

DETECTION OF AUTISTIC SPECTRUM DISORDER

Importing libraries

```
In [1]: import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
from plotly.subplots import make_subplots
import plotly.graph_objects as go
```

Loading Dataset

```
In [2]: ad = pd.read_csv("csv_result-Autism-Adolescent-Data.csv",na_values=['?'])
ch = pd.read_csv("csv_result-Autism-Child-Data.csv",na_values=['?'])
```

Understanding the data

Sample records

```
In [3]: ad.sample(5)
```

id			A1_Score	A2_Score	A3_Score	A4_Score	A5_Score	A6_Score	A7_Score	A8_Score	A9_Score	...	gender	ethn
23	24	1	0	1	1	1	1	1	1	1	1	...	m	Middle East
101	102	1	0	0	1	1	1	1	1	1	1	...	f	White European
57	58	1	1	1	1	1	1	1	1	1	0	...	m	White European
60	61	1	1	1	1	1	1	1	1	1	1	...	f	White European
46	47	0	1	1	1	1	1	1	1	1	1	...	m	Asian

5 rows x 22 columns

```
In [4]: ch.sample(5)
```

id			A1_Score	A2_Score	A3_Score	A4_Score	A5_Score	A6_Score	A7_Score	A8_Score	A9_Score	...	gender	ethn
60	61	1	0	1	0	1	0	0	0	0	0	...	m	Turkish
268	269	1	0	1	0	1	1	1	0	1	1	...	m	NaN
41	42	0	1	0	0	0	0	0	0	0	0	...	m	Middle East
264	265	1	0	1	1	1	1	1	0	1	1	...	m	White European
126	127	1	1	1	1	0	1	0	1	0	0	...	m	NaN

5 rows x 22 columns

Number of tuples and fields

```
In [5]: print("Adolescents :", ad.shape)
        print("Children    :", ch.shape)
```

```
Adolescents : (104, 22)
Children    : (292, 22)
```

Information of all fields

```
In [6]: ad.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 104 entries, 0 to 103
Data columns (total 22 columns):
id                104 non-null int64
A1_Score          104 non-null int64
A2_Score          104 non-null int64
A3_Score          104 non-null int64
A4_Score          104 non-null int64
A5_Score          104 non-null int64
A6_Score          104 non-null int64
A7_Score          104 non-null int64
A8_Score          104 non-null int64
A9_Score          104 non-null int64
A10_Score         104 non-null int64
age              104 non-null int64
gender            104 non-null object
ethnicity         98 non-null object
jundice           104 non-null object
austim            104 non-null object
contry_of_res     104 non-null object
used_app_before   104 non-null object
result           104 non-null int64
age_desc         104 non-null object
relation          98 non-null object
Class/ASD        104 non-null object
dtypes: int64(13), object(9)
memory usage: 18.0+ KB
```

In [7]: `ch.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 292 entries, 0 to 291
Data columns (total 22 columns):
id                292 non-null int64
A1_Score          292 non-null int64
A2_Score          292 non-null int64
A3_Score          292 non-null int64
A4_Score          292 non-null int64
A5_Score          292 non-null int64
A6_Score          292 non-null int64
A7_Score          292 non-null int64
A8_Score          292 non-null int64
A9_Score          292 non-null int64
A10_Score         292 non-null int64
age              288 non-null float64
gender            292 non-null object
ethnicity         249 non-null object
jundice           292 non-null object
austim            292 non-null object
contry_of_res     292 non-null object
used_app_before   292 non-null object
result           292 non-null int64
age_desc          292 non-null object
relation          249 non-null object
Class/ASD         292 non-null object
dtypes: float64(1), int64(12), object(9)
memory usage: 50.3+ KB
```

Data description

```
In [8]: ad.describe()
```

	id	A1_Score	A2_Score	A3_Score	A4_Score	A5_Score	A6_Score	A7_Score	A8_Score	A9_Score	A10_Sc
count	104.000000	104.000000	104.000000	104.000000	104.000000	104.000000	104.000000	104.000000	104.000000	104.000000	104.000000
mean	52.500000	0.730769	0.538462	0.653846	0.701923	0.750000	0.769231	0.519231	0.615385	0.788462	0.653846
std	30.166206	0.445708	0.500933	0.478047	0.459629	0.43511	0.423365	0.502050	0.488860	0.410377	0.478047
min	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	26.750000	0.000000	0.000000	0.000000	0.000000	0.750000	1.000000	0.000000	0.000000	1.000000	0.000000
50%	52.500000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
75%	78.250000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
max	104.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

```
In [9]: ch.describe()
```

	id	A1_Score	A2_Score	A3_Score	A4_Score	A5_Score	A6_Score	A7_Score	A8_Score	A9_Score	A10_Sc
count	292.000000	292.000000	292.000000	292.000000	292.000000	292.000000	292.000000	292.000000	292.000000	292.000000	292.000000
mean	146.500000	0.633562	0.534247	0.743151	0.551370	0.743151	0.712329	0.606164	0.496575	0.493151	0.726027
std	84.437354	0.482658	0.499682	0.437646	0.498208	0.437646	0.453454	0.489438	0.500847	0.500811	0.446761
min	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	73.750000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	146.500000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.000000	0.000000	1.000000
75%	219.250000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
max	292.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

Number of null values in each column

```
In [10]: ad.isnull().sum()
```

```
id          0
A1_Score    0
A2_Score    0
A3_Score    0
A4_Score    0
A5_Score    0
A6_Score    0
A7_Score    0
A8_Score    0
A9_Score    0
A10_Score   0
age         0
gender      0
ethnicity    6
jundice     0
austim      0
contry_of_res 0
used_app_before 0
result      0
age_desc    0
relation    6
Class/ASD   0
dtype: int64
```

