1.Gender==

scs$sex <- factor(scs$sex, levels = c(1,0,-1), labels = c("Male", "Female","Unspecify"))

2.AGE

scs$age <- cut(scs$age,

breaks = c(0,5,10,15,20,25,30,35,40,

45,50,55,60,65,70,75,80,85,

90, Inf),

labels = c("0-4","5-9","10-14","15-19","20-24",

"25-29","30-34", "35-39","40-44","45-49",

"50-54","55-59","60-64","65-69","70-74","75-79",

"80-84","85-89","90+"),right = FALSE)

3. Date

as.POSIXct(scs$date, tryFormats = c("%d/%m/%Y %H:%M", "%d/%m/%Y", "%Y-%m-%d"))

4.IF ELSE

scs$survey\_type <- ifelse(scs$postal== "yes", "postal",

ifelse(scs$web == "yes", "web", "unknown"))

5. Colour Panel

Acustom\_colors <- custom\_colors <- c("#1f77b4", "#ff7f0e", "#2ca02c", "#d62728", "#9467bd",

"#8c564b", "#e377c2", "#7f7f7f", "#bcbd22", "#17becf",

"#f0027f", "#666666", "#a6cee3", "#1f78b4", "#b2df8a",

"#33a02c", "#fb9a99", "#e31a1c", "#fdbf6f", "#ff7f00",

"#cab2d6", "#6a3d9a", "#ffff99", "#b15928", "#008080")

6. bar plot for univariate distributions

# barplot of sex distribution

Gender <- ggplot(scs,aes(x=sex) +

geom\_bar(fill= "steelblue", width = 0.5)+

labs(title = "Sex Distribution", x= "sex", y="count"))

7. aggregated\_data <- merged\_data %>% group\_by(Age\_Group, Year\_of\_diagnosis) %>% summarise( total\_diagnoses = n(), late\_diagnoses = sum(Late\_Diagnosis), percent\_late\_diagnoses = (late\_diagnoses / total\_diagnoses) \* 100 )

percentage\_positive <- hiv\_data %>% group\_by(year, region) %>% summarise(total\_tests = n(), positive\_tests = sum(test\_result == 1), positivity\_rate = (positive\_tests / total\_tests) \* 100)

8. # Summarise the new HIV diagnoses by age group and region aggregated\_diagnoses <- df %>% group\_by(age\_group, region) %>% summarise(new\_hiv\_diagnoses = sum(New\_HIV\_Diagnosis))

**Step 4: Create a Bar Plot of Percentage Positivity by Year and Region**

r

Copy code

# Create a bar plot

ggplot(percentage\_positive, aes(x = year, y = positivity\_rate, fill = region)) +

geom\_bar(stat = "identity", position = "dodge") +

labs(title = "HIV Positivity Rate by Year and Region",

x = "Year",

y = "Positivity Rate (%)") +

theme\_minimal()

**Step 5: Identify the Age Group and Sex with the Highest HIV Positivity Rate in the Last Year**

r

Copy code

# Filter for the last available year

last\_year <- max(hiv\_data$year)

last\_year\_data <- hiv\_data %>%

filter(year == last\_year) %>%

group\_by(age\_group, sex) %>%

summarise(total\_tests = n(),

positive\_tests = sum(test\_result == 1),

positivity\_rate = (positive\_tests / total\_tests) \* 100) %>%

arrange(desc(positivity\_rate))

# View the age group and sex with the highest positivity rate

print(last\_year\_data)

Assesssing distribution for numerical data

# Histogram

ggplot(hiv\_data, aes(x = years\_on\_ART)) +

geom\_histogram(bins = 30, fill = "blue", color = "black") +

labs(title = "Distribution of Years on ART", x = "Years on ART", y = "Frequency")

# Density Plot

ggplot(hiv\_data, aes(x = years\_on\_ART)) +

geom\_density(fill = "blue", alpha = 0.5) +

labs(title = "Density Plot of Years on ART")

# Boxplot

ggplot(hiv\_data, aes(y = years\_on\_ART)) +

geom\_boxplot() +

labs(title = "Boxplot of Years on ART", y = "Years on ART")

7. Handling missing data by multiple imputation method

Multiple Imputation For missing data

First stage Impute

M = mice(df, m = 10, maxit = 10, print = FALSE)

#Running the models on each imputation data set

G1=with(M,(Model)

#Pooling the imputations

COEFM=summary(pool(G1))

8. Analysis and modelling of surveillance data

Models for count data eg number of cases, incidence and mortality rates use either

a. Poisson or negative binomial. Choose binomial if

there is overdispersion- difference between mean and variation is wide. ie mean(df$count) -var(df$count) is wide.

DO NOT FORGET TO ACCOUNT FOR POPULATION SIZE OR TIME OF OBSERVATION USING ---OFFSET()

# Model the incidence rate per population and time using Poisson regression

poisson\_model <- glm(infections ~ predictor\_variable + offset(log(population)) + offset(log(time\_at\_risk)), family = poisson, data = data)

# Summary of the model

summary(poisson\_model)

# Load MASS package for negative binomial regression

library(MASS)

# Fit the Negative Binomial model

nb\_model <- glm.nb(infections ~ predictor\_variable + offset(log(population)) + offset(log(time\_at\_risk)), data = data)

# Summary of the model summary(nb\_model)

b. Binomial -for a yes or no outcome, risk factors, prediction, evaluation of effectiveness, sick or not sick etc