Virtual Internship Program

Author :- Prateek Kumar Singh

Beginer Level Tasks

Task-3 Music Recommendation

Music recommender systems can suggest songs to users based on their listening patterns.

Datasetlink :- https://www.kaggle.com/c/kkbox-music-recommendation-challenge/data

1. Importing The Libraries

pip install recommender-system

In [15]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
```

2. Load Music Data

In [38]:
```
train = pd.read_csv('train.csv')
songs = pd.read_csv('songs.csv')
members = pd.read_csv('members.csv')
```

In [39]:
```
df_train.head()
```

Out[39]:

	msno	song_id	source_system_tab	source_screen_name	source_type	target
0	FGtllVqz18RPiwJj/edr2gV78zirAiY/9SmYvia+kCg=	BBzumQNXUHKdEBOB7mAJuzok+IJA1c2Ryg/yzTF6tik=	explore	Explore	online-playlist	1
1	Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=	bhp/MpSNoqoxOIB+/l8WPqu6jldth4DIpCm3ayXnJqM=	my library	Local playlist more	local-playlist	1
2	Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=	JNWfrrC7zNN7BdMpsISKa4Mw+xVJYNnxXh3/Epw7QgY=	my library	Local playlist more	local-playlist	1
3	Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=	2A87tzfnJTSWqD7gIZHisolhe4DMdzkbd6LzO1KHjNs=	my library	Local playlist more	local-playlist	1
4	FGtllVqz18RPiwJj/edr2gV78zirAiY/9SmYvia+kCg=	3qm6XTZ6MOCU11x8FIVbAGH5l5uMkT3/ZalWG1oo2Gc=	explore	Explore	online-playlist	1

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

Out[40]:

	song_id	song_length	genre_ids	artist_name	composer	lyricist	language
0	CXoTN1eb7AI+DntdU1vbcwGRV4SCIDxZu+YD8JP8r4E=	247640	465	張信哲 (Jeff Chang)	董貞	何啟弘	3.0
1	o0kFgae9QtnYgRkVPqLJwa05zIhRlUjfF7O1tDw0ZDU=	197328	444	BLACKPINK	TEDDY| FUTURE BOUNCE| Bekuh BOOM	TEDDY	31.0
2	DwVvVurfpuz+XPuFvucclVQEyPqcpUkHR0ne1RQzPs0=	231781	465	SUPER JUNIOR	NaN	NaN	31.0
3	dKMBWoZyScdxSkihKG+Vf47nc18N9q4m58+b4e7dSSE=	273554	465	S.H.E	湯小康	徐世珍	3.0
4	W3bqWd3T+VeHFzHAUfARgW9AvVRaF4N5Yzm4Mr6Eo/o=	140329	726	貴族精選	Traditional	Traditional	52.0

In [41]:
```
df_songs_extra.head()
```

Out[41]:

	song_id	name	isrc
0	LP7pLJoJFBvyuUwvu+oLzjT+bI+UeBPURCecJsX1jjs=	我們	TWUM71200043
1	ClazTFnk6r0Bnuie44bocdNMM3rdlrq0bCGAsGUWcHE=	Let Me Love You	QMZSY1600015
2	u2ja/bZE3zhCGxvbbOB3zOoUjx27u40cf5g09UXMoKQ=	原諒我	TWA530887303
3	92Fqsy0+p6+RHe2EoLKjHahORHR1Kq1TBJoClW9v+Ts=	Classic	USSM11301446
4	0QFmz/+rJy1Q56C1DuYqT9hKKqi5TUqx0sN0IwvoHrw=	愛投羅網	TWA471306001

In [42]:
```
df_test.head()
```

Out[42]:

	id	msno	song_id	source_system_tab	source_screen_name	source_type
0	0	V8ruy7SGk7tDm3zA51DPpn6qutt+vmKMBKa21dp54uM=	WmHKgKMlp1lQMecNdNvDMkvIycZYHnFwDT72I5sIssc=	my library	Local playlist more	local-library
1	1	V8ruy7SGk7tDm3zA51DPpn6qutt+vmKMBKa21dp54uM=	y/rsZ9DC7FwK5F2PK2D5mj+aOBUJAjuu3dZ14NgE0vM=	my library	Local playlist more	local-library
2	2	/uQAlrAkaczV+nWCd2sPF2ekvXPRipV7q0l+gbLuxjw=	8eZLFOdGVdXBSqoAv5nsLigeH2BvKXzTQYtUM53I0k4=	discover	NaN	song-based-playlist
3	3	1a6oo/iXKatxQx4eS9zTVD+KlSVaAFbTIqVvwLC1Y0k=	ztCf8thYsS4YN3GcIL/bvoxLm/T5mYBVKOO4C9NiVfQ=	radio	Radio	radio
4	4	1a6oo/iXKatxQx4eS9zTVD+KlSVaAFbTIqVvwLC1Y0k=	MKVMpslKcQhMaFEgcEQhEfi5+RZhMYlU3eRDpySrH8Y=	radio	Radio	radio

3. Create New data

In [43]:
```
res = df_train.merge(df_songs[['song_id','song_length','genre_ids','artist_name','language']], on=['song_id'], how='left')
res.head()
```

Out[43]:

	msno	song_id	source_system_tab	source_screen_name	source_type	target	song_length	genre_ids	artist_
0	FGtllVqz18RPiwJj/edr2gV78zirAiY/9SmYvia+kCg=	BBzumQNXUHKdEBOB7mAJuzok+IJA1c2Ryg/yzTF6tik=	explore	Explore	online-playlist	1	NaN	NaN	
1	Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=	bhp/MpSNoqoxOIB+/l8WPqu6jldth4DIpCm3ayXnJqM=	my library	Local playlist more	local-playlist	1	NaN	NaN	
2	Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=	JNWfrrC7zNN7BdMpsISKa4Mw+xVJYNnxXh3/Epw7QgY=	my library	Local playlist more	local-playlist	1	225396.0	1259	