```
In [11]: ch.isnull().sum()
```

```
id          0
A1_Score    0
A2_Score    0
A3_Score    0
A4_Score    0
A5_Score    0
A6_Score    0
A7_Score    0
A8_Score    0
A9_Score    0
A10_Score   0
age         4
gender      0
ethnicity   43
jundice     0
austim      0
contry_of_res 0
used_app_before 0
result      0
age_desc    0
relation    43
Class/ASD   0
dtype: int64
```

Exploratory Data Analysis

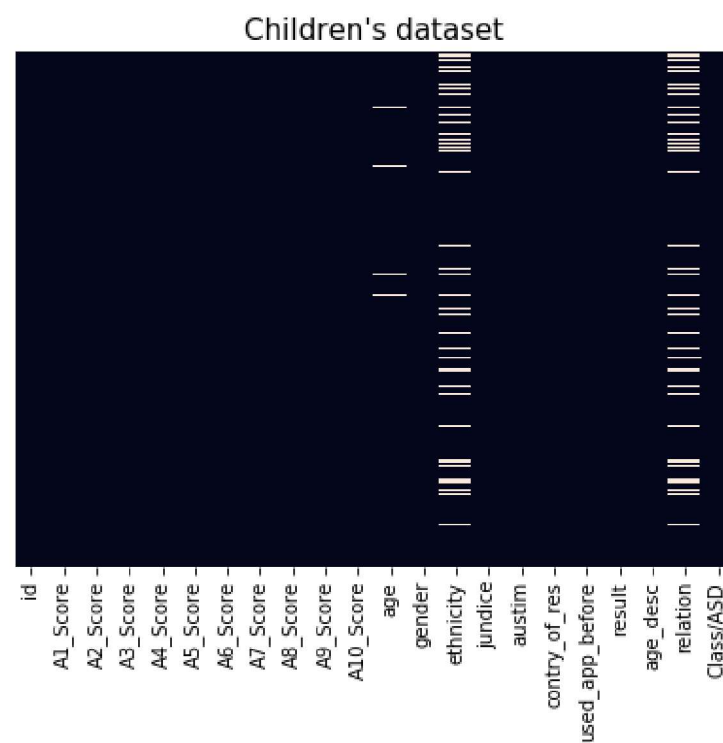
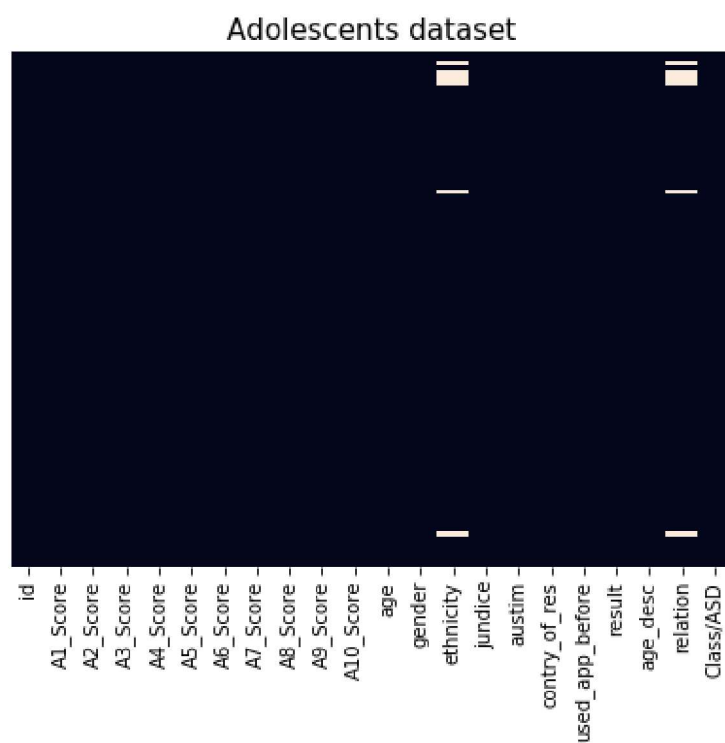
Heatmap showing missing values in each column


```

In [12]: fig, ax = plt.subplots(1,2,figsize=(15,5))
sns.heatmap(ad.isnull(),yticklabels=False,cbar=False,cmap='rocket',ax=ax[0])
ax[0].set_title('Adolescents dataset',fontsize = 15)
sns.heatmap(ch.isnull(),yticklabels=False,cbar=False,cmap='rocket',ax=ax[1])
ax[1].set_title("Children's dataset",fontsize = 15)

Text(0.5, 1, "Children's dataset")

```



ASD count

```
In [13]: print("No of adolescents diagonised with ASD = ",len(ad[ad['Class/ASD'] == 'YES']))  
print("No of adolescents not diagonised with ASD = ",len(ad[ad['Class/ASD'] == 'NO']))  
print("\nNo of children diagonised with ASD = ",len(ch[ch['Class/ASD'] == 'YES']))  
print("No of children not diagonised with ASD = ",len(ch[ch['Class/ASD'] == 'NO']))
```

No of adolescents diagonised with ASD = 63

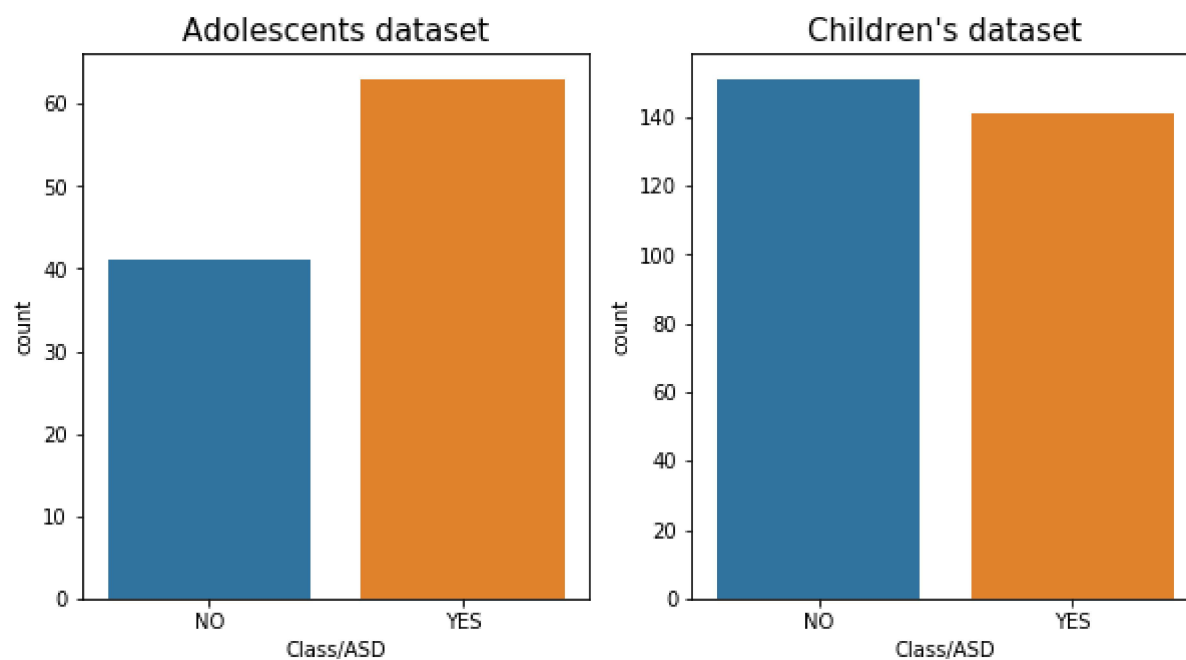
No of adolescents not diagonised with ASD = 41

No of children diagonised with ASD = 141

No of children not diagonised with ASD = 151

```
In [14]: fig, ax = plt.subplots(1,2,figsize=(10,5))
sns.countplot(ad['Class/ASD'],ax=ax[0])
ax[0].set_title('Adolescents dataset',fontsize = 15)
sns.countplot(ch['Class/ASD'],ax=ax[1])
ax[1].set_title("Children's dataset",fontsize = 15)
```

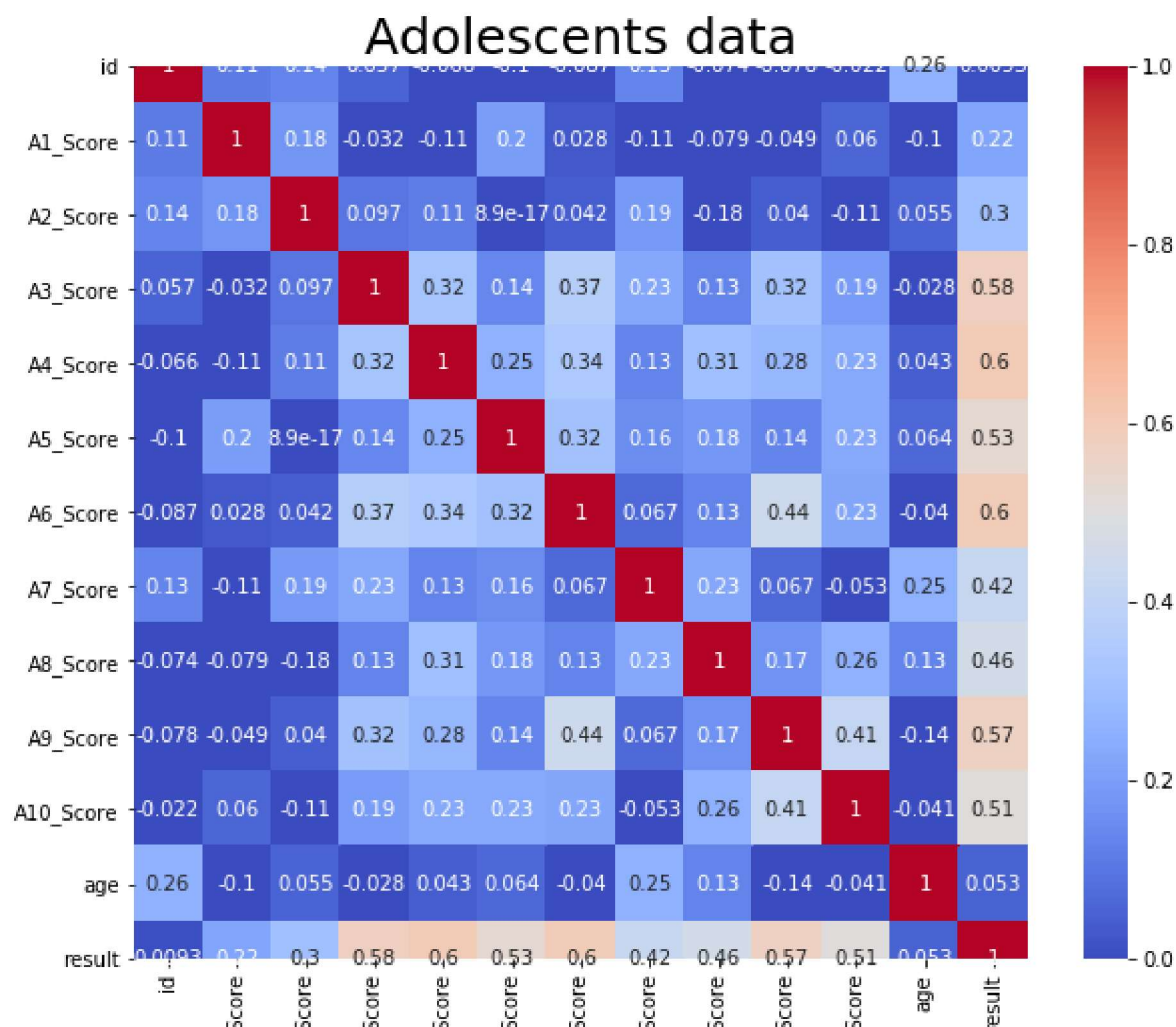
```
Text(0.5, 1.0, "Children's dataset")
```

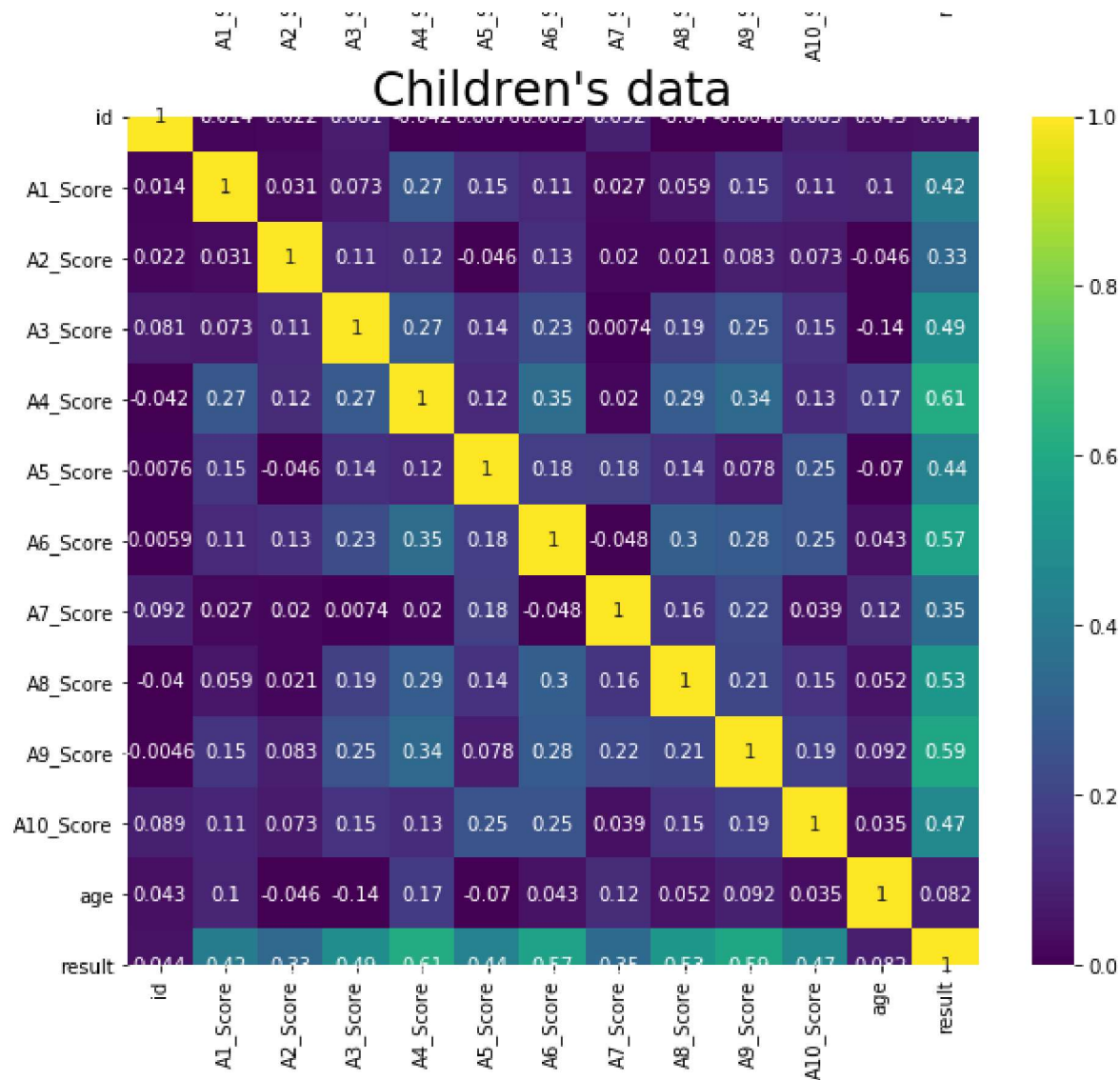


Correlaion

