Impact Evaluation Presentation

By: Prashanthi Subbiah

Agenda

1. Introduction
2. Data Cleaning
3. Data Analysis
4. Results and Going Forward

Introduction

Objectives and Tools Used

Objectives and Tools Used

Aim: to evaluate the impact of childcare services on the mental well-being of Ugandan women working in markets in Kampala

3 groups: Control, Market-based services (T1), Community-based (T2)

Relevant outcome variables: PHQ4 (depression and anxiety), Locus of Control (anxiety) and Cantril's Ladder (happiness)

All of the coding was done on R Studio

https://forum.posit.co/t/cant-open-rstudio-with-a-yellow-triangle/14735

Data Cleaning

PHQ4

PHQ4

Each participant rated how much they agreed with the following statements on a scale of 0 (least) to 3 (most):

1. Little interest or pleasure in doing things
2. Feeling down, depressed, or hopeless
3. Feeling nervous, anxious, or on edge
4. Not being able to stop or control worrying

PHQ4

Each participant rated how much they agreed with the following statements on a scale of 0 (least) to 3 (most):

1. Little interest or pleasure in doing things
2. Feeling down, depressed, or hopeless
3. Feeling nervous, anxious, or on edge
4. Not being able to stop or control worrying

Depression: Sum of answers to 1 and 2 - out of 6

Anxiety: Sum of answers to 3 and 4 - out of 6

Total Wellbeing: Sum of answers to 1 through 4 - out of 12

```r
## Making sure none of the variables have NA responses
end_df_1 <- end_df %>%
  filter(!is.na(well1),
         !is.na(well2),
         !is.na(well3),
         !is.na(well4))


## Recoding wellbeing to fit standard procedures
end_df_1 <- end_df_1 %>%
  mutate(well_1 = as.numeric(well1) - 1,
         well_2 = as.numeric(well2) - 1,
         well_3 = as.numeric(well3) - 1,
         well_4 = as.numeric(well4) - 1) %>%
  mutate(dep = well_1 + well_2,
         anx = well_3 + well_4)

unique(is.na(end_df_1$well4)) # Returns FALSE, so no NAs

## Creating a Variable that returns the aggregate of the responses to the above
## questions

end_df_1 <- end_df_1 %>%
  mutate(phq_4 = well_1 + well_2 + well_3 + well_4)

## Naming similarly to midline and endline surveys, and switching signs for
## methodological purposes

end_df_1 <- end_df_1 %>%
  mutate(tot_phq_4 = -phq_4,
         dep_f = -dep,
         anx_f = -anx)
```

```
## Making sure none of the variables have NA responses
end_df_1 <- end_df %>%
  filter(!is.na(well1),
         !is.na(well2),
         !is.na(well3),
         !is.na(well4))

## Recoding wellbeing to fit standard procedures
end_df_1 <- end_df_1 %>%
  mutate(well_1 = as.numeric(well1) - 1,
         well_2 = as.numeric(well2) - 1,
         well_3 = as.numeric(well3) - 1,
         well_4 = as.numeric(well4) - 1) %>%
  mutate(dep = well_1 + well_2,
         anx = well_3 + well_4)

unique(is.na(end_df_1$well4)) # Returns FALSE, so no NAs

## Creating a Variable that returns the aggregate of the responses to the above
## questions

end_df_1 <- end_df_1 %>%
  mutate(phq_4 = well_1 + well_2 + well_3 + well_4)

## Naming similarly to midline and endline surveys, and switching signs for
## methodological purposes

end_df_1 <- end_df_1 %>%
  mutate(tot_phq_4 = -phq_4,
         dep_f = -dep,
         anx_f = -anx)
```

Locus of Control

Locus of Control

Participants responded 1 (least agreement) to 5 (most agreement) to set of 9 questions, then standardized each column

The entire sample size was divided into 9 batches - a total of 81 columns

Locus of Control

Participants responded 1 (least agreement) to 5 (most agreement) to set of 9 questions, then standardized each column

The entire sample size was divided into 9 batches - a total of 81 columns

Aggregated all of the scores into a column for each question, resulting in a total of 9 columns

Using 9 x-variables could lead to more variance in regression, potential correlation

Locus of Control

Participants responded 1 (least agreement) to 5 (most agreement) to set of 9 questions, then standardized each column

The entire sample size was divided into 9 batches - a total of 81 columns

Aggregated all of the scores into a column for each question, resulting in a total of 9 columns

Using 9 x-variables could lead to more variance in regression, potential correlation

