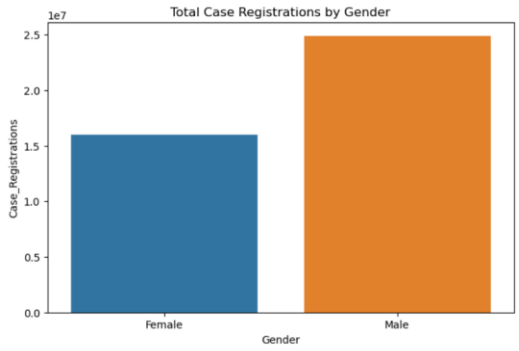
```
Index(['Category', 'States/UT's', 'Districts', 'No._of_CSCs', 'Female', 'Male',
       'Total', 'General', 'OBC', 'SC', 'ST', 'Total.1'],
      dtype='object')
```

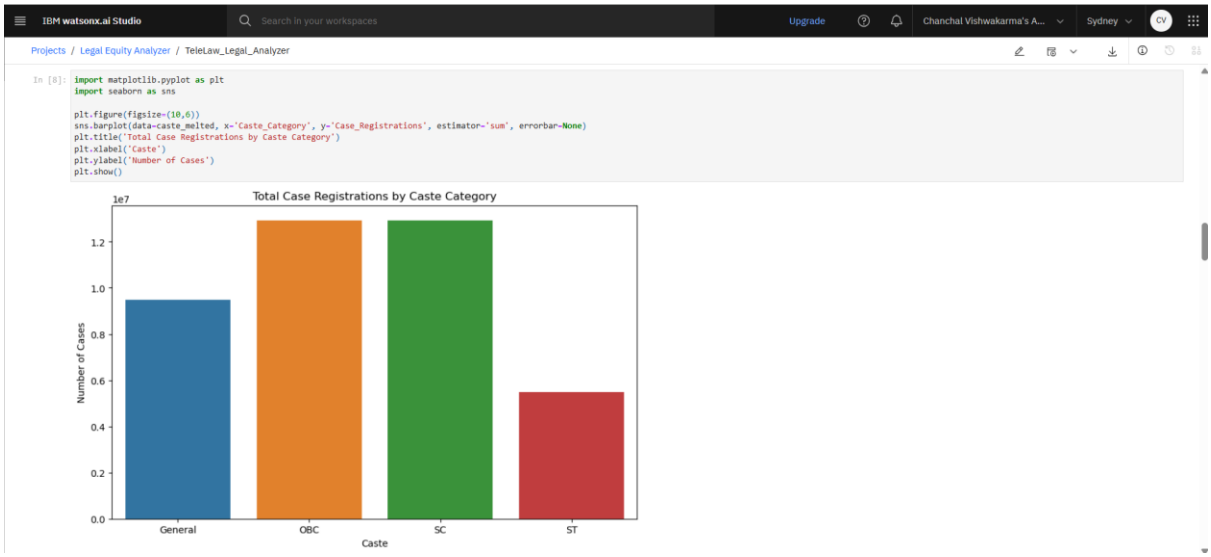
In [5]:
```
gender_df = df_1[['Districts', 'Female', 'Male']].copy()

gender_melted = gender_df.melt(id_vars='Districts',
                               value_vars=['Female', 'Male'],
                               var_name='Gender',
                               value_name='Case_Registrations')
```