```
In [15]: fig, ax = plt.subplots(2,1,figsize=(10,18))
sns.heatmap(ad.corr(),annot=True,cmap='coolwarm',vmin=0, vmax=1,ax=ax[0])
ax[0].set_title('Adolescents data',fontsize = 25)
sns.heatmap(ch.corr(),annot=True,cmap='viridis',vmin=0, vmax=1,ax=ax[1])
ax[1].set_title("Children's data",fontsize = 25)
```

```
Text(0.5, 1, "Children's data")
```

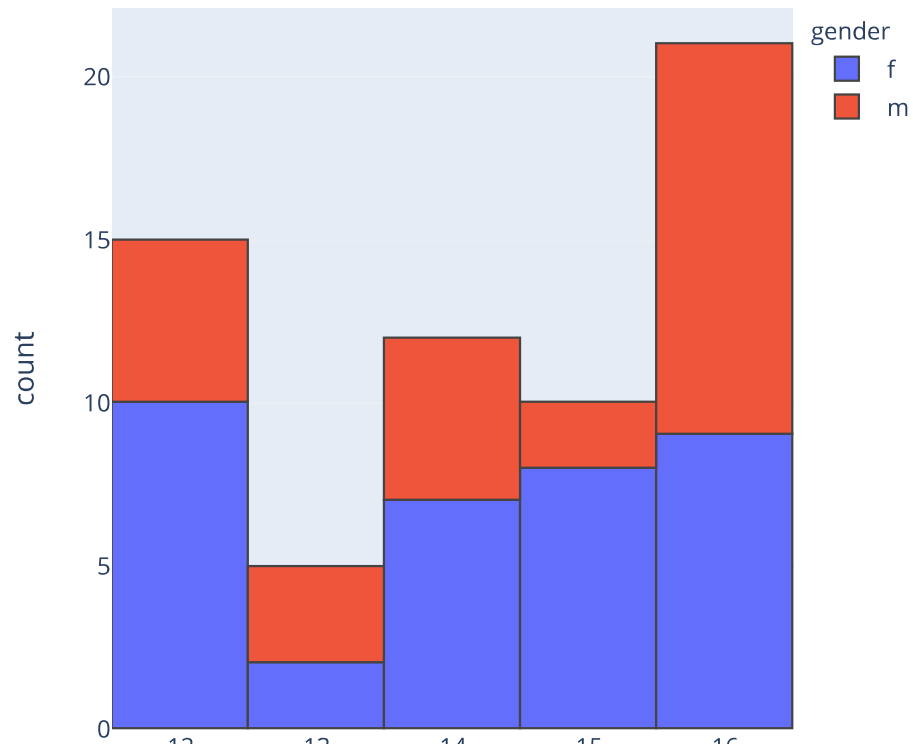




Age distribution of ASD positive over gender

```
In [16]: fig = px.histogram(ad[ad['Class/ASD'] == 'YES'], x="age", color="gender")
fig.update_layout(
    autosize=False,
    width=500,
    height=500,
    title = "Age distribution of ASD positive adolescents over gender"
)
fig.update_traces(marker_line_width=1)
fig.show()
```

Age distribution of ASD positive adolescents over gender



12

13

14

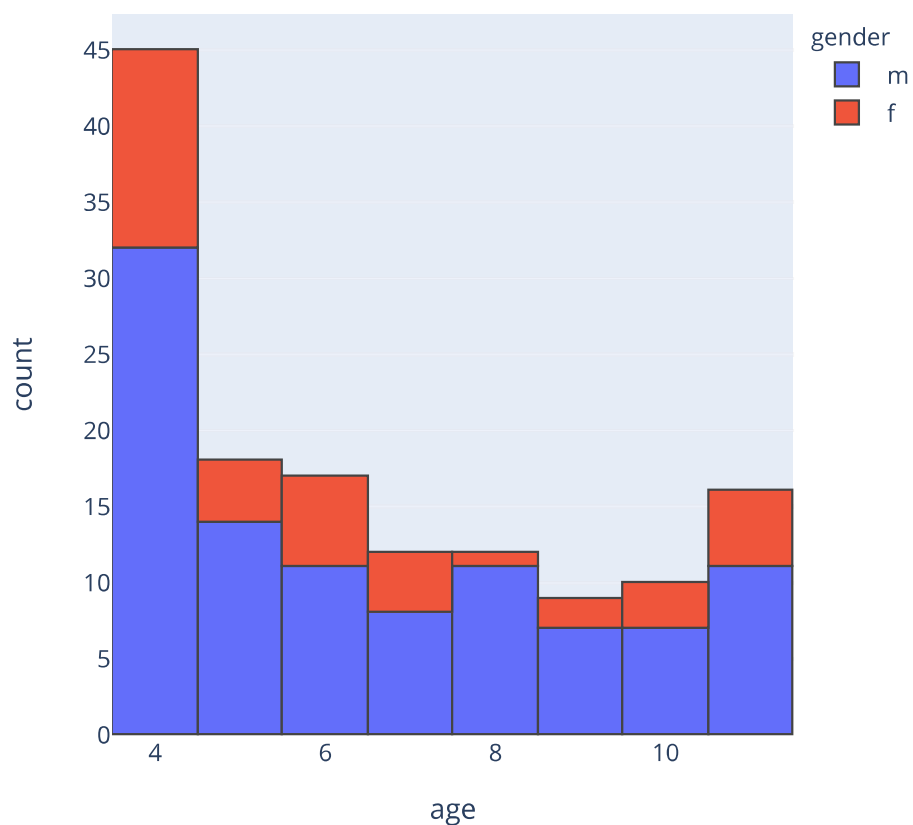
15

16

age