Dimension reduction method PCA applied to summarize into pca_external (3 questions), pca_internal (6 questions), and pca_total (all 9)

```
recode_string <- function(value) {
    case_when(
        is.na(value) ~ NA,  # Handle NA values
        value == "Completely agree (Okiliziganyiza ddala)" ~ 5,
        value == "Mostly agree (Okiliziganya)" ~ 4,
        value == "Neither agree nor disagree (Toli ku ludda olukiliziganya ate era toli ku lutakiliziganya)" ~ 3,
        value == "Mostly disagree (Tokilizigannya)" ~ 2,
        value == "Completely disagree (Tokiliziganyiza ddala)" ~ 1,
        value == "" ~ NA,
    )
}
```

Initial answers recoded as numeric scores

```
# Create the base DataFrame with standardized columns and new columns as needed
base_df_3 <- base_df_2 %>%
  select(std_1_loc5_o1_1_label.y, std_1_loc5_o2_1_label.y, std_1_loc5_o3_1_label.y, std_1_loc5_o4_1_label.y,
std_1_loc5_o5_1_label.y,
         std_1_loc5_o6_1_label.y, std_1_loc5_o7_1_label.y, std_1_loc5_o8_1_label.y, std_1_loc5_o9_1_label.y) %>%
  mutate(loc5_o1_1_N = ifelse(is.na(std_1_loc5_o1_1_label.y), NA, std_1_loc5_o1_1_label.y),
         loc5_o2_1_N = ifelse(is.na(std_1_loc5_o2_1_label.y), NA, std_1_loc5_o2_1_label.y),
         loc5_o3_1_N = ifelse(is.na(std_1_loc5_o3_1_label.y), NA, std_1_loc5_o3_1_label.y),
         loc5_o4_1_N = ifelse(is.na(std_1_loc5_o4_1_label.y), NA, std_1_loc5_o4_1_label.y),
         loc5_o5_1_N = ifelse(is.na(std_1_loc5_o5_1_label.y), NA, std_1_loc5_o5_1_label.y),
         loc5_o6_1_N = ifelse(is.na(std_1_loc5_o6_1_label.y), NA, std_1_loc5_o6_1_label.y),
         loc5_o7_1_N = ifelse(is.na(std_1_loc5_o7_1_label.y), NA, std_1_loc5_o7_1_label.y),
         loc5_o8_1_N = ifelse(is.na(std_1_loc5_o8_1_label.y), NA, std_1_loc5_o8_1_label.y),
         loc5_o9_1_N = ifelse(is.na(std_1_loc5_o9_1_label.y), NA, std_1_loc5_o9_1_label.y))

# Calculate row sums with handling for rows where all values are NA
base_df_3 <- base_df_3 %>%
  select(loc5_o1_1_N, loc5_o2_1_N, loc5_o3_1_N, loc5_o4_1_N, loc5_o5_1_N, loc5_o6_1_N, loc5_o7_1_N, loc5_o8_1_N, loc5_o9_1_N)

# Making sure all rows that have NA in all columns return sum as NA

base_df_3$loc5_1_Sum <- apply(base_df_3, 1, function(row) {
  if (all(is.na(row))) {
    return(NA)
  } else {
    return(sum(row, na.rm = TRUE))
  }
})

sum(is.na(base_df_3$loc5_1_Sum))

# Checking with variable from do file - same number of NAs
sum(is.na(base_df$loc5_1))

base_df_2$loc5_1_Stand = base_df_3$loc5_1_Sum
```

Aggregation of answers to a single question into 1 column after standardization

```r
recode_string <- function(value) {
    case_when(
        is.na(value) ~ NA,  # Handle NA values
        value == "Completely agree (Okiliziganyiza ddala)" ~ 5,
        value == "Mostly agree (Okiliziganya)" ~ 4,
        value == "Neither agree nor disagree (Toli ku ludda olukiliziganya ate era toli ku lutakiliziganya)" ~ 3,
        value == "Mostly disagree (Tokilizigannya)" ~ 2,
        value == "Completely disagree (Tokiliziganyiza ddala)" ~ 1,
        value == "" ~ NA,
    )
}
```

Initial answers recoded as numeric scores

```r
# Create the base DataFrame with standardized columns and new columns as needed
base_df_3 <- base_df_2 %>%
  select(std_1_loc5_o1_1_label.y, std_1_loc5_o2_1_label.y, std_1_loc5_o3_1_label.y, std_1_loc5_o4_1_label.y,
std_1_loc5_o5_1_label.y,
         std_1_loc5_o6_1_label.y, std_1_loc5_o7_1_label.y, std_1_loc5_o8_1_label.y, std_1_loc5_o9_1_label.y) %>%
  mutate(loc5_o1_1_N = ifelse(is.na(std_1_loc5_o1_1_label.y), NA, std_1_loc5_o1_1_label.y),
         loc5_o2_1_N = ifelse(is.na(std_1_loc5_o2_1_label.y), NA, std_1_loc5_o2_1_label.y),
         loc5_o3_1_N = ifelse(is.na(std_1_loc5_o3_1_label.y), NA, std_1_loc5_o3_1_label.y),
         loc5_o4_1_N = ifelse(is.na(std_1_loc5_o4_1_label.y), NA, std_1_loc5_o4_1_label.y),
         loc5_o5_1_N = ifelse(is.na(std_1_loc5_o5_1_label.y), NA, std_1_loc5_o5_1_label.y),
         loc5_o6_1_N = ifelse(is.na(std_1_loc5_o6_1_label.y), NA, std_1_loc5_o6_1_label.y),
         loc5_o7_1_N = ifelse(is.na(std_1_loc5_o7_1_label.y), NA, std_1_loc5_o7_1_label.y),
         loc5_o8_1_N = ifelse(is.na(std_1_loc5_o8_1_label.y), NA, std_1_loc5_o8_1_label.y),
         loc5_o9_1_N = ifelse(is.na(std_1_loc5_o9_1_label.y), NA, std_1_loc5_o9_1_label.y))

# Calculate row sums with handling for rows where all values are NA
base_df_3 <- base_df_3 %>%
  select(loc5_o1_1_N, loc5_o2_1_N, loc5_o3_1_N, loc5_o4_1_N, loc5_o5_1_N, loc5_o6_1_N, loc5_o7_1_N, loc5_o8_1_N, loc5_o9_1_N)

# Making sure all rows that have NA in all columns return sum as NA

base_df_3$loc5_1_Sum <- apply(base_df_3, 1, function(row) {
  if (all(is.na(row))) {
    return(NA)
  } else {
    return(sum(row, na.rm = TRUE))
  }
})

sum(is.na(base_df_3$loc5_1_Sum))

# Checking with variable from do file - same number of NAs
sum(is.na(base_df$loc5_1))

base_df_2$loc5_1_Stand = base_df_3$loc5_1_Sum
```

Aggregation of answers to a single question into 1 column after standardization

Cantril’s Ladder

Cantril’s Ladder

Participants assigned scores to standard set of 2 questions from 0 (least happy) to 10 (most happy):

- ladder_now: Women’s happiness today
- ladder_five: Expected women’s happiness in 5 years

Calculated ladder_diff: Difference between expected happiness and current happiness scores

```
# Creating ladder_diff
end_df_1 <- end_df_1 %>%
  mutate(ladder_diff = ladder_five - ladder_now)
```

Creating the Wellbeing Index

Creating the Wellbeing Index

The wellbeing index is a sum all 9 of these variables, then standardized by subtracting the control mean and dividing by the standard deviation of the control group.

```
# Compute the composite index
end_df_3 <- end_df_3 %>%
  mutate(w_index = tot_phq_4 + dep_f + anx_f + pca_locus + pca_internal +
           pca_external + ladder_now + ladder_five + ladder_diff)

# Calculate the mean and standard deviation
control_value_df <- end_df_3 %>%
  filter(control == 1)

mean_control_value <- mean(control_value_df$w_index)

sd_control_value <- sd(control_value_df$w_index)

# Standardize the variable
end_df_3$w_index <- (end_df_3$w_index - mean_control_value) / sd_control_value

# Summary
summary(end_df_3$w_index)

# Checking for NAs
sum(is.na(end_df_3$w_index))
```

Creating the Wellbeing Index

The wellbeing index is a sum all 9 of these variables, then standardized by subtracting the control mean and dividing by the standard deviation of the control group.

Sum all 9 columns

```
# Compute the composite index
end_df_3 <- end_df_3 %>%
  mutate(w_index = tot_phq_4 + dep_f + anx_f + pca_locus + pca_internal +
           pca_external + ladder_now + ladder_five + ladder_diff)

# Calculate the mean and standard deviation
control_value_df <- end_df_3 %>%
  filter(control == 1)

mean_control_value <- mean(control_value_df$w_index)

sd_control_value <- sd(control_value_df$w_index)

# Standardize the variable
end_df_3$w_index <- (end_df_3$w_index - mean_control_value) / sd_control_value

# Summary
summary(end_df_3$w_index)

# Checking for NAs
sum(is.na(end_df_3$w_index))
```

Subtract Control Mean and divide by Control SD

Data Analysis

Lee Bounds

Lee Bounds

There was differential attrition - participants who left the study at different stages belonged to a particular treatment group

Lee Bounds are a statistical method to exclude outliers resulting from this.

This includes calculating the number of observations to trim:

1. trimmed_upper: copy of dataframe with lowest observations trimmed
2. trimmed_lower: copy of dataframe with highest observations trimmed

Reference: https://blogs.worldbank.org/en/impactevaluations/lee-bounds-in-practice

Choosing Control Variables

Choosing Control Variables

Machine Learning Technique used to choose which x-variables to regress the wellbeing index on: LASSO

Non-zero coefficients - yield x-variables to be used, to minimize variance

```r
# LASSO for outcome (y) and treatment (d)
lasso_y <- cv.glmnet(X, y, alpha = 1)  # LASSO for outcome
lasso_d <- cv.glmnet(X, d, alpha = 1, family = "binomial") # LASSO for treatment

# Combine selected variables
selected_vars <- union(
  which(coef(lasso_y, s = "lambda.min")[-1] != 0),
  which(coef(lasso_d, s = "lambda.min")[-1] != 0)
)
```

ANCOVA Fixed Effects Regressions

ANCOVA Fixed Effects Regressions

Fun part - running regressions!

For each stage of the study - baseline, midline, and endline - the wellbeing index is regressed upon the baseline wellbeing index, treatment status, and controls obtained from previous stage

	Wellbeing Index	Lower Lee Bound	Upper Lee Bound
Pooled	0.170	0.070	0.300
Treatment 1	0.250	0.080	0.275
Treatment 2	0.150	0.075	0.205

These results are for demonstrative purposes only as actual data is confidential

	Wellbeing Index	Lower Lee Bound	Upper Lee Bound
Pooled	-0.060	-0.200	0.100
Treatment 1	0.010	-0.119	0.275
Treatment 2	-0.110	0.075	0.205

None of these endline results were statistically significant and bounds were less than 0, making the treatment effect uncertain

These results are for demonstrative purposes only as actual data is confidential

Panel Model

Panel Model

To corroborate results, the wellbeing indexes from all 3 stages were concatenated into one column, regressed on variables from before

	Wellbeing Index
Pooled	0.050
Treatment 1	0.030
Treatment 2	0.040

Fell in ranges, but not statistically significant

These results are for demonstrative purposes only as actual data is confidential

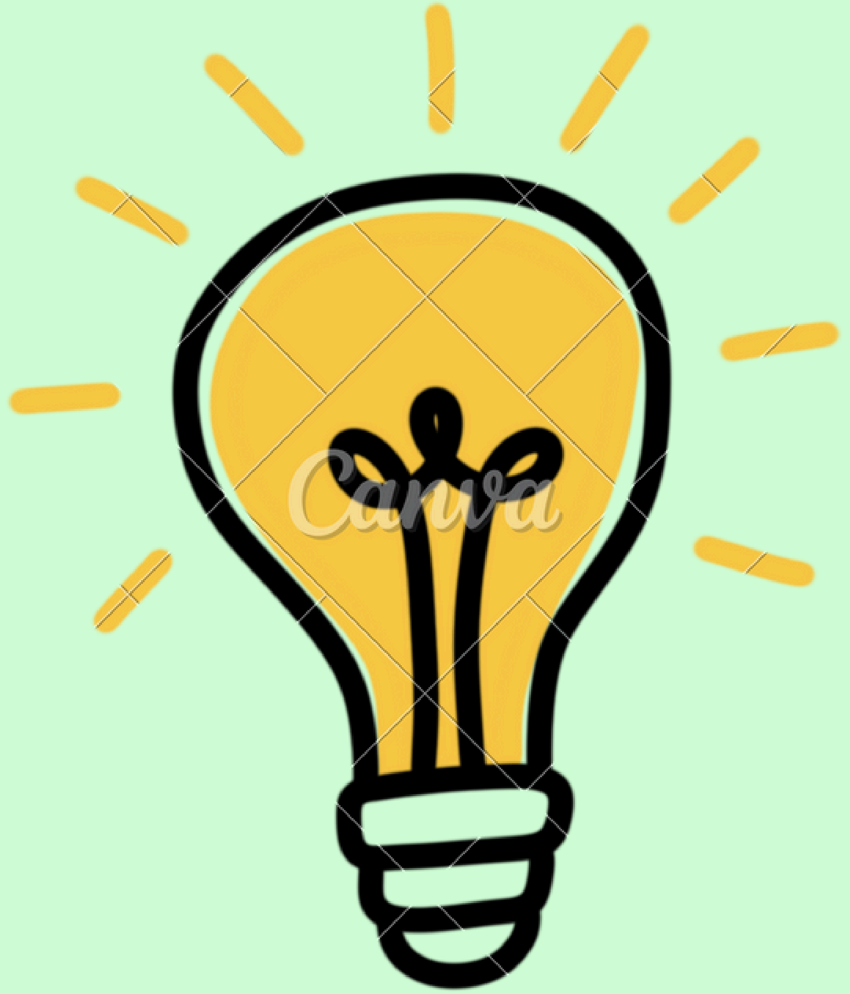
Results and Going Forward

Findings

Findings

Overall, treatment effect was positive and significant at midline, just before the treatment ended. This did not last after it stopped

Market-based treatment had a higher positive effect than Community-based, which would call for further research into social dynamics of this section of Ugandan society



This could be studied with respect to other similar interventions - such as the provision of childcare services out of local schools in Algeria to create a validation framework

Thank You!