# CHILD MALNUTRITION_COUNTRY WISE

Data Analysis *By ANKIT SHARMA*

Introduction

There are three key physiological measures of undernourishment and undernutrition in children. The measures discussed in the sub-sections below are:

● Stunting – being too short for one’s age;

● Wasting – being dangerously thin for one’s height;

● Underweight – low weight-for-age in children.

● Overweight - more weight for-age

1. Too little height-for-age: Stunting

Children who are stunted are determined as having a height which falls two standard deviations below the median height-for-age of the World Health Organization’s Child Growth Standards.

Stunting is an indicator of severe malnutrition. Unlike wasting and low weight-for-age, the impacts of stunting on child development are considered to be largely irreversible beyond the first 1000 days of a child’s life. It can have severe impacts on both cognitive and physical development throughout an individual’s life.

Stunting can be caused by a range of compounding factors including nutritional intake of the child, as well as the mother during pregnancy, the recurrence of infectious diseases and infections from poor hygiene practices.

The global map of the prevalence of childhood stunting is shown as the share of the under-5 population who are defined as stunted. Note that many countries report stunting prevalence through periodic health and demographic surveys, meaning that this data is often not available on an annual basis. The year of the latest published estimates vary by country so you may have to use the time scrollbar to find the most up-to-date figure for a given country.

2. Too little weight-for-height: Wasting

Wasting is defined as being dangerously thin for one’s height, and is generally a sign (especially in children) of rapid weight loss. A child is classified as wasted if his or her weight-for-height is more than two standard deviations below the median for the international reference population ages 0-59 months. The factors which contribute to this weight loss are associated with measures related to both diet and nutrition, and infection. As a result, wasting is often compounded by conditions of poor nutrition, feeding practices as well as inadequate sanitary conditions.

Unlike stunting, wasting can be treated through improved nutritional intakes, healthcare interventions and treatment of infection.

In 2015, South Sudan experienced the highest prevalence of wasting, with 22.7 percent of under-5s defined as wasted. The prevalence of wasting is typically highest across Sub-Saharan Africa and South Asia, with countries such as India, Sri Lanka, Djibouti, Sudan and Niger recording some of the highest levels (greater than 15 percent).

The share of children suffering from wasting has been declining. If we compare our global map in the early 2000s to a decade later, we see the number of countries with a prevalence greater than 15 percent has fallen. However, the nature of wasting- exemplified often by rapid weight loss-means that particular short-term events which impact food supplies can disrupt long-term trends. This is particularly prevalent in countries with poor political stability; for example, we see a large spike in childhood wasting in the Democratic Republic of Congo during the late 1990s-early 2000s during the Second Congo War.

3. Underweight children

Undernourishment, or the incidence of being underweight for age, can include children who are stunted, wasted or suffering from insufficient energy intake over a longer period of time.

In the dataset we see the share of children under-5 who are defined as underweight for their age across world regions since 1990. Overall, we see a steady decline at the global level, falling from around 25 percent in 1990 to 15 percent in 2015.

South Asia- despite having the highest regional prevalence- has made significant progress over the last few decades, reducing undernourishment by 20 percentage points from 1990-2017. The rate of undernourishment in Sub-Saharan Africa has also fallen notably, from 30 percent in 1990 to below 20 percent in 2017. Rates in East Asia, Latin America, North Africa and the Middle East are notably lower than South Asia and Sub-Saharan Africa, but have also seen significant declines, each more than halving the prevalence of undernourishment since 1990.

4. Overweight children

Moderate and severe: % aged 0-59 months who are above two standard deviations from median weight-for-height

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt # this is used for the plot the graph
import seaborn as sns # used for plot interactive graph.
import plotly.offline as py
py.init_notebook_mode(connected=True)
import plotly.graph_objs as go
import warnings
warnings.filterwarnings('ignore')
from pylab import rcParams
# figure size in inches
%matplotlib inline
```

## Three Datasets are downloaded from UNICEF site and used for Exploratory Data Analysis (EDA)

```
def overview():
  df = pd.read_csv("/content/country-wise-average.csv")
  df_world = pd.read_csv("/content/World_Malnutrition_Data.csv")
  df_region = pd.read_csv("/content/Region_Data.csv")
  print("First 5 lines of data \n\n")
  print(df.head())
  print("\n")
  print(df_world.head())
  print("\n")
  print(df_region.head())
  return df, df_world, df_region
df, df_world, df_region=overview()
```

```
First 5 lines of data


       Country  Income Classification  Severe Wasting    Wasting  Overweight  \
0  AFGHANISTAN                    0.0        3.033333  10.350000    5.125000
1      ALBANIA                    2.0        4.075000   7.760000   20.800000
2      ALGERIA                    2.0        2.733333   5.942857   12.833333
```

```
3      ANGOLA                    1.0        2.400000   6.933333    2.550000
4   ARGENTINA                    2.0        0.200000   2.150000   11.125000

    Stunting  Underweight  U5 Population ('000s)
0  47.775000    30.375000            4918.561500
1  24.160000     7.700000             232.859800
2  19.571429     7.342857            3565.213143
3  42.633333    23.600000            3980.054000
4  10.025000     2.600000            3613.651750


               Country                  Indicator     Sex  TIME_PERIOD  \
0          Afghanistan  Under-five mortality rate  Female         1970
1          Afghanistan  Under-five mortality rate    Male         1970
2          Afghanistan  Under-five mortality rate   Total         1970
3 United Arab Emirates  Under-five mortality rate  Female         1970
4 United Arab Emirates  Under-five mortality rate    Male         1970

    OBS_VALUE             Unit of Measure
0  295.370962  Deaths per 1000 live births
1  309.649534  Deaths per 1000 live births
2  302.709104  Deaths per 1000 live births
3   90.916017  Deaths per 1000 live births
4  104.215923  Deaths per 1000 live births


   Geographic Area                        Indicator    Sex  TIME_PERIOD  \
0  Northern Africa  Height-for-age <-2 SD (stunting)  Total         1985
1  Northern Africa  Height-for-age <-2 SD (stunting)  Total         1986
2  Northern Africa  Height-for-age <-2 SD (stunting)  Total         1987
3  Northern Africa  Height-for-age <-2 SD (stunting)  Total         1988
4  Northern Africa  Height-for-age <-2 SD (stunting)  Total         1989

   OBS_VALUE Unit of Measure
0         31         Percent
1         30         Percent
2         30         Percent
3         29         Percent
4         29         Percent
```

## To check Whether there are null values in datasets or not

```
print(df.isnull().sum())
print(df_world.isnull().sum())
print(df_region.isnull().sum())
```

```
Country                   0
Income Classification     0
Severe Wasting           12
Wasting                   2
Overweight                3
Stunting                  1
Underweight               2
```

```
U5 Population ('000s)     0
dtype: int64
Country            0
Indicator          0
Sex                0
TIME_PERIOD        0
OBS_VALUE          0
Unit of Measure    0
dtype: int64
Geographic Area    0
Indicator          0
Sex                0
TIME_PERIOD        0
OBS_VALUE          0
Unit of Measure    0
dtype: int64
```

## How I Dealt with Null values:-

I will usually remove rows with NaN values if the percentage of NaN values are below 5%

For df, we can remove NaN values in wasting, overweight, stunting and underweight.

However for severe wasting, I will replace them with median values.

```
from sklearn.impute import SimpleImputer
import numpy as np
df= df.dropna(subset = ['Wasting', 'Overweight', 'Stunting', 'Underweight'])
imputer = SimpleImputer(missing_values = np.nan, strategy = 'median')
df[['Severe Wasting']] = imputer.fit_transform(df[['Severe Wasting']])
print("Null values after updating dataset")
print("\n")
print(df.isnull().sum())
```

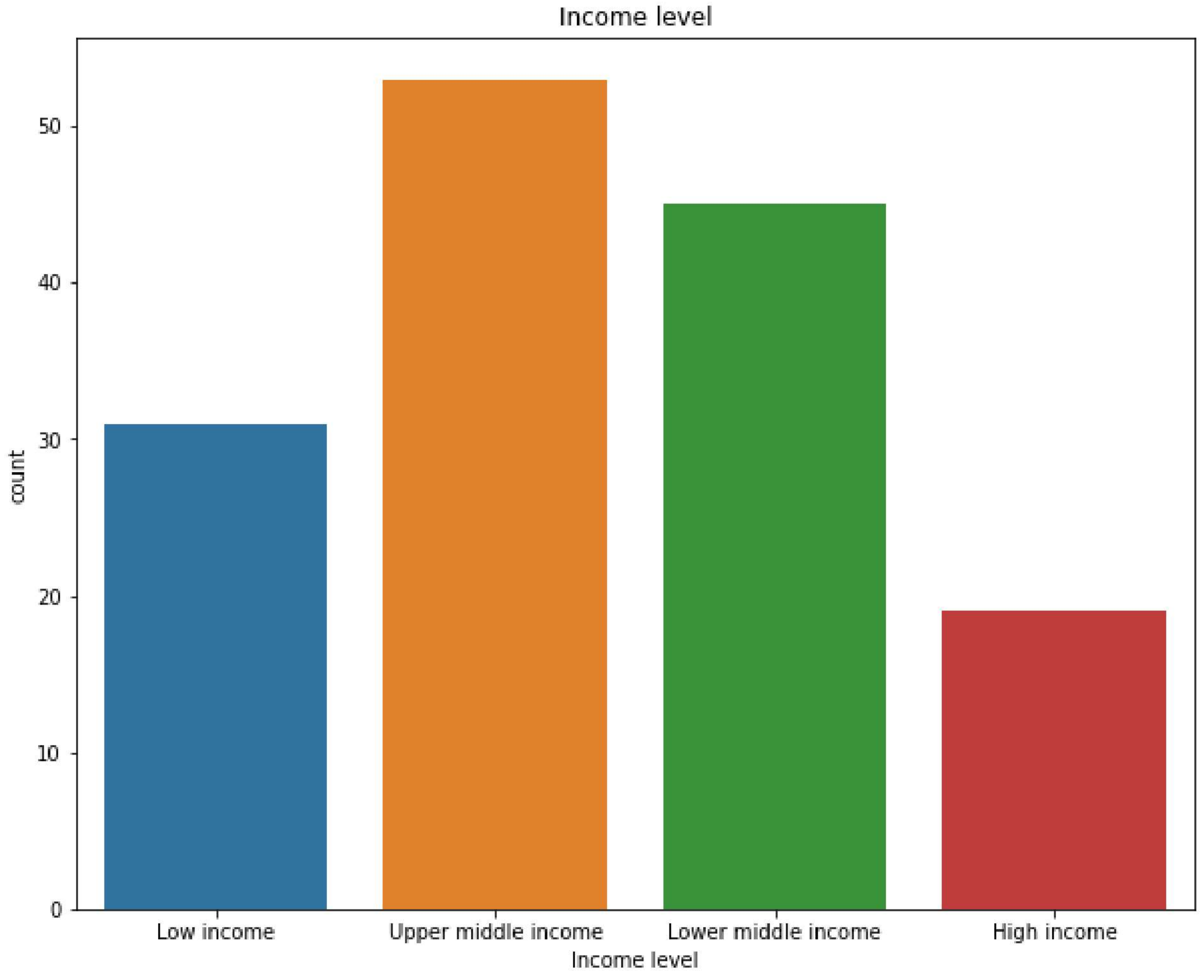
```
Null values after updating dataset


Country                  0
Income Classification    0
Severe Wasting           0
Wasting                  0
Overweight               0
Stunting                 0
Underweight              0
U5 Population ('000s)    0
dtype: int64
```

### Income level

Here, we are converting those numbers in income classification to their real meaning. This allows readers to understand the context without having to constantly go back the description.

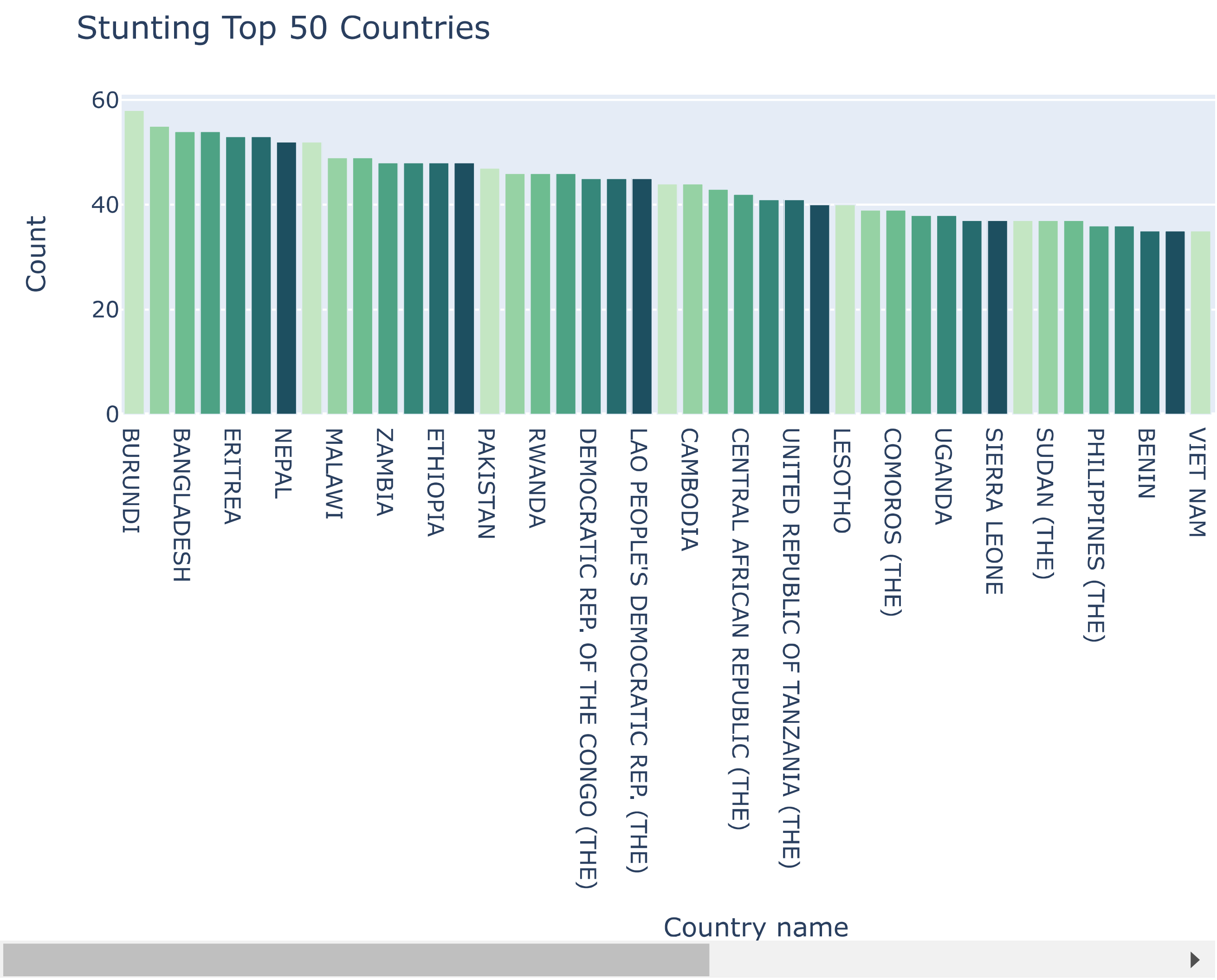
```
df['Income Classification'] = df['Income Classification'].astype('int')

# Create a new column to represent income level
def func(row):
    if row == 0:
        return 'Low income'
    elif row == 1:
        return 'Lower middle income'
    elif row == 2:
        return 'Upper middle income'
    else:
        return 'High income'
df['Income level'] = df.apply(lambda x: func(x['Income Classification']), axis=1)