```
In [17]: fig = px.histogram(ch[ch['Class/ASD'] == 'YES'], x="age", color="gender")
fig.update_layout(
    autosize=False,
    width=500,
    height=500,
    title = "Age distribution of ASD positive children over gender"
)
fig.update_traces(marker_line_width=1)
fig.show()
```

Age distribution of ASD positive children over gender



Country wise distribution of ASD postive individuals

```
In [18]: ad_yes = ad[ad['Class/ASD'] == 'YES']  
         ch_yes = ch[ch['Class/ASD'] == 'YES']
```

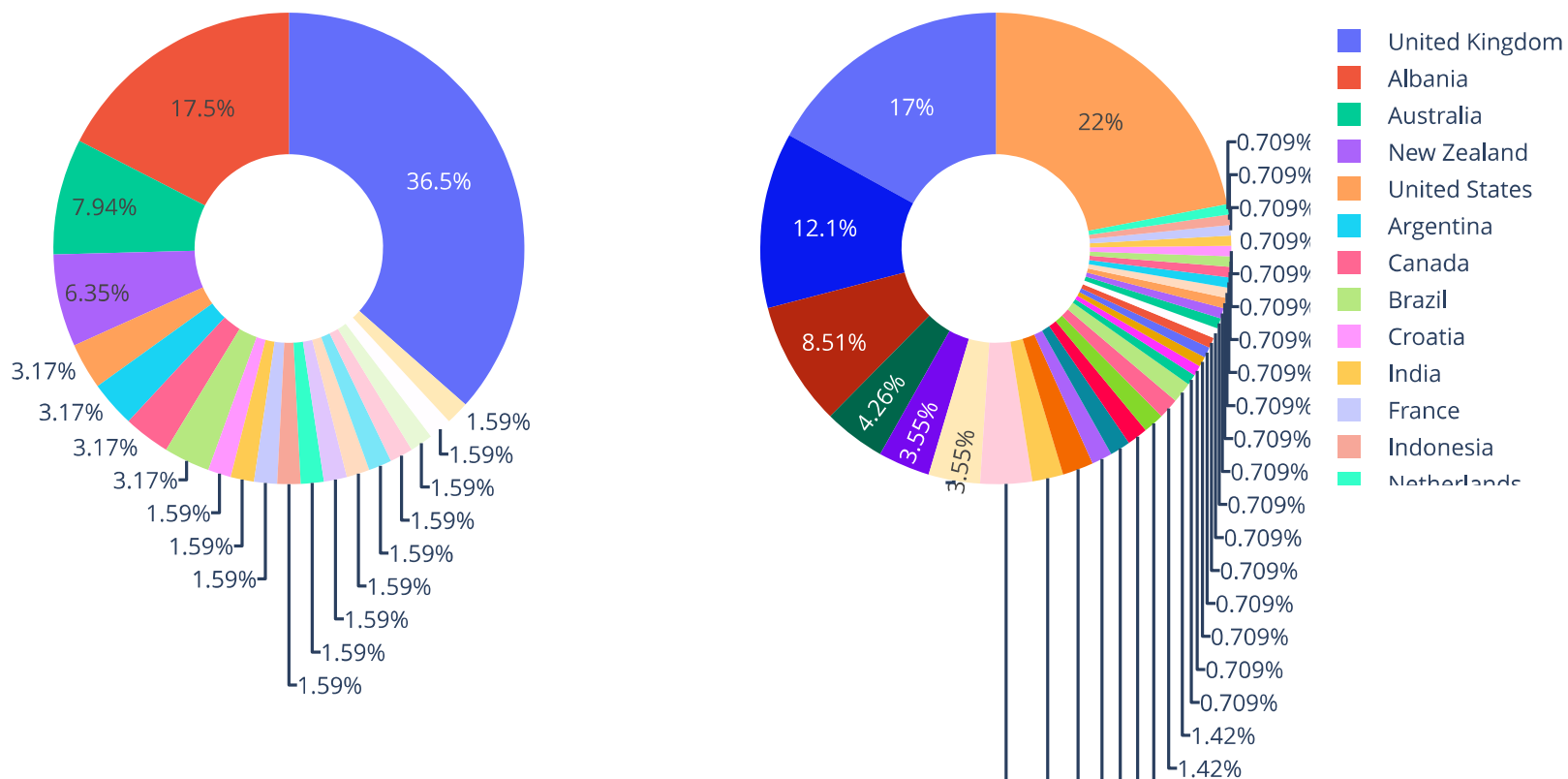
```

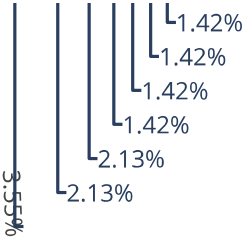
In [19]: labels1 = ad_yes['contry_of_res'].unique().tolist()
values1 = ad_yes['contry_of_res'].value_counts().tolist()
labels2 = ch_yes['contry_of_res'].unique().tolist()
values2 = ch_yes['contry_of_res'].value_counts().tolist()

fig = make_subplots(rows=1, cols=2, specs=[[{'type':'domain'}, {'type':'domain'}]])

fig.add_trace(go.Pie(labels=labels1, values=values1, name="Adolescents  "),1, 1)
fig.add_trace(go.Pie(labels=labels2, values=values2, name="Children"),1, 2)
fig.update_traces(hole=.4, hoverinfo="label+percent+name")
fig.update_layout(margin=dict(t=0, b=0, l=0, r=0))
fig.show()