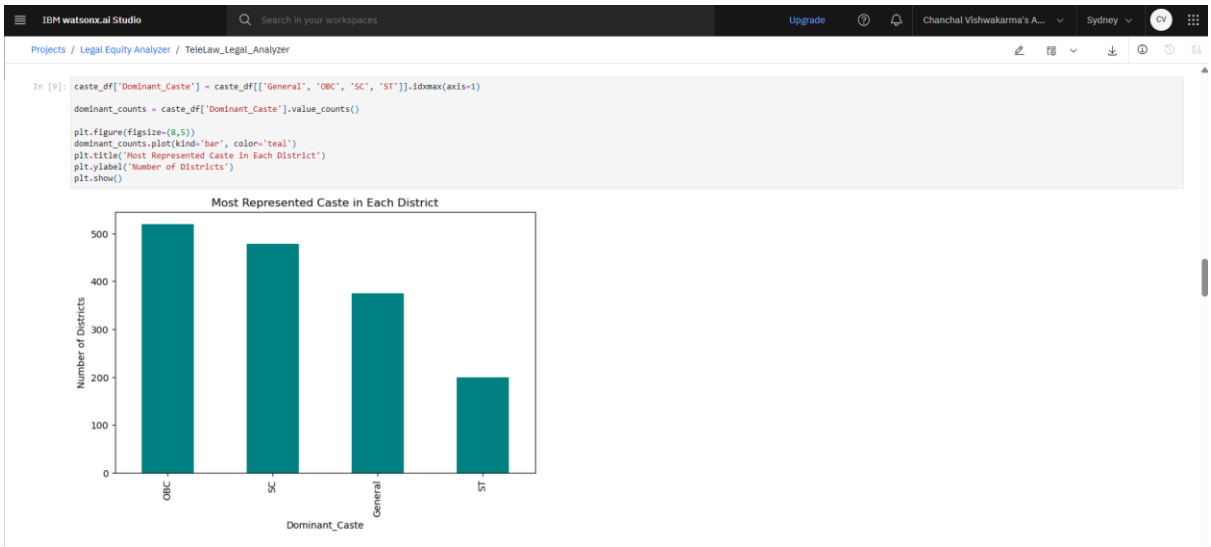
IBM watsonx.ai Studio Search in your workspaces Upgrade Chanchal Vishwakarma's A... Sydney CV

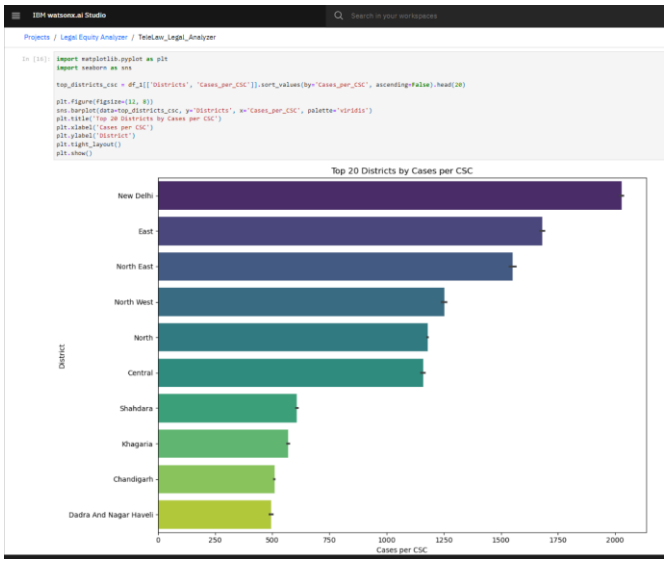
Projects / Legal Equity Analyzer / TeleLaw_Legal_Analyzer

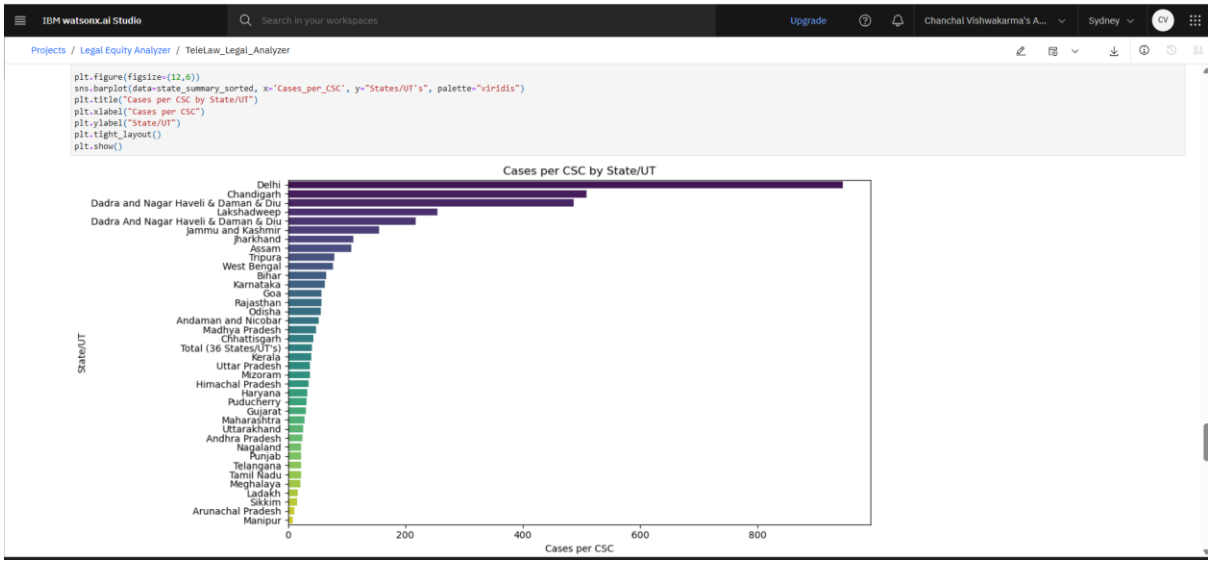
In [6]:
```
import seaborn as sns
import matplotlib.pyplot as plt

plt.figure(figsize=(8,5))
sns.barplot(data=gender_melted, x='Gender', y='Case_Registrations', estimator='sum', errorbar=None)
plt.title('Total Case Registrations by Gender')
plt.show()
```




IBM watsonx.ai Studio
Search in your workspaces
Upgrade
Chanchal Vishwakarma's A...
Sydney
Projects / Legal Equity Analyzer / TeleLaw_Legal_Analyzer
In [8]:
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(10,6))
sns.barplot(data=caste_melted, x='Caste_Category', y='Case_Registrations', estimator='sum', errorbar=None)
plt.title('Total Case Registrations by Caste Category')
plt.xlabel('Caste')
plt.ylabel('Number of Cases')
plt.show()
Total Case Registrations by Caste Category
1e7
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Number of Cases
General
OBC
SC
ST
Caste


IBM watsonx.ai Studio
Search in your workspaces
Upgrade
Chanchal Vishwakarma's A...
Sydney
Projects / Legal Equity Analyzer / TeleLaw_Legal_Analyzer
In [9]:
caste_df['Dominant_Caste'] = caste_df[['General', 'OBC', 'SC', 'ST']].idxmax(axis=1)
dominant_counts = caste_df['Dominant_Caste'].value_counts()
plt.figure(figsize=(8,5))
dominant_counts.plot(kind='bar', color='teal')
plt.title('Most Represented Caste in Each District')
plt.ylabel('Number of Districts')
plt.show()
Most Represented Caste in Each District
500
400
300
200
100
0
Number of Districts
OBC
SC
General
ST
Dominant_Caste

