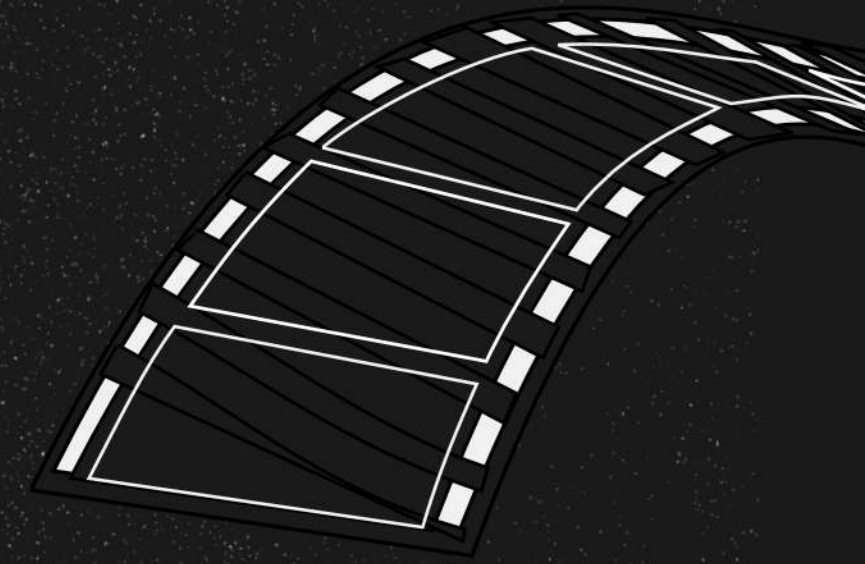
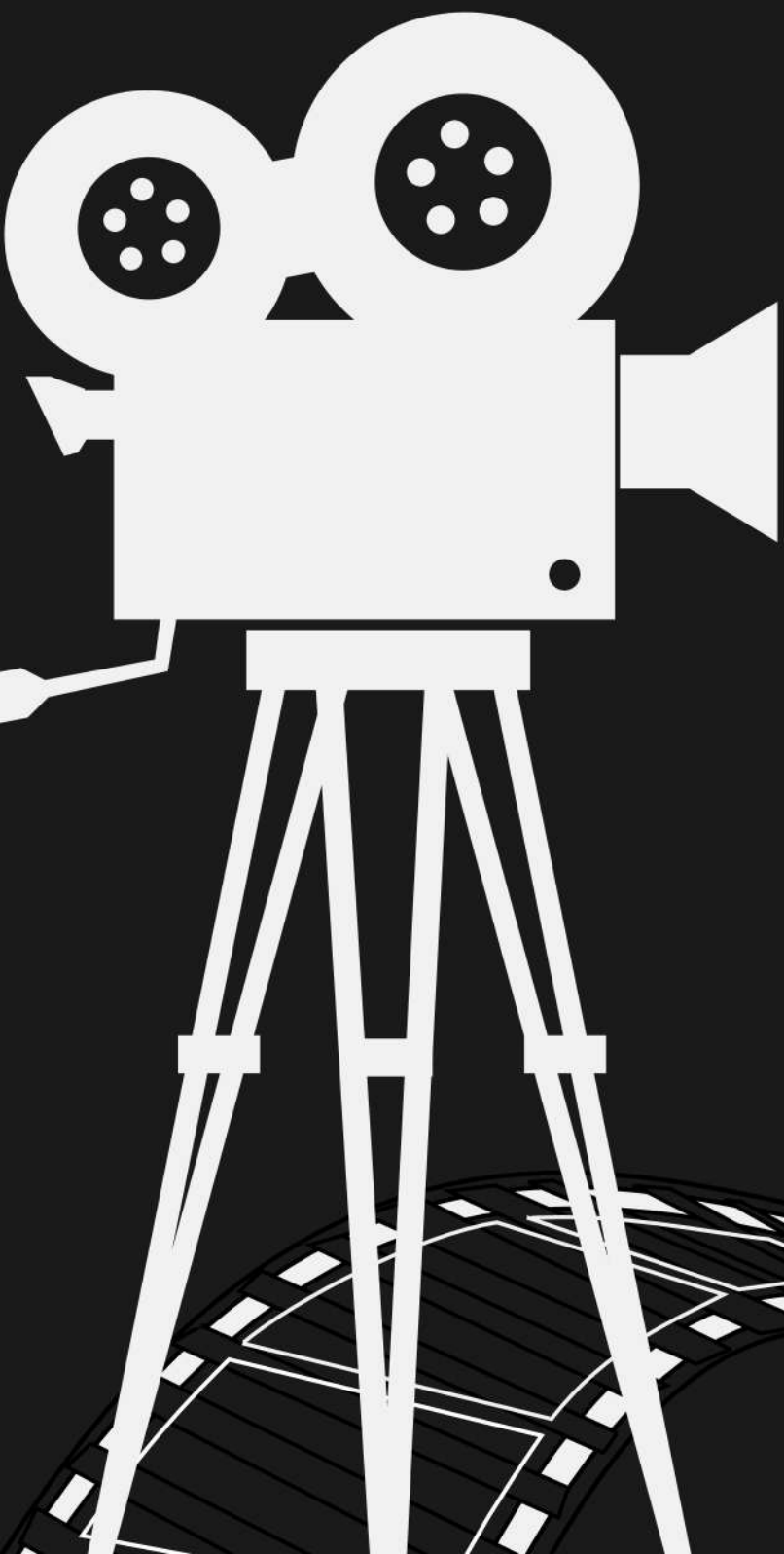