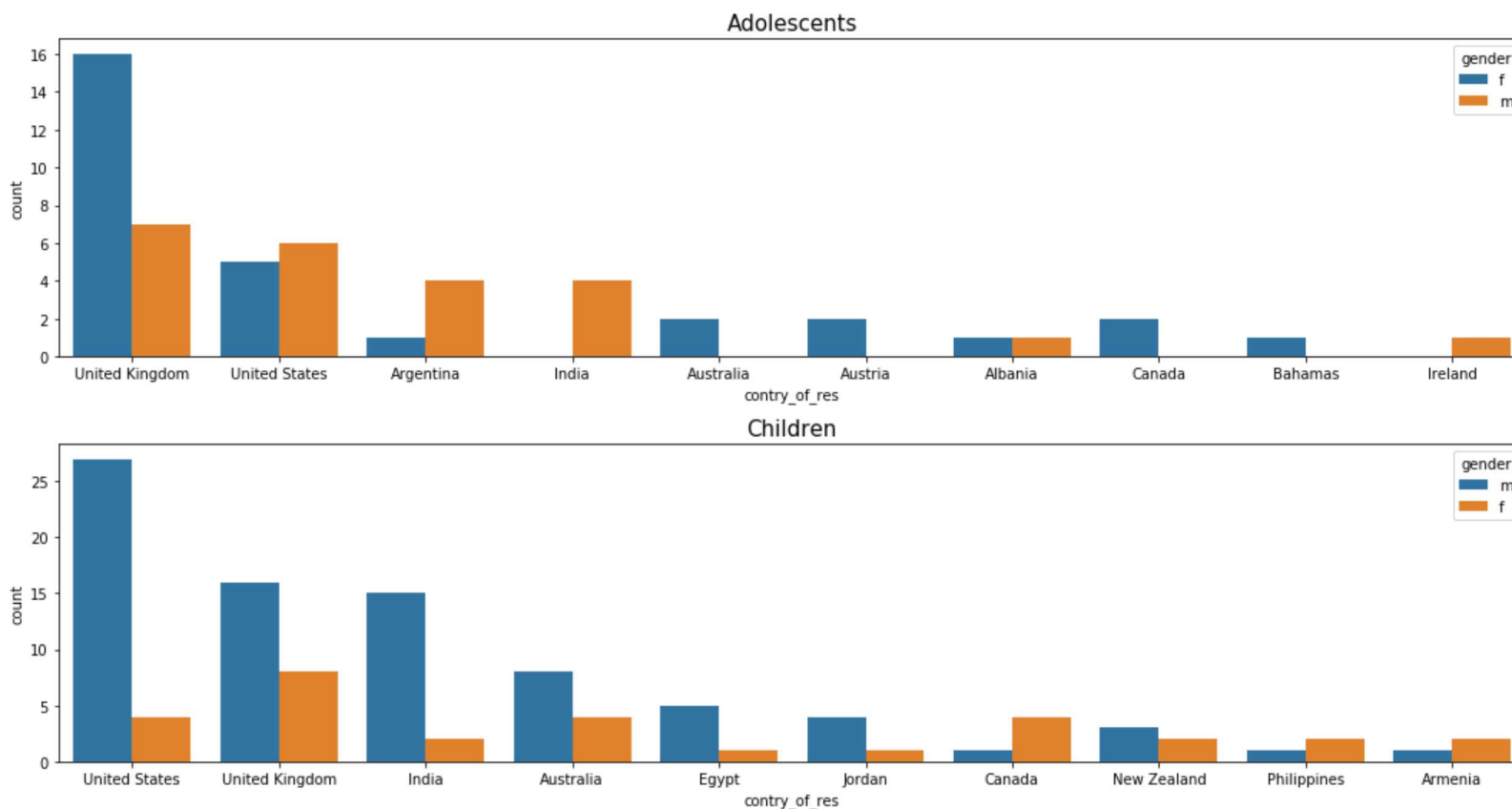
```





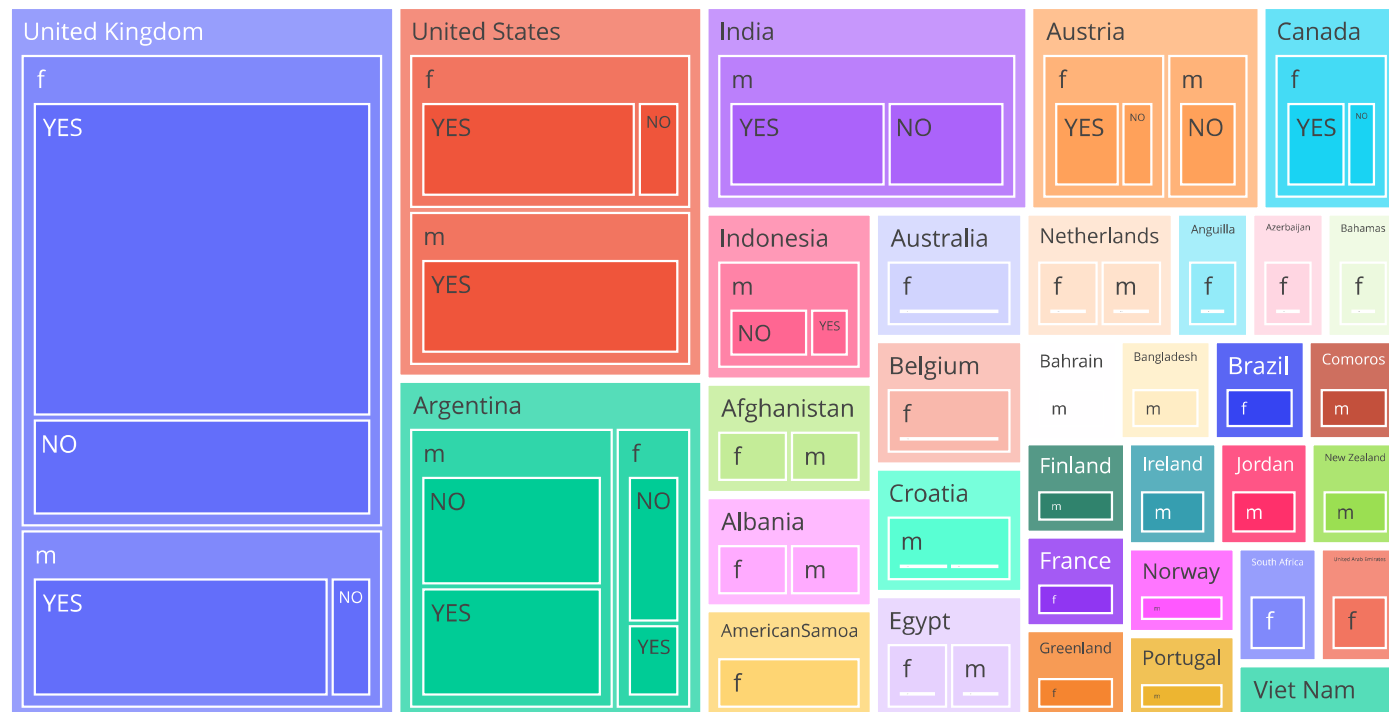
ASD postive individuals in top 10 countries

