

```
In [1]: from __future__ import print_function
import time
import openapi_client
from openapi_client.rest import ApiException
from openapi_client.api.default_api import DefaultApi
from pprint import pprint
import requests
import plotly.express as px
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import seaborn as sns
```

```
In [2]: def get_data(url):

    category = str(url)
    response = requests.get(category)
    data = response.json()
    data_dict = data["items"]

    data = pd.DataFrame.from_records(data_dict)

    return data
```

```
In [3]: asylum_decisions = get_data('https://api.unhcr.org/population/v1/asylum-decision')
asylum_applications = get_data('https://api.unhcr.org/population/v1/asylum-applications')
demographics = get_data('https://api.unhcr.org/population/v1/demographics/?&yearFrom=2010')
population = get_data('https://api.unhcr.org/population/v1/population/?&yearFrom=2010')
```

```
In [4]: asylum_decisions = asylum_decisions.dropna()
asylum_decisions[['year', 'dec_recognized', 'dec_other', 'dec_rejected', 'dec_claims']]
asylum_decisions = asylum_decisions.drop(['dec_pc', 'coo_iso', 'coa', 'coa_iso'], axis=1)
asylum_decisions = asylum_decisions[asylum_decisions.coo != 'UKN']
asylum_decisions
```

```
Out[4]:
```

	year	coo_id	coo_name	coo	coa_id	coa_name	dec_recognized	dec_other	dec_rejec
0	2010	2	Afghanistan	AFG	11	Australia	11	0	
1	2010	3	Albania	ALB	11	Australia	0	0	
2	2010	8	Egypt	ARE	11	Australia	22	0	
3	2010	14	Bahrain	BAH	11	Australia	0	0	
4	2010	20	Bangladesh	BGD	11	Australia	10	0	
...	
65811	2023	91	Iran (Islamic Rep. of)	IRN	89	Indonesia	5	0	
65812	2023	177	Sudan	SUD	96	Jordan	0	0	
65813	2023	211	Yemen	YEM	185	Syrian Arab Rep.	5	0	

	year	coo_id	coo_name	coo	coa_id	coa_name	dec_recognized	dec_other	dec_rejec
65814	2023	147	Pakistan	PAK	188	Thailand	0	0	
65815	2023	174	Viet Nam	SRV	188	Thailand	0	0	

65208 rows × 11 columns

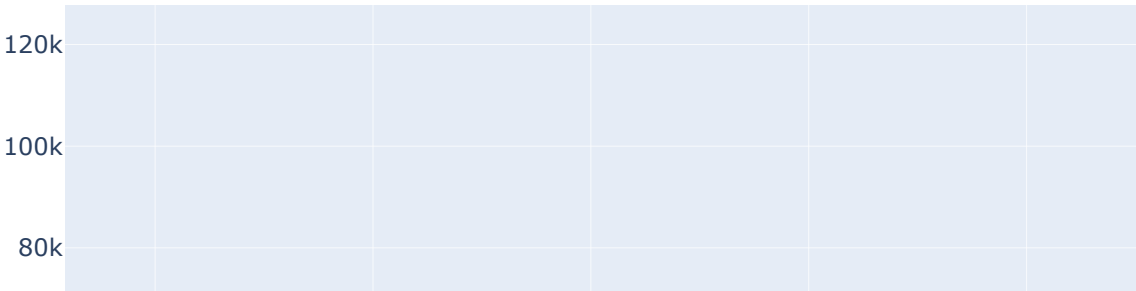
```
In [5]: asylum_applications['year'] = asylum_applications['year'].astype(int)
asylum_applications = asylum_applications.drop(['coo_iso', 'coa', 'coa_iso', 'pr
asylum_applications = asylum_applications[asylum_applications.coo != 'UKN']
asylum_applications = asylum_applications.dropna()
asylum_applications
```