# Plot countplot
plt.figure(figsize = (10,8))
sns.countplot(data = df, x = 'Income level').set_title('Income level')
```

```
Text(0.5, 1.0, 'Income level')
```



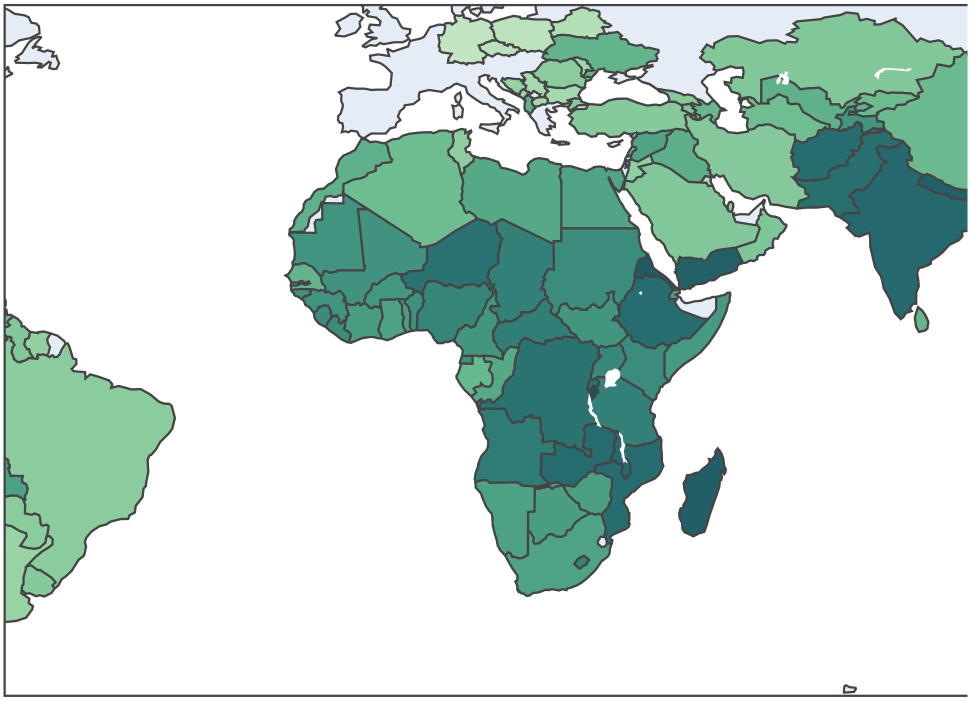
## Stunting – Moderate and severe: % of children aged 0–59 months who are below minus two standard deviations from median height-for-age

```
import plotly.express as px
name = df.groupby("Country")["Stunting"].mean().sort_values(ascending=False).index[:50]
yax = df.groupby("Country")["Stunting"].mean().sort_values(ascending=False).round()[:50]

fig = px.bar(df, y=yax, x=name, color = name, color_discrete_sequence=px.colors.sequential.Bl
fig.update_layout(
    title="Stunting Top 50 Countries",
    xaxis_title="Country name",
    yaxis_title="Count"

)

fig.show()
```



```
import plotly.graph_objs as go
x = df.groupby(["Country"])["Stunting"].mean()
layout = dict(geo = {'scope':'world'},title="Stunting percentage")
map = go.Figure(data = [dict(type = 'choropleth',
            locations = x.index,
            locationmode = 'country names',
            colorscale= px.colors.sequential.Blugrn,
            text= x.index,
            z=x,
            colorbar = {'title':'stunting %' })],layout = layout)

map.show()
```

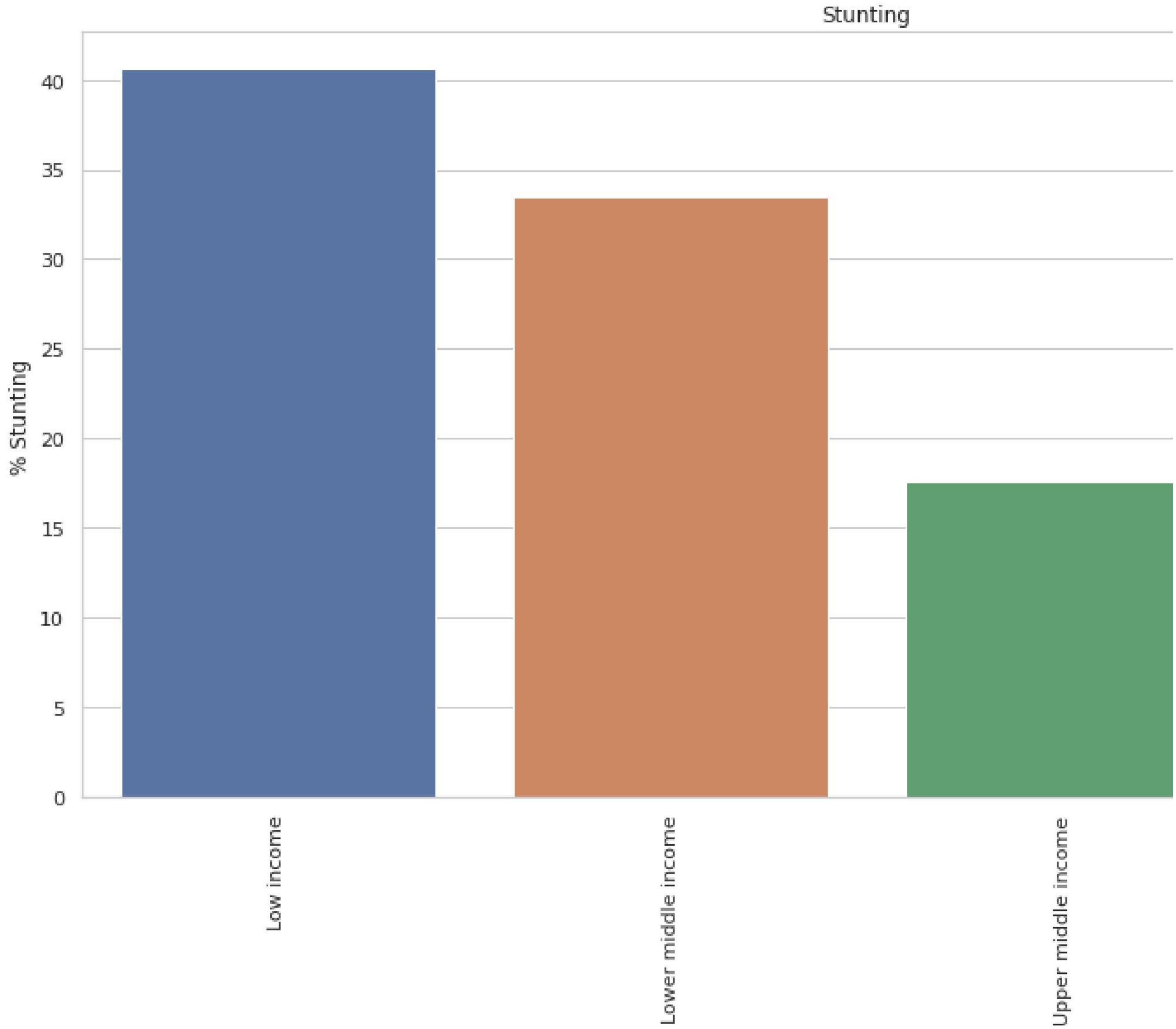
Stunting percentage



## Relation between Income Level and Stunting

```
plt.figure(figsize=(16, 8))
x = df.groupby(["Income level"])["Stunting"].mean().sort_values(ascending=False)
sns.set(style="whitegrid")
ax = sns.barplot(x.index, x)
ax.set_title('Stunting')
ax.set_ylabel('% Stunting')
ax.set_xlabel('Income level')
plt.xticks(rotation = 90)
```