```
In [20]: fig, ax = plt.subplots(2,1,figsize=(15,8))
sns.countplot(x='contry_of_res',data=ad_yes,order= ad_yes['contry_of_res'].value_counts().index[:10],hue='gender')
ax[0].set_title('Adolescents',fontsize = 15)
sns.countplot(x='contry_of_res',data=ch_yes,order= ch_yes['contry_of_res'].value_counts().index[:10],hue='gender')
ax[1].set_title("Children",fontsize = 15)
plt.tight_layout()
```

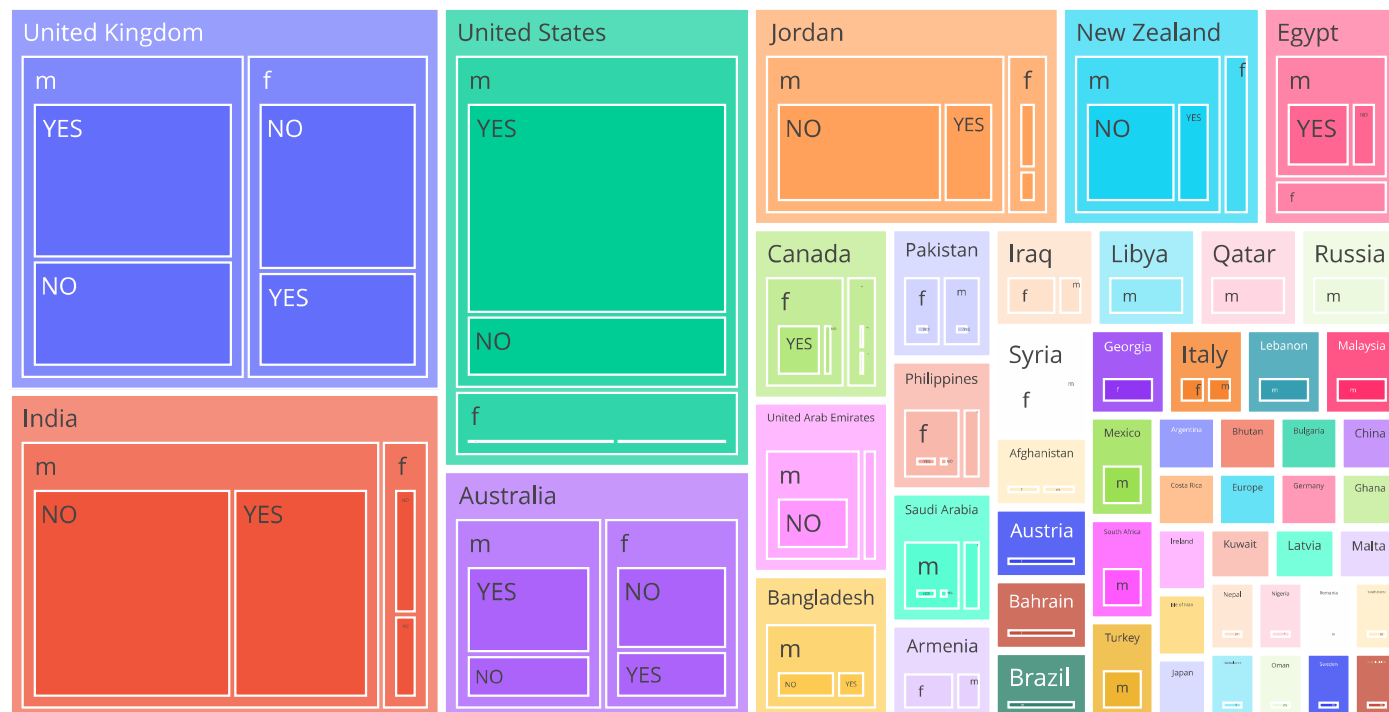


Distribution of ASD positive and ASD neagative individuals based on Country and gender

```
In [21]: fig = px.treemap(ad, path=['contry_of_res','gender','Class/ASD'])
fig.show()
```



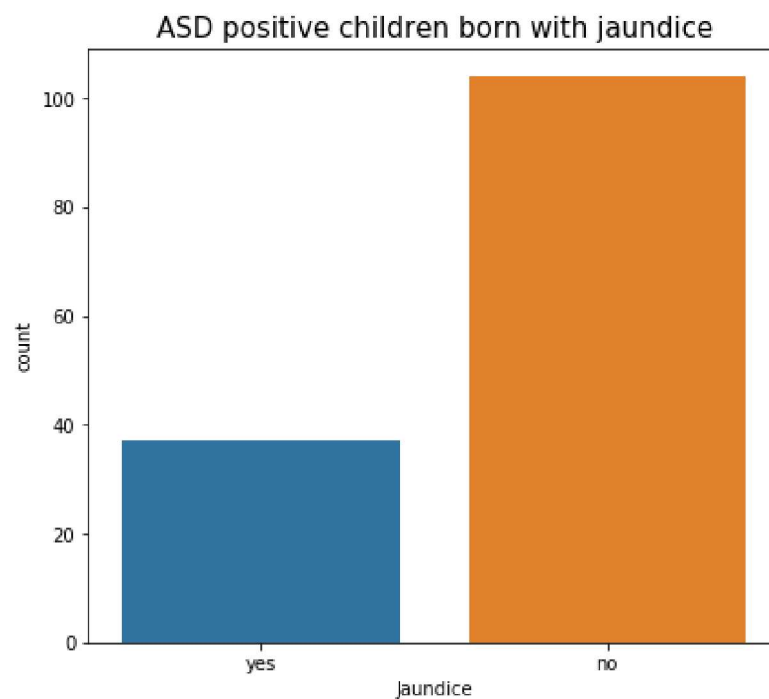
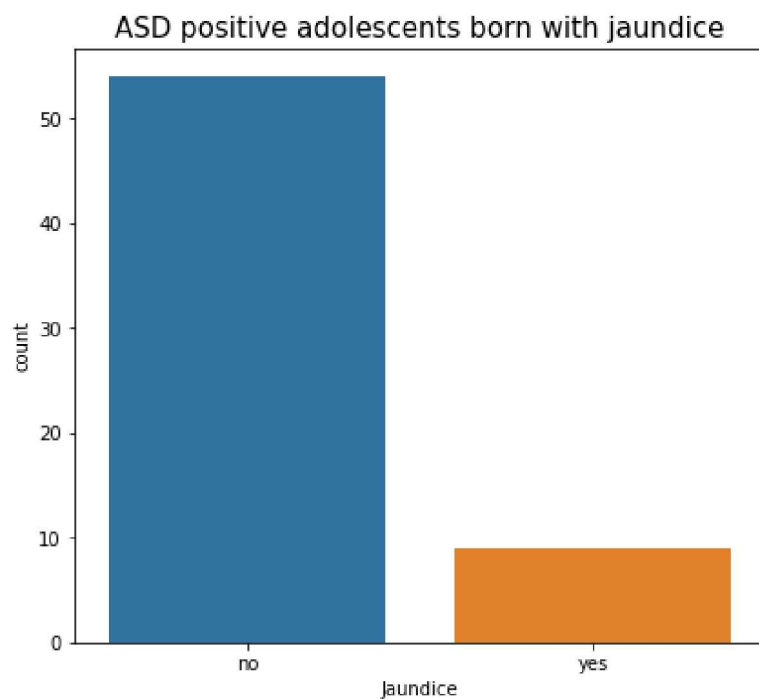
```
In [22]: fig = px.treemap(ch, path=['contry_of_res','gender','Class/ASD'])
fig.show()
```



Relation between Jaundice born individuals and ASD positive individuals