3	Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=	2A87tzfnJTSWqD7gIZHisolhe4DMdzkbd6LzO1KHjNs=	my library	Local playlist more	local-playlist	1	NaN	NaN	
4	FGtllVqz18RPiwJj/edr2gV78zirAiY/9SmYvia+kCg=	3qm6XTZ6MOCU11x8FIVbAGH5l5uMkT3/ZalWG1oo2Gc=	explore	Explore	online-playlist	1	187802.0	1011	Brett

In [44]:
```
train = res.merge(df_songs_extra,on=['song_id'],how = 'left')
train.head()
```

Out[44]:

	msno	song_id	source_system_tab	source_screen_name	source_type	target	song_length	genre_ids	artist_
0	FGtllVqz18RPiwJj/edr2gV78zirAiY/9SmYvia+kCg=	BBzumQNXUHKdEBOB7mAJuzok+IJA1c2Ryg/yzTF6tik=	explore	Explore	online-playlist	1	NaN	NaN	
1	Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=	bhp/MpSNoqoxOIB+/l8WPqu6jldth4DIpCm3ayXnJqM=	my library	Local playlist more	local-playlist	1	NaN	NaN	
2	Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=	JNWfrrC7zNN7BdMpsISKa4Mw+xVJYNnxXh3/Epw7QgY=	my library	Local playlist more	local-playlist	1	225396.0	1259	
3	Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=	2A87tzfnJTSWqD7gIZHisolhe4DMdzkbd6LzO1KHjNs=	my library	Local playlist more	local-playlist	1	NaN	NaN	
4	FGtllVqz18RPiwJj/edr2gV78zirAiY/9SmYvia+kCg=	3qm6XTZ6MOCU11x8FIVbAGH5l5uMkT3/ZalWG1oo2Gc=	explore	Explore	online-playlist	1	187802.0	1011	Brett

In [45]:
```
song_id = train.loc[:,["name","target"]]
song1 = song_id.groupby(["name"],as_index=False).count().rename(columns = {"target":"listen_count"})
song1.head()
```

Out[45]:

	name	listen_count
0	24karats –type S-(與放浪兄弟EXILE、DOBELMAN合作超跳嘻哈單曲)	1
1	A-CHA	2
2	Draw Me Close To You	1
3	I Can Only Imagine	1
4	I Concentrate On You - STACEY KENT and JIM TO...	11

In [46]:
```
dataset=train.merge(song1,on=['name'],how= 'left')
```

In [47]:
```
df=pd.DataFrame(dataset)
```

In [48]:
```
df.drop(columns=['source_system_tab','source_screen_name','source_type','target','isrc'],axis=1,inplace=True)
df=df.rename(columns={'msno':'user_id'})
```

4. Loading New data

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

Out[49]:

	user_id	song_id	song_length	genre_ids	artist_name	language	name	listen_count
0	FGtllVqz18RPiwJj/edr2gV78zirAiY/9SmYvia+kCg=	BBzumQNXUHKdEBOB7mAJuzok+IJA1c2Ryg/yzTF6tik=	NaN	NaN	NaN	NaN	Good Grief	51.0
1	Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=	bhp/MpSNoqoxOIB+/l8WPqu6jldth4DIpCm3ayXnJqM=	NaN	NaN	NaN	NaN	Lords of Cardboard	1.0
2	Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=	JNWfrrC7zNN7BdMpsISKa4Mw+xVJYNnxXh3/Epw7QgY=	225396.0	1259	Nas	52.0	Hip Hop Is Dead(Album Version (Edited))	1.0
3	Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=	2A87tzfnJTSWqD7gIZHisolhe4DMdzkbd6LzO1KHjNs=	NaN	NaN	NaN	NaN	Disco Africa	1.0
4	FGtllVqz18RPiwJj/edr2gV78zirAiY/9SmYvia+kCg=	3qm6XTZ6MOCU11x8FIVbAGH5l5uMkT3/ZalWG1oo2Gc=	187802.0	1011	Brett Young	52.0	Sleep Without You	56.0

5. Preprocessing of data

In [50]:
```
df.shape
```

Out[50]:
```
(1048575, 8)
```

In [51]:
```
#checking null values
df.isnull().sum()
```

Out[51]:
```
user_id              0
song_id              0
song_length     178938
genre_ids       192122
artist_name     178938
language        178945
name                93
listen_count        93
dtype: int64
```

In [52]:
```
#fill the null values
df['song_length'].fillna('0',inplace=True)
df['genre_ids'].fillna('0',inplace=True)
df['artist_name'].fillna('none',inplace=True)
df['language'].fillna('0',inplace=True)
df['name'].fillna('none',inplace=True)
df['listen_count'].fillna('0',inplace=True)
```

In [53]:
```
#Recheck null values
df.isnull().sum()
```

Out[53]:
```
user_id         0
song_id         0
song_length     0
genre_ids       0
artist_name     0
language        0
name            0
listen_count    0
dtype: int64
```

In [32]:
```
print("Total no of songs:",len(df))
```