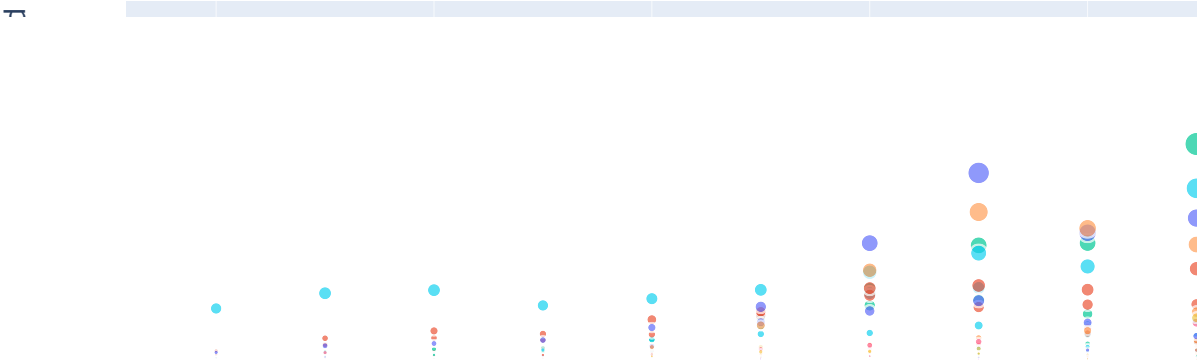
	year	coo_id	coo_name	coo	coa_id	coa_name	app_type	applied
0	2010	2	Afghanistan	AFG	11	Australia	A	14
1	2010	3	Albania	ALB	11	Australia	A	9
2	2010	8	Egypt	ARE	11	Australia	A	86
3	2010	14	Bahrain	BAH	11	Australia	A	5
4	2010	20	Bangladesh	BGD	11	Australia	A	42
...
72517	2023	172	Somalia	SOM	8	Egypt	R	23
72518	2023	179	South Sudan	SSD	8	Egypt	R	68
72519	2023	199	Uganda	UGA	8	Egypt	R	5
72520	2023	34	Central African Rep.	CAR	39	Cameroon	R	16
72521	2023	161	Rwanda	RWA	39	Cameroon	R	5

71871 rows × 8 columns

```
In [6]: usa_app = asylum_applications.loc[asylum_applications['coa_name'] == 'United Sta
px.scatter(data_frame=usa_app, x='year', y='applied', color='coo_name', hover_da
```

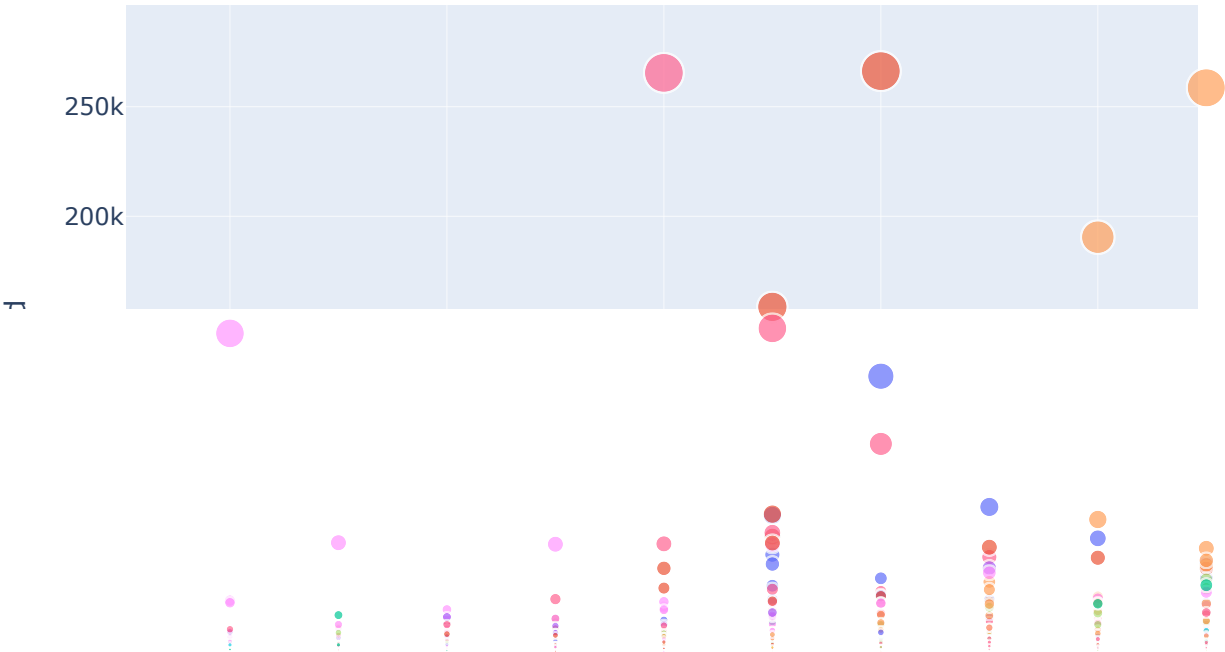
Asylum Applications to USA





```
In [23]: px.scatter(data_frame= asylum_applications, x='year', y='applied', color='country')
```

