SDG INDICATORS CALCULATIONS - Zambia Example

INTRODUCTION

This R markdown template will take through computation of some SDG 3, 5 and 8 key indicators namely;

- SDG Indicator 3.7.2: Adolescent birth rate (aged 10–14 years; aged 15–19 years) per 1,000 women in that age group.
- SDG INDICATOR 5.3.1: Proportion of women aged 20–24 years who were married or in a union before age 15 and before age 18.
- SDG INDICATOR 8.6.1: Proportion of youth (aged 15-24 years) not in education, employment or training.

LOAD LIBRARIES

```
if(!require("pacman")) install.packages("pacman")

pacman::p_load(tidyverse, haven, forcats, gt, sf, ggrepel, patchwork)
```

LOAD DATA

This represent 10% from the Zambia 2010 population and houses census data whose sole purpose is for this exercise.

```
demographics <- haven::read_sav("input/DataExercise/SPSS files/DemographicsDIST.sav")

constituency_shape <- sf::read_sf("input/TUC/zambia_constituency_reproj_pop_GHS-DU-TUC.shp")
```

DATA DESCRIPTION & EXPLORATION

Description

There are two main datasets used in this exercise.

- **demographics** : 10% of Zambia 2010 individual census data with 1,321,973 records and 92 variables. This will be used to compute the SDG indicators using variables such as age, sex, activity in the past 12 months, child bird in the past 12 months and constituency.
- **constituency_shape** : Zambia constituency boundaries (admin 4) with 150 records and 34 variables including information on the DEGURBA classification. This will be used to map the SDG indicators computed from the demographic data and then later to disaggregate the indicators my DEGURBA.

Exploration

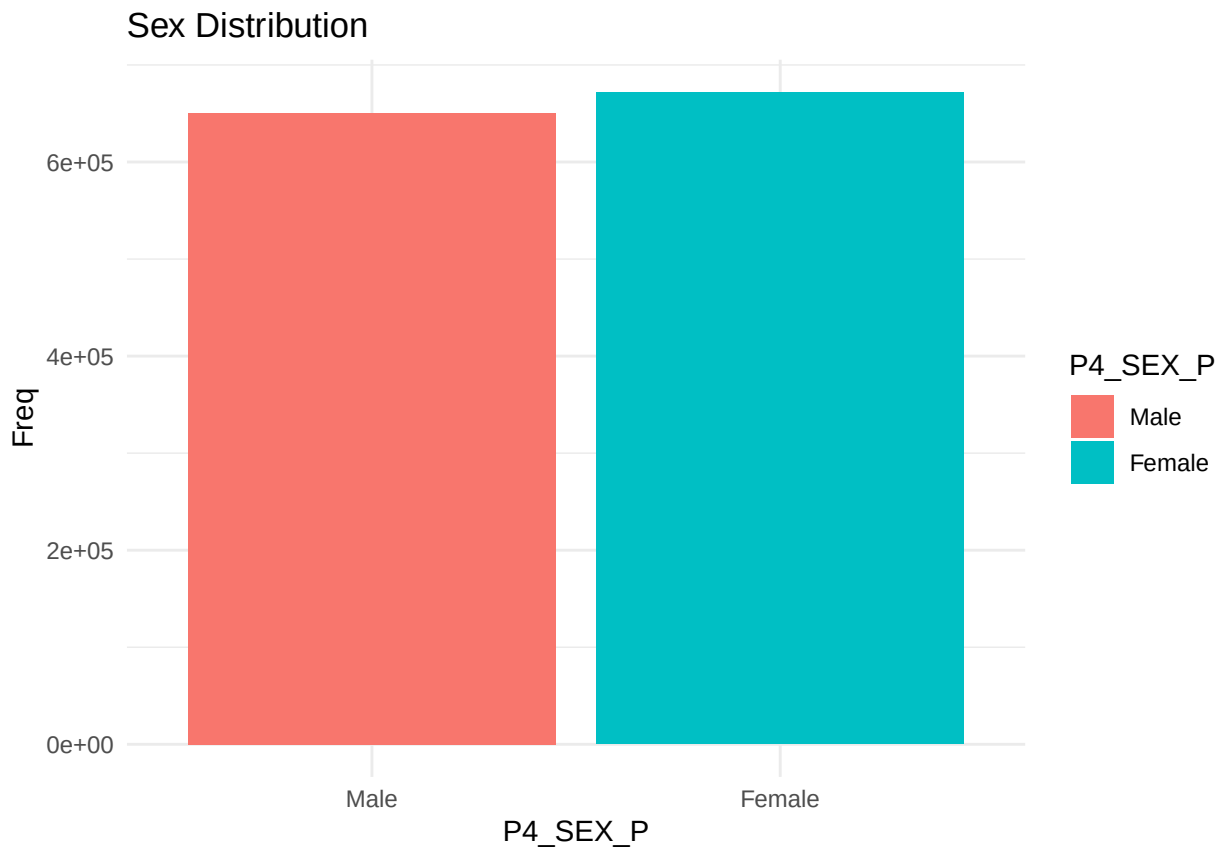
In this section, we are exploring the key variables that will be used in this exercise. These variables are

- *P4_SEX_P = Sex*

```
# Sex unique values
demographics$P4_SEX_P |> str()
```

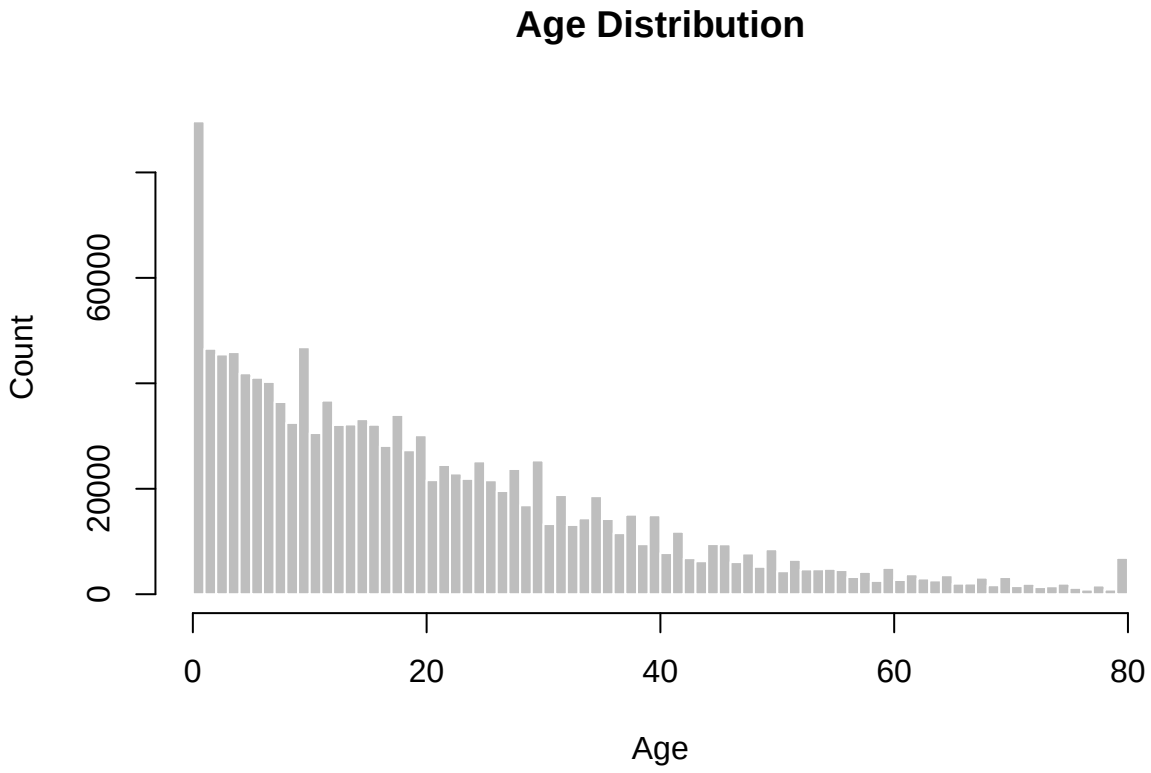
```
##  dbl+lbl [1:1321973] 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 1, 2, 1, 1, 1, ...
##  @ label        : chr "Sex"
##  @ format.spss  : chr "F1.0"
##  @ display_width: int 10
##  @ labels       : Named num [1:2] 1 2
##   ..- attr(*, "names")= chr [1:2] "Male" "Female"
```

```
# Sex distribution
demographics |>
  mutate(P4_SEX_P = as_factor(P4_SEX_P)) |>
  dplyr::group_by(P4_SEX_P) |>
  count(name = "Freq") |>
  tibble::tibble() |>
  ggplot2::ggplot() +
  ggplot2::geom_bar(aes(x = P4_SEX_P, fill = P4_SEX_P, y = Freq),
                    stat = "identity") +
  ggplot2::labs(title = "Sex Distribution") +
  ggplot2::theme_minimal()
```



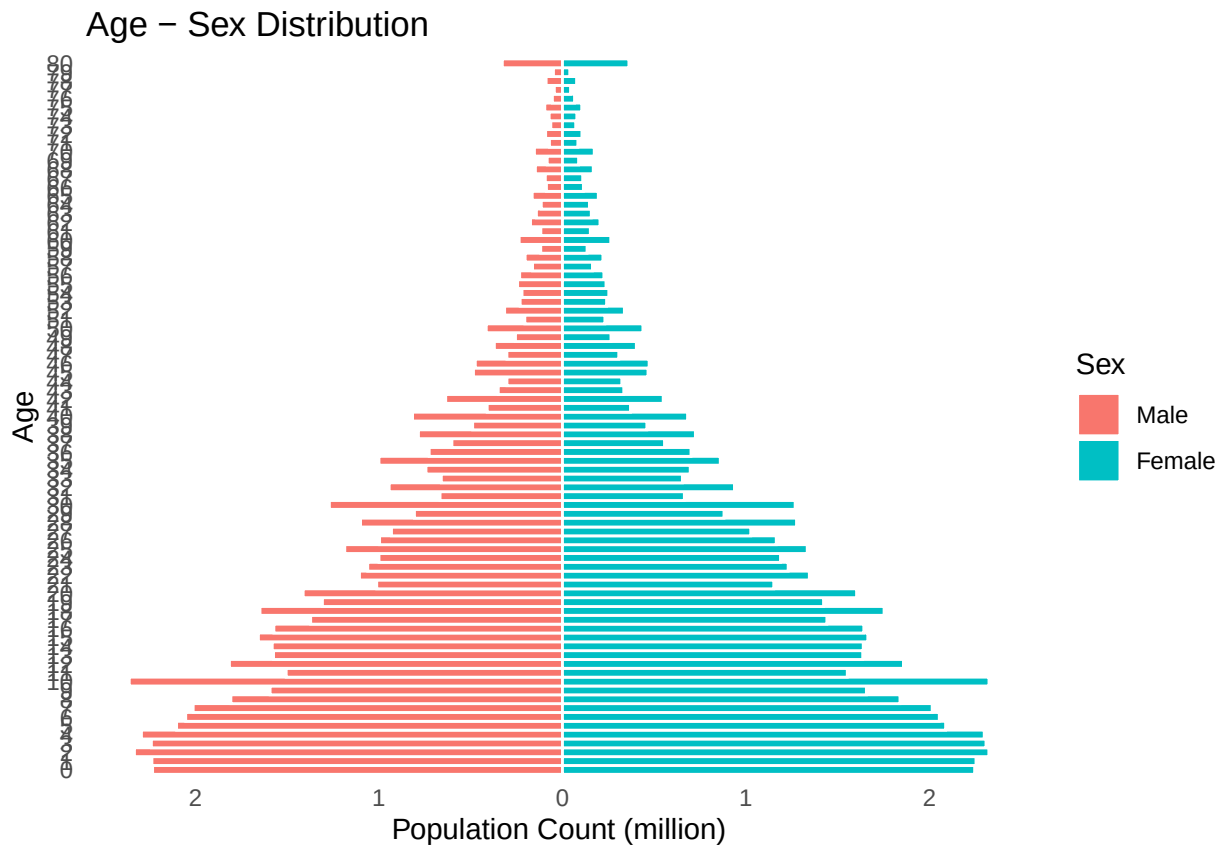
- $P5\_AGE\_P = Age$

```
# Age distribution
demographics$P5_AGE_P |> hist(col = "grey",
                              main = "Age Distribution",
                              xlab = "Age",
                              ylab = "Count",
                              border = "white",
                              breaks = length(unique(demographics$P5_AGE_P)))
```




```
# Age distribution by Sex
demographics |>
  group_by(P4_SEX_P, P5_AGE_P) |>
  summarise(pop_count = n()/10000) |>
  mutate(
    pop_count = case_when(
      P4_SEX_P == 1 ~ -pop_count,
      P4_SEX_P == 2 ~ pop_count
    ),
    sex_label = factor(P4_SEX_P, levels = c(1, 2), labels = c("Male", "Female")),
    age_factor = factor(P5_AGE_P, levels = sort(unique(P5_AGE_P)))
  ) |>
  ggplot2::ggplot(aes(x = pop_count, y = age_factor, fill = sex_label)) +
  ggplot2::geom_bar(stat = "identity", width = 1, , color = "white") +
  ggplot2::scale_x_continuous(labels = abs) +
```

```
  ggplot2::labs(title = "Age - Sex Distribution",
                x = "Population Count (million)",
                y = "Age",
                fill = "Sex") +
  ggplot2::theme_minimal() +
  ggplot2::theme(
    panel.grid = element_blank()
  )
```



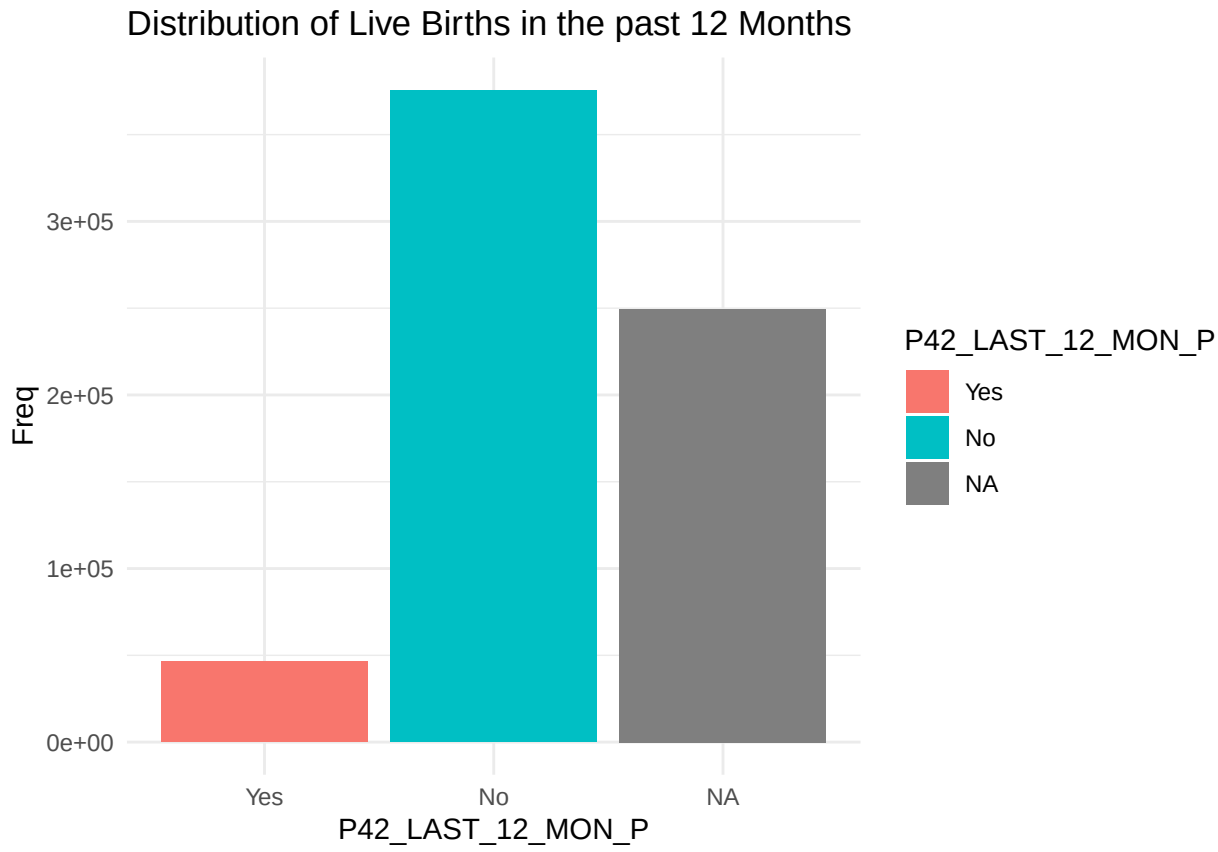
- *P42_LAST_12_MON_P = Live births in the past 12 months*

```
# Unique values
demographics$P42_LAST_12_MON_P |> str()
```

```
##  dbl+lbl [1:1321973]  2,  2,  2,  2, NA,  2,  2, NA,  2,  2,  2,  2, NA, NA...
##  @ label        : chr "Any live births in last 12 months?"
##  @ format.spss  : chr "F1.0"
##  @ display_width: int 19
##  @ labels       : Named num [1:2] 1 2
##   ..- attr(*, "names")= chr [1:2] "Yes" "No"
```

```
# distribution of live births in females
demographics |>
  dplyr::filter(P4_SEX_P == 2) |>
```

```
  mutate(P42_LAST_12_MON_P = as_factor(P42_LAST_12_MON_P)) |>
  dplyr::group_by(P42_LAST_12_MON_P) |>
  count(name = "Freq") |>
  tibble::tibble() |>
  ggplot2::ggplot() +
  ggplot2::geom_bar(aes(x = P42_LAST_12_MON_P , fill = P42_LAST_12_MON_P, y = Freq),
                    stat = "identity") +
  ggplot2::labs(title = "Distribution of Live Births in the past 12 Months") +
  ggplot2::theme_minimal()
```



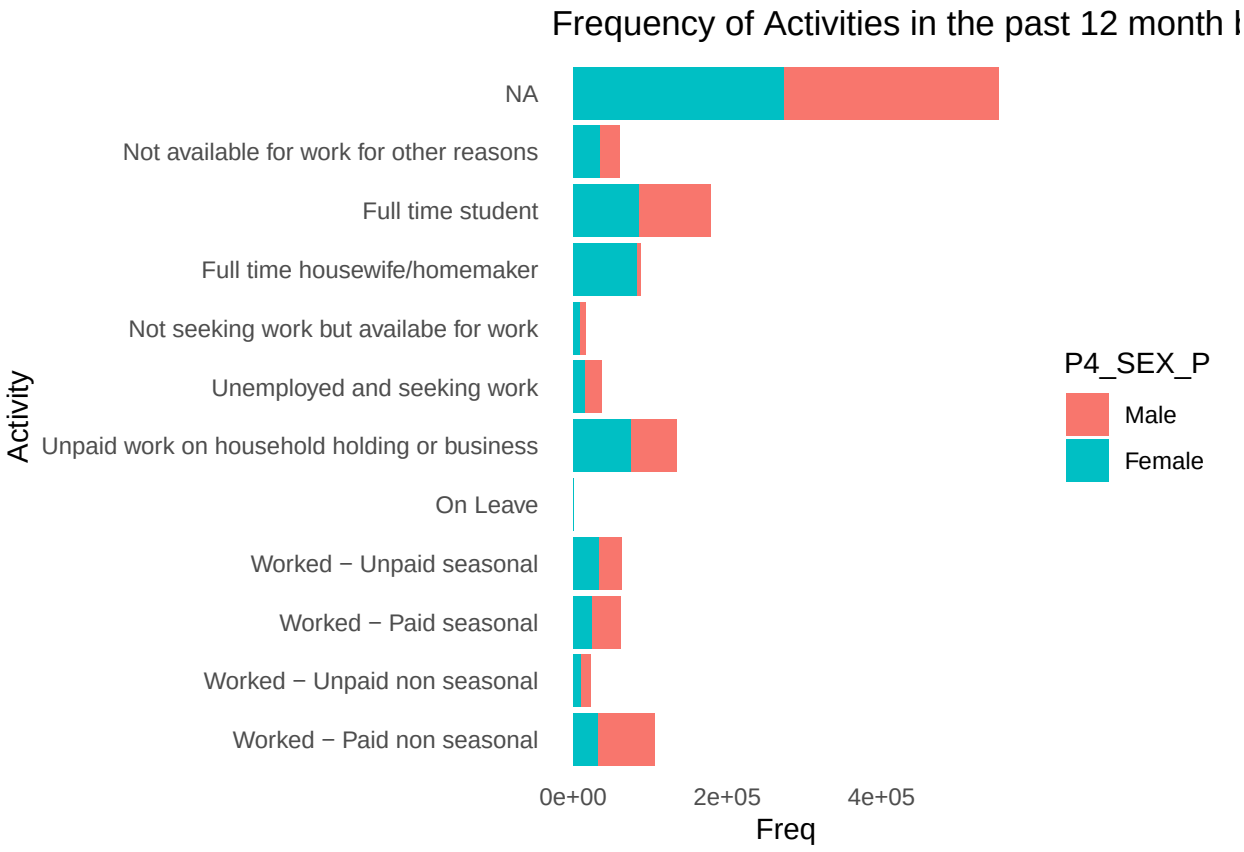
- *P32_ACTIVITY_LAST_12_MONTHS_P = Activity in the last 12 months*

```
# Unique Activities
demographics$P32_ACTIVITY_LAST_12_MONTHS_P |> unique()
```

```
## <labelled<double>[12]>: Activity last twelve months
##  [1] NA  7 10  2  8  9  4  6 11  3  1  5
##
## Labels:
##  value                                    label
##      1                 Worked - Paid non seasonal
##      2               Worked - Unpaid non seasonal
##      3                     Worked - Paid seasonal
##      4                   Worked - Unpaid seasonal
```

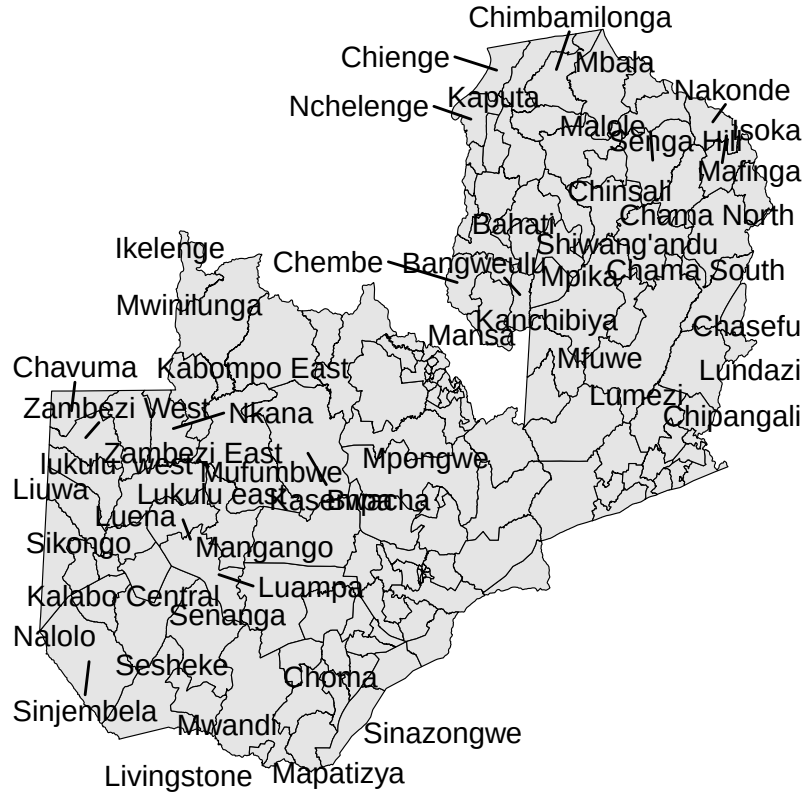
```
##      5                                      On Leave
##      6 Unpaid work on household holding or business
##      7                   Unemployed and seeking work
##      8       Not seeking work but availabe for work
##      9                 Full time housewife/homemaker
##     10                              Full time student
##     11     Not available for work for other reasons
```

```r
# activity distribution
demographics |>
  mutate(P32_ACTIVITY_LAST_12_MONTHS_P = as_factor(P32_ACTIVITY_LAST_12_MONTHS_P),
         P4_SEX_P = as_factor(P4_SEX_P)
         ) |>
  dplyr::group_by(P32_ACTIVITY_LAST_12_MONTHS_P, P4_SEX_P) |>
  count(name = "Freq") |>
  ggplot2::ggplot() +
  ggplot2::geom_bar(aes(y = P32_ACTIVITY_LAST_12_MONTHS_P,
                        fill = P4_SEX_P,
                        x = Freq),
                    stat = "identity") +
  ggplot2::theme_minimal() +
  ggplot2::theme(
    panel.grid = element_blank()
  ) +
  ggplot2::labs(
    title = "Frequency of Activities in the past 12 month by Sex",
    y = "Activity"
  )
```



- *CONST_P = Constituency (Admin 4)*

```
constituency_shape |>
  ggplot2::ggplot() +
  ggplot2::geom_sf(color = "black") +
  ggrepel::geom_text_repel(data = constituency_shape,
                           aes(label = NAME1_, geometry = geometry),
                           stat = "sf_coordinates") +
  ggplot2::labs(title = "Zambia Constituencies 2010") +
  ggplot2::theme_void()
```



SUSTAINABLE DEVELOPMENT GOALS (SDG) CALCULATION

SDG 3 : Good Health and Well Being

SDG TARGET 3.7: By 2030, ensure universal access to sexual and reproductive health-care services, including for family planning, information and education, and the integration of reproductive health into national strategies and programmes.

SDG Indicator 3.7.2: Adolescent birth rate (aged 10–14 years; aged 15–19 years) per 1,000 women in that age group adolescent_birth_rate_10_14 = (number of women aged 10 - 14 with live birth in the past 12 months / number of women aged 10 - 14)*1000

adolescent_birth_rate_14_19 = (number of women aged 14 - 19 with live birth in the past 12 months / number of women aged 14 - 19)*1000

Input Variables Definition:

- *P4_SEX_P = SexP4_SEX_P*
- *P5_AGE_P = Age*
- *P42_LAST_12_MON_P = Live births in the past 12 months*
- *CONST_P = Constituency (Admin 4)*

Output Variables Definition:

- *total.ado.10_14 = Total female adolescents aged 10 to 14*
- *total.ado.15_19 = Total female adolescents aged 15 to 19*
- *total.ado.birth.10_14 = Total female adolescents aged 10 to 14 with live birth in the past 12 months*
- *total.ado.birth.15_19 = Total female adolescents aged 15 to 19 with live birth in the past 12 months*
- *abr.10_14 = Adolescent aged 10 to 14 birth rate*
- *abr.15_19 = Adolescent aged 15 to 19 birth rate*

Methodology:

1- Filter the Sex to Female by using the P4_SEX_P == 2

2- Group by the constituency

3- Compute summary statistics

Adolescent Birth Rate by Constituency

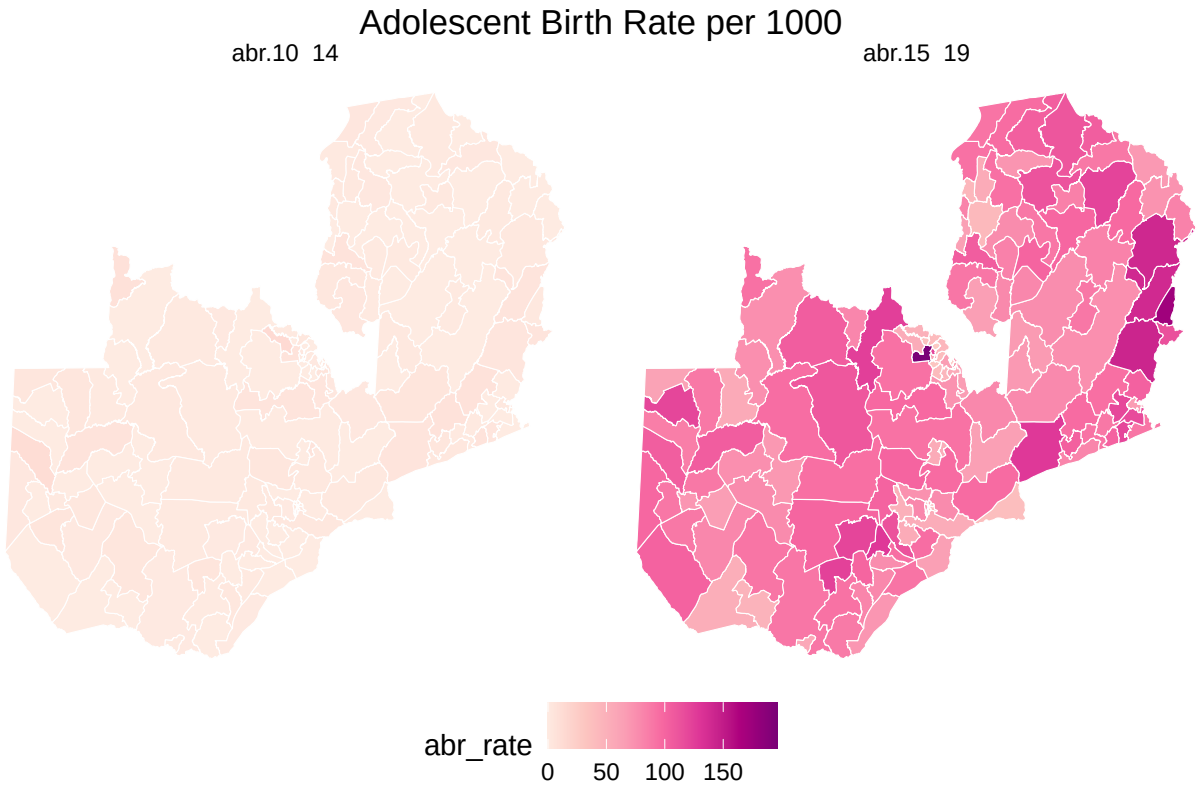
```
adolescent_birth_rate <-
  demographics |>
  dplyr::filter(P4_SEX_P == 2) |>
  dplyr::group_by(CONST_P) |>
  dplyr::summarise(
    total.ado.10_14 = sum(dplyr::between(P5_AGE_P, 10, 14), na.rm = TRUE),
    total.ado.15_19 = sum(dplyr::between(P5_AGE_P, 15, 19), na.rm = TRUE),
    total.ado.birth.10_14 = sum(dplyr::between(P5_AGE_P, 10, 14) &
                                  P42_LAST_12_MON_P == 1, na.rm = TRUE),
    total.ado.birth.15_19 = sum(dplyr::between(P5_AGE_P, 15, 19) &
                                  P42_LAST_12_MON_P == 1, na.rm = TRUE),
    abr.10_14 = round((total.ado.birth.10_14 / total.ado.10_14) * 1000,2),
    abr.15_19 = round((total.ado.birth.15_19 / total.ado.15_19) * 1000,2)
  ) |>
  dplyr::mutate(
    CONST_P_name = forcats::as_factor(CONST_P)
  ) |>
  dplyr::select(
    CONST_P_name,
    everything()
  )

# Print table
adolescent_birth_rate |>
  head() |>
  gt::gt()
```

Visualization

```
constituency_shape |>
  dplyr::select(NAME1_, geometry) |>
  dplyr::left_join(
    adolescent_birth_rate,
    by = c("NAME1_" = "CONST_P_name")
  ) |>
```

Constituency	Constituency	total.ado.10_14	total.ado.15_19	total.ado.birth.10_14	total.ado.birt
Chisamba	1	753	622	2	
Katuba	2	613	523	0	
Keembe	3	883	740	4	
Bwacha	4	588	559	0	
Kabwe Central	5	881	950	6	
Kapiri Mposhi	6	1828	1507	2	

```
  tidyr::pivot_longer(
    cols = c("abr.10_14", "abr.15_19"),
    names_to = "abr_cat",
    values_to = "abr_rate"
  ) |>
  ggplot2::ggplot() +
  ggplot2::geom_sf(aes(fill = abr_rate), color = "white") +
  ggplot2::scale_fill_distiller(palette = "RdPu", direction = 1) +
  ggplot2::theme_void() +
  ggplot2::labs(title = "Adolescent Birth Rate per 1000") +
  ggplot2::theme(
    legend.position = "bottom",
    plot.title = element_text(hjust = 0.5)
  ) +
  ggplot2::facet_grid(~abr_cat)
```



SDG 5 : GENDER EQUALITY

SDG TARGET 5.3 : Eliminate all harmful practices, such as child, early and forced marriage and female genital mutilations

SDG INDICATOR 5.3.1: Proportion of women aged 20–24 years who were married or in a union before age 15 and before age 18 child_marriage_before_15 = (number of women aged 20-24 who were married before age 15 / total number of women aged 20-24)* 100

child_marriage_before_18 = (number of women aged 20-24 who were married before age 18 / total number of women aged 20-24)* 100

Input Variables Definition:

- *P4_SEX_P = Sex (1 - Male, 2 - Female)*
- *P5_AGE_P = Age*
- *CONST_P = Constituency*

Output Variable Definition:

- *total.girls.20_24 = Total number of females aged 20 to 24*
- *married.before.15 = Total number of females married before 15*

Constituency	Constituency	total.girls.20_24	married.before.15	married.before.18	prop.cm.before
Chisamba	1	459	9	112	0
Katuba	2	376	13	101	0
Keembe	3	567	29	176	0
Bwacha	4	414	15	91	0
Kabwe Central	5	716	8	85	0
Kapiri Mposhi	6	1153	57	344	0

- *married.before.18 = Total number of females married before 18*
- *prop.cm.before.15 = Proportion of child marriages before 15*
- *prop.cm.before.18 = Proportion of child marriages before 18*