# INDIA IMDB

by ឌីណូ



# สมาชิก

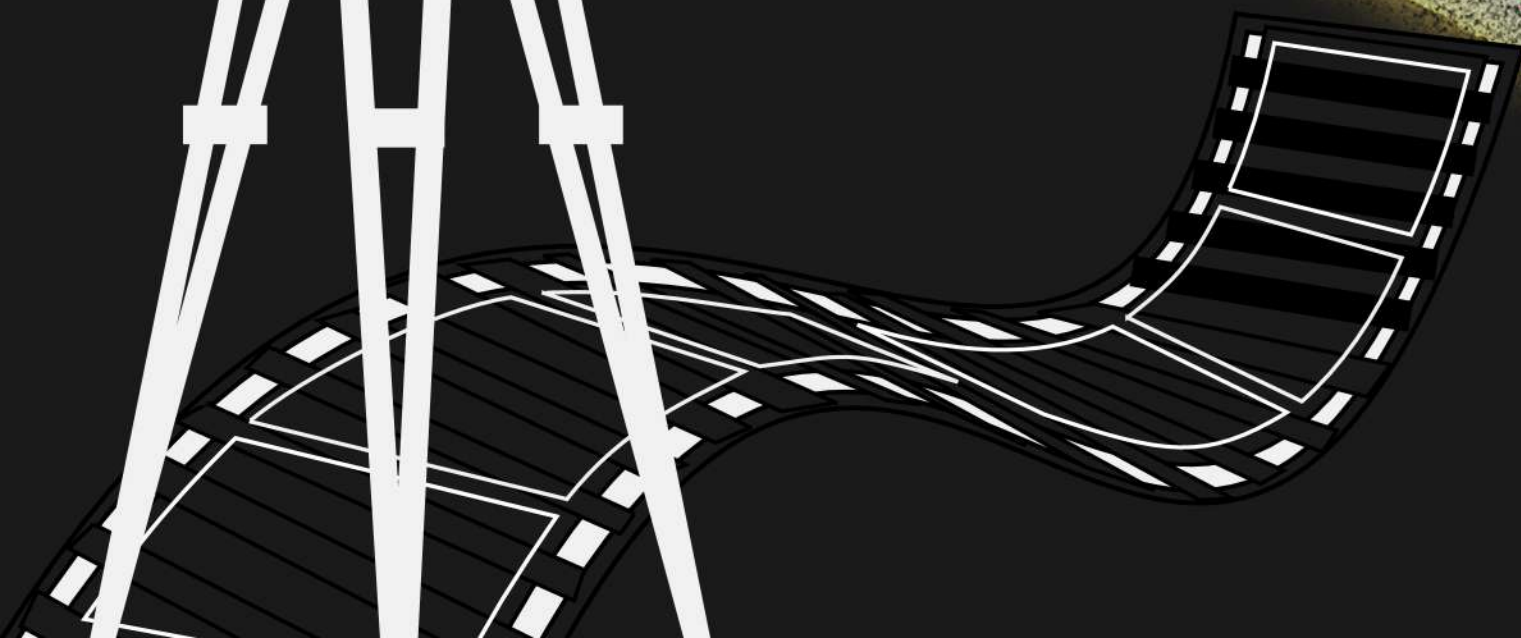


5 นั้ชชา ปะทะมชาติ

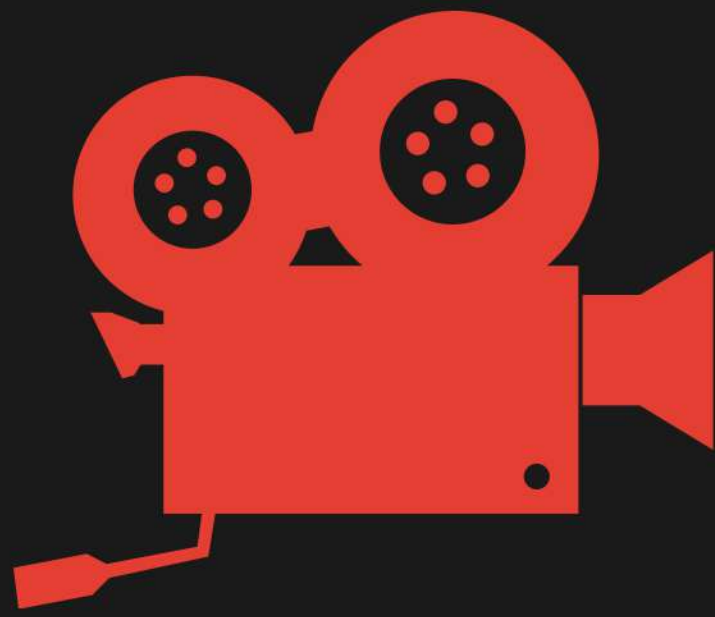
12 พรรณรณณ ราชคณน

15 รัชชานนท พันภาพสินธุ์

27 อริสรา ย้งออยู่







# Hw. 9

## สร้าง RADAR CHART

ผู้กำกับที่มีคะแนนสูงสุด 5 อันดับแรก จากข้อมูลหนัง INDIA

G1

กราฟแสดงจำนวนหนังที่ผู้กำกับแต่ละคนสร้างขึ้น

G2

จำนวนโหวตเฉลี่ยที่ผู้กำกับแต่ละคนได้จากหนังที่สร้างขึ้น

G3

คะแนนเฉลี่ยที่ผู้กำกับแต่ละคนได้จากหนังที่สร้างขึ้น

G4

ความยาวหนังเฉลี่ยของหนังที่ผู้กำกับแต่ละคนสร้างขึ้น

G5

ประเภทหนังที่ผู้กำกับแต่ละคนสร้างขึ้น

# การจัดการข้อมูล

```
[8] #ตรวจสอบค่า null  
data_india.isnull().sum()
```



	0
Name	0
Year	528
Duration	8269
Genre	1877
Rating	7590
Votes	7589
Director	525
Actor 1	1617
Actor 2	2384
Actor 3	3144

dtype: int64

```
[9] data_india = data_india.dropna(subset=['Rating', 'Votes', 'Duration', 'Genre', 'Actor 1', 'Actor 2'])
```



# การจัดการข้อมูล

```
data_india['Year'] = data_india['Year'].str.replace('(', '').str.replace(')', '').astype(int)
data_india['Duration'] = data_india['Duration'].str.replace(' min', '').astype(int)
data_india['Votes'] = data_india['Votes'].str.replace(',', '')
```

```
# ตรวจสอบและแปลงคอลัมน์ Votes เป็น float
data_india['Votes'] = pd.to_numeric(data_india['Votes'], errors='coerce')
```

data\_india

	Name	Year	Duration	Genre	Rating	Votes	Director	Actor 1	Actor 2	Actor 3
1	#Gadhvi (He thought he was Gandhi)	2019	109	Drama	7.0	8	Gaurav Bakshi	Rasika Dugal	Vivek Ghamande	Arvind Jangid
3	#Yaaram	2019	110	Comedy, Romance	4.4	35	Ovais Khan	Prateik	Ishita Raj	Siddhant Kapoor
5	...Aur Pyaar Ho Gaya	1997	147	Comedy, Drama, Musical	4.7	827	Rahul Rawail	Bobby Deol	Aishwarya Rai Bachchan	Shammi Kapoor
6	...Yahaan	2005	142	Drama, Romance, War	7.4	1086	Shoojit Sircar	Jimmy Sheirgill	Minissha Lamba	Yashpal Sharma
8	?: A Question Mark	2012	82	Horror, Mystery, Thriller	5.6	326	Allyson Patel	Yash Dave	Muntazir Ahmad	Kiran Bhatia
...	...	...	...	...	...	...	...	...	...	...

# การจัดการข้อมูล

[16] #หลังdropnaแล้วข้อมูลเหลือกี่%จากเดิม

```
print(f"ข้อมูลเหลือ {(len(data_india) / len(pd.read_csv('/content/drive/MyDrive/data_viz_2024/IMDb Movie
```

➞ ข้อมูลเหลือ 36.79% จากเดิม

▶ data\_india.isnull().sum()

➞

	0
--	---

Name	0
------	---

Year	0
------	---

Duration	0
----------	---

Genre	0
-------	---

Rating	0
--------	---

Votes	0
-------	---

Director	0
----------	---

Actor 1	0
---------	---

Actor 2	0
---------	---

Actor 3	46
---------	----

dtype: int64

# การจัดการข้อมูล

```
▶ # prompt: split value in column genre by ',' and make more column for each of those

import pandas as pd
# Split the 'Genre' column by ',' and create new columns
genre_split = data_india['Genre'].str.split(',', expand=True)

# Rename the new columns
genre_split.columns = ['Genre1', 'Genre2', 'Genre3']

# Concatenate the new columns with the original DataFrame
data_india = pd.concat([data_india, genre_split], axis=1)

# Display the updated DataFrame
data_india.head()
```

```
▶ # prompt: delete space in the value in Genre1 Genre2 Genre3

for col in ['Genre1', 'Genre2', 'Genre3']:
    data_india[col] = data_india[col].str.strip() if data_india[col].dtype == 'object' else data_india[col]
```



# การจัดการข้อมูล

```
# prompt: เลือกผู้กำกับที่มี Rating และ Votes มากกว่า 75%
# top_directors = movie_df[(movie_df['Rating'] >= rating_q3) & (movie_df['Votes'] >= tes_q3)][['Director']].value_counts()
# และ คัดเลือกคะแนนถ่วงน้ำหนัก Votes และ rating ให้ rating 60% และ Votes 40% ถ่วงให้ไม่เกิน 100%
# Calculate the 75th percentile for Rating and Votes
rating_q3 = data_india['Rating'].quantile(0.75)
tes_q3 = data_india['Votes'].quantile(0.75)

# Filter the DataFrame to include only directors with Rating and Votes above the 75th percentile
top_directors = data_india[(data_india['Rating'] >= rating_q3) & (data_india['Votes'] >= tes_q3)][['Director']].value_counts()
top_directors
```

Director	count
Yash Chopra	11
Ram Gopal Varma	10
Hrishikesh Mukherjee	10
Anurag Kashyap	10
Nagesh Kukunoor	8
...	...
Shubhashish Bhutiani	1
Leslee Udwin	1
Abhay Chopra	1
Abbas Tyrewala	1
Milind Dhaimade	1

298 rows × 1 columns



# การจัดการข้อมูล

```
data_india[data_india['Director'] == 'Hrishikesh Mukherjee']
```

	Name	Year	Duration	Genre	Rating	Votes	Director	Actor 1	Actor 2	Actor 3	Genre1	Genre2	Genre3
479	Aashiq	1962	150	Romance	6.7	64	Hrishikesh Mukherjee	Raj Kapoor	Padmini	Nanda	Romance	None	None
487	Aashirwad	1968	146	Drama	7.3	203	Hrishikesh Mukherjee	Ashok Kumar	Sanjeev Kumar	Sumita Sanyal	Drama	None	None
563	Abhimaan	1973	122	Drama, Musical	7.9	2665	Hrishikesh Mukherjee	Amitabh Bachchan	Jaya Bachchan	Asrani	Drama	Musical	None
806	Alaap	1977	160	Drama, Musical	6.7	327	Hrishikesh Mukherjee	Amitabh Bachchan	Rekha	Chhayadevi	Drama	Musical	None

```
# prompt: ดึงข้อมูลของ Yash Chopra, Ram Gopal Varma, Hrishikesh Mukherjee, Anurag Kashyap, Nagesh Kukunoor มาเป็นdataframeใหม่
```

```
directors_to_extract = ['Yash Chopra', 'Ram Gopal Varma', 'Hrishikesh Mukherjee', 'Anurag Kashyap', 'Nagesh Kukunoor']
top = data_india[data_india['Director'].isin(directors_to_extract)].copy()
top
```