Asylum Applications



In [8]:

```
demographics[['year', 'f_0_4', 'f_5_11', 'f_12_17', 'f_18_59', 'f_60', 'f_other'],
demographics = demographics.drop(['coo', 'coo_iso', 'coa', 'coa_iso'], axis = 1)
demographics = demographics.dropna()
demographics
```

Out[8]:

	year	coo_id	coo_name	coa_id	coa_name	f_0_4	f_5_11	f_12_17	f_18_59	f_60	f
0	2010	2	Afghanistan	2	Afghanistan	70435	180578	103913	272816	14923	
1	2010	91	Iran (Islamic Rep. of)	2	Afghanistan	0	0	0	16	0	
2	2010	92	Iraq	2	Afghanistan	0	0	0	0	0	
3	2010	147	Pakistan	2	Afghanistan	0	0	0	7	0	
4	2010	8	Egypt	3	Albania	5	0	0	0	0	
...	
64455	2022	207	Venezuela (Bolivarian Republic of)	224	Aruba	0	0	0	0	0	
64456	2022	224	Aruba	224	Aruba	0	0	0	0	0	
64457	2022	207	Venezuela (Bolivarian Republic of)	254	Curacao	0	0	0	0	0	
64458	2022	254	Curacao	254	Curacao	0	0	0	0	0	
64459	2022	207	Venezuela (Bolivarian Republic of)	255	Sint Maarten (Dutch part)	0	0	0	0	0	

64460 rows x 20 columns



In [31]:

```
ax = demographics.groupby('year', as_index=False).sum()
ax
```

Out[31]:

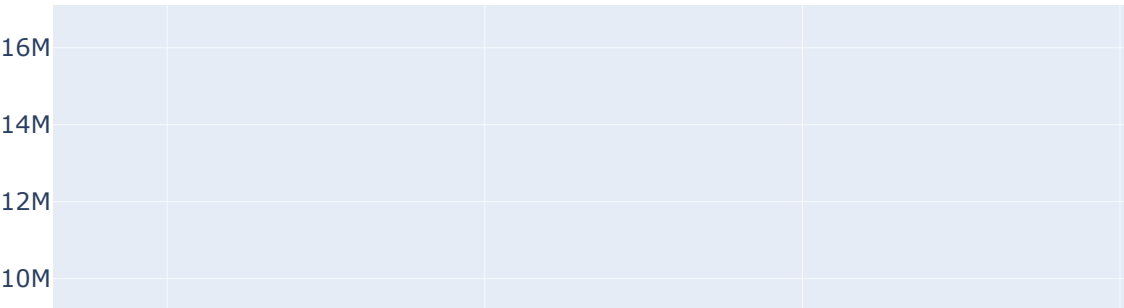
	year	coo_id	coa_id	f_0_4	f_5_11	f_12_17	f_18_59	f_60	f_other	f_tota
0	2010	435423	430467	825988	1233425	905704	3088782	335609	3353415	974292
1	2011	436098	427416	965743	1389627	1033612	3469224	371045	3706594	1093584
2	2012	454168	445254	875040	1454786	1062724	3017112	310150	3501253	1022106
3	2013	477788	464086	1205398	1632987	1058049	3321821	291143	4095684	1160508
4	2014	501199	483106	1250374	1805103	1184284	3671946	329846	6177087	1441864
5	2015	529098	498818	1535707	2268862	1472882	4620148	403974	1665330	1196690
6	2016	530226	499136	1785662	2586197	1621779	4946759	475506	5624292	1704019

	year	coo_id	coa_id	f_0_4	f_5_11	f_12_17	f_18_59	f_60	f_other	f_tota
7	2017	553337	510243	2073913	2968050	1859113	5510242	631625	7295969	2033891
8	2018	567870	531699	2609643	3292382	2139690	6778462	650736	8590885	2406179
9	2019	592301	551524	3232464	4176209	2748655	8436568	667947	9395730	2865757
10	2020	600819	557862	3156037	4002046	3035940	10897461	983130	7679638	2975425
11	2021	607965	567703	3218508	4504514	3388485	12637836	1297246	4954297	3000088
12	2022	653378	608391	3539943	5052005	3896957	16019860	2256670	11350271	4211570