```
In [23]: fig, ax = plt.subplots(1,2,figsize=(15,6))
sns.countplot(x='jaundice',data=ad[ad['Class/ASD'] == 'YES'],ax=ax[0])
ax[0].set_title('ASD positive adolescents born with jaundice', fontsize=15)
ax[0].set_xlabel('Jaundice')
sns.countplot(x='jaundice',data=ch[ch['Class/ASD'] == 'YES'],ax=ax[1])
ax[1].set_title('ASD positive children born with jaundice', fontsize=15)
ax[1].set_xlabel('Jaundice')
```

```
Text(0.5, 0, 'Jaundice')
```



Count of ethnicity

```
In [24]: print("\tAdolescents\n",ad_yes['ethnicity'].value_counts())  
print("\n\tChildren\n",ch_yes['ethnicity'].value_counts())
```

```
      Adolescents  
White-European    30  
Others            10  
Asian             8  
Black             4  
Middle Eastern    4  
Latino            3  
Hispanic          2  
South Asian       1  
Name: ethnicity, dtype: int64
```

```
      Children  
White-European    62  
Asian            22  
Black            9  
South Asian       8  
Middle Eastern    8  
Hispanic          6  
Others            5  
Latino            5  
Pasifika          1  
Name: ethnicity, dtype: int64
```



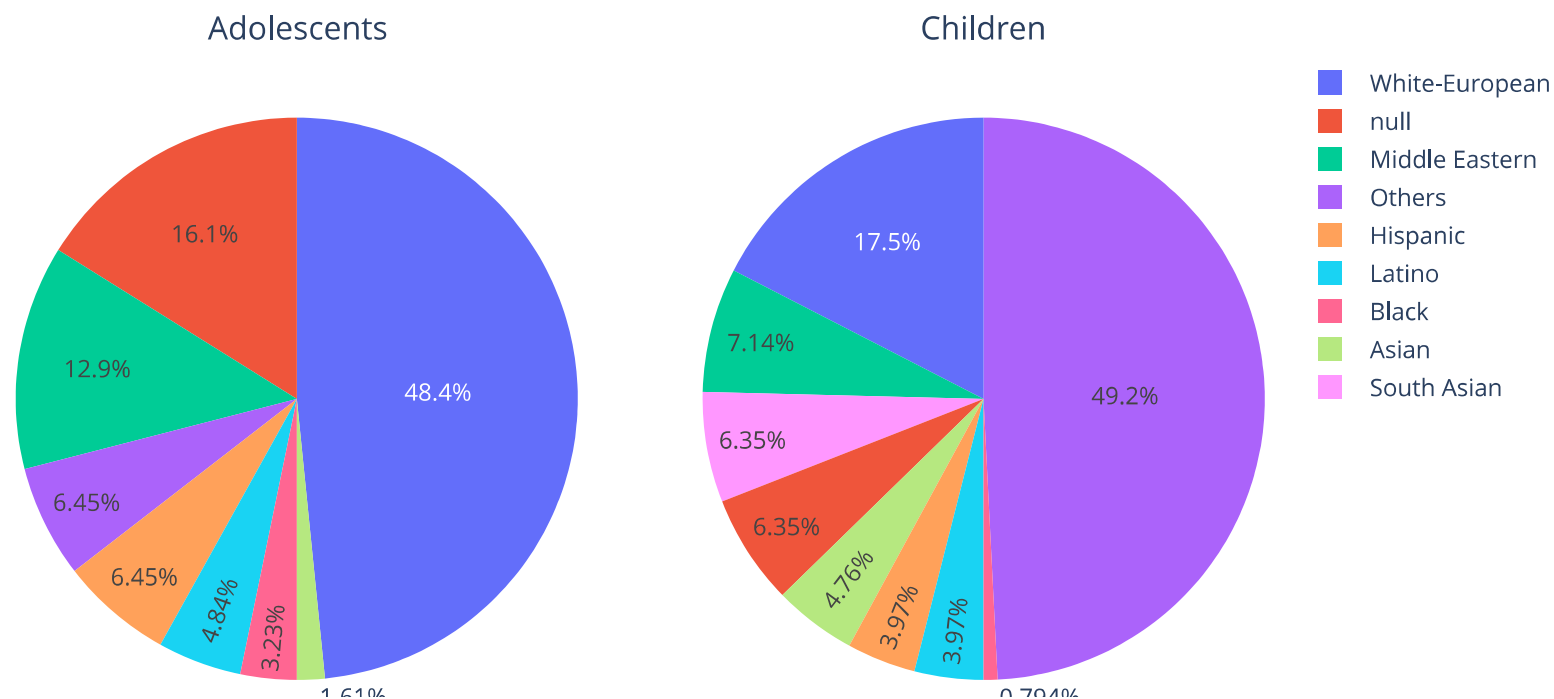
```

In [25]: labels1 = ad_yes['ethnicity'].unique().tolist()
values1 = ad_yes['ethnicity'].value_counts().tolist()
labels2 = ch_yes['ethnicity'].unique().tolist()
values2 = ch_yes['ethnicity'].value_counts().tolist()

fig = make_subplots(rows=1, cols=2, subplot_titles=("Adolescents", "Children"), specs=[[{'type': 'domain'}], {'type': 'domain'}])

fig.add_trace(go.Pie(labels=labels1, values=values1, name="Adolescents"), 1, 1)
fig.add_trace(go.Pie(labels=labels2, values=values2, name="Children"), 1, 2)
fig.show()

```



Distribution based on ethnicity and gender

```
In [26]: fig, ax = plt.subplots(1,2,figsize=(25,8))
sns.countplot(x='ethnicity',hue='gender',data=ad_yes,ax=ax[0])
ax[0].set_title('Adolescents',fontsize = 15)
sns.countplot(x='ethnicity',hue='gender',data=ad_yes,ax=ax[1])
ax[1].set_title('Children',fontsize = 15)
```

```
Text(0.5, 1.0, 'Children')
```