	Name	Year	Duration	Genre	Rating	Votes	Director	Actor 1	Actor 2	Actor 3	Genre1	Genre2	Genre3
22	12 O'Clock	2021	105	Horror, Thriller	4.4	63	Ram Gopal Varma	Mithun Chakraborty	Makrand Deshpande	Dalip Tahil	Horror	Thriller	None
73	3 Deewarein	2003	115	Crime, Drama	7.8	2322	Nagesh Kukunoor	Naseeruddin Shah	Jackie Shroff	Juhi Chawla	Crime	Drama	None
125	8 x 10 Tasveer	2009	122	Action, Fantasy, Mystery	5.4	5459	Nagesh Kukunoor	Akshay Kumar	Ayesha Takia	Javed Jaffrey	Action	Fantasy	Mystery



# การจัดการข้อมูล

```
# prompt: Using dataframe top: สร้าง bar นับจำนวน Genre1 Genre2 Genre3 ของ director ใช้ px

import plotly.express as px

# Group by director and count the number of each genre
genre_counts = top.groupby(['Director', 'Genre1'])['Name'].count().reset_index(name='Count')
genre_counts2 = top.groupby(['Director', 'Genre2'])['Name'].count().reset_index(name='Count')
genre_counts3 = top.groupby(['Director', 'Genre3'])['Name'].count().reset_index(name='Count')

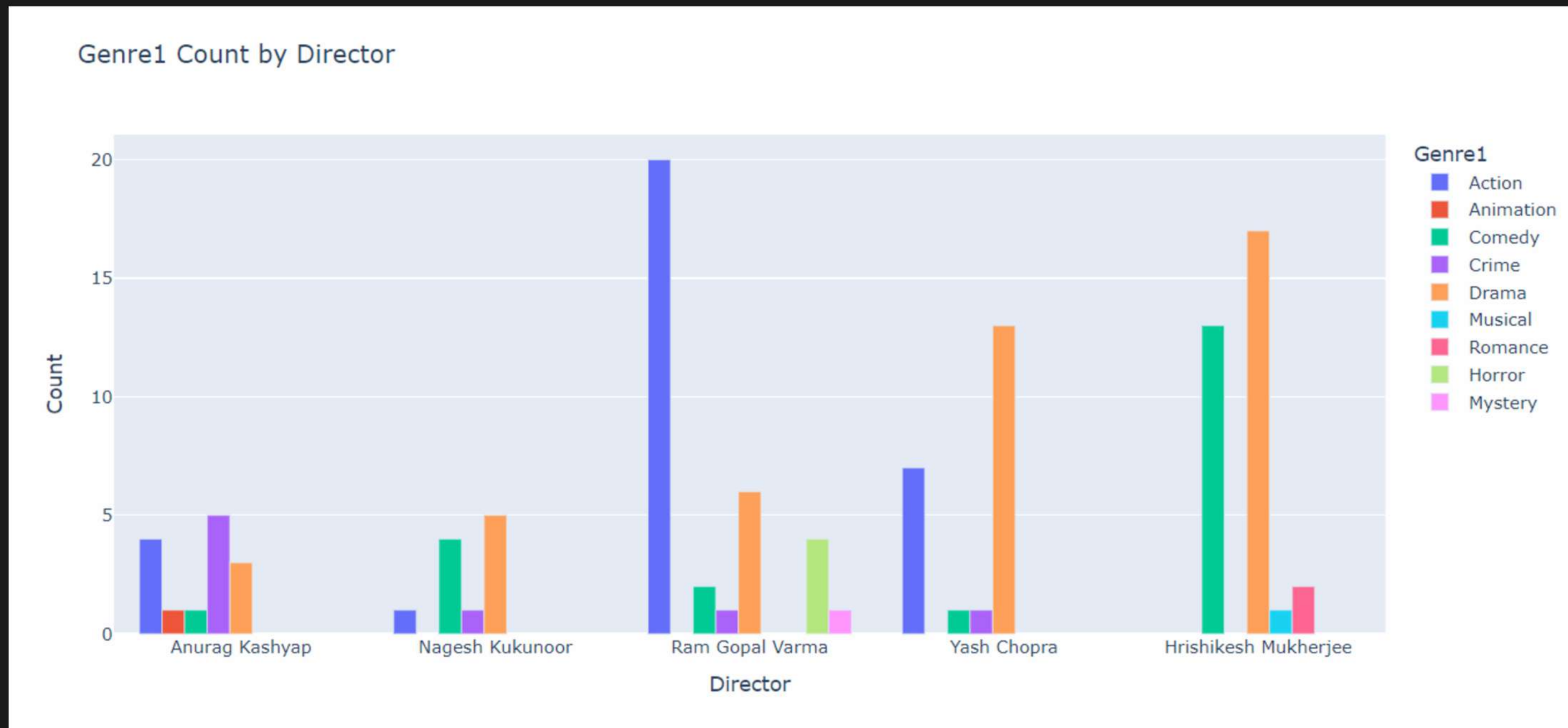
# Create a bar chart using Plotly Express
fig = px.bar(genre_counts, x='Director', y='Count', color='Genre1', barmode='group', title='Genre1 Count by Director')
fig.show()

fig2 = px.bar(genre_counts2, x='Director', y='Count', color='Genre2', barmode='group', title='Genre2 Count by Director')
fig2.show()

fig3 = px.bar(genre_counts3, x='Director', y='Count', color='Genre3', barmode='group', title='Genre3 Count by Director')
fig3.show()
```