```
Total no of songs: 1048575
```

6. Subset of The Dataset

In [54]:
```
df = df.head(10000)
df['song'] = df['name'].map(str) + " - " + df['artist_name']
```

7. Most Popular Songs in The Dataset

In [55]:
```
song_gr = df.groupby(['song']).agg({'listen_count': 'count'}).reset_index()
grouped_sum = song_gr['listen_count'].sum()
song_gr['percentage']  = song_gr['listen_count'].div(grouped_sum)*100
song_gr.sort_values(['listen_count', 'song'], ascending = [0,1])
```

Out[55]:

	song	listen_count	percentage
3190	告白氣球 - none	62	0.62
3655	帥到分手 - 周湯豪 (NICKTHEREAL)	54	0.54
5229	謝謝妳愛我 (Thanks For Your Love) - 謝和弦 (R-chord)	53	0.53
427	Closer - The Chainsmokers	38	0.38
2884	你，好不好？ (How Have You Been?) - none	38	0.38
...	...	...	...
5617	착한여자(GOOD TO YOU) - 2NE1	1	0.01
5619	촌스럽게 왜 이래 - K.Will	1	0.01
5620	칼로리 송 (통통한거라 믿고 싶겠지) Calorie Song - none	1	0.01
5621	헤어질걸 알기에(i know| we will breake up) - none	1	0.01
5622	Ｌｉａｒ - ONE OK ROCK	1	0.01

5623 rows × 3 columns

8. Unique Users in The Dataset

In [56]:
```
users = df['user_id'].unique()
print("The no. of unique users:", len(users))
```

```
The no. of unique users: 1622
```

8. Number of Unique Songs in The Dataset

In [57]:
```
songs = df['song'].unique()
len(songs)
```

Out[57]:
```
5623
```

8. Create a Song Recommender

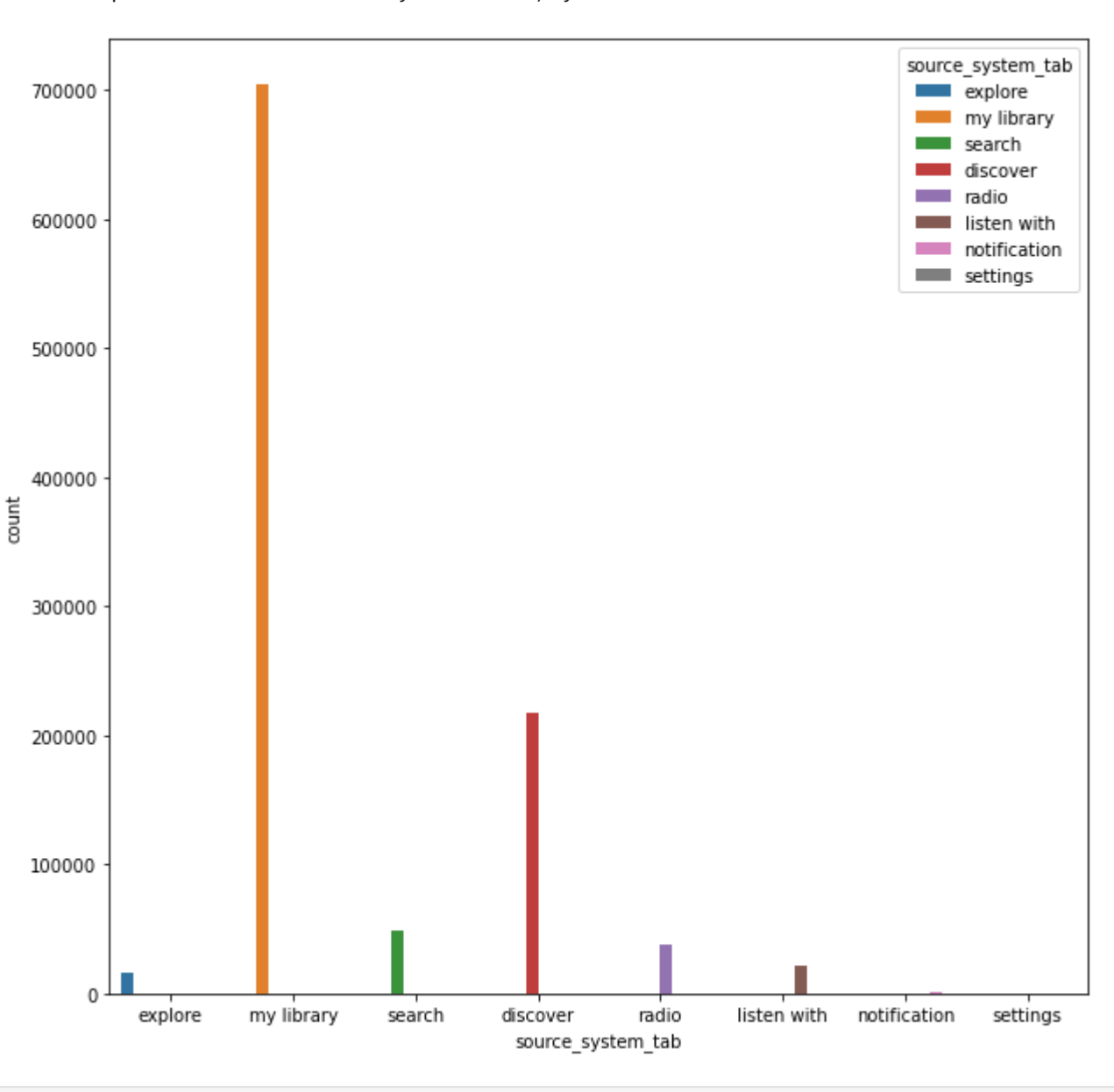
In [58]:
```
train_data, test_data = train_test_split(df, test_size = 0.20, random_state=0)
print(train.head(5))
```