In [32]:

```
px.scatter(ax, x = 'year', y=['f_0_4', 'f_5_11', 'f_12_17', 'f_18_59', 'f_60', 'm_0_4
```

Number of Applications by Year



In [9]:

```
population[['year', 'refugees', 'asylum_seekers', 'returned_refugees', 'idps', '
population = population.drop(['coo', 'coo_iso', 'coa', 'coa_iso', 'oip', 'hst'],
population = population.dropna()
population
```

Out[9]:

	year	coo_id	coo_name	coa_id	coa_name	refugees	asylum_seekers	returned_refuge
0	2010	2	Afghanistan	2	Afghanistan	0		0

	year	coo_id	coo_name	coa_id	coa_name	refugees	asylum_seekers	returned_refuge
1	2010	91	Iran (Islamic Rep. of)	2	Afghanistan	30	21	
2	2010	92	Iraq	2	Afghanistan	6	0	
3	2010	147	Pakistan	2	Afghanistan	6398	9	
4	2010	8	Egypt	3	Albania	5	0	
...	
70745	2023	207	Venezuela (Bolivarian Republic of)	254	Curacao	0	0	
70746	2023	254	Curacao	254	Curacao	0	0	
70747	2023	207	Venezuela (Bolivarian Republic of)	255	Sint Maarten (Dutch part)	0	131	
70748	2023	92	Iraq	262	Unknown	0	0	1
70749	2023	200	Ukraine	262	Unknown	0	0	170

70750 rows x 12 columns

In []:

In [10]:

```
asylum_grouped = asylum_decisions.groupby('coa_name').sum().reset_index()
pop_grouped = population.groupby('coa_name').sum().reset_index()

asylum_grouped = asylum_grouped.drop(['year', 'coo_id', 'coa_id'], axis = 1)
```

In [11]:

```
pop_grouped = pop_grouped.drop(['year', 'coo_id', 'coa_id'], axis = 1)
```

In [19]:

```
merged_df = pd.merge(asylum_grouped, pop_grouped, on='coa_name')
merged_df.corr()
```

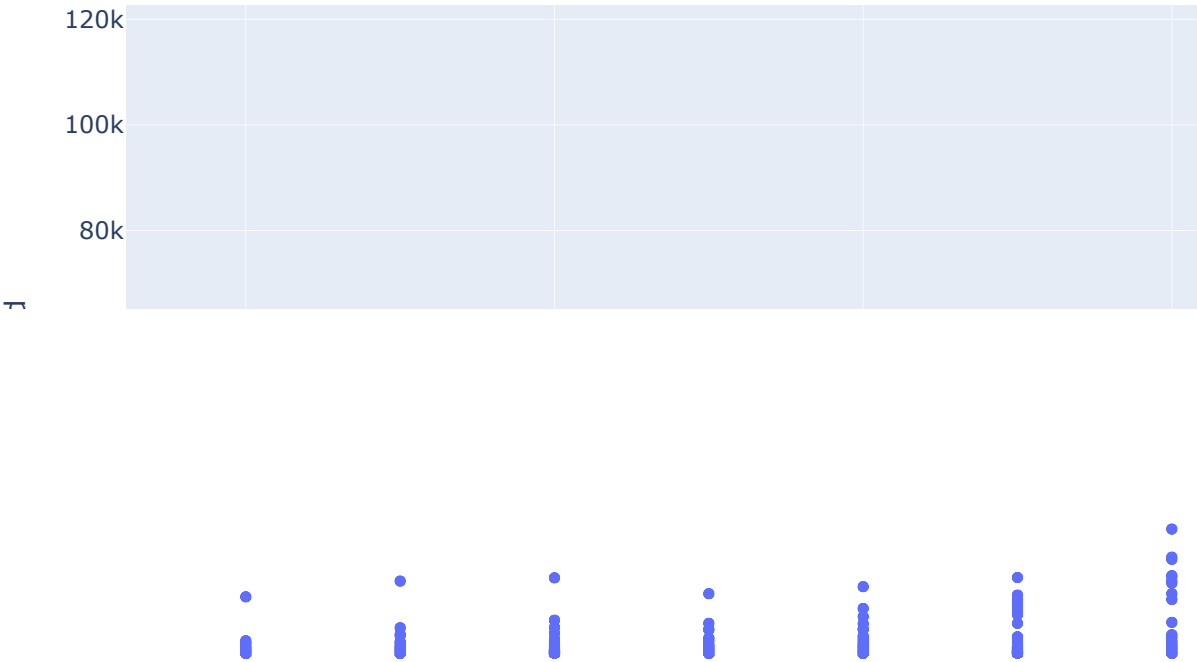
Out[19]:

	dec_recognized	dec_other	dec_rejected	dec_closed	dec_total	refugees	a:
dec_recognized	1.000000	0.699957	0.782173	0.870113	0.935127	0.407414	
dec_other	0.699957	1.000000	0.657895	0.626796	0.794029	0.217782	
dec_rejected	0.782173	0.657895	1.000000	0.702324	0.915963	0.228852	
dec_closed	0.870113	0.626796	0.702324	1.000000	0.898985	0.369765	
dec_total	0.935127	0.794029	0.915963	0.898985	1.000000	0.337073	
refugees	0.407414	0.217782	0.228852	0.369765	0.337073	1.000000	

	dec_recognized	dec_other	dec_rejected	dec_closed	dec_total	refugees	a:
asylum_seekers	0.628568	0.264677	0.543942	0.767214	0.645064	0.288089	
returned_refugees	0.189878	0.022466	0.032057	0.133828	0.101144	0.737635	
idps	-0.033496	-0.047448	-0.062319	-0.031046	-0.050854	0.092930	
returned_idps	-0.014911	-0.045612	-0.059511	-0.045679	-0.049529	0.230440	
stateless	-0.017439	0.033228	-0.030907	-0.026995	-0.019015	0.042172	
ooc	-0.008709	-0.033147	-0.034559	-0.024710	-0.029386	0.009638	

```
In [22]: fig = px.scatter(usa_app, x='year', y='applied', title='Number of Applications by Year')
fig.show()
```

Number of Applications by Year



```
In [14]: #usa_app['applied'] = ((usa_app['applied'] - usa_app['applied'].mean()) / usa_app['applied'].std()) * 100000
#usa_app
```

In [15]:

usa_app

Out[15]:

	year	coo_id	coo_name	coo	coa_id	coa_name	app_type	applied
4959	2017	3	Albania	ALB	202	United States of America	A	5
4960	2017	8	Egypt	ARE	202	United States of America	A	29
4961	2017	9	Argentina	ARG	202	United States of America	A	5
4962	2017	20	Bangladesh	BGD	202	United States of America	A	5
4963	2017	27	Brazil	BRA	202	United States of America	A	52
...
52935	2023	203	Uzbekistan	UZB	202	United States of America	N	1566
52936	2023	207	Venezuela (Bolivarian Republic of)	VEN	202	United States of America	N	61995
52937	2023	211	Yemen	YEM	202	United States of America	N	119
52938	2023	213	Zambia	ZAM	202	United States of America	N	56
52939	2023	214	Zimbabwe	ZIM	202	United States of America	N	315

4557 rows × 8 columns

In [16]:

```
x = usa_app.year
X = np.array([[1] * len(usa_app),
              x]).T
y = np.array(usa_app.applied)
X[0:4508,]
```

Out[16]:

```
array([[ 1, 2017],
       [ 1, 2017],
       [ 1, 2017],
       ...,
       [ 1, 2023],
       [ 1, 2023],
       [ 1, 2023]])
```

In [17]:

```
b = np.matmul(np.linalg.inv(np.matmul(X.T, X)), np.matmul(X.T, y))
np.set_printoptions(suppress=True) #this just prevents python from printing it c
b
```

Out[17]:

```
array([-191290.24548709,      95.18494156])
```


In [18]:

```
plot = sns.regplot(data= usa_app, x='year', y='applied', fit_reg=True, ci=None,  
                  line_kws={'label': "y={1:.2f}+{0:.2f}x".format(b[0],b[1])});  
plot.set_ylim(0,1800)  
plot.legend();  
sns.move_legend(plot, "upper right")
```