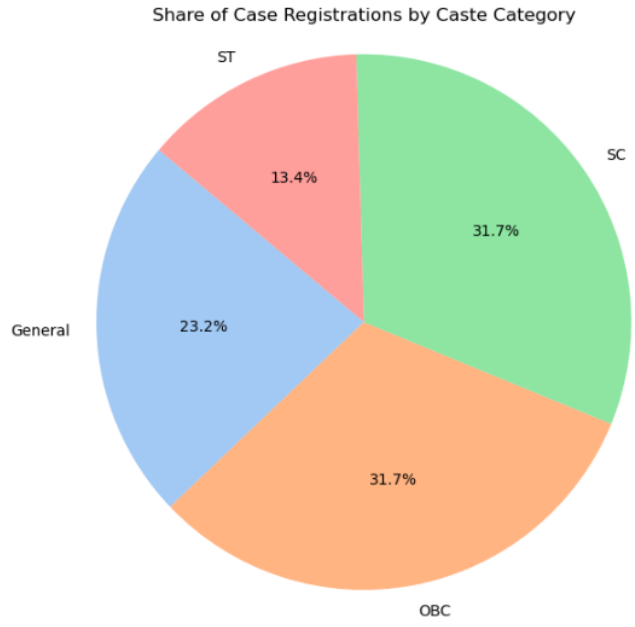

IBM watsonx.ai Studio
Projects / Legal Equity Analyzer / TeleLaw_Legal_Analyzer
In [16]:
import matplotlib.pyplot as plt
import seaborn as sns
top_districts_csc = df_1[['Districts', 'Cases_per_CSC']].sort_values(by='Cases_per_CSC', ascending=False).head(20)
plt.figure(figsize=(12, 8))
sns.barplot(data=top_districts_csc, y='Districts', x='Cases_per_CSC', palette='viridis')
plt.title('Top 20 Districts by Cases per CSC')
plt.xlabel('Cases per CSC')
plt.ylabel('District')
plt.tight_layout()
plt.show()
Top 20 Districts by Cases per CSC
New Delhi
East
North East
North West
North
Central
Shahdara
Khagaria
Chandigarh
Dadra And Nagar Haveli
District
0
250
500
750
1000
1250
1500
1750
2000
Cases per CSC



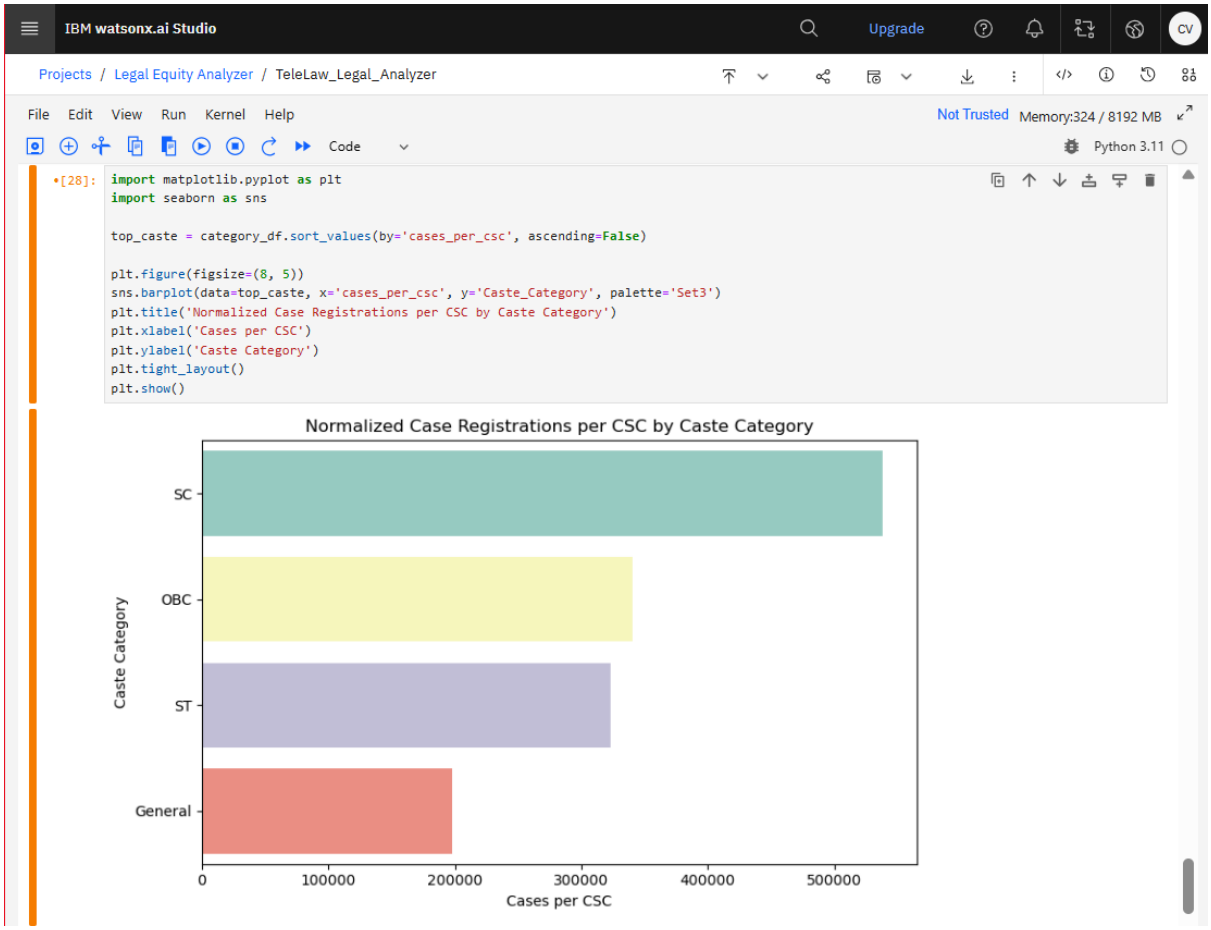
IBM watsonx.ai Studio
Search in your workspaces
Projects / Legal Equity Analyzer / TeleLaw_Legal_Analyzer

In [19]:
```
category_totals = {
    'General': df_1['General'].sum(),
    'OBC': df_1['OBC'].sum(),
    'SC': df_1['SC'].sum(),
    'ST': df_1['ST'].sum()
}

category_df = pd.DataFrame(list(category_totals.items()), columns=['Caste_Category', 'Total_Cases'])
```

In [20]:
```
plt.figure(figsize=(7,7))
plt.pie(category_df['Total_Cases'], labels=category_df['Caste_Category'],
        autopct='%1.1f%%', startangle=140, colors=sns.color_palette('pastel'))
plt.title('Share of Case Registrations by Caste Category')
plt.axis('equal')
plt.show()
```

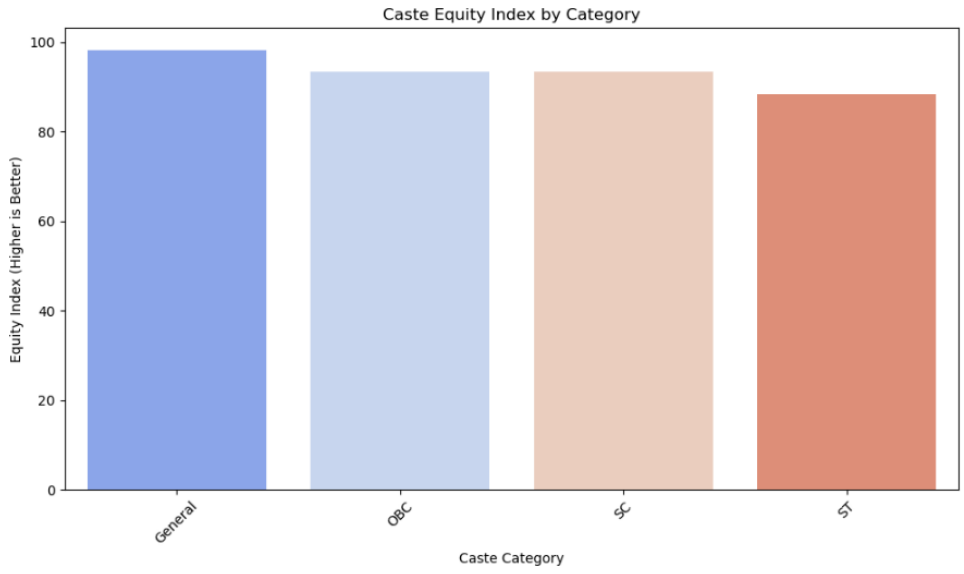


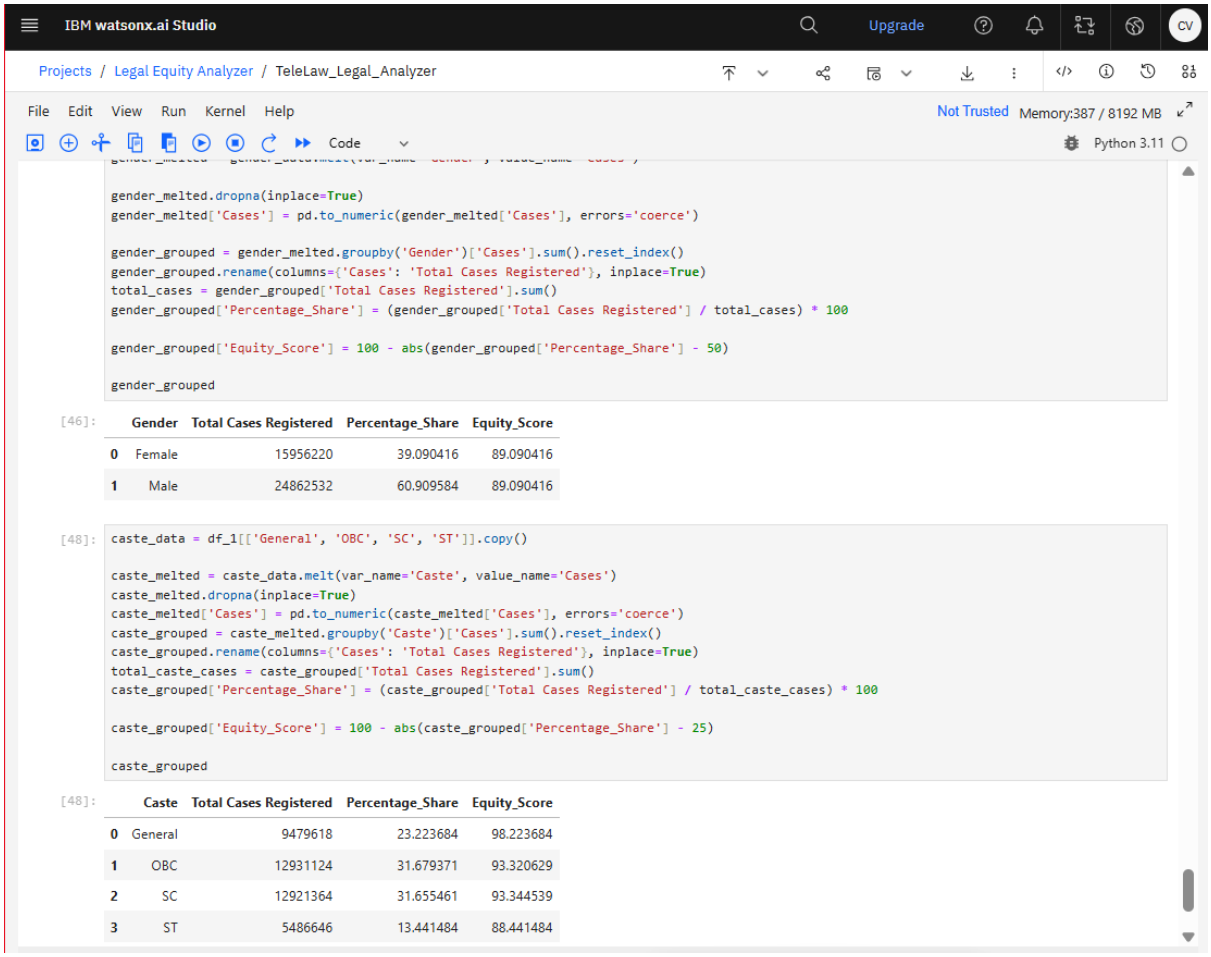

IBM watsonx.ai Studio
Search in your workspaces
Upgrade
Chanchal Vishwakarma's A...
Sydney
Projects / Legal Equity Analyzer / TeleLaw_Legal_Analyzer
In [21]:
plt.figure(figsize=(8,5))
sns.barplot(data=category_df, x='Caste_Category', y='Total_Cases', palette='Set2')
plt.title('Total Case Registrations by Caste Category')
plt.xlabel('Caste Category')
plt.ylabel('Total Cases')
plt.tight_layout()
plt.show()
Total Case Registrations by Caste Category
1e7
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Total Cases
General
OBC
SC
ST
Caste Category