```
                                           msno  \
0  FGtllVqz18RPiwJj/edr2gV78zirAiY/9SmYvia+kCg=
1  Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=
2  Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=
3  Xumu+NIjS6QYVxDS4/t3SawvJ7viT9hPKXmf0RtLNx8=
4  FGtllVqz18RPiwJj/edr2gV78zirAiY/9SmYvia+kCg=

                                        song_id source_system_tab  \
0  BBzumQNXUHKdEBOB7mAJuzok+IJA1c2Ryg/yzTF6tik=           explore
1  bhp/MpSNoqoxOIB+/l8WPqu6jldth4DIpCm3ayXnJqM=        my library
2  JNWfrrC7zNN7BdMpsISKa4Mw+xVJYNnxXh3/Epw7QgY=        my library
3  2A87tzfnJTSWqD7gIZHisolhe4DMdzkbd6LzO1KHjNs=        my library
4  3qm6XTZ6MOCU11x8FIVbAGH5l5uMkT3/ZalWG1oo2Gc=           explore

    source_screen_name      source_type  target  song_length genre_ids  \
0              Explore  online-playlist       1          NaN       NaN
1  Local playlist more   local-playlist       1          NaN       NaN
2  Local playlist more   local-playlist       1     225396.0      1259
3  Local playlist more   local-playlist       1          NaN       NaN
4              Explore  online-playlist       1     187802.0      1011

  artist_name  language                                     name  \
0         NaN       NaN                               Good Grief
1         NaN       NaN                       Lords of Cardboard
2         Nas      52.0  Hip Hop Is Dead(Album Version (Edited))
3         NaN       NaN                             Disco Africa
4  Brett Young      52.0                        Sleep Without You