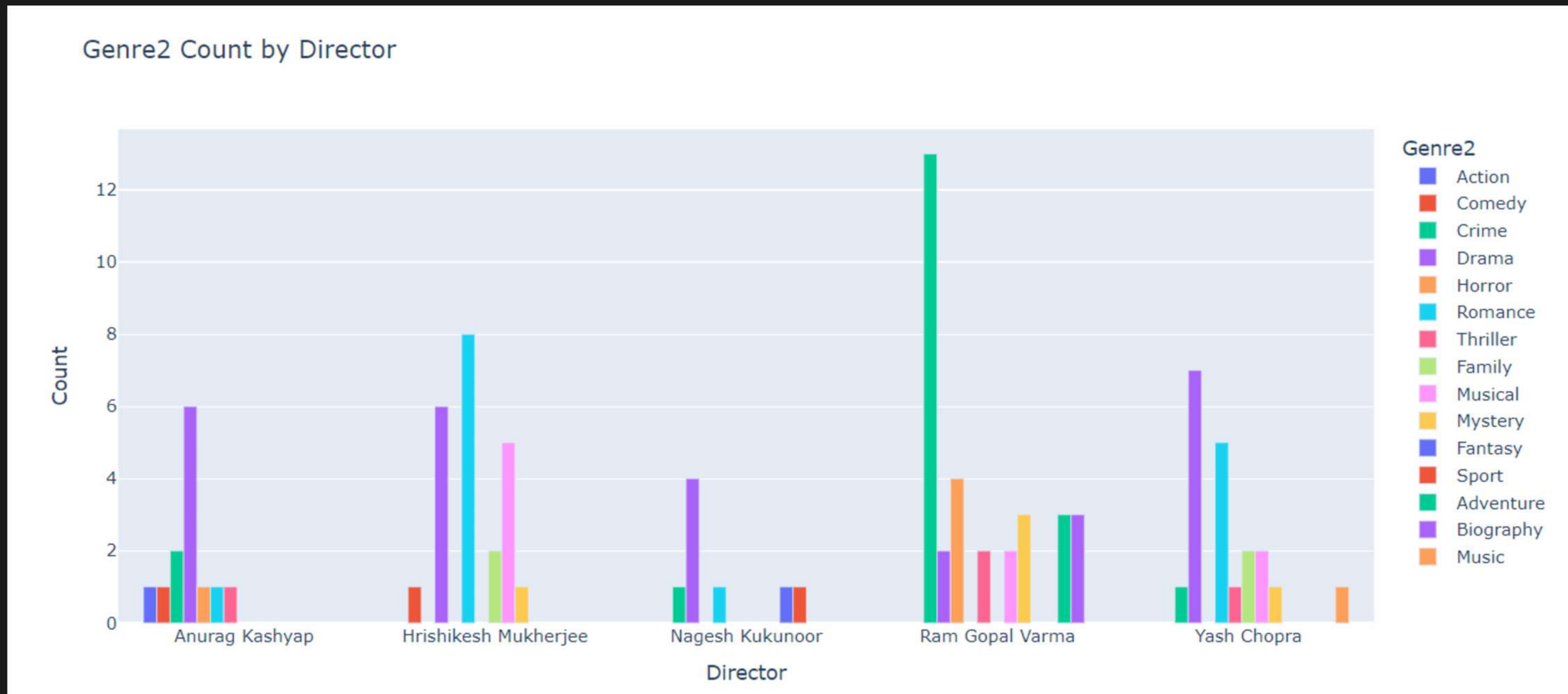


# การจัดการข้อมูล



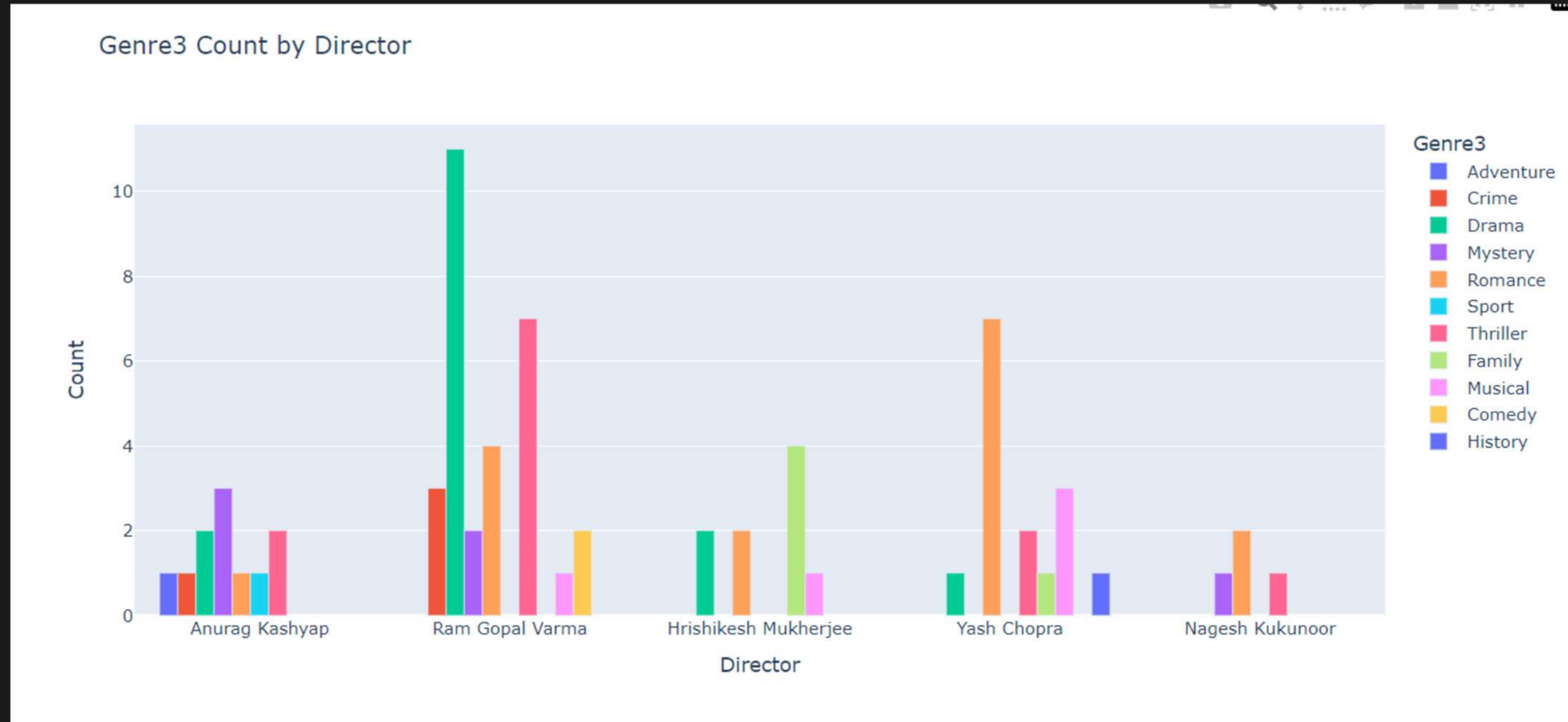


# การจัดการข้อมูล





# การจัดการข้อมูล





# การจัดการข้อมูล

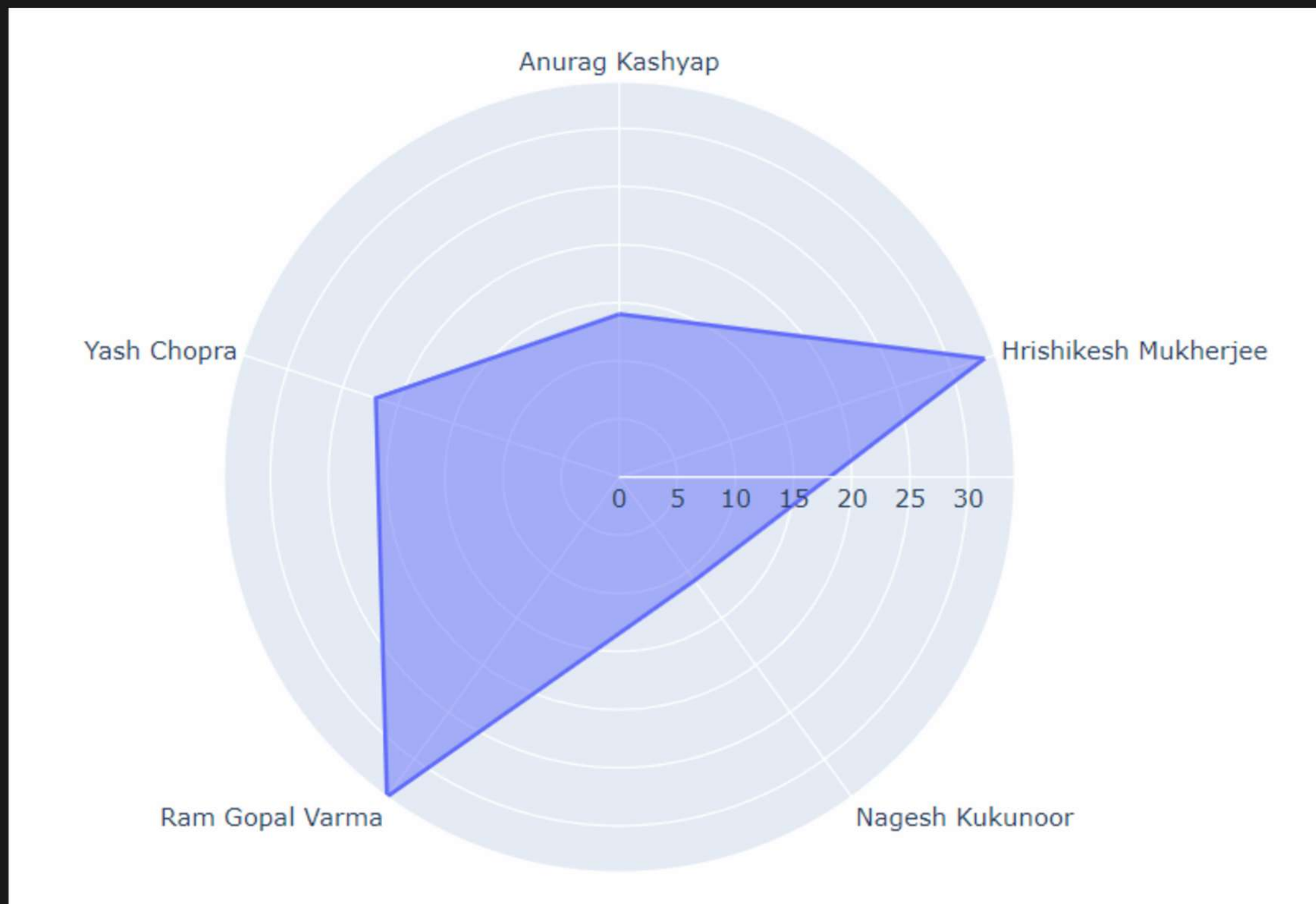
```
# prompt: Using dataframe top: plot radar chart count 'Name' that 'Director' own

# Assuming you want to plot a radar chart showing the count of movies directed by each director.
import plotly.express as px

# Group by 'Director' and count the number of movies directed by each.
director_counts = top.groupby('Director')['Name'].count().reset_index()

# Create the radar chart using Plotly Express.
fig = px.line_polar(director_counts, r='Name', theta='Director', line_close=True)
fig.update_traces(fill='toself')
fig.show()
```

