ESS Data - Immigration

The Key ESS Variables for this study

imsmetn: Allow many/few immigrants of same race/ethnic group as majority

imdfetn: Allow many/few immigrants of different race/ethnic group from majority

impcntr: Allow many/few immigrants from poorer countries outside Europe

imbgeco: Immigration bad or good for country's economy

imueclt: Country's cultural life undermined or enriched by immigrants

imwbcnt: Immigrants make country worse or better place to live

In [40]:
```
import os
os.chdir(r'C:\Users\deepa\OneDrive - University of Limerick\Sociology and Data Analyti

import pandas as pd
import matplotlib.pyplot as plt

data = pd.read_csv("ESS-subset.csv")

data = data[['name', 'essround', 'edition', 'proddate', 'idno', 'cntry', 'imsmetn', 'i
#data.to_excel("ESS-Immigration-data.xlsx")
data
```

```
C:\Users\deepa\AppData\Local\Temp\ipykernel_36704\1738252775.py:7: DtypeWarning: Colu
mns (506,507,508,509,510,515,516,517,518,519,537,538,539,540,547,548,549,550,551,552,
553,554,1990,2587,2588,2589,2590,2591,2592,2593,2594,2595,2596,2597,2598,2599,2600,26
01) have mixed types. Specify dtype option on import or set low_memory=False.
  data = pd.read_csv("ESS-subset.csv")
```

Out[40]:

	name	essround	edition	proddate	idno	cntry	imsmetn	imdfetn	impcntr	imbgeco	imu
0	ESS1e06_7	1	6.7	23.11.2023	101	IE	2	3	3	88	
1	ESS1e06_7	1	6.7	23.11.2023	102	IE	1	3	3	3	
2	ESS1e06_7	1	6.7	23.11.2023	104	IE	2	2	2	7	
3	ESS1e06_7	1	6.7	23.11.2023	105	IE	1	3	3	88	
4	ESS1e06_7	1	6.7	23.11.2023	106	IE	2	2	3	8	
...	...	...	...	...	...	...	...	...	...	...	
22228	ESS10e03_2	10	3.2	02.11.2023	92993	IE	3	3	3	5	
22229	ESS10e03_2	10	3.2	02.11.2023	93009	IE	3	4	4	1	
22230	ESS10e03_2	10	3.2	02.11.2023	93031	IE	2	3	2	7	
22231	ESS10e03_2	10	3.2	02.11.2023	93040	IE	1	1	1	10	
22232	ESS10e03_2	10	3.2	02.11.2023	93065	IE	1	1	1	8	

22233 rows × 12 columns

mapping the values based on the ESS Codebook

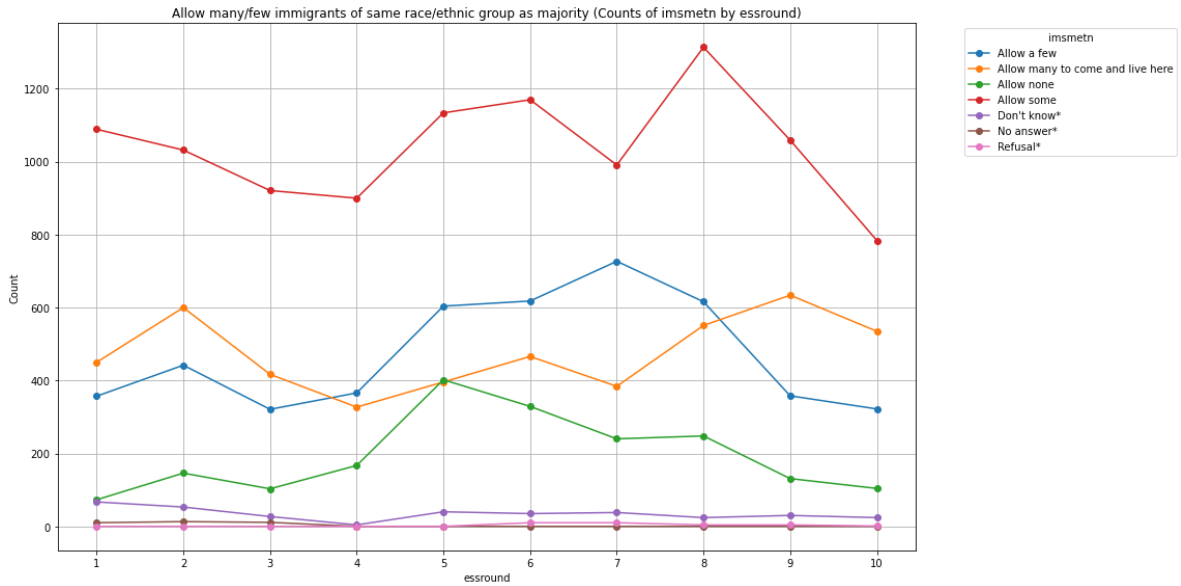
In [9]:
```
# Allow many/few immigrants of same race/ethnic group as majority
data['imsmetn'].replace({
    1: 'Allow many to come and live here',
    2: 'Allow some',
    3: 'Allow a few',
    4: 'Allow none',
    7: 'Refusal*',
    8: "Don't know*",
    9: 'No answer*'
}, inplace=True)

print("frequency table")
frequency_table = data['imsmetn'].value_counts()
print(frequency_table)
```

```
frequency table
Allow some                          10394
Allow many to come and live here     4760
Allow a few                          4731
Allow none                           1943
Don't know*                           342
No answer*                             34
Refusal*                               29
Name: imsmetn, dtype: int64
```