          isrc
0  GBUM71602854
1  US3C69910183
2  USUM70618761
3  GBUQH1000063
4  QM3E21606003
```

9. Data Visualization

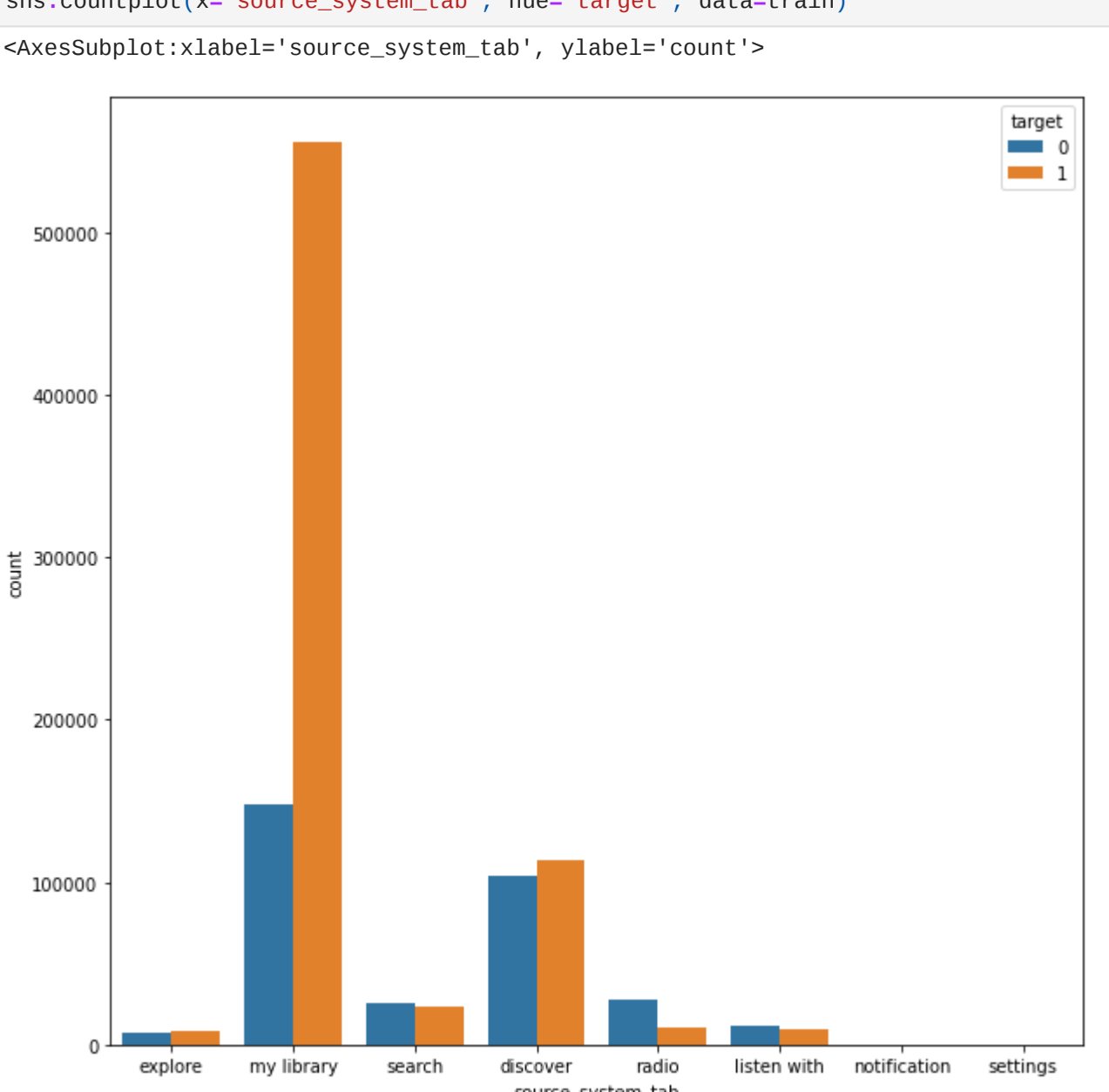
In [59]:
```
plt.figure(figsize=(10,10))
sns.countplot(x='source_system_tab', hue='source_system_tab', data=train)
```

Out[59]:
```
<AxesSubplot:xlabel='source_system_tab', ylabel='count'>
```



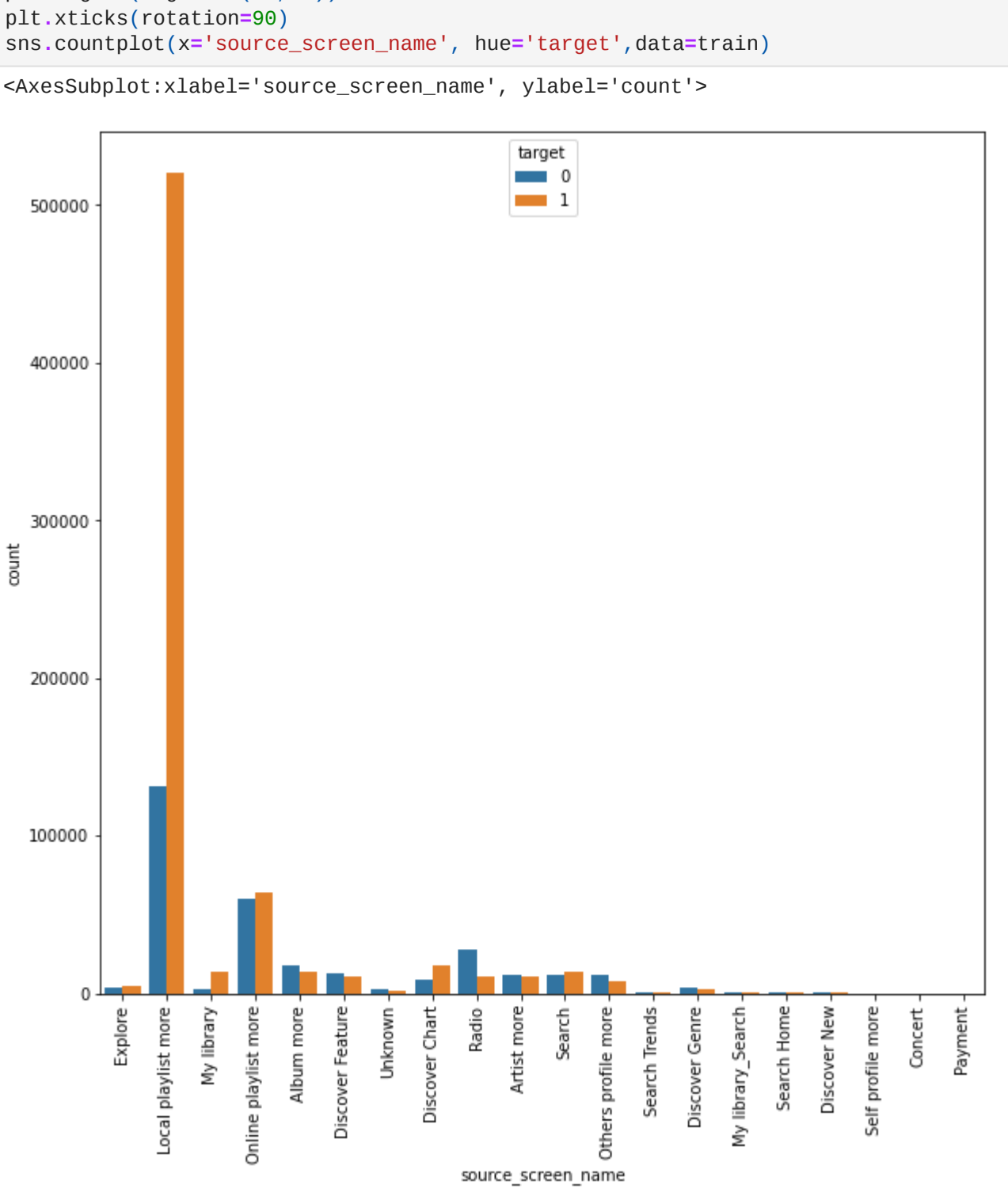
In [60]:
```
plt.figure(figsize=(10,10))
sns.countplot(x='source_system_tab', hue='target', data=train)
```

Out[60]:
```
<AxesSubplot:xlabel='source_system_tab', ylabel='count'>
```



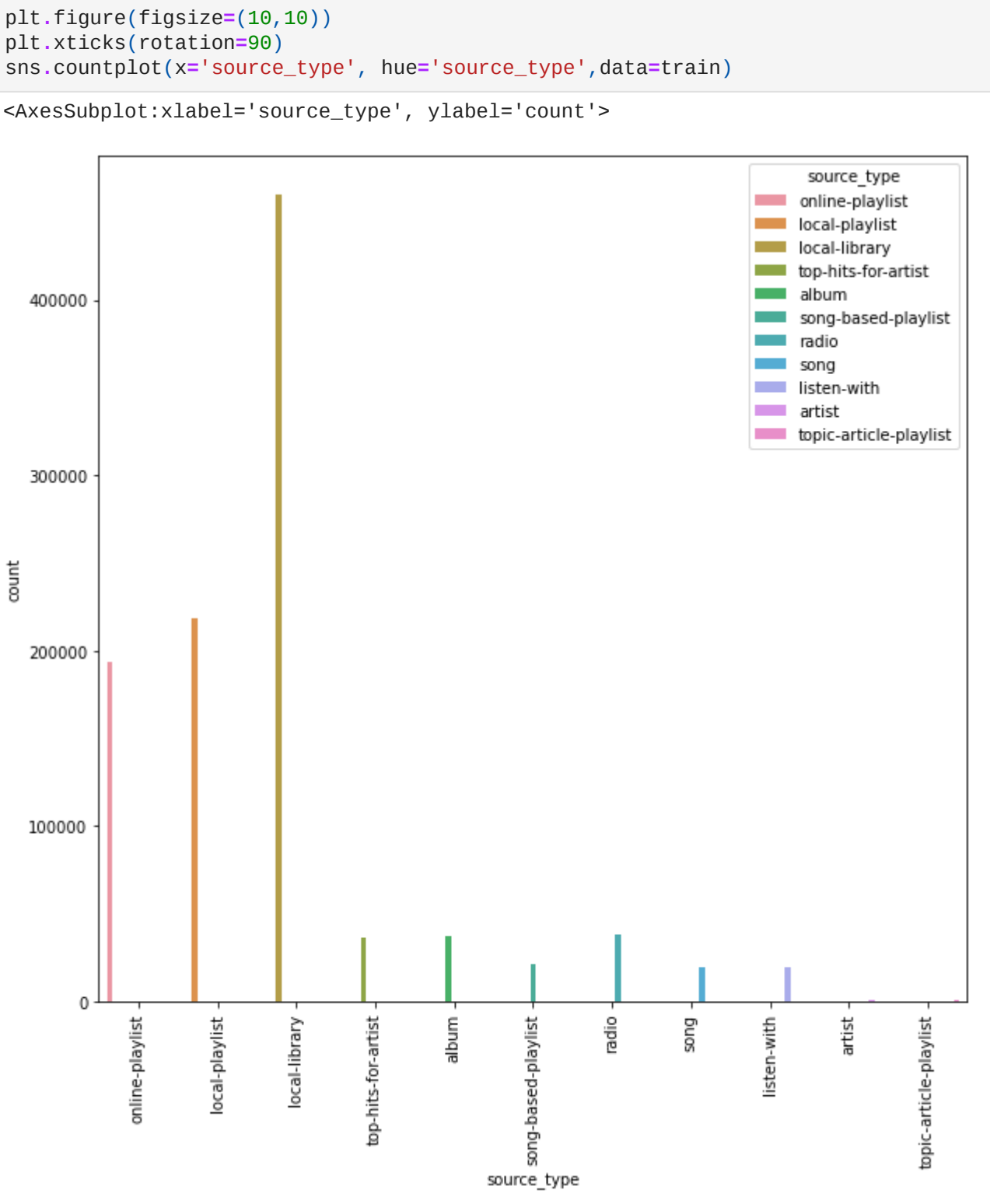
In [61]:
```
plt.figure(figsize=(10,10))
plt.xticks(rotation=90)
sns.countplot(x='source_screen_name', hue='target',data=train)
```

Out[61]:
```
<AxesSubplot:xlabel='source_screen_name', ylabel='count'>
```



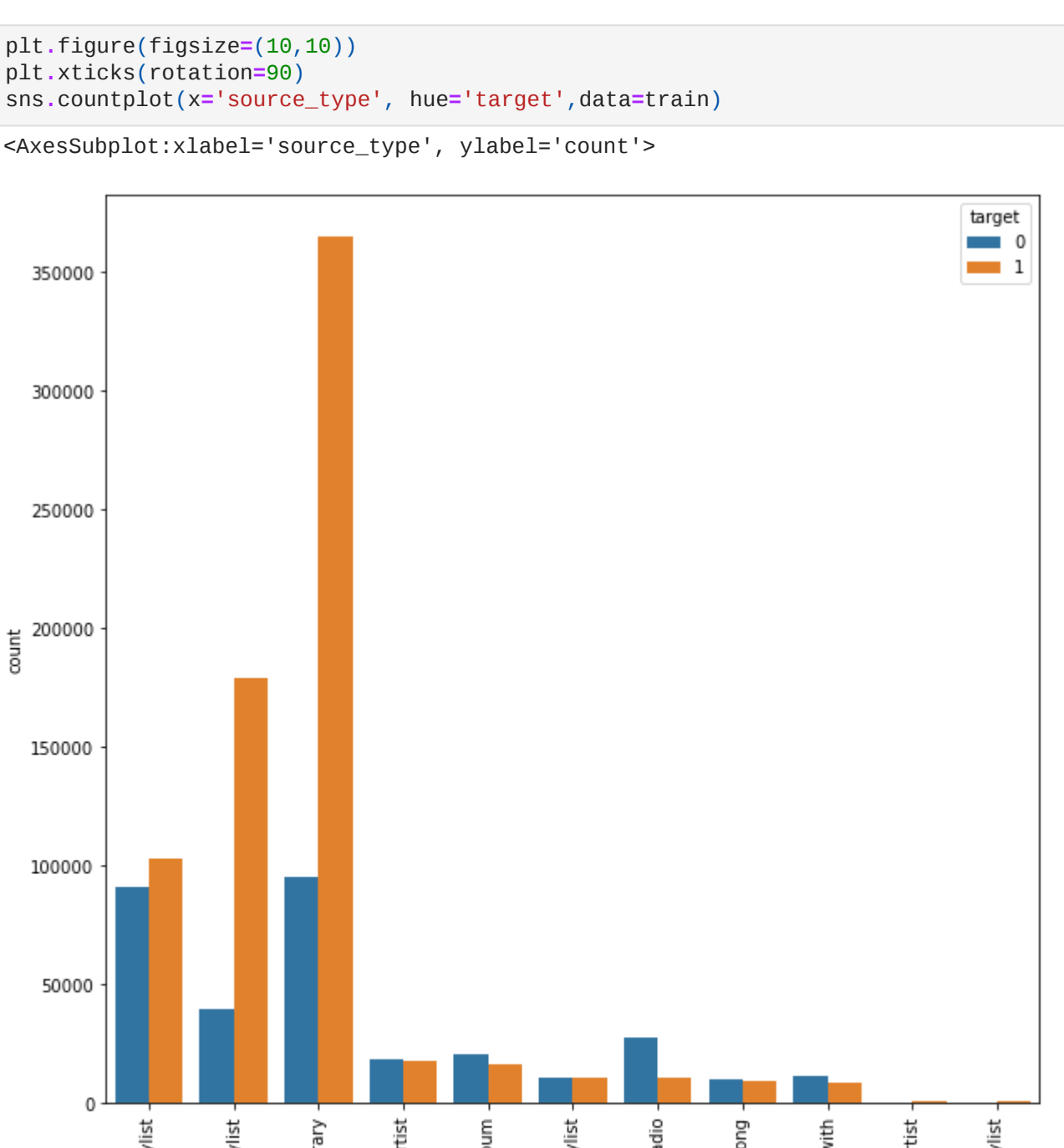