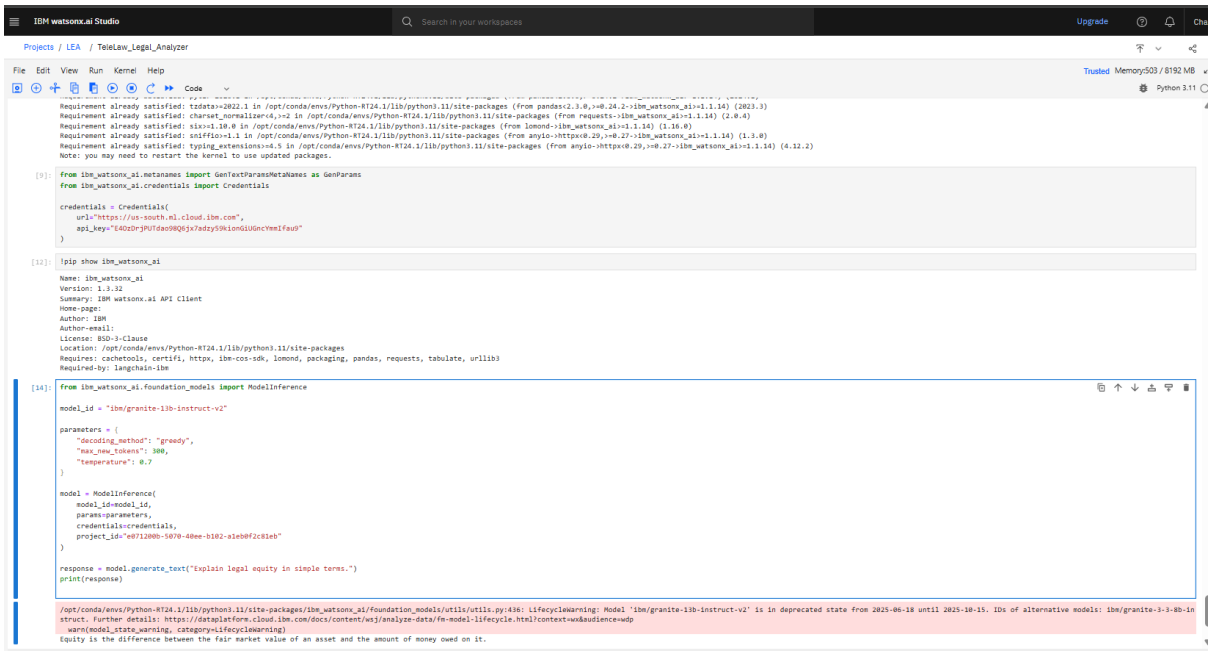

IBM watsonx.ai Studio
Upgrade
CV
Projects / Legal Equity Analyzer / TeleLaw_Legal_Analyzer
File Edit View Run Kernel Help
Not Trusted
Memory:324 / 8192 MB
Code
Python 3.11
•[28]:
import matplotlib.pyplot as plt
import seaborn as sns
top_caste = category_df.sort_values(by='cases_per_csc', ascending=False)
plt.figure(figsize=(8, 5))
sns.barplot(data=top_caste, x='cases_per_csc', y='Caste_Category', palette='Set3')
plt.title('Normalized Case Registrations per CSC by Caste Category')
plt.xlabel('Cases per CSC')
plt.ylabel('Caste Category')
plt.tight_layout()
plt.show()
Normalized Case Registrations per CSC by Caste Category
SC
OBC
ST
General
Caste Category
0
100000
200000
300000
400000
500000
Cases per CSC

```
category_df['Caste_Equity_Index'] = 100 - category_df['Deviation']
```