# การจัดการข้อมูล

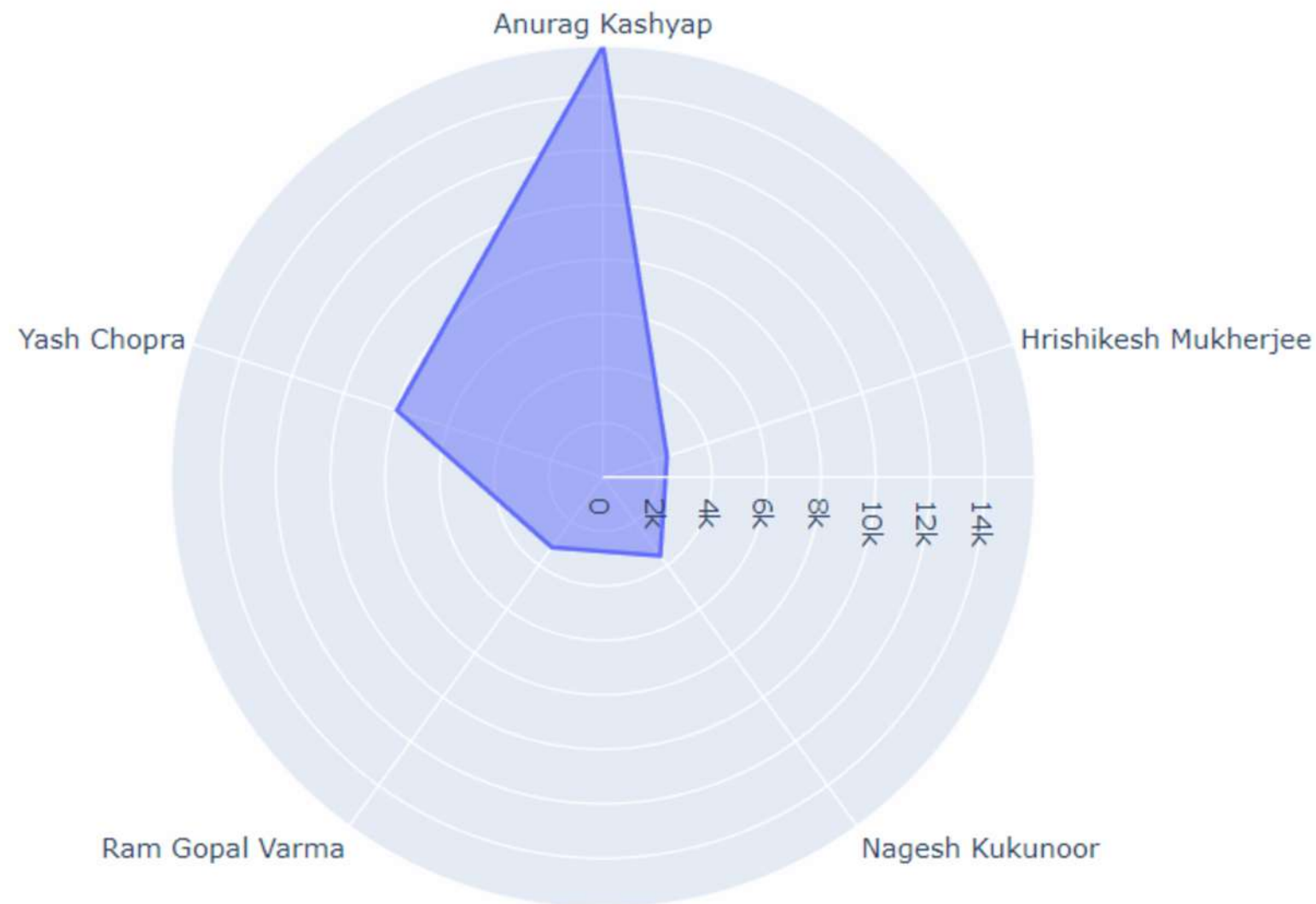




# การจัดการข้อมูล

```
director_votes = top.groupby('Director')['Votes'].mean().reset_index()

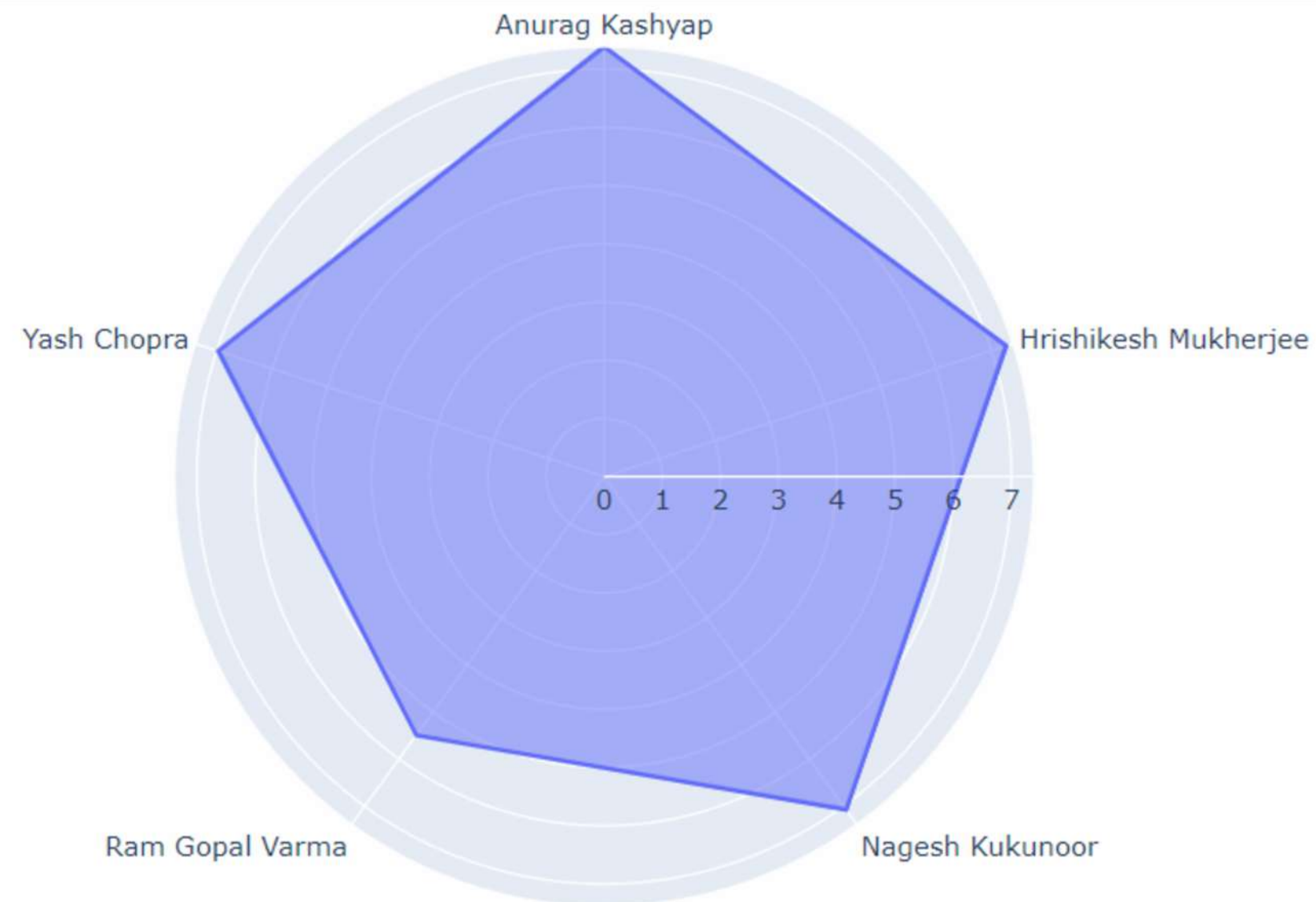
# Create the radar chart using Plotly Express.
fig = px.line_polar(director_votes, r='Votes', theta='Director', line_close=True)
fig.update_traces(fill='toself')
fig.show()
```