In [62]:
```
plt.figure(figsize=(10,10))
plt.xticks(rotation=90)
sns.countplot(x='source_type', hue='source_type',data=train)
```

Out[62]:
```
<AxesSubplot:xlabel='source_type', ylabel='count'>
```



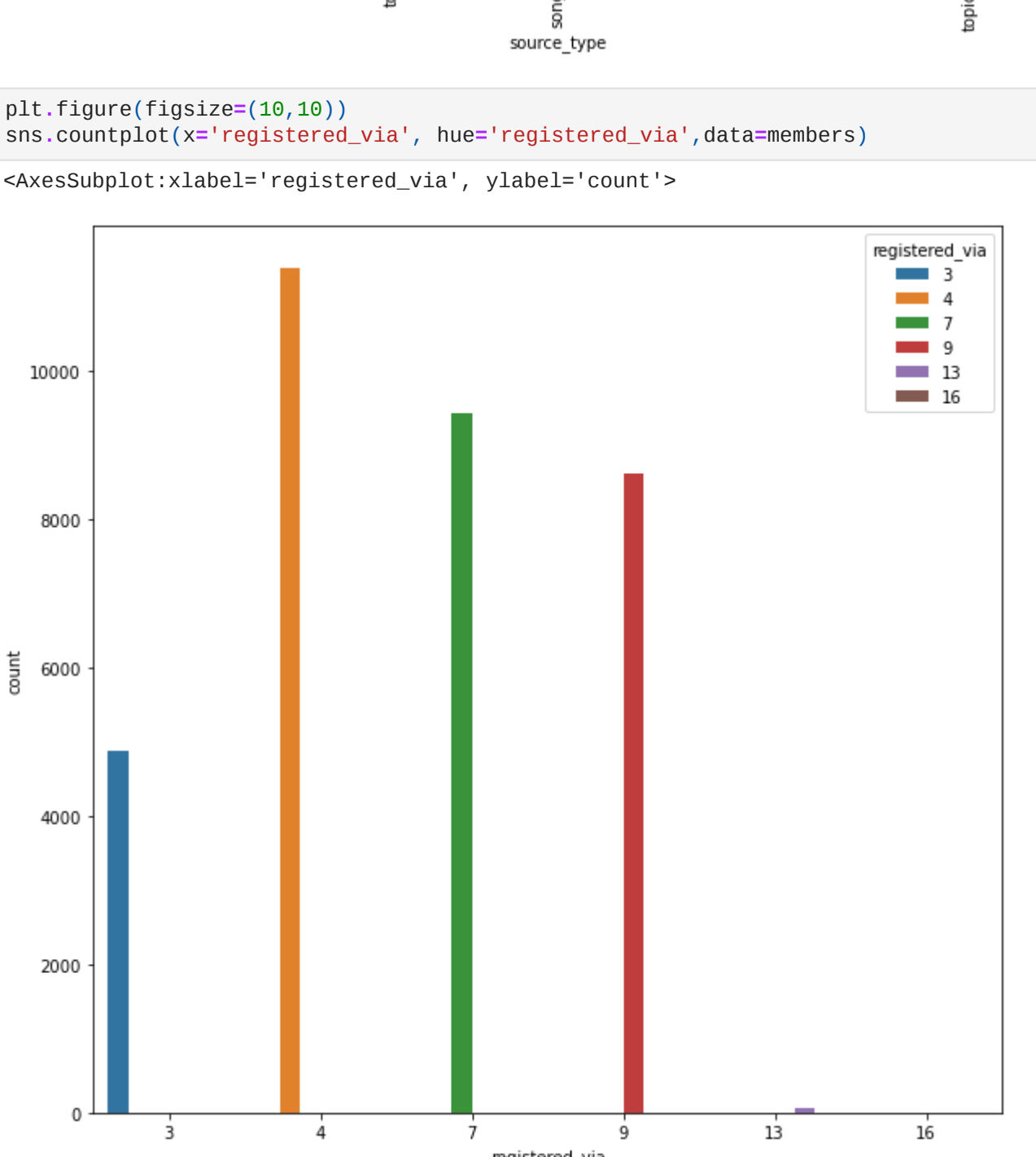
In [63]:
```
plt.figure(figsize=(10,10))
plt.xticks(rotation=90)
sns.countplot(x='source_type', hue='target',data=train)
```

Out[63]:
```
<AxesSubplot:xlabel='source_type', ylabel='count'>
```



In [65]:
```
plt.figure(figsize=(10,10))
sns.countplot(x='registered_via', hue='registered_via',data=members)
```

Out[65]:
```
<AxesSubplot:xlabel='registered_via', ylabel='count'>
```



10. data Cleaning

In [66]:
```
ntr = 7000
nts = 3000
names=['msno','song_id','source_system_tab','source_screen_name','source_type','target']
test1 = pd.read_csv('train.csv',names=names,skiprows=ntr,nrows=nts)
```

In [67]:
```