```
(array([0, 1, 2, 3]), <a list of 4 Text major ticklabel objects>)
```



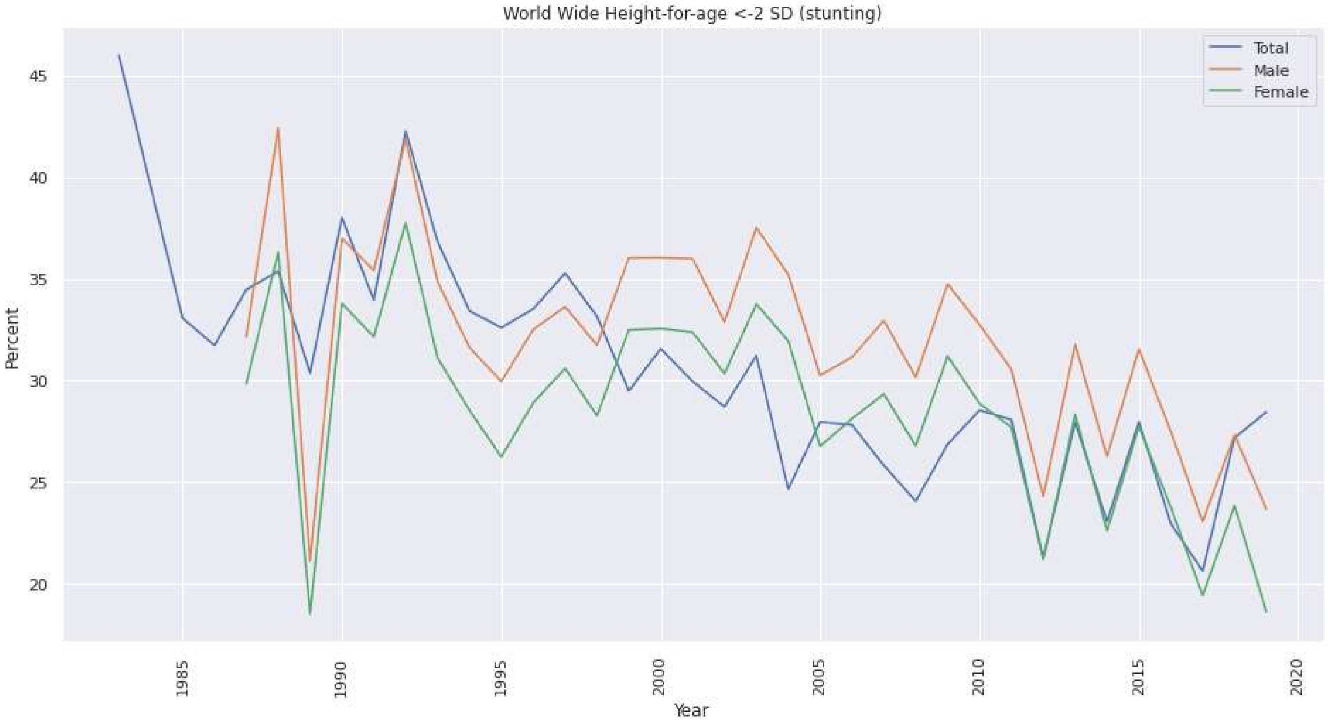
From Graph we can say as the income level increases Stunting decreases

## Time series of World Wide Height-for-age <-2 SD (stunting)

## Representation of rate of Stunting over the time period

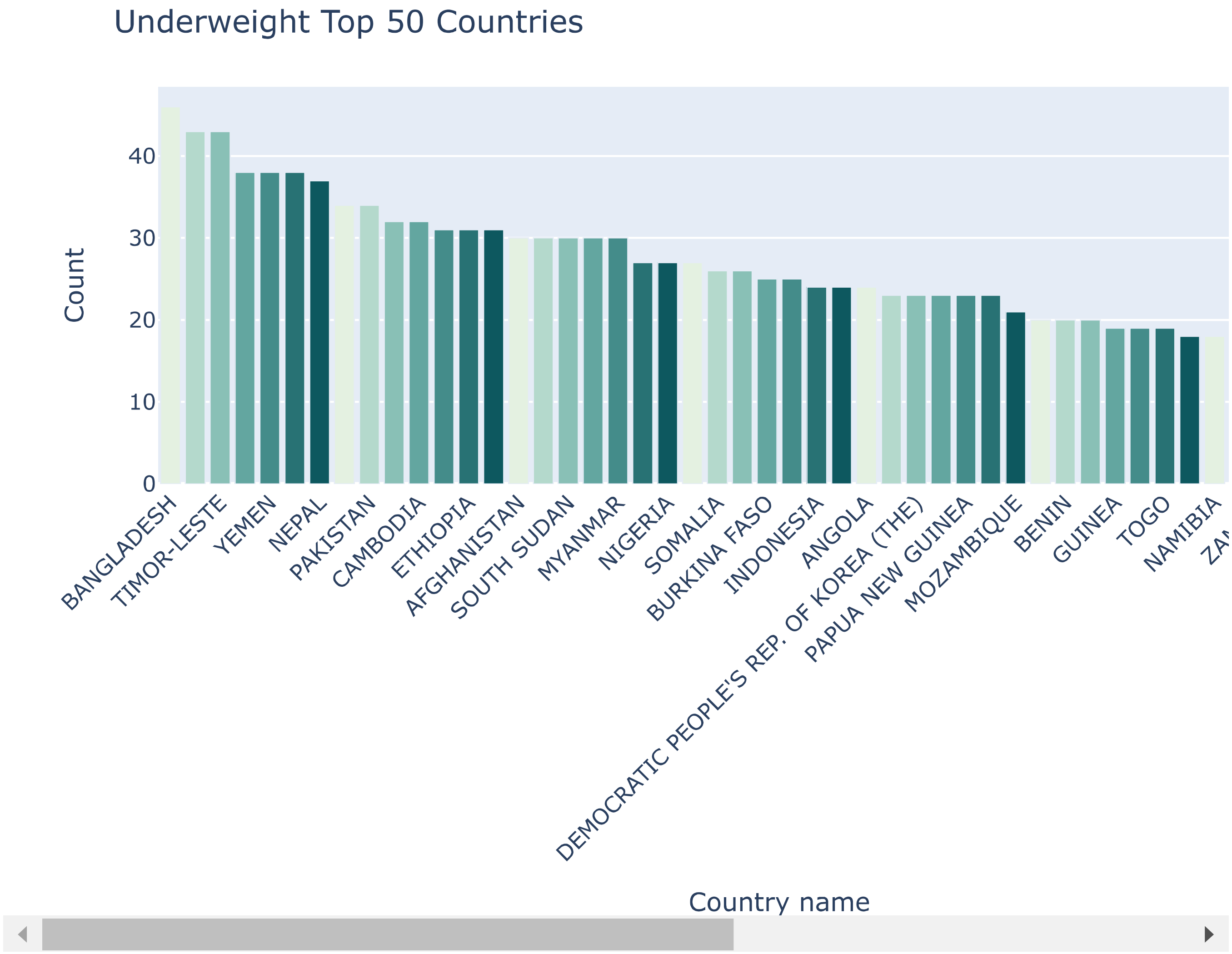
```
df_world.rename(columns = {"TIME_PERIOD" : "Year", "OBS_VALUE" : "value"}, inplace =True)
df_world_mort = df_world[df_world["Indicator"] == "Under-five mortality rate"]
df_world_stunting = df_world[df_world.Indicator == "Height-for-age <-2 SD (stunting)"]
df_world_underwt = df_world[df_world.Indicator == "Weight-for-age <-2 SD (Underweight)"]
df_world_mort.drop(["Indicator"],axis="columns",inplace = True)
plt.figure(figsize=(16, 8))
sns.set(style="darkgrid")
cols = ["Total","Male","Female"]
for col in cols:
    x = df_world_stunting[df_world_stunting["Sex"] == col].groupby("Year")["value"].mean()
    ax = sns.lineplot(x.index, x, label=col)
ax.set_title('World Wide Height-for-age <-2 SD (stunting)')
ax.set_ylabel('Percent')
ax.set_xlabel('Year')
ax.legend()
plt.xticks(rotation = 90)
```

```
     (array([1980., 1985., 1990., 1995., 2000., 2005., 2010., 2015., 2020.,
             2025.]), <a list of 10 Text major ticklabel objects>)
```



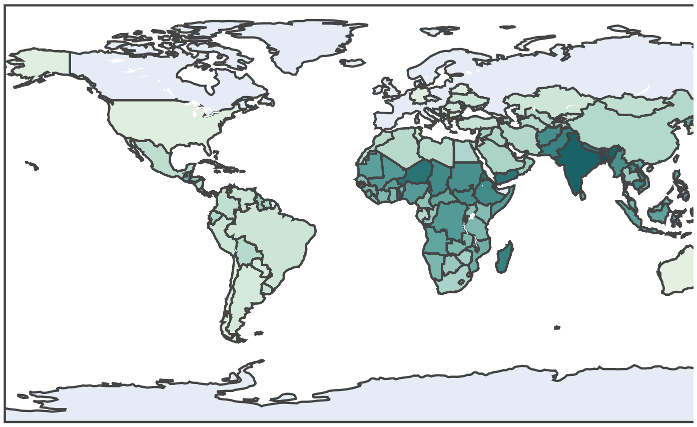
## Underweight – Moderate and severe: % of children aged 0–59 months who are below minus two standard deviations from median weight-for-age