# การจัดการข้อมูล

```
director_Rating = top.groupby('Director')['Rating'].mean().reset_index()

# Create the radar chart using Plotly Express.
fig = px.line_polar(director_Rating, r='Rating', theta='Director', line_close=True)
fig.update_traces(fill='toself')
fig.show()
```

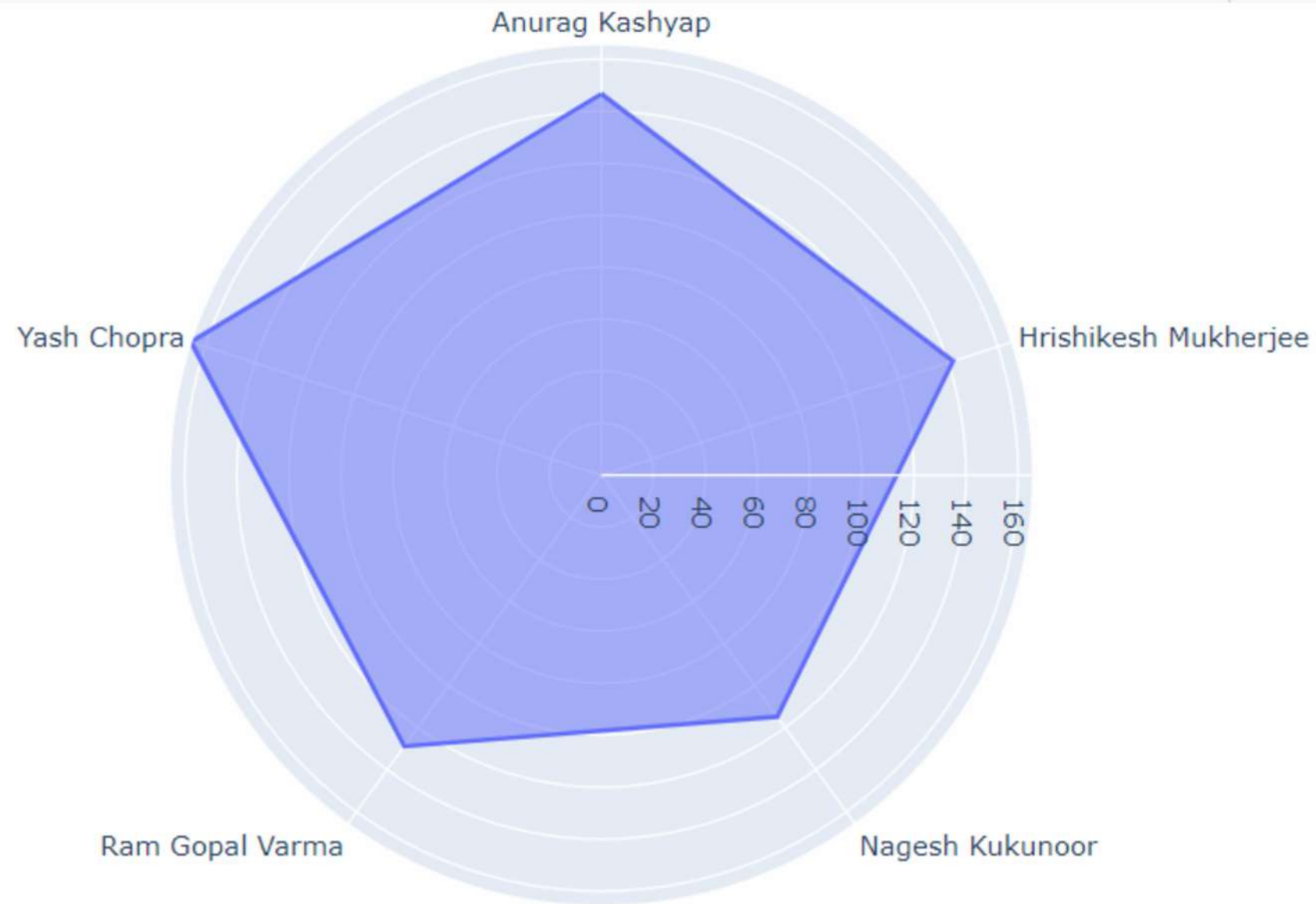




# การจัดการข้อมูล

```
director_Duration = top.groupby('Director')['Duration'].mean().reset_index()

# Create the radar chart using Plotly Express.
fig = px.line_polar(director_Duration, r='Duration', theta='Director', line_close=True)
fig.update_traces(fill='toself')
fig.show()
```



# การจัดการข้อมูล

data\_india



	Name	Year	Duration	Genre	Rating	Votes	Director	Actor 1	Actor 2	Actor 3	Genre1	Genre2	Genre3
1	#Gadhvi (He thought he was Gandhi)	2019	109	Drama	7.0	8	Gaurav Bakshi	Rasika Dugal	Vivek Ghamande	Arvind Jangid	Drama	None	None
3	#Yaaram	2019	110	Comedy, Romance	4.4	35	Ovais Khan	Prateik	Ishita Raj	Siddhant Kapoor	Comedy	Romance	None
5	...Aur Pyaar Ho Gaya	1997	147	Comedy, Drama, Musical	4.7	827	Rahul Rawail	Bobby Deol	Aishwarya Rai Bachchan	Shammi Kapoor	Comedy	Drama	Musical
6	...Yahaan	2005	142	Drama, Romance, War	7.4	1086	Shoojit Sircar	Jimmy Sheirgill	Minissha Lamba	Yashpal Sharma	Drama	Romance	War
8	?: A Question Mark	2012	82	Horror, Mystery, Thriller	5.6	326	Allyson Patel	Yash Dave	Muntazir Ahmad	Kiran Bhatia	Horror	Mystery	Thriller



# การจัดการข้อมูล

```
data_india[data_india['Director'] == 'Saif Ali Sayeed']
```

	Name	Year	Duration	Genre	Rating	Votes	Director	Actor 1	Actor 2	Actor 3	Genre1	Genre2	Genre3
8339	Love Qubool Hai	2020	94	Drama, Romance	10.0	5	Saif Ali Sayeed	Ahaan Jha	Mahesh Narayan	Rajasree Rajakumari	Drama	Romance	None

```
allgenre = list(top['Genre1'])+list(top['Genre2'])+list(top['Genre3'])  
allgenre
```

```
[32] # prompt: allgenreที่ไม่ซ้ำ
```

```
unique_genres = list(set([genre for genre in allgenre if genre is not None]))  
print(unique_genres)
```

```
len(set(allgenre))
```

```
18
```

# Radar Chart

```
import pandas as pd
import matplotlib.pyplot as plt
from math import pi

df = pd.DataFrame(top)

# รายการแนวนอนที่ต้องการพิจารณา
unique_genres = list(set([genre for genre in allgenre if genre is not None]))
# คำนวณจำนวนหนังในแต่ละ Genre ที่ผู้กำกับแต่ละคนทำ
director_genre_count = {director: {genre: 0 for genre in unique_genres} for director in df['Director'].unique()}

# นับจำนวนหนังของผู้กำกับแต่ละคนในแต่ละ Genre
for _, row in df.iterrows():
    movie_genres = {row['Genre1'], row['Genre2'], row['Genre3']}
    movie_genres = {g for g in movie_genres if g != ''} # เอา genre ที่ไม่ว่าง

    director = row['Director']
    for genre in movie_genres:
        if genre in director_genre_count[director]:
            director_genre_count[director][genre] += 1

# เลือกผู้กำกับที่ต้องการเปรียบเทียบ (ตัวอย่าง Director A และ Director B)
#selected_directors = ['Director A', 'Director B'] #This line caused the error
selected_directors = df['Director'].unique()[:5] # Select the first two directors from your dataframe instead
```