[35]:
```
plt.figure(figsize=(10,6))
sns.barplot(data=category_df, x='Caste_Category', y='Caste_Equity_Index', palette='coolwarm')
plt.title('Caste Equity Index by Category')
plt.xlabel('Caste Category')
plt.ylabel('Equity Index (Higher is Better)')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```




IBM watsonx.ai Studio
Upgrade
CV
Projects / Legal Equity Analyzer / TeleLaw_Legal_Analyzer
File Edit View Run Kernel Help
Not Trusted Memory:387 / 8192 MB
Code
Python 3.11
gender_melted.dropna(inplace=True)
gender_melted['Cases'] = pd.to_numeric(gender_melted['Cases'], errors='coerce')
gender_grouped = gender_melted.groupby('Gender')['Cases'].sum().reset_index()
gender_grouped.rename(columns={'Cases': 'Total Cases Registered'}, inplace=True)
total_cases = gender_grouped['Total Cases Registered'].sum()
gender_grouped['Percentage_Share'] = (gender_grouped['Total Cases Registered'] / total_cases) * 100
gender_grouped['Equity_Score'] = 100 - abs(gender_grouped['Percentage_Share'] - 50)
gender_grouped
[46]:
Gender Total Cases Registered Percentage_Share Equity_Score
0 Female 15956220 39.090416 89.090416
1 Male 24862532 60.909584 89.090416
[48]:
caste_data = df_1[['General', 'OBC', 'SC', 'ST']].copy()
caste_melted = caste_data.melt(var_name='Caste', value_name='Cases')
caste_melted.dropna(inplace=True)
caste_melted['Cases'] = pd.to_numeric(caste_melted['Cases'], errors='coerce')
caste_grouped = caste_melted.groupby('Caste')['Cases'].sum().reset_index()
caste_grouped.rename(columns={'Cases': 'Total Cases Registered'}, inplace=True)
total_caste_cases = caste_grouped['Total Cases Registered'].sum()
caste_grouped['Percentage_Share'] = (caste_grouped['Total Cases Registered'] / total_caste_cases) * 100
caste_grouped['Equity_Score'] = 100 - abs(caste_grouped['Percentage_Share'] - 25)
caste_grouped
[48]:
Caste Total Cases Registered Percentage_Share Equity_Score
0 General 9479618 23.223684 98.223684
1 OBC 12931124 31.679371 93.320629
2 SC 12921364 31.655461 93.344539
3 ST 5486646 13.441484 88.441484


IBM watsonx.ai Studio
Projects / LEA / TeleLaw_Legal_Analyzer
File Edit View Run Kernel Help
Trusted Memory:503 / 8192 MB
Python 3.11
Requirement already satisfied: tzdata>=2022.1 in /opt/conda/envs/Python-RT24.1/lib/python3.11/site-packages (from pandas<2.3.0,>=0.24.2->ibm_watsonx_ai>=1.1.14) (2023.3)
Requirement already satisfied: charset_normalizer<4,>=2 in /opt/conda/envs/Python-RT24.1/lib/python3.11/site-packages (from requests->ibm_watsonx_ai>=1.1.14) (2.0.4)
Requirement already satisfied: six>=1.10.0 in /opt/conda/envs/Python-RT24.1/lib/python3.11/site-packages (from lomond->ibm_watsonx_ai>=1.1.14) (1.16.0)
Requirement already satisfied: sniffio>=1.1 in /opt/conda/envs/Python-RT24.1/lib/python3.11/site-packages (from anyio->httpx<0.29,>=0.27->ibm_watsonx_ai>=1.1.14) (1.3.0)
Requirement already satisfied: typing_extensions>=4.5 in /opt/conda/envs/Python-RT24.1/lib/python3.11/site-packages (from anyio->httpx<0.29,>=0.27->ibm_watsonx_ai>=1.1.14) (4.12.2)
Note: you may need to restart the kernel to use updated packages.
[9]:
from ibm_watsonx_ai.metanames import GenTextParamsMetaNames as GenParams
from ibm_watsonx_ai.credentials import Credentials
credentials = Credentials(
url="https://us-south.ml.cloud.ibm.com",
api_key="E4OzDrjPUTdao98Q6jx7adzy59kionGlUGncYmmlfau9"
)
[12]:
!pip show ibm_watsonx_ai
Name: ibm_watsonx_ai
Version: 1.3.32
Summary: IBM watsonx.ai API Client
Home-page:
Author: IBM
Author-email:
License: BSD-3-Clause
Location: /opt/conda/envs/Python-RT24.1/lib/python3.11/site-packages
Requires: cachetools, certifi, httpx, ibm-cos-sdk, lomond, packaging, pandas, requests, tabulate, urllib3
Required-by: langchain-ibm
[14]:
from ibm_watsonx_ai.foundation_models import ModelInference
model_id = "ibm/granite-13b-instruct-v2"
parameters = {
"decoding_method": "greedy",
"max_new_tokens": 300,
"temperature": 0.7
}
model = ModelInference(
model_id=model_id,
params=parameters,
credentials=credentials,
project_id="e071200b-5070-40ee-b102-a1eb0f2c81eb"
)
response = model.generate_text("Explain legal equity in simple terms.")
print(response)
/opt/conda/envs/Python-RT24.1/lib/python3.11/site-packages/ibm_watsonx_ai/foundation_models/utils/utils.py:436: LifecycleWarning: Model 'ibm/granite-13b-instruct-v2' is in deprecated state from 2025-06-18 until 2025-10-15. IDs of alternative models: ibm/granite-3-3-8b-instruct. Further details: https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/fm-model-lifecycle.html?context=wx&audience=wdp
warn(model_state_warning, category=LifecycleWarning)
Equity is the difference between the fair market value of an asset and the amount of money owed on it.