```
name1 = df.groupby("Country")["Underweight"].mean().sort_values(ascending=False).index[:50]
yax1 = df.groupby("Country")["Underweight"].mean().sort_values(ascending=False).round()[:50]

fig = px.bar(df, y=yax1, x=name1, color = name1, color_discrete_sequence=px.colors.sequential
fig.update_layout(
    title="Underweight Top 50 Countries",
    xaxis_title="Country name",
    yaxis_title="Count"
)
fig.update_xaxes(tickangle=-45)

fig.show()
```



```
x = df.groupby(["Country"])["Underweight"].mean()
layout = dict(geo = {'scope':'world'},title="Underweight Percentage")
map = go.Figure(data = [dict(type = 'choropleth',
            locations = x.index,
            locationmode = 'country names',
            colorscale= px.colors.sequential.Mint,
            text= x.index,
            z=x,
            colorbar = {'title':'stunting %' })],layout = layout)
map.show()
```

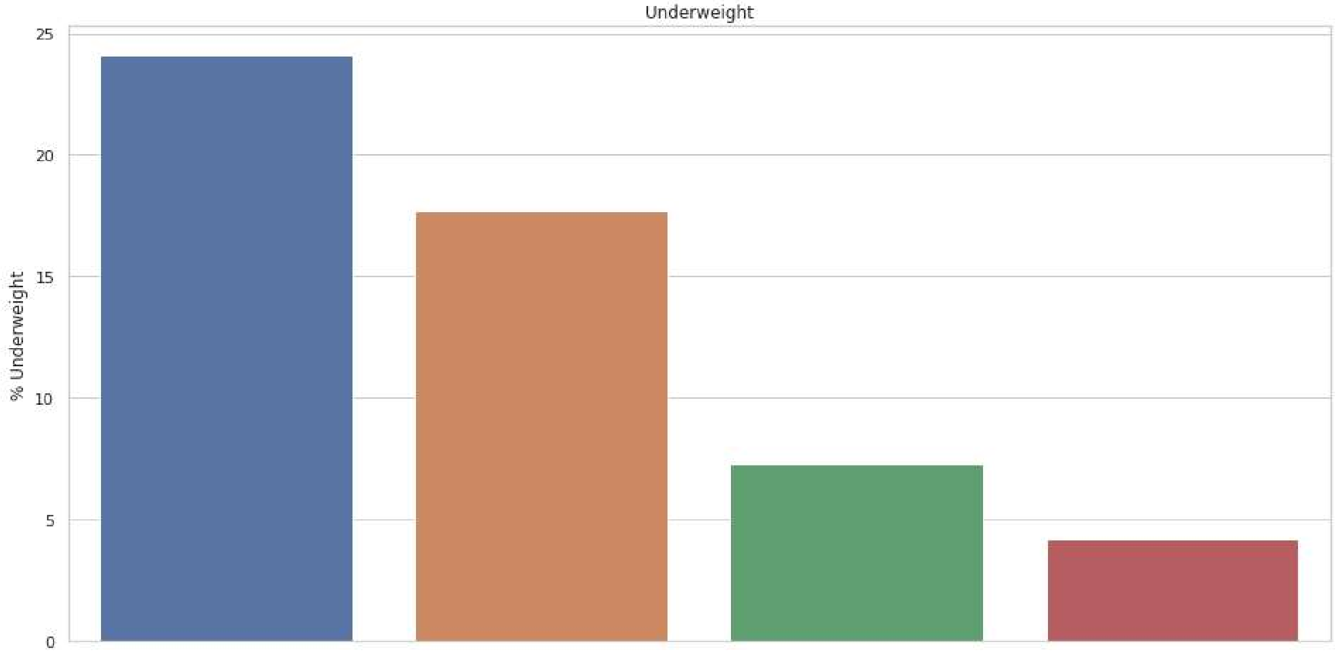
Underweight Percentage



## Relation between Income Level and Underweight

```
plt.figure(figsize=(16, 8))
x = df.groupby(["Income level"])["Underweight"].mean().sort_values(ascending=False)
sns.set(style="whitegrid")
ax = sns.barplot(x.index, x)
ax.set_title('Underweight')
ax.set_ylabel('% Underweight')
ax.set_xlabel('Income Classification')
plt.xticks(rotation = 90)
```

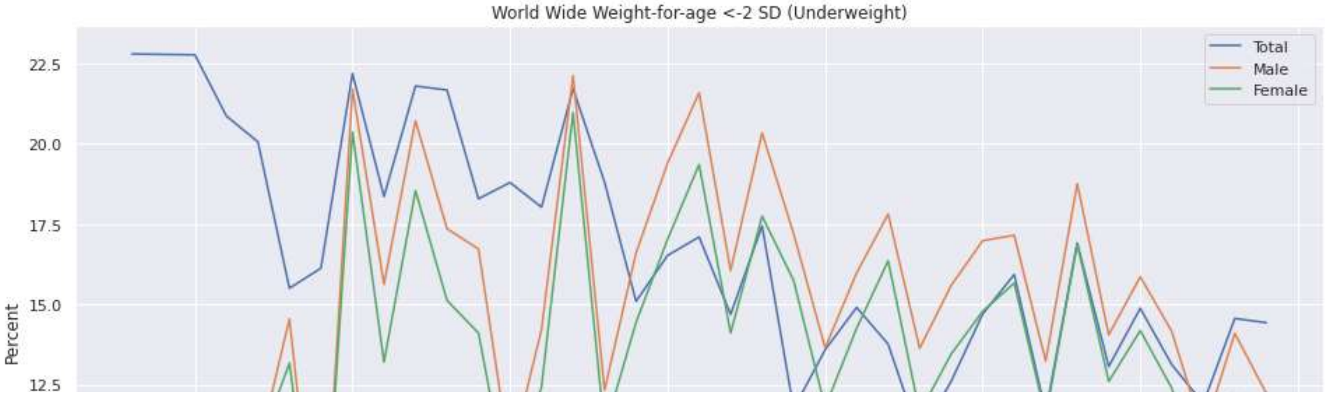
```
(array([0, 1, 2, 3]), <a list of 4 Text major ticklabel objects>)
```



## Time series of World Wide Weight-for-age <-2 SD (Underweight)

```
plt.figure(figsize=(16, 8))
sns.set(style="darkgrid")
cols = ["Total","Male","Female"]
for col in cols:
    x = df_world_underwt[df_world_underwt["Sex"] == col].groupby("Year")["value"].mean()
    ax = sns.lineplot(x.index, x, label=col)
ax.set_title('World Wide Weight-for-age <-2 SD (Underweight)')
ax.set_ylabel('Percent')
ax.set_xlabel('Year')
ax.legend()
plt.xticks(rotation = 90)
```

```
(array([1980., 1985., 1990., 1995., 2000., 2005., 2010., 2015., 2020.,
        2025.]), <a list of 10 Text major ticklabel objects>)
```

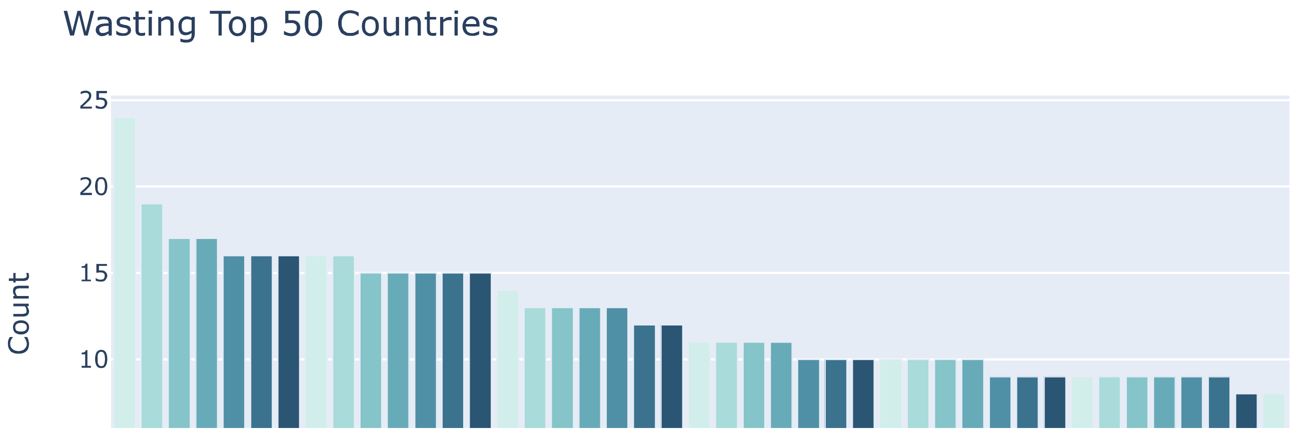


Double-click (or enter) to edit

Underheight (Wasting)

Wasting – Moderate and severe: % of children aged 0–59 months who are below minus two standard deviations from median weight-for-height