In [23]:
```
# Create the cross-tabulation
crosstab = pd.crosstab(data['essround'], data['imsmetn'])

# Plot the counts of 'imsmetn' categories for each 'essround'
fig, ax = plt.subplots(figsize=(16, 8))
crosstab.plot(kind='line', marker='o', ax=ax)
plt.title('Allow many/few immigrants of same race/ethnic group as majority (Counts of
plt.xlabel('essround')
plt.ylabel('Count')
plt.xticks(crosstab.index)
plt.legend(title='imsmetn', bbox_to_anchor=(1.05, 1), loc='upper left')
plt.grid(True)
plt.tight_layout()
plt.show()
```



In [37]:
```
crosstab_table = pd.crosstab(data['essround'], data['imsmetn'])
crosstab_table_df = pd.DataFrame(crosstab_table)
crosstab_table_df
```

Out[37]:

imsmetn	Allow a few	Allow many to come and live here	Allow none	Allow some	Don't know*	No answer*	Refusal*
essround							
1	357	450	73	1089	67	10	0
2	442	600	146	1032	53	13	0
3	321	417	103	921	27	11	0
4	366	327	167	900	4	0	0
5	604	396	402	1134	40	0	0
6	618	466	329	1170	35	0	10
7	727	384	240	991	38	0	10
8	616	551	248	1314	24	0	4
9	358	634	131	1059	30	0	4
10	322	535	104	784	24	0	1

In [31]:
```
# cross-tabulation
crosstab = pd.crosstab(data['essround'], data['imsmetn'])

# total count of respondents for each 'essround'
total_respondents = crosstab.sum(axis=1)

# proportions of 'imsmetn' categories for each 'essround'
proportions = crosstab.div(total_respondents, axis=0)
proportions
```

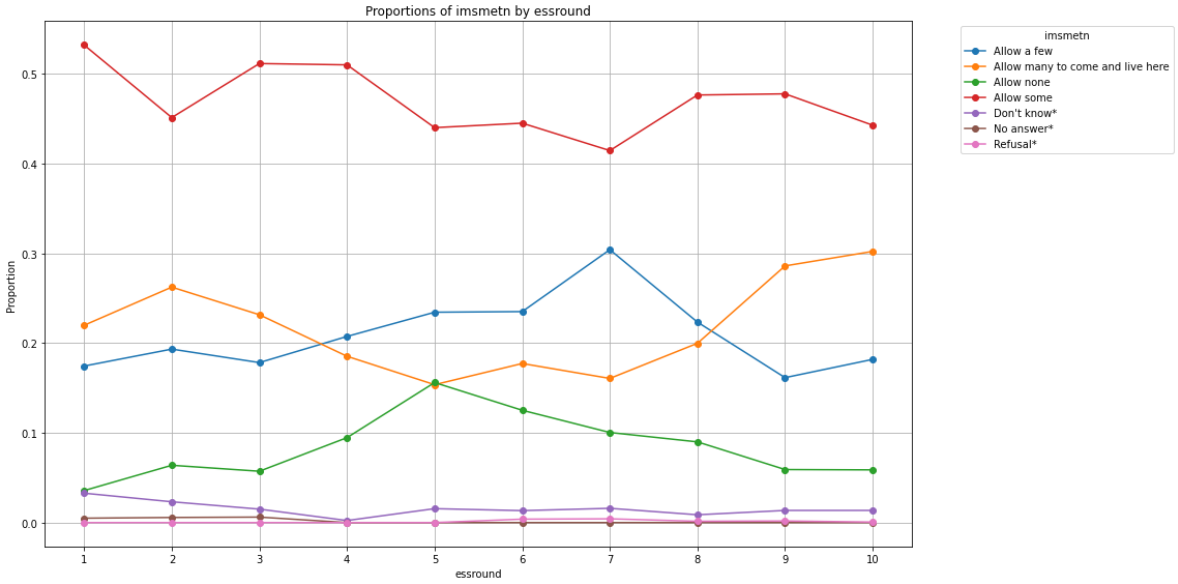
Out[31]:

imsmetn	Allow a few	Allow many to come and live here	Allow none	Allow some	Don't know*	No answer*	Refusal*
essround							
1	0.174487	0.219941	0.035679	0.532258	0.032747	0.004888	0.000000
2	0.193351	0.262467	0.063867	0.451444	0.023185	0.005687	0.000000
3	0.178333	0.231667	0.057222	0.511667	0.015000	0.006111	0.000000
4	0.207483	0.185374	0.094671	0.510204	0.002268	0.000000	0.000000
5	0.234472	0.153727	0.156056	0.440217	0.015528	0.000000	0.000000
6	0.235160	0.177321	0.125190	0.445205	0.013318	0.000000	0.003805
7	0.304184	0.160669	0.100418	0.414644	0.015900	0.000000	0.004184
8	0.223431	0.199855	0.089953	0.476605	0.008705	0.000000	0.001451
9	0.161552	0.286101	0.059116	0.477888	0.013538	0.000000	0.001805
10	0.181921	0.302260	0.058757	0.442938	0.013559	0.000000	0.000565

In [32]:
```
# 'imsmetn' categories
fig, ax = plt.subplots(figsize=(16, 8))

for category in proportions.columns:
    proportions[category].plot(ax=ax, marker='o', label=category)

ax.set_title('Proportions of imsmetn by essround')
ax.set_xlabel('essround')
ax.set_ylabel('Proportion')
plt.xticks(proportions.index)
plt.legend(title='imsmetn', bbox_to_anchor=(1.05, 1), loc='upper left')
plt.grid(True)
plt.tight_layout()
plt.show()
```



In [19]:
```
# Allow many/few immigrants of different race/ethnic group from majority
data['imdfetn'].replace({
    1: 'Allow many to come and live here',
    2: 'Allow some',
    3: 'Allow a few',
    4: 'Allow none',
    7: 'Refusal*',
    8: "Don't know*",
    9: 'No answer*'
}, inplace=True)

frequency_table = data['imdfetn'].value_counts()
print(frequency_table)
```

```
Allow some                          9765
Allow a few                         6057
Allow many to come and live here    3332
Allow none                          2619
Don't know*                          389
No answer*                            42
Refusal*                              29
Name: imdfetn, dtype: int64
```

In [20]:
```
# Allow many/few immigrants from poorer countries outside Europe
data['impcntr'].replace({
    1: 'Allow many to come and live here',
    2: 'Allow some',
    3: 'Allow a few',
    4: 'Allow none',
    7: 'Refusal*',
    8: "Don't know*",
    9: 'No answer*'
}, inplace=True)

frequency_table = data['impcntr'].value_counts()
print(frequency_table)
```

```
Allow some                          9405
Allow a few                         6284
Allow many to come and live here    3097
Allow none                          2969
Don't know*                          357
Refusal*                              86
No answer*                            35
Name: impcntr, dtype: int64
```

In [21]:
```
# Immigration bad or good for country's economy
data['imbgeco'].replace({
    0: 'Bad for the economy',
    1: '1',
    2: '2',
    3: '3',
    4: '4',
    5: '5',
    6: '6',
    7: '7',
    8: '8',
    9: '9',
    10: 'Good for the economy',
    77: 'Refusal*',
    88: "Don't know*",
    99: 'No answer*'
}, inplace=True)

frequency_table = data['imbgeco'].value_counts()
print(frequency_table)
```

```
5                     4218
7                     3145
8                     2851
6                     2426
3                     1791
4                     1775
2                     1449
Good for the economy  1123
Bad for the economy   1018
9                     1001
1                      910
Don't know*            490
No answer*              19
Refusal*                17
Name: imbgeco, dtype: int64
```

In [22]:
```
# Country's cultural life undermined or enriched by immigrants
data['imueclt'].replace({
    0: 'Cultural life undermined',
    1: '1',
    2: '2',
    3: '3',
    4: '4',
    5: '5',
    6: '6',
    7: '7',
    8: '8',
    9: '9',
    10: 'Cultural life enriched',
    77: 'Refusal*',
    88: "Don't know*",
    99: 'No answer*'
})

frequency_table = data['imueclt'].value_counts()
print(frequency_table)
```

```
5     4023
7     3448
8     3222
6     2342
4     1742
3     1591
10    1407
9     1280
2     1188
0      712
88     652
1      591
99      23
77      12
Name: imueclt, dtype: int64
```

In [23]:
```
# Immigrants make country worse or better place to live
data['imwbcnt'].replace({
    0: 'Worse place to live',
    1: '1',
    2: '2',
    3: '3',
    4: '4',
    5: '5',
    6: '6',
    7: '7',
    8: '8',
    9: '9',
    10: 'Better place to live',
    77: 'Refusal*',
    88: "Don't know*",
    99: 'No answer*'
}, inplace=True)

frequency_table = data['imwbcnt'].value_counts()
print(frequency_table)
```

```
5                       4584
7                       3205
8                       3144
6                       2509
4                       1726
3                       1597
Better place to live    1262
2                       1183
9                       1135
Worse place to live      759
1                        609
Don't know*              488
No answer*                21
Refusal*                  11
Name: imwbcnt, dtype: int64
```

In [ ]:

In [ ]: