obj4_all_tests.r

shivs

2026-01-07

```
# obj 3. To analyze the impact of PMAGY on education and health facilities in the s
elected villages.

# import packages
library(readxl)
library(ltm)
```

```
## Loading required package: MASS
```

```
## Loading required package: msm
```

```
## Loading required package: polycor
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:MASS':
##
##     select
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
# Load the purrr library for iteration
library(purrr)

post_pmagy_file_path <- "/home/shivs/r-analysis-pmagy/post_pmagy.xlsx"

### lisd Section tests:

print("lisd Section tests: ")
```

```
## [1] "lisd Section tests: "
```

```
#POST PMAGY lisd
lisd_POST <-data.frame(read_excel(post_pmagy_file_path,sheet = "Sheet1",range = "CJ
1:CN487")) #lisd_POST
colnames(lisd_POST) <- paste('POST', colnames(lisd_POST), sep = '_')
#PRE PMAGY lisd
lisd_PRE <-data.frame(read_excel("/home/shivs/r-analysis-pmagy/pre_pmagy_obj4_lis
d.xlsx",range = "A1:E487")) #lisd_PRE
colnames(lisd_PRE) <- paste('PRE', colnames(lisd_PRE), sep = '_')


#calculate Cronbach's Alpha:
print(cronbach.alpha(lisd_POST))
```

```
##
## Cronbach's alpha for the 'lisd_POST' data-set
##
## Items: 5
## Sample units: 486
## alpha: 0.081
```

```
print(cronbach.alpha(lisd_PRE))
```

```
##
## Cronbach's alpha for the 'lisd_PRE' data-set
##
## Items: 5
## Sample units: 486
## alpha: -0.065
```

```
#

#Variance Inflation Factor (VIF):
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
library(caTools)
# Fit a regression model on post data
lisd_POST_w_index <-data.frame(read_excel(post_pmagy_file_path,sheet = "Sheet1",ran
ge = "CJ1:CO487"))
colnames(lisd_POST_w_index) <- paste('POST', colnames(lisd_POST_w_index), sep =
'_')
model_all <- lm(POST_LISD_INDEX ~ ., data = lisd_POST_w_index)
# Calculate VIF
vif_results <- car::vif(model_all)

print("VIF results for post data:")
```

```
## [1] "VIF results for post data:"
```

```
print(vif_results)
```

```
##                                                                    POST_LISD.1..
..Skill.development.training.programs.are.accessible.
##
1.061232
##                                        POST_LISD.2...Career.counselling...Governme
nt.job.coachings.classes.are.available.and.effective.
##
1.008113
## POST_LISD.3...we.are.satisfied.with.the.opportunities.for.starting.a.business.in
.the.village..including.access.to.loans.and.training.
##
1.041113
##                                                  POST_LISD.3....Skill.training
.has.positively.impacted.employment.in.our.household.
##
1.031550
##                                               POST_LISD.4.....A.member.of.our.ho
usehold.is.currently.part.of.a.Self.Help.Group..SHG..
##
1.015522
```

```
# Fit a regression model on pre data
lisd_PRE_w_index <-data.frame(read_excel("/home/shivs/r-analysis-pmagy/pre_pmagy_ob
j4_lisd.xlsx",range = "A1:F487"))
colnames(lisd_PRE_w_index) <- paste('PRE', colnames(lisd_PRE_w_index), sep = '_')
model_all <- lm(PRE_LISD_INDEX ~ ., data = lisd_PRE_w_index)
# Calculate VIF
vif_results <- car::vif(model_all)
print("VIF results for pre data:")
```

```
## [1] "VIF results for pre data:"
```

```
print(vif_results)
```

```
##                                                                    PRE_LISD.1...
.Skill.development.training.programs.are.accessible.
##
1.011790
##                                        PRE_LISD.2...Career.counselling...Governmen
t.job.coachings.classes.are.available.and.effective.
##
1.007277
## PRE_LISD.3...we.are.satisfied.with.the.opportunities.for.starting.a.business.in.
the.village..including.access.to.loans.and.training.
##
1.003468
##                                                  PRE_LISD.3....Skill.training.
has.positively.impacted.employment.in.our.household.
##
1.009637
##                                               PRE_LISD.4.....A.member.of.our.hou
sehold.is.currently.part.of.a.Self.Help.Group..SHG..
##
1.017514
```

```
#TODO: Visualizing VIF Values from here https://www.geeksforgeeks.org/r-language/vi
f-function-in-r/
#

# Using theShapiro–Wilk Test :
print("Shapiro–Wilk Test results for pre data:")
```

```
## [1] "Shapiro–Wilk Test results for pre data:"
```

```
print(shapiro.test(lisd_PRE_w_index$PRE_LISD_INDEX))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  lisd_PRE_w_index$PRE_LISD_INDEX
## W = 0.95163, p-value = 1.618e-11
```

```
print("Shapiro–Wilk Test results for post data:")
```

```
## [1] "Shapiro–Wilk Test results for post data:"
```

```
print(shapiro.test(lisd_POST_w_index$POST_LISD_INDEX))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  lisd_POST_w_index$POST_LISD_INDEX
## W = 0.94583, p-value = 2.469e-12
```

```
#

# Paired Samples Wilcoxon Test :

# combine two data frames horizontally
lisd_pre_post_w_index_hcomb <- cbind(lisd_PRE_w_index, lisd_POST_w_index)

# # Define a function to perform paired Paired Samples Wilcoxon Test between pairs
of columns
paired_wilcoxon_test <- function(before, after) {
  wilcox.test(before, after, paired = TRUE)
}

# # Apply paired Paired Samples Wilcoxon Test to each pair of pre and post columns
results_wilcoxon_test <- map2(lisd_pre_post_w_index_hcomb[grepl("PRE", names(lisd_p
re_post_w_index_hcomb))],
                lisd_pre_post_w_index_hcomb[grepl("POST", names(lisd_pre_post_w_ind
ex_hcomb))],
                paired_wilcoxon_test)

# # Print the results
print("Print the results for column/question wise Paired Samples Wilcoxon Test")
```

```
## [1] "Print the results for column/question wise Paired Samples Wilcoxon Test"
```

```
print(results_wilcoxon_test)
```

```
## $PRE_LISD.1....Skill.development.training.programs.are.accessible.
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  before and after
## V = 130, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
##
##
## $PRE_LISD.2...Career.counselling...Government.job.coachings.classes.are.availabl
e.and.effective.
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  before and after
## V = 559, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
##
##
## $PRE_LISD.3...we.are.satisfied.with.the.opportunities.for.starting.a.business.i
n.the.village..including.access.to.loans.and.training.
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  before and after
## V = 2068, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
##
##
## $PRE_LISD.3....Skill.training.has.positively.impacted.employment.in.our.househol
d.
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  before and after
## V = 4485, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
##
##
## $PRE_LISD.4.....A.member.of.our.household.is.currently.part.of.a.Self.Help.Grou
p..SHG..
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  before and after
## V = 20, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
##
##
## $PRE_LISD_INDEX
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  before and after
```

```
## V = 0, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