```
name2 = df.groupby("Country")["Wasting"].mean().sort_values(ascending=False).index[:50]
yax2 = df.groupby("Country")["Wasting"].mean().sort_values(ascending=False).round()[:50]

fig = px.bar(df, y=yax2, x=name2, color = name2, color_discrete_sequence=px.colors.sequential
fig.update_layout(
    title="Wasting Top 50 Countries",
    xaxis_title="Country name",
    yaxis_title="Count"
)
fig.update_xaxes(tickangle=-45)

fig.show()
```

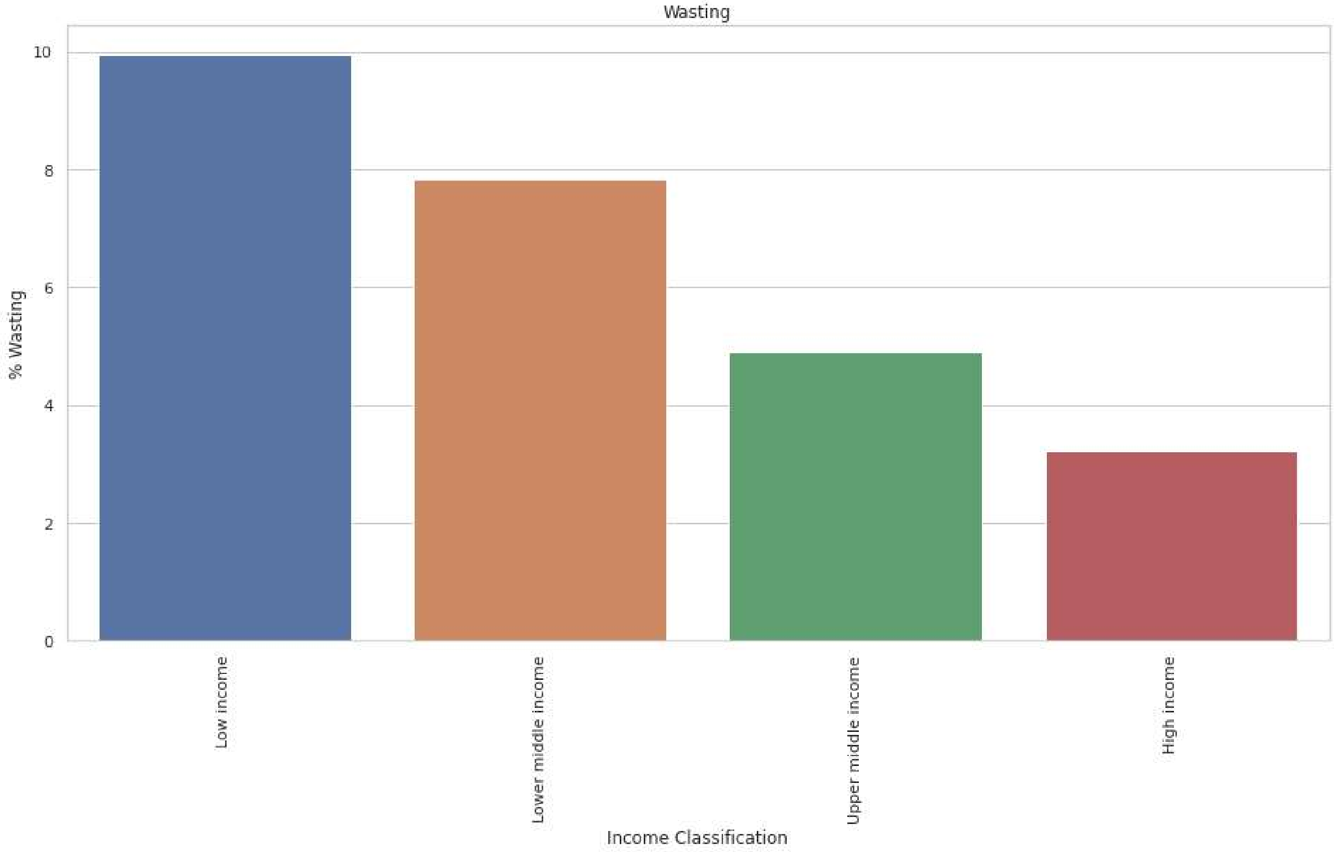


```
x = df.groupby(["Country"])["Wasting"].mean()
layout = dict(geo = {'scope':'world'},title="Wasting percentage")
map = go.Figure(data = [dict(type = 'choropleth',
            locations = x.index,
            locationmode = 'country names',
            colorscale= px.colors.sequential.Teal,
            text= x.index,
            z=x,
            colorbar = {'title':'stunting %' })],layout = layout)
map.show()
```

## Relation between Income Level and Wasting

```
plt.figure(figsize=(16, 8))
x = df.groupby(["Income level"])["Wasting"].mean().sort_values(ascending=False)
sns.set(style="whitegrid")
ax = sns.barplot(x.index, x)
ax.set_title('Wasting')
ax.set_ylabel('% Wasting')
ax.set_xlabel('Income Classification')
plt.xticks(rotation = 90)
```