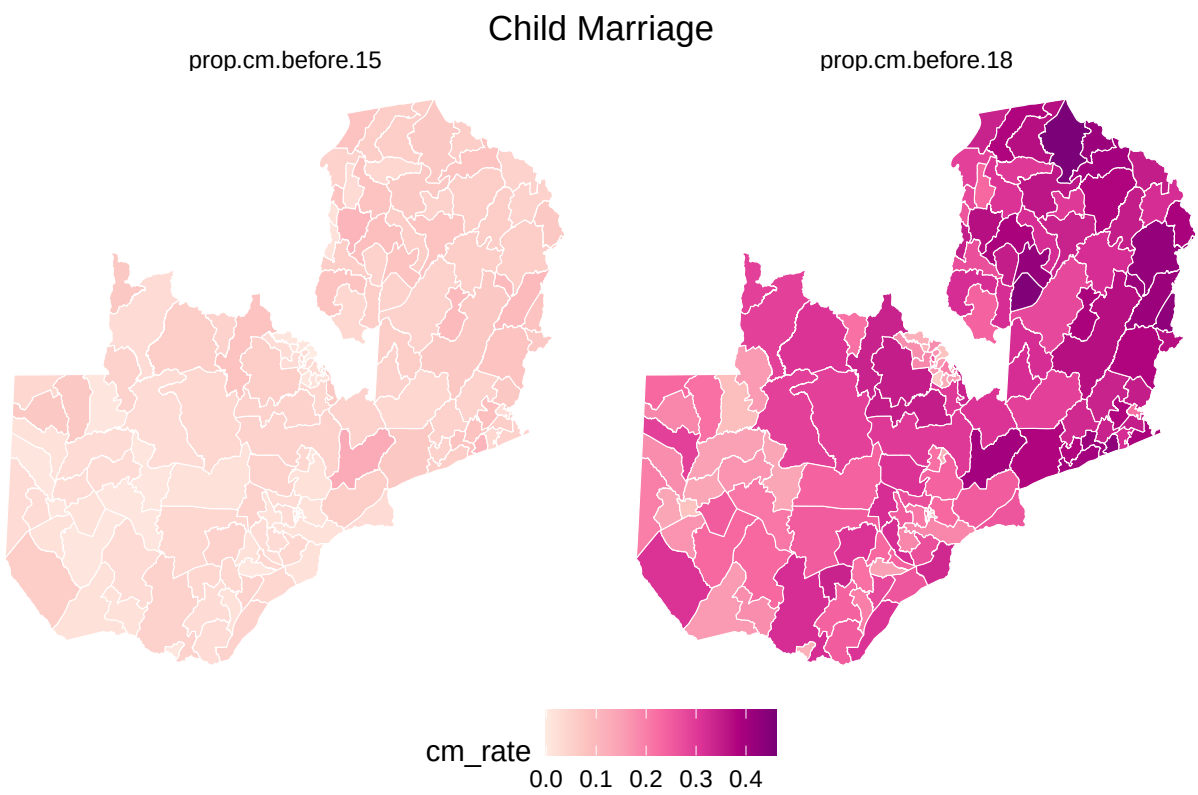
Child Marriage by Constituency (Admin 4 Zambia)

```
child_marriage <-
  demographics |>
  dplyr::filter(P5_AGE_P >= 20, P5_AGE_P <= 24, P4_SEX_P == 2) |>
  dplyr::group_by(CONST_P) |>
  dplyr::summarise(
    total.girls.20_24 = n(),
    married.before.15 = sum(P37_AGE_FIRST_MARRAIGE_P < 15, na.rm = T),
    married.before.18 = sum(P37_AGE_FIRST_MARRAIGE_P < 18, na.rm = T),
    prop.cm.before.15 = round(married.before.15 / total.girls.20_24, 2),
    prop.cm.before.18 = round(married.before.18 / total.girls.20_24, 2)
  ) |>
  dplyr::mutate(
    CONST_P_name = forcats::as_factor(CONST_P)
  ) |>
  dplyr::select(
    CONST_P_name,
    everything()
  )

# print table
child_marriage |>
  head() |>
  gt::gt()
```

Visualization

```
constituency_shape |>
  dplyr::select(NAME1_, geometry) |>
  dplyr::left_join(
    child_marriage,
    by = c("NAME1_" = "CONST_P_name")
  ) |>
  tidyr::pivot_longer(
    cols = c("prop.cm.before.15", "prop.cm.before.18"),
    names_to = "child_marriage",
```

```
  values_to = "cm_rate"
) |>
ggplot2::ggplot() +
ggplot2::geom_sf(aes(fill = cm_rate), color = "white") +
ggplot2::scale_fill_distiller(palette = "RdPu", direction = 1) +
ggplot2::theme_void() +
ggplot2::labs(title = "Child Marriage") +
ggplot2::theme(
  legend.position = "bottom",
  plot.title = element_text(hjust = 0.5)
) +
ggplot2::facet_grid(~child_marriage)
```



SDG 8 : DECENT WORK AND ECONOMIC GROWTH

SDG TARGET 8.6 : By 2020, substantially reduce the proportion of youth not in employment, education or training

SDG INDICATOR 8.6.1: Proportion of youth (aged 15-24 years) not in education, employment or training NEET rate (%) = (Youth – Youth in employment – Youth not in employment but in education or training) / Youth *100

Input Variables Definition :

- *P32_ACTIVITY_LAST_12_MONTHS_P = Activity in the last 12 months*

- *P5_AGE_P = Age*
- *CONST_P = Constituency (Admin 4)*

Output Variables Definition :

- *total.youth.15_24 = Total number of people aged 15 to 24 (Youth)*
- *total.neet = Number of youths not employed, not in education or training*
- *rate.need = proportion of youths not employed, not in education or training*