# Radar Chart

```
# เตรียมข้อมูลสำหรับ Radar Chart
categories = unique_genres
N = len(categories)

angles = [n / float(N) * 2 * pi for n in range(N)]
angles += angles[:1] # ทำให้กราฟปิดวงกลม

fig, ax = plt.subplots(figsize=(6, 6), subplot_kw=dict(polar=True))

for director in selected_directors:
    values = list(director_genre_count[director].values())
    values += values[:1] # ทำให้กราฟปิดวงกลม

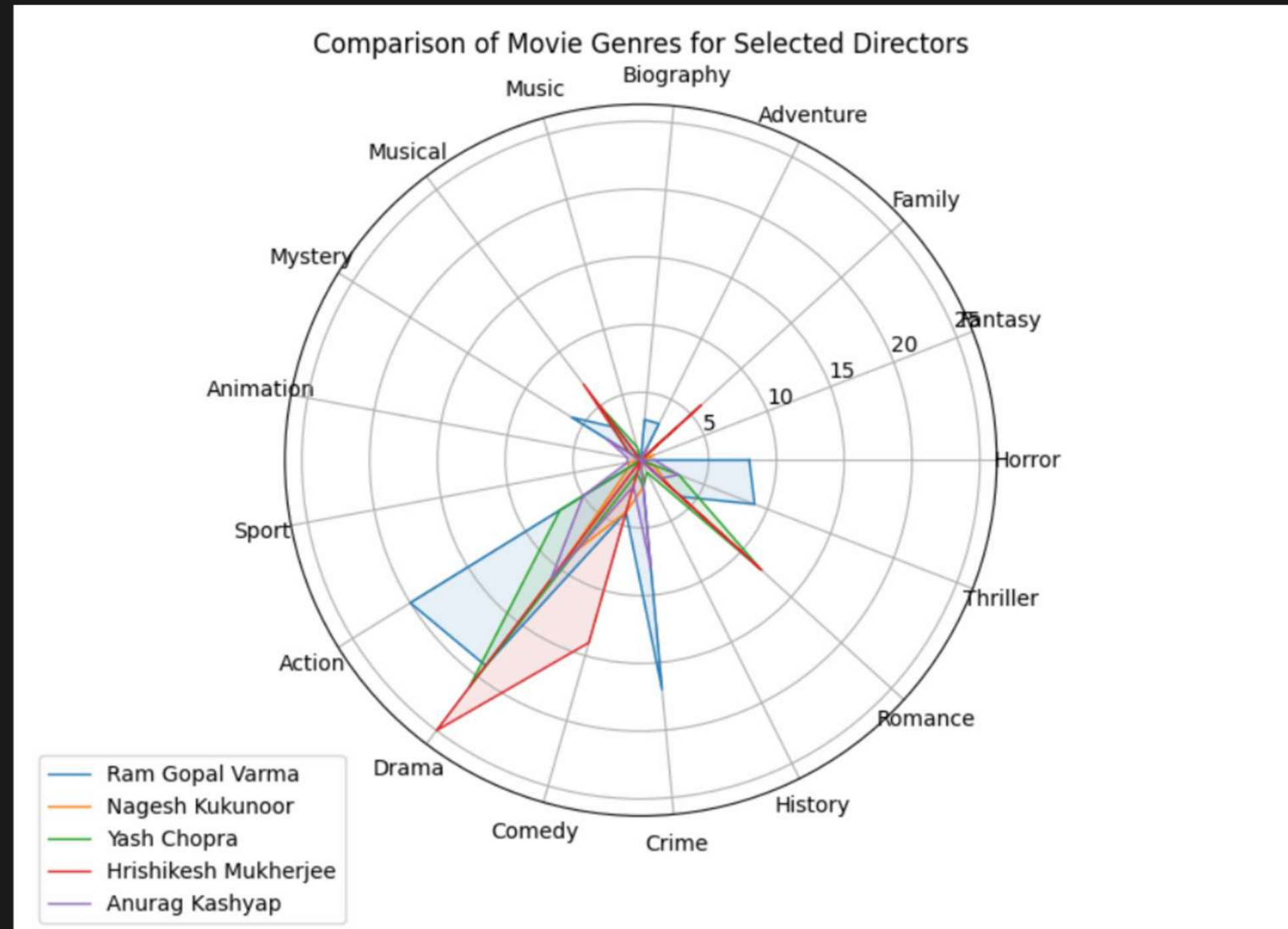
    ax.plot(angles, values, linewidth=1, linestyle='solid', label=director)
    ax.fill(angles, values, alpha=0.1)

# ตั้งค่าแกนและชื่อของแต่ละ Genre
plt.xticks(angles[:-1], categories)

# เพิ่มชื่อเรื่องและแสดงผลลัพธ์
plt.title('Comparison of Movie Genres for Selected Directors')
ax.legend(loc='upper right', bbox_to_anchor=(0.1, 0.1))

plt.show()
```

# Radar Chart





# Radar Chart

```
import pandas as pd
import matplotlib.pyplot as plt
from math import pi

# ตัวอย่างการอ่านข้อมูลจากไฟล์ CSV (ถ้ามีการโหลดข้อมูลจากไฟล์อื่น)
# df = pd.read_csv('your_data.csv')

# สำหรับไฟล์ที่คุณให้มา สามารถอ่านข้อมูลนี้แทนได้:
df = pd.DataFrame(top) # นำข้อมูลจาก 'top' ที่คุณเมื่ออยู่แล้ว

# รายการแนวหนังที่ต้องการพิจารณา
allgenre = df[['Genre1', 'Genre2', 'Genre3']].values.flatten().tolist()
unique_genres = list(set([genre for genre in allgenre if genre != '' and genre is not None]))

# คำนวณจำนวนหนังในแต่ละ Genre ที่ผู้กำกับแต่ละคนทำ
director_genre_count = {director: {genre: 0 for genre in unique_genres} for director in df['Director'].unique()}

# นับจำนวนหนังของผู้กำกับแต่ละคนในแต่ละ Genre
for _, row in df.iterrows():
    movie_genres = {row['Genre1'], row['Genre2'], row['Genre3']}
    movie_genres = {g for g in movie_genres if g != ''} # เอา genre ที่ไม่ว่าง

    director = row['Director']
    for genre in movie_genres:
        if genre in director_genre_count[director]:
            director_genre_count[director][genre] += 1
```