```
(array([0, 1, 2, 3]), <a list of 4 Text major ticklabel objects>)
```

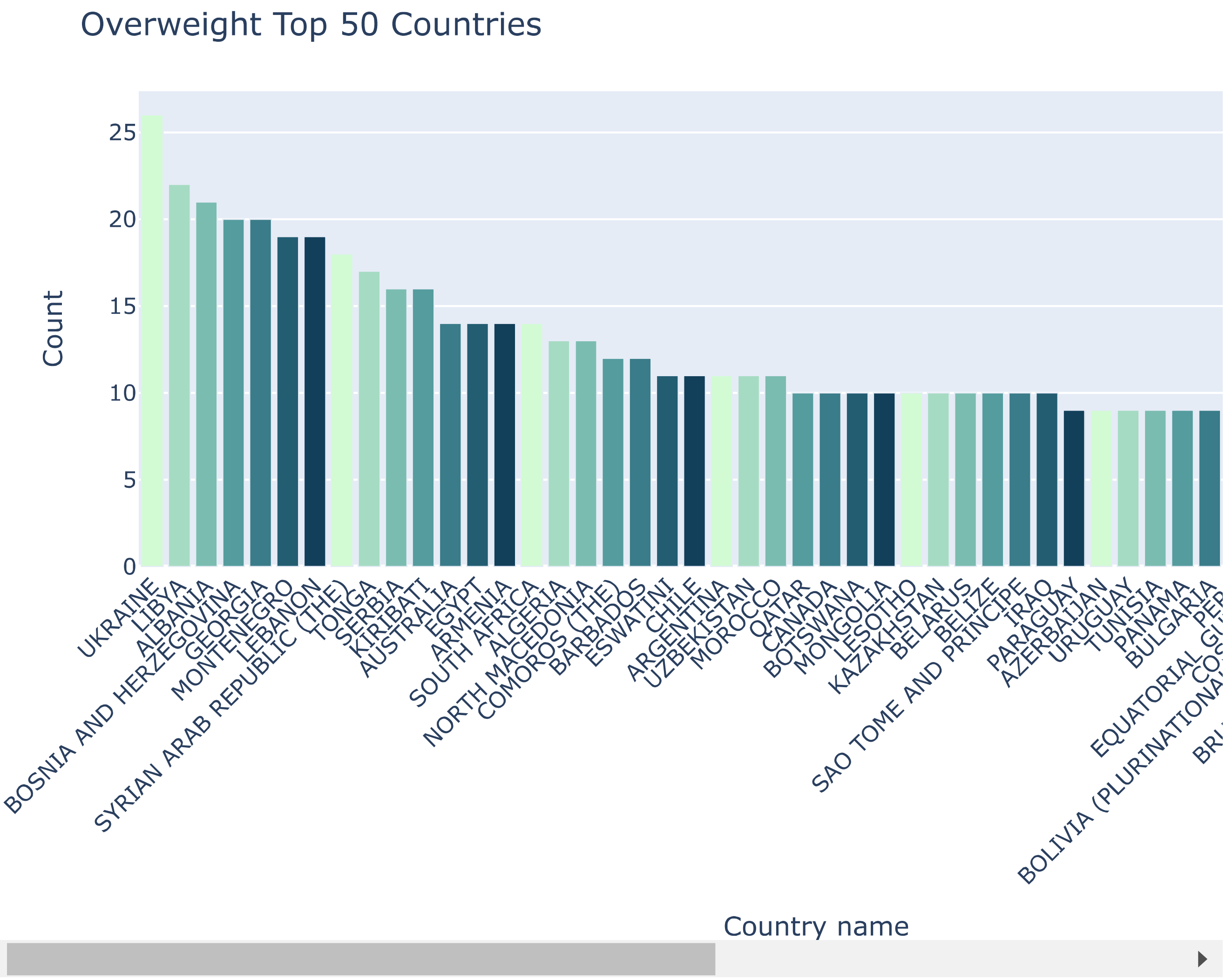


Double-click (or enter) to edit

Overweight

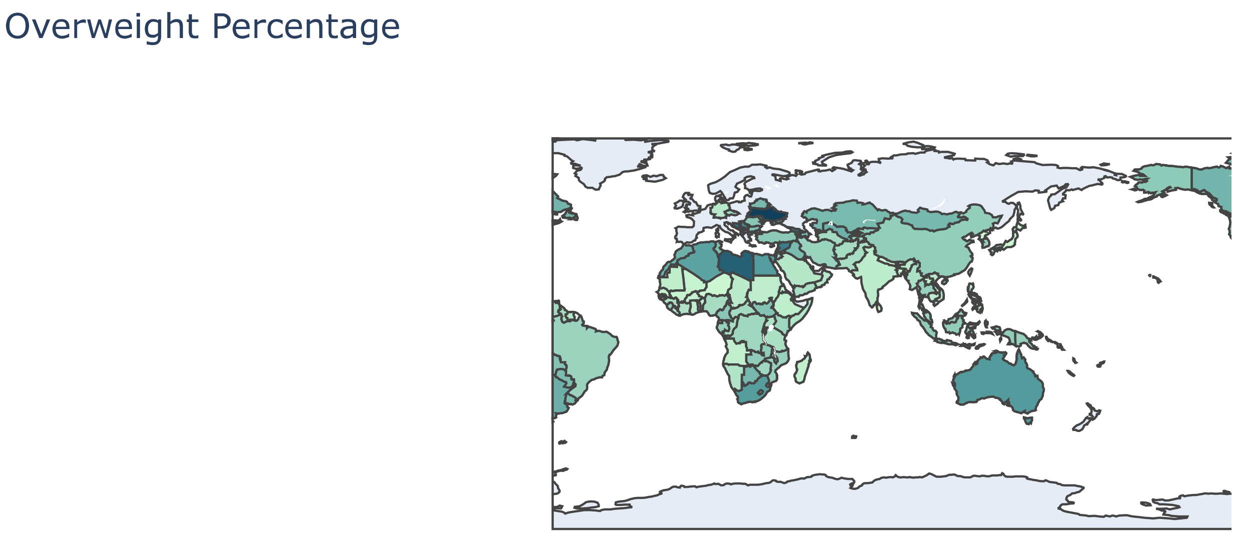
Overweight – Moderate and severe: % aged 0-59 months who are above two standard deviations from median weight-for-height

```
name3 = df.groupby("Country")["Overweight"].mean().sort_values(ascending=False).index[:50]
yax3 = df.groupby("Country")["Overweight"].mean().sort_values(ascending=False).round()[:50]

fig = px.bar(df, y=yax3, x=name3, color = name3, color_discrete_sequence=px.colors.sequential
fig.update_layout(
    title="Overweight Top 50 Countries",
    xaxis_title="Country name",
    yaxis_title="Count"
)
fig.update_xaxes(tickangle=-45)

fig.show()
```



Double-click (or enter) to edit

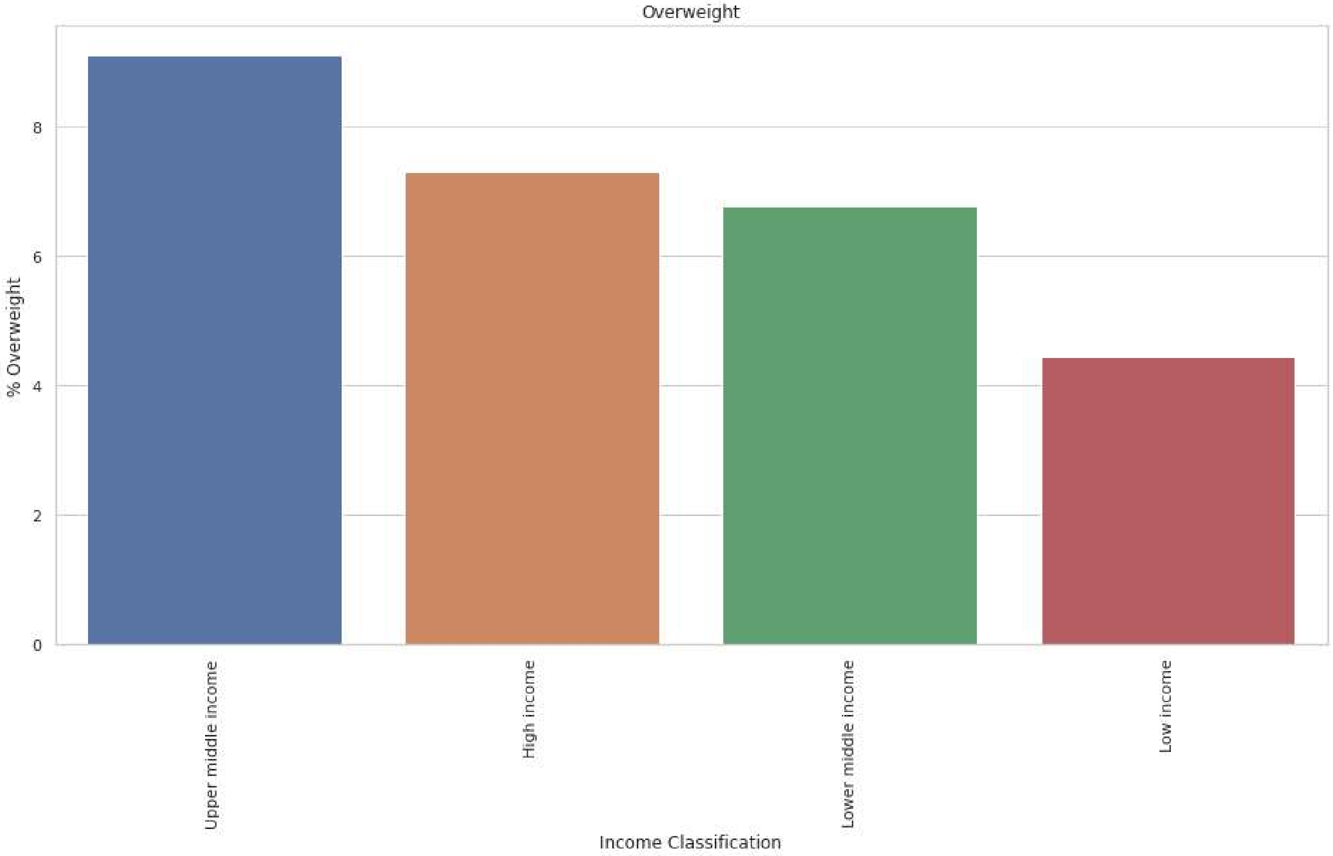
```
x = df.groupby(["Country"])["Overweight"].mean()
layout = dict(geo = {'scope':'world'},title="Overweight Percentage")
map = go.Figure(data = [dict(type = 'choropleth',
            locations = x.index,
            locationmode = 'country names',
            colorscale= px.colors.sequential.Darkmint,
            text= x.index,
            z=x,
            colorbar = {'title':'stunting %' })],layout = layout)
map.show()
```



## Relation between Income Level and Overweight