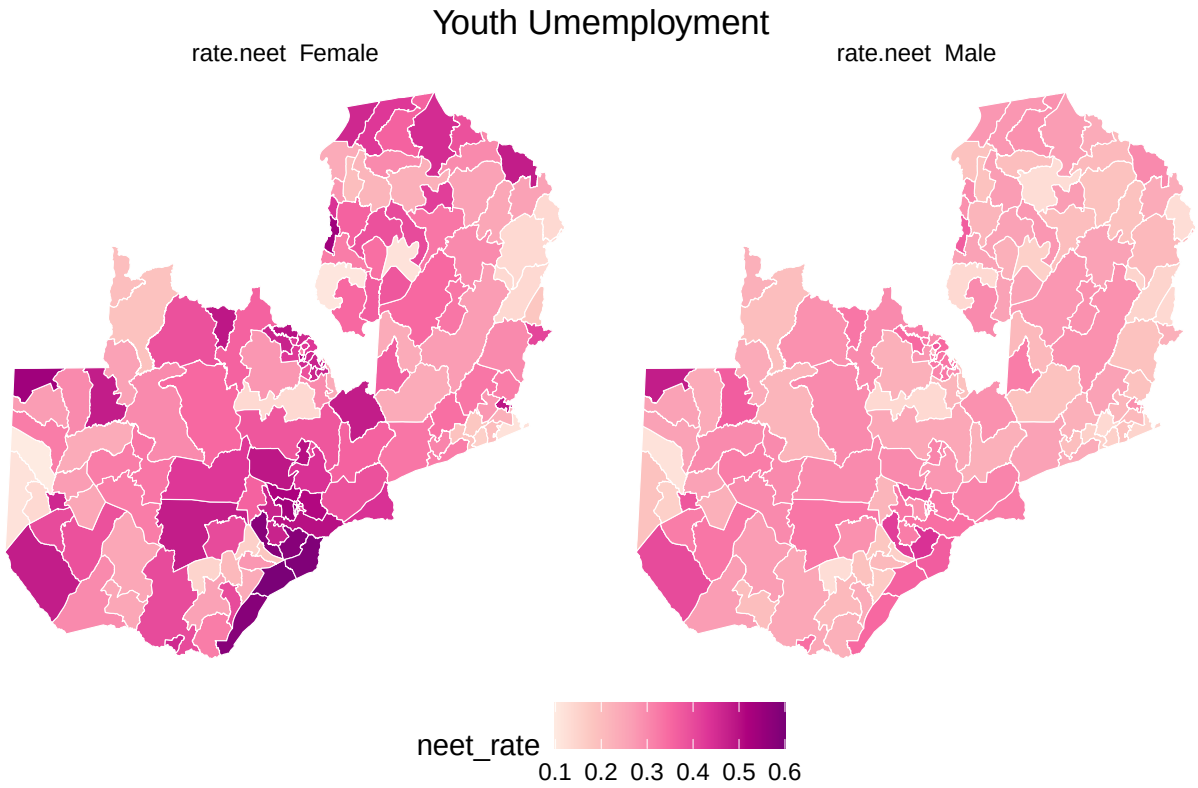
Youth Unemployment by Constituency (Admin 4 Zambia) & Sex

```
youth_umemplyment <-
  demographics |>
  dplyr::mutate(
    Employ = case_when(
      P32_ACTIVITY_LAST_12_MONTHS_P %in% 1:6 ~ 1,
      P32_ACTIVITY_LAST_12_MONTHS_P %in% 7:9 ~ 2,
      P32_ACTIVITY_LAST_12_MONTHS_P == 10 ~ 3,
      T ~ 2
    )
  ) |>
  dplyr::filter(P5_AGE_P >= 15, P5_AGE_P <= 24) |>
  dplyr::group_by(CONST_P, P4_SEX_P) |>
  summarise(
    total.youth.15_24 = n(),
    total.neet = sum(Employ == 2, na.rm = TRUE),
    rate.neet = round(total.neet / total.youth.15_24, 2)
  ) |>
  dplyr::mutate(
    CONST_P_name = forcats::as_factor(CONST_P),
    P4_SEX_P = forcats::as_factor(P4_SEX_P)
  ) |>
  dplyr::select(
    CONST_P_name,
    everything()
  ) |>
  tidyr::pivot_wider(
    names_from =  P4_SEX_P,
    values_from = 4:6
  )

# print table
youth_umemplyment |>
  tibble::tibble() |>
  head() |>
  gt::gt()
```

Visualization

CONST_P_name	Constituency	total.youth.15_24_Male	total.youth.15_24_Female	total.neet_M
Chisamba	1	1078	1081	3
Katuba	2	834	899	3
Keembe	3	1227	1307	3
Bwacha	4	903	973	2
Kabwe Central	5	1444	1666	4
Kapiri Mposhi	6	2571	2660	6

```
constituency_shape |>
  dplyr::select(NAME1_, geometry) |>
  dplyr::left_join(
    youth_umemplyment,
    by = c("NAME1_" = "CONST_P_name")
  ) |>
  tidyr::pivot_longer(
    cols = c("rate.neet_Male", "rate.neet_Female"),
    names_to = "neet",
    values_to = "neet_rate"
  ) |>
  ggplot2::ggplot() +
  ggplot2::geom_sf(aes(fill = neet_rate), color = "white") +
  ggplot2::scale_fill_distiller(palette = "RdPu", direction = 1) +
  ggplot2::theme_void() +
  ggplot2::labs(title = "Youth Umemployment") +
  ggplot2::theme(
    legend.position = "bottom",
    plot.title = element_text(hjust = 0.5)
  ) +
  ggplot2::facet_grid(~neet)
```



EXPORT RESULTS SDG RESULTS TO CSV

1- Merge all results into one data frame

```
indicators_df <-
  child_marriage |>
  dplyr::left_join(
    adolescent_birth_rate,
    by = c("CONST_P_name", "CONST_P")
  ) |>
  dplyr::left_join(
    youth_umemplyment,
    by = c("CONST_P_name", "CONST_P")
  )
# print table
indicators_df |>
  head() |>
  gt::gt()
```

2- Export the results into a CSV file for further analysis

```
readr::write_csv(x = indicators_df, file = "output/indicators.csv")
```

CONST_P_name	Constituency	total.girls.20_24	married.before.15	married.before.18	prop.cm.be
Chisamba	1	459	9	112	
Katuba	2	376	13	101	
Keembe	3	567	29	176	
Bwacha	4	414	15	91	
Kabwe Central	5	716	8	85	
Kapiri Mposhi	6	1153	57	344	

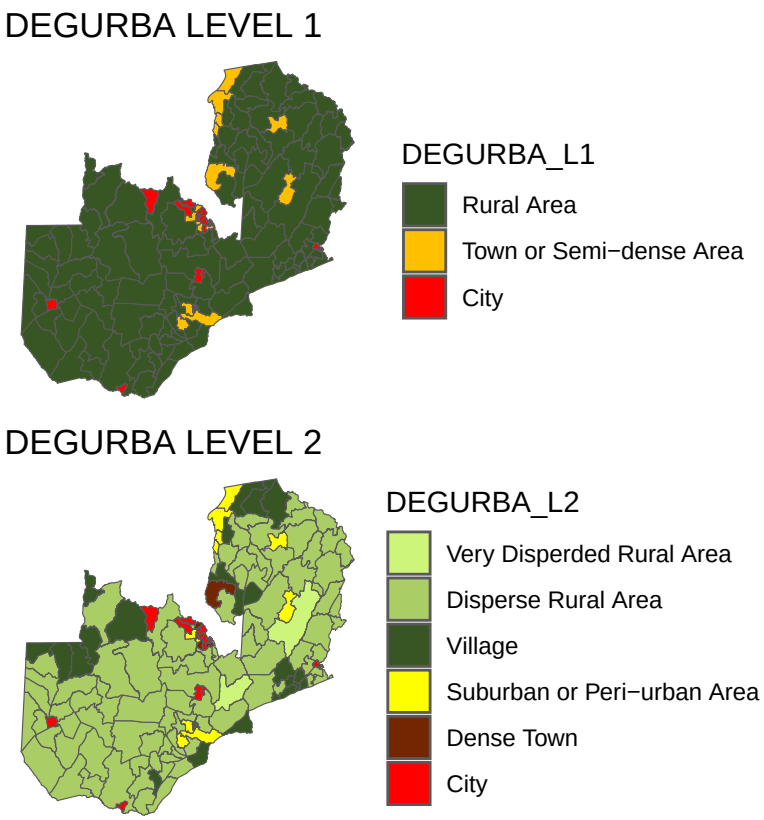
OPTIONAL: APPLICATION OF DEGURBA TO SDG INDICATORS IN R

In this section, we are Going to classify our admin data by DEGURBA classification generated from the application of the DEGURBA methodology.