test = test1.drop(['target'],axis=1)
ytr = np.array(test1['target'])
```

In [68]:
```
test_name = ['id','msno','song_id','source_system_tab','source_screen_name','source_type']
test['id']=np.arange(nts)
test = test[test_name]
```

In [71]:
```
members['registration_year'] = members['registration_init_time'].apply(lambda x: int(str(x)[0:4]))
members['registration_month'] = members['registration_init_time'].apply(lambda x: int(str(x)[4:6]))
members['registration_date'] = members['registration_init_time'].apply(lambda x: int(str(x)[6:8]))
```

In [72]:
```
members['expiration_year'] = members['expiration_date'].apply(lambda x: int(str(x)[0:4]))
members['expiration_month'] = members['expiration_date'].apply(lambda x: int(str(x)[4:6]))
members['expiration_date'] = members['expiration_date'].apply(lambda x: int(str(x)[6:8]))
members = members.drop(['registration_init_time'], axis=1)
```

In [73]:
```
members_cols = members.columns
train = train.merge(members[members_cols], on='msno', how='left')
test = test.merge(members[members_cols], on='msno', how='left')
```

In [74]:
```
train = train.fillna(-1)
test = test.fillna(-1)
```

In [75]:
```
import gc
del members, songs; gc.collect();
import warnings
warnings.filterwarnings('ignore')
```

In [76]:
```
cols = list(train.columns)
cols.remove('target')
```

In [81]:
```
from tqdm import tqdm
from sklearn.preprocessing import LabelEncoder
```

In [83]:
```
unique_songs = range(max(train['song_id'].max(), test['song_id'].max()))
song_popularity = pd.DataFrame({'song_id': unique_songs, 'popularity':0})
train_sorted = train.sort_values('song_id')
train_sorted.reset_index(drop=True, inplace=True)
test_sorted = test.sort_values('song_id')
test_sorted.reset_index(drop=True, inplace=True)
```

Thank You