```
plt.figure(figsize=(16, 8))
x = df.groupby(["Income level"])["Overweight"].mean().sort_values(ascending=False)
sns.set(style="whitegrid")
ax = sns.barplot(x.index, x)
ax.set_title('Overweight')
ax.set_ylabel('% Overweight')
ax.set_xlabel('Income Classification')
plt.xticks(rotation = 90)
```

```
(array([0, 1, 2, 3]), <a list of 4 Text major ticklabel objects>)
```

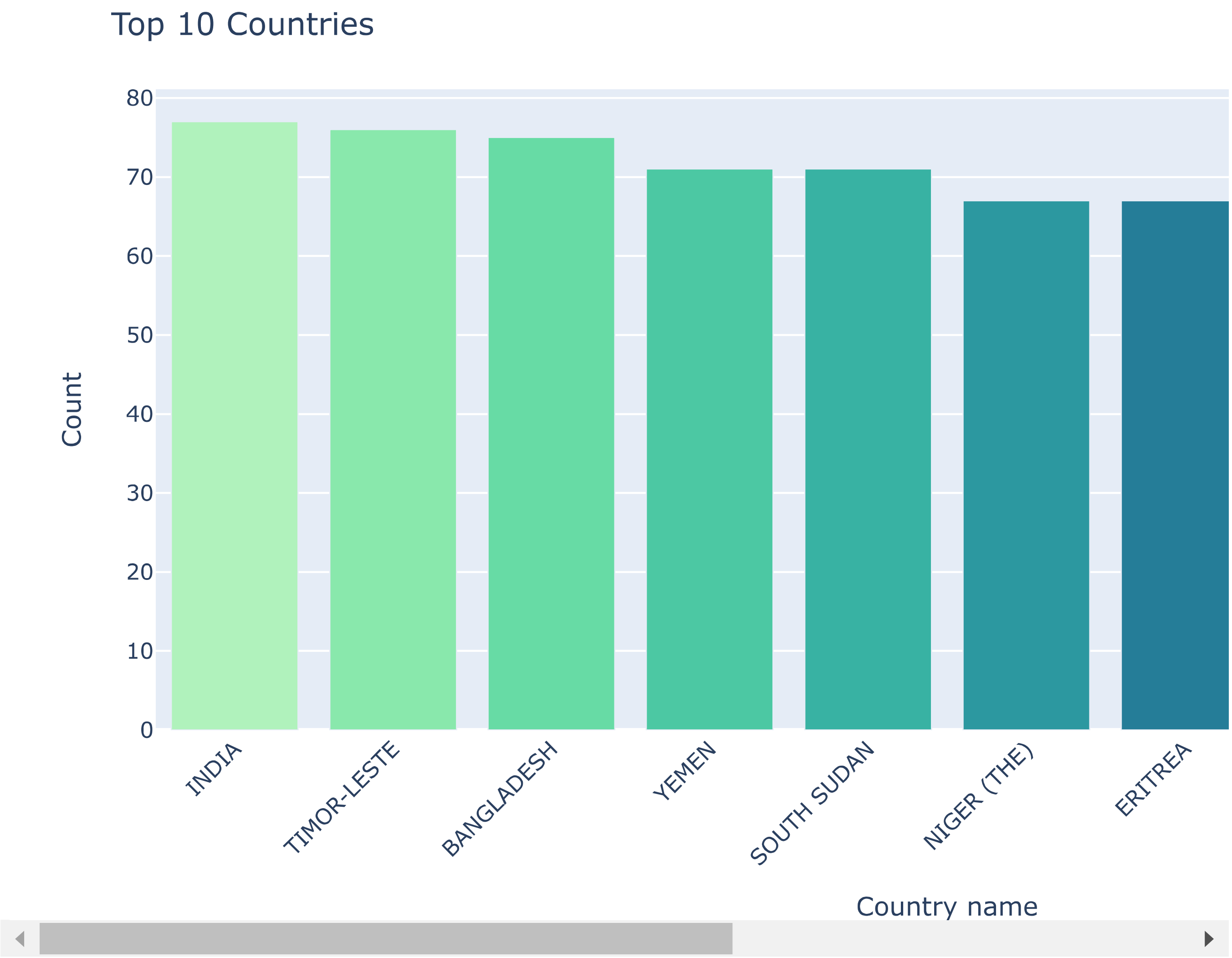


## Top 10 Country with Childhood Malnutrition

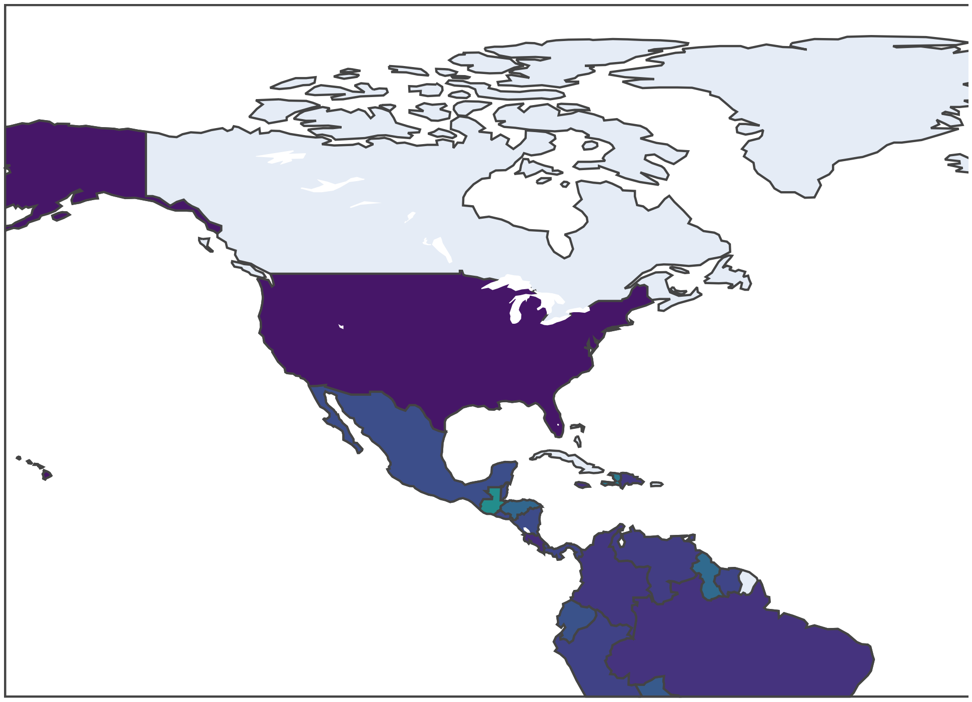
Lastly we will see which 10 countries which has highest average of four symptoms

```
df['Average'] = df.Overweight + df.Underweight + df.Wasting + df.Stunting / 4
```

```
name3 = df.groupby("Country")["Average"].mean().sort_values(ascending=False).index[:10]
yax3 = df.groupby("Country")["Average"].mean().sort_values(ascending=False).round()[:10]

fig = px.bar(df, y=yax3, x=name3, color = name3, color_discrete_sequence=px.colors.sequential
fig.update_layout(
    title="Top 10 Countries",
    xaxis_title="Country name",
    yaxis_title="Count"
)
fig.update_xaxes(tickangle=-45)

fig.show()
```



```
x = df.groupby(["Country"])["Average"].mean()
layout = dict(geo = {'scope':'world'},title="Overall Percentage")
map = go.Figure(data = [dict(type = 'choropleth',
            locations = x.index,
            locationmode = 'country names',
            colorscale= px.colors.sequential.Viridis,
            text= x.index,

            z=x,
            colorbar = {'title':'stunting %' })],layout = layout)
map.show()
```



Overall Percentage

Conclusion

India is having highest Childhood malnutrition after which comes Timor Leste (right to Indonesia in the map).

While we get access to everything, there are many children who cannot even afford to eat one meal a day so let us pray for them and help them maximum as we can 🙏🏻🙏🏻

✓ 0s completed at 2:25 AM ● ✕