Step 0: Classification of Constituencies by DEGURBA

```
# degurba level 1
degurba_l1 <- constituency_shape |>
  dplyr::mutate(
    DEGURBA_L1 = factor(DEGURBA_L1,
                        levels = c(1,2,3),
                        labels = c("Rural Area",
                                   "Town or Semi-dense Area",
                                   "City"
                                   )
                        )
    ) |>
  ggplot2::ggplot() +
  ggplot2::geom_sf(aes(fill = DEGURBA_L1)) +
  ggplot2::scale_fill_manual(
    values = c("#375623","#FFC000","red")
    ) +
  ggplot2::theme_void() +
  ggplot2::labs(title = "DEGURBA LEVEL 1")

# degurba level 2
degurba_l2 <- constituency_shape |>
  dplyr::mutate(
    DEGURBA_L2 = factor(DEGURBA_L2,
                        levels = c(11,12,13,21,23,30),
                        labels = c("Very Disperded Rural Area",
                                   "Disperse Rural Area",
                                   "Village",
                                   "Suburban or Peri-urban Area",
                                   "Dense Town",
                                   "City"
                                   )
                        )
    ) |>
  ggplot2::ggplot() +
  ggplot2::geom_sf(aes(fill = DEGURBA_L2)) +
  ggplot2::scale_fill_manual(
    values = c("#cdf57a","#abcd66","#375623","#ffff00","#732600", "red")
```

```
    ) +
  ggplot2::theme_void() +
  ggplot2::labs(title = "DEGURBA LEVEL 2")

# use patchwork to paste both plot side by side

degurba_l1 / degurba_l2
```



Step 1: Join the Constituency Shapes and The SDG Indicators

```
sdg_degurba <-
  constituency_shape |>
  dplyr::select(NAME1_, DEGURBA_L1, DEGURBA_L2) |>
  dplyr::left_join(
    indicators_df,
    by = c("NAME1_" = "CONST_P_name")
  ) |>
  tibble::as.tibble() |>
  dplyr::select(-geometry)

# print table
sdg_degurba |>
  head() |>
  gt::gt()
```

Step 2 : SDG dissagregation by DEGURBA

NAME1_	DEGURBA_L1	DEGURBA_L2	Constituency	total.girls.20_24	married.before.15	r
Itezhi Tezhi	1	12	130	310	13	
Mulobezi	1	12	148	135	4	
Sesheke	1	12	150	213	4	
Sinjembela	1	12	147	454	26	
Katombola	1	12	121	463	21	
Mwandi	1	12	149	128	2	

```
sdg_degurba_diss <-
  sdg_degurba |>
  tidyr::pivot_longer(
    cols = c("DEGURBA_L1", "DEGURBA_L2"),
    names_to = "degurba_level",
    values_to = "degurba_class"
  ) |>
  dplyr::group_by(degurba_level, degurba_class) |>
  dplyr::summarise(
    total.girls.20_24 = sum(total.girls.20_24),
    married.before.15   = sum(married.before.15),
    married.before.18 = sum(married.before.18),
    prop.cm.before.15   = round((married.before.15 / total.girls.20_24), 2),
    prop.cm.before.18 = round((married.before.18 / total.girls.20_24), 2),
    total.ado.10_14 = sum(total.ado.10_14),
    total.ado.15_19 = sum(total.ado.15_19),
    total.ado.birth.10_14   = sum(total.ado.birth.10_14),
    total.ado.birth.15_19   = sum(total.ado.birth.15_19),
    abr.10_14   = round((total.ado.birth.10_14/total.ado.10_14)*1000, 2),
    abr.15_19   = round((total.ado.birth.15_19/total.ado.15_19)*1000, 2),
    total.youth.15_24_Male = sum(total.youth.15_24_Male),
    total.youth.15_24_Female = sum(total.youth.15_24_Female),
    total.neet_Male = sum(total.neet_Male),
    total.neet_Female   = sum(total.neet_Female),
    rate.neet_Male = round((total.neet_Male / total.youth.15_24_Male), 2),
    rate.neet_Female = round((total.youth.15_24_Female / total.neet_Female), 2)
  )


sdg_degurba_diss <-
  sdg_degurba_diss |>
  dplyr::mutate(
    degurba_class = factor(degurba_class,
                           levels = c(1,2,3,11,12,13,21,23,30),
                           labels = c("Rural Area",
                                      "Town or Semi-dense Area",
                                      "City",
                                      "Very Disperded Rural Area",
                                      "Disperse Rural Area",
                                      "Village",
                                      "Suburban or Peri-urban Area",
                                      "Dense Town",
```

```
                                        "city"
                                        )
                            )
    )
```

Step 3: Visualization

```
sdg_degurba_diss |>
  gt::gt() |>
  gt::tab_header(title = "SDG INDICATOR BY DEGURBA",
                 subtitle = "Zambia 2010 Pop & House Census") |>
  gt::data_color(
    columns = c("prop.cm.before.15",
                "prop.cm.before.18"
                ),
    method = "numeric",
    palette = "OrRd",
    domain = c(min(sdg_degurba_diss$prop.cm.before.15),
               max(sdg_degurba_diss$prop.cm.before.18))
  ) |>
  gt::data_color(
    columns = c("abr.10_14",
                "abr.15_19"
                ),
    method = "numeric",
    palette = "OrRd",
    domain = c(min(sdg_degurba_diss$abr.10_14),
               max(sdg_degurba_diss$abr.15_19))
  ) |>
  gt::data_color(
    columns = c("rate.neet_Male",
                "rate.neet_Female"
                ),
    method = "numeric",
    palette = "OrRd",
    domain = c(min(sdg_degurba_diss$rate.neet_Male),
               max(sdg_degurba_diss$rate.neet_Female))
  )
```

[1]

[1] *UNFPA - SDG DISSAGREGATION BY DEGURBA | by Derrick DEMEVENG | Tanzania DEGURBA Workshop - May*

SDG INDICATOR BY DEGURBA

Zambia 2010 Pop & House Census

degurba_class	total.girls.20_24	married.before.15	married.before.18	prop.cm.before.
DEGURBA_L1				
Rural Area	37060	1822	11142	0.
Town or Semi-dense Area	6963	284	1628	0.
City	21146	423	3272	0.
DEGURBA_L2				
Very Disperded Rural Area	265	27	102	0.
Disperse Rural Area	29834	1412	8907	0.
Village	6961	383	2133	0.
Suburban or Peri-urban Area	5663	254	1428	0.
Dense Town	1300	30	200	0.
city	21146	423	3272	0.