# Radar Chart

```
# เลือกผู้กำกับที่ต้องการเปรียบเทียบ (ตัวอย่างใช้ 5 คนแรก)
selected_directors = df['Director'].unique()[:5]

# เตรียมข้อมูลสำหรับ Radar Chart
categories = unique_genres
N = len(categories)

# สร้างมุมสำหรับการวาด Radar Chart
angles = [n / float(N) * 2 * pi for n in range(N)]
angles += angles[:1] # ทำให้กราฟปิดวงกลม

# สร้างกราฟ
fig, ax = plt.subplots(figsize=(8, 8), subplot_kw=dict(polar=True))

# Loop เพื่อเพิ่มเส้นและพื้นที่สำหรับผู้กำกับแต่ละคน
for director in selected_directors:
    values = list(director_genre_count[director].values())
    values += values[:1] # ทำให้กราฟปิดวงกลม

    ax.plot(angles, values, linewidth=2, linestyle='solid', label=director)
    ax.fill(angles, values, alpha=0.2)

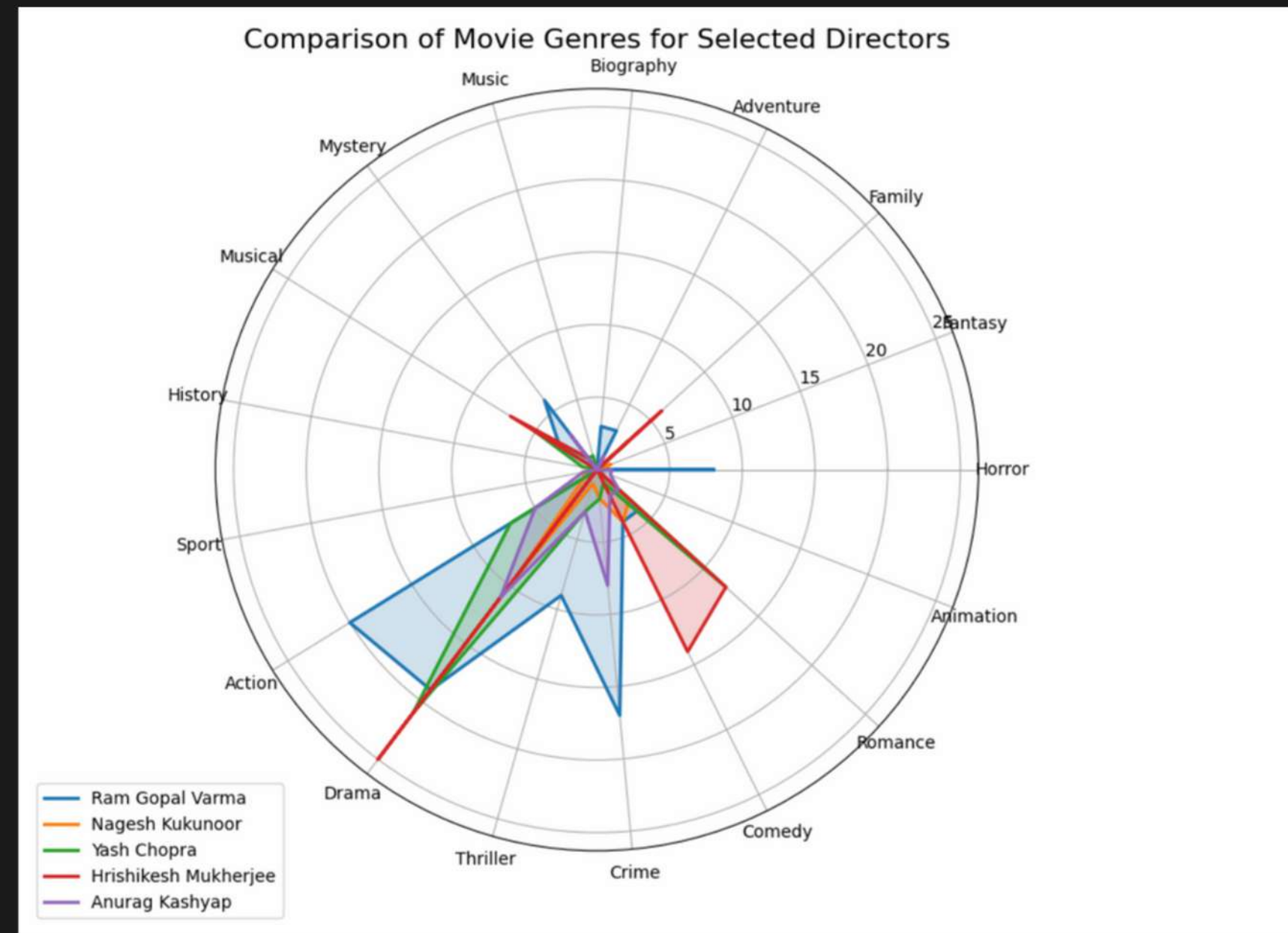
# ตั้งค่าแกนและชื่อของแต่ละ Genre
plt.xticks(angles[:-1], categories)

# เพิ่มชื่อเรื่องและแสดงผลลัพธ์
plt.title('Comparison of Movie Genres for Selected Directors', size=16, color='black')
ax.legend(loc='upper right', bbox_to_anchor=(0.1, 0.1))

# แสดงกราฟ
plt.show()
```



# Radar Chart



# Radar Chart

```
import pandas as pd
import plotly.graph_objects as go

# เตรียมข้อมูล
df = pd.DataFrame(top)

# รายการแนวหนังที่ต้องการพิจารณา
allgenre = df[['Genre1', 'Genre2', 'Genre3']].values.flatten().tolist()
unique_genres = list(set([genre for genre in allgenre if genre != '' and genre is not None]))

# คำนวณจำนวนหนังในแต่ละ Genre ที่ผู้กำกับแต่ละคนทำ
director_genre_count = {director: {genre: 0 for genre in unique_genres} for director in df['Director'].unique()}

# นับจำนวนหนังของผู้กำกับแต่ละคนในแต่ละ Genre
for _, row in df.iterrows():
    movie_genres = {row['Genre1'], row['Genre2'], row['Genre3']}
    movie_genres = {g for g in movie_genres if g != ''} # เอา genre ที่ไม่ว่าง

    director = row['Director']
    for genre in movie_genres:
        if genre in director_genre_count[director]:
            director_genre_count[director][genre] += 1

# เลือกผู้กำกับที่ต้องการเปรียบเทียบ
selected_directors = df['Director'].unique()[:5]

# เตรียมข้อมูลสำหรับ Radar Chart
categories = unique_genres
```



# Radar Chart

```
# สร้างกราฟ Radar แบบโต้ตอบ
fig = go.Figure()

for director in selected_directors:
    values = list(director_genre_count[director].values())
    values += values[:1] # ทำให้กราฟปิดวงกลม

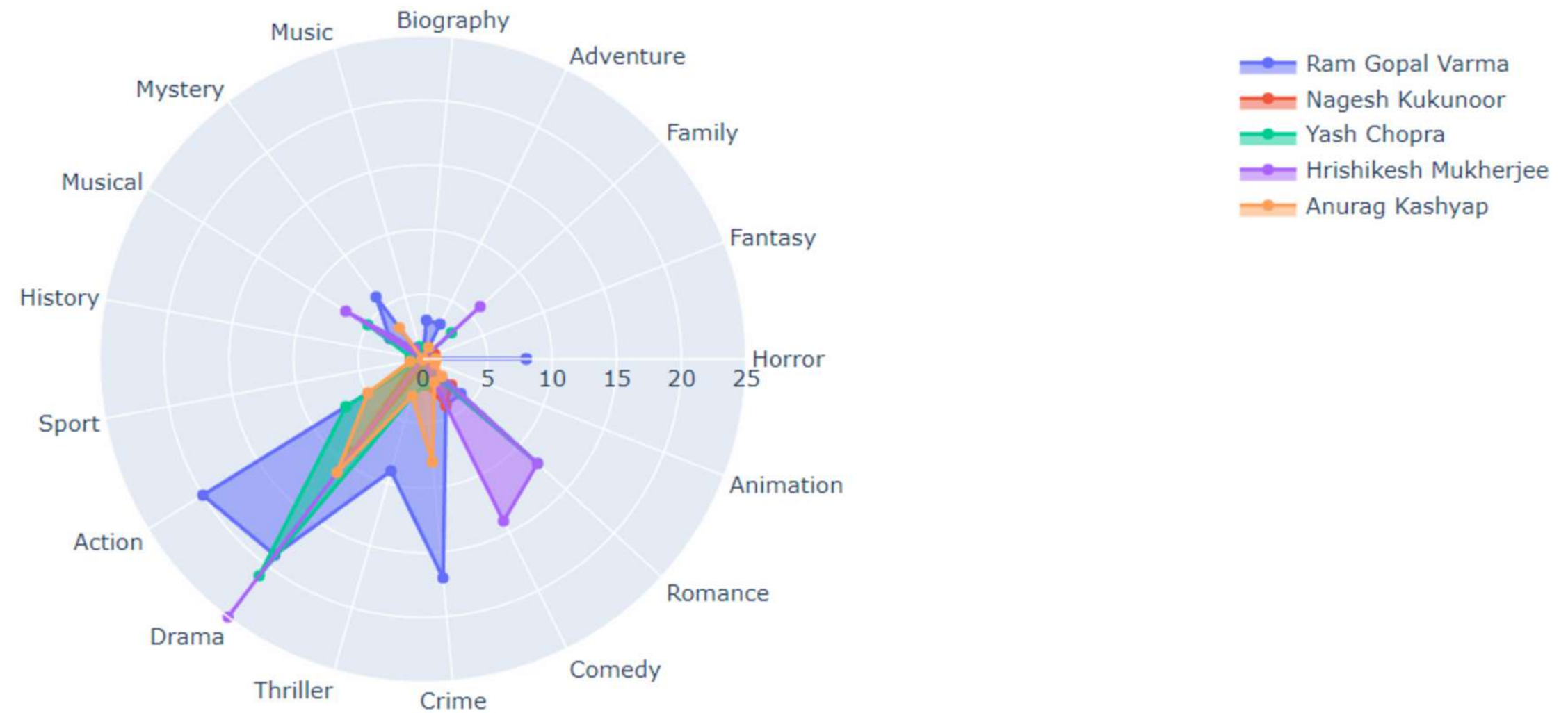
    fig.add_trace(go.Scatterpolar(
        r=values,
        theta=categories + [categories[0]], # เพื่อปิดกราฟ
        fill='toself',
        name=director
    ))

# ตั้งค่า layout
fig.update_layout(
    polar=dict(
        radialaxis=dict(visible=True, range=[0, max(max(director_genre_count[director].values()) for director in selected_directors)])
    ),
    title="Comparison of Movie Genres for Selected Directors",
    showlegend=True
)

# แสดงกราฟ
fig.show()
```

# Radar Chart

Comparison of Movie Genres for Selected Directors







**THANK  
YOU**