```
#

## cohen's d test :
#Step 1:
#install.packages("effsize")
library(effsize)

#Step 2:

## Define a function to perform cohen d Test between pairs of columns
cohen_d_test <- function(before, after) {
  cohen.d(before, after)
}

    ## Apply paired Paired Samples Wilcoxon Test to each pair of pre and post colum
ns
    results_cohen_d_test <- map2(lisd_pre_post_w_index_hcomb[grepl("PRE", names(lis
d_pre_post_w_index_hcomb))],
                      lisd_pre_post_w_index_hcomb[grepl("POST", names(lisd_pre_post_w
_index_hcomb))],
                      cohen_d_test)
print("Print the results for column/question wise cohen d Test")
```

```
## [1] "Print the results for column/question wise cohen d Test"
```

```
print(results_cohen_d_test)
```

```
## $PRE_LISD.1....Skill.development.training.programs.are.accessible.
##
## Cohen's d
##
## d estimate: -4.389146 (large)
## 95 percent confidence interval:
##     lower     upper
## -4.621550 -4.156743
##
##
## $PRE_LISD.2...Career.counselling...Government.job.coachings.classes.are.availabl
e.and.effective.
##
## Cohen's d
##
## d estimate: -3.034932 (large)
## 95 percent confidence interval:
##     lower     upper
## -3.219579 -2.850285
##
##
## $PRE_LISD.3...we.are.satisfied.with.the.opportunities.for.starting.a.business.i
n.the.village..including.access.to.loans.and.training.
##
## Cohen's d
##
## d estimate: -2.14882 (large)
## 95 percent confidence interval:
##     lower     upper
## -2.306919 -1.990722
##
##
## $PRE_LISD.3....Skill.training.has.positively.impacted.employment.in.our.househol
d.
##
## Cohen's d
##
## d estimate: -1.355679 (large)
## 95 percent confidence interval:
##     lower     upper
## -1.495281 -1.216077
##
##
## $PRE_LISD.4.....A.member.of.our.household.is.currently.part.of.a.Self.Help.Grou
p..SHG..
##
## Cohen's d
##
## d estimate: -2.942353 (large)
## 95 percent confidence interval:
##     lower     upper
## -3.124007 -2.760698
##
##
```

```
## $PRE_LISD_INDEX
##
## Cohen's d
##
## d estimate: -5.785104 (large)
## 95 percent confidence interval:
##     lower     upper
## -6.071717 -5.498491
```

```
### lisd Section tests COMPLETE
print("lisd Section tests COMPLETE")
```

```
## [1] "lisd Section tests COMPLETE"
```