Package 'WSCdata'

October 23, 2025

Title A New Four-Arm Within-Study Comparison Data on Math and Vocabulary Training

Version 0.1.1

Description This dataset was collected using a new four-arm within-study comparison design. The study aimed to examine the impact of a mathematics training intervention and a vocabulary study session on post-test scores in mathematics and vocabulary, respectively. The innovative four-arm within-study comparison design facilitates both experimental and quasi-experimental identification of average causal effects.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

LazyData true

Suggests testthat (>= 3.0.0), tidyverse

Config/testthat/edition 3

URL https://github.com/jzangela/WSCdata

BugReports https://github.com/jzangela/WSCdata/issues

NeedsCompilation no

Author Bryan Keller [aut, cph],
Sangbaek Park [aut, ctb],
Jingru Zhang [aut, cre]

Maintainer Jingru Zhang <jzhang2637@wisc.edu>

Repository CRAN

Date/Publication 2025-10-23 21:30:02 UTC

Contents

AMAS_WSC *AMAS Dataset*

Description

This dataset contains measurements of math anxiety using the Abbreviated Math Anxiety Scale (AMAS), collected from 2,200 Amazon Mechanical Turk workers (Keller et al., 2022) This scale consists of 9 items that assess mathematics anxiety, a negative emotional response associated with mathematics-related activities.

Usage

```
data(AMAS_WSC)
```

Format

AMAS_WSC is a data frame with 2200 cases (rows) and 9 variables (columns). The variables are named item_1, item_2, ... , item_9.

References

Hopko, D., Mahadevan, R., Bare, R. L., & Hunt, M. K. (2003). The abbreviated math anxiety scale (AMAS): Construction, validity, and reliability. Assessment, 10, 78-182.

Examples

```
AMAS_WSC
```

BDI_WSC *BDI Dataset*

Description

The dataset contains measurements of attitudes and symptoms of depression using the 13-item short form of the Beck Depression Inventory (BDI), collected from 2,200 Amazon Mechanical Turk workers (Keller et al., 2022)

Usage

```
data(BDI_WSC)
```

Format

BDI_WSC is a data frame with 2200 cases (rows) and 13 variables (columns). The variables are named item_1, item_2, ... , item_13.

References

Beck, A. T., & Beck, R. W. (1972). Screening Depressed Patients in Family Practice: A Rapid Technic. Postgraduate Medicine, 52(6), 81-85. https://doi.org/10.1080/00325481.1972.11713319

Examples

```
BDI_WSC
```

Big5_WSC *Big 5 Dataset*

Description

This dataset contains measurements of the Big Five personality trait measurements using the Big Five Inventory (BFI), collected from 2,200 Amazon Mechanical Turk workers (Keller et al., 2022). The items include both positively and negatively worded statements; therefore, reverse coding is required for certain items prior to conducting any psychometric analyses.

Usage

```
data(Big5_WSC)
```

Format

Big5_WSC is a data frame with 2200 cases (rows) and 44 variables (columns). The variables are named O1-O10, C1-C9, E1-E8, A1-A9, and N1-N8.

Details

The Big Five personality test comprises five scales: Openness (O) with 10 items, Conscientiousness (C) with 9 items, Extraversion (E) with 8 items, Agreeableness (A) with 9 items, and Neuroticism (N) with 8 items.

References

John, O. P., & Srivastava, S. (1999). The big five trait taxonomy: History, measurement, and theoretical perspectives. In L. A. Pervin & O. P. John (Eds.), Handbook of personality: Theory and research (pp. 102-138). New York, NY: Guilford Press.

Examples

```
Big5_WSC
```

GSES_WSC *GSES Dataset*

Description

The dataset contains measurements of perceived self-efficacy related to coping and adaptation abilities, using six out of the ten items from the General Self-Efficacy Scale (GSES)-specifically item 2, 3, 6, 5, 7, and 10. The data were collected from 2,200 Amazon Mechanical Turk workers (Keller et al., 2022).

Usage

```
data(GSES_WSC)
```

Format

GSES_WSC is a data frame with 2200 cases (rows) and 6 variables (columns). The variables are named item_1, item_2, ... , item_6.

References

Schwarzer, R., & Jerusalem, M. (1995). General Self-Efficacy Scale (GSE) Database record. APA PsycTests. https://doi.org/10.1037/t00393-000

Examples

```
GSES_WSC
```

Math_Post_WSC *Math Post-test Dataset*

Description

This dataset contains mathematics posttest data prepared by the authors (Keller et al., 2022), collected from the 2,200 Amazon Mechanical Turk workers, This test consists of 15 items: five assessing general mathematics aptitude and ten focused on properties of exponents.

Usage

```
data(Math_Post_WSC)
```

Format

Math_Post_WSC is a data frame with 2200 cases (rows) and 15 variables (columns). The variables are named item_1, item_2, ... , item_15.

Examples

```
Math_Post_WSC
```

Math_Pre_WSC *Math Pre-test Dataset*

Description

This dataset contains mathematics pretest data prepared by the authors (Keller et al., 2022), collected from 2,200 Amazon Mechanical Turk workers. The test draws on a combination of self-written items and items from the ETS Kit of Factor-Referenced Cognitive Tests. It consists of 12 items: seven assessing general mathematics aptitude, and five focusing on simplifying algebraic expressions involving exponents.

Usage

```
data(Math_Pre_WSC)
```

Format

Math_Pre_WSC is a data frame with 2200 cases (rows) and 12 variables (columns). The variables are named item_1, item_2, ... , item_12.

References

R. B. Ekstrom, J. W. French, H. H. Harman, and D. Dermen. Manual for Kit of Factor- Referenced Cognitive Tests. Educational Testing Service, 1976.

Examples

```
Math_Pre_WSC
```

`Math_Train_WSC` *Math Training Dataset*

Description

This dataset contains practice questions presented during mathematics intervention to 2,200 Amazon Mechanical Turk workers. A total 10 items assessed participants' understanding of rules for simplifying exponential expressions. For participants who received the vocabulary intervention, all items are recorded as missing.

Usage

```
data(Math_Train_WSC)
```

Format

Math_Train_WSC is a data frame with 2200 cases (rows) and 10 variables (columns). The variables are named item_1, item_2, ... , item_10.

Examples

```
Math_Train_WSC
```

`MCS_WSC` *MSC Dataset*

Description

This dataset contains measurements of confidence in understanding and in the ability to simplify algebraic expressions involving exponents. The 5-point Likert-type items were developed by Keller et al. (2022) and administered to 2,200 Amazon Mechanical Turk workers. The dataset includes Math Confidence Scale (MCS) measurements taken both before and after a math training intervention. The final variable, named 'type', indicates whether the data were collected prior to or following the intervention.

Usage

```
data(MCS_WSC)
```

Format

MCS_WSC is a data frame with 2200 cases (rows) and 7 variables (columns). The variables are named item_1, item_2 ...., item_6, and type ("pre" or "post").

item_1 I understand the meaning of a factor in an algebraic expression.

item_2 I understand the difference between a base and an expression in an algebraic expression.

item_3 I can simplify algebraic expressions that involve multiplying exponential expressions with the same base.

item_4 I can simplify algebraic expressions that involve dividing exponential expressions with the same base.

item_5 I can simplify algebraic expressions that involve raising an expression to a power.

item_6 I can simplify algebraic expressions that involve negative exponents.

type Either "pre" or "post", indicating assessment timing.

Examples

```
MCS_WSC
```

Vocab_Post_WSC *Vocabulary Post-test Dataset*

Description

This dataset contains vocabulary posttest for the 2,200 Amazon Mechanical Turk workers. This test consists of 24 items that ask participants to match words with their definitions. Twelve of the fifty words presented during the vocabulary study session also appeared on the 24-item vocabulary posttest.

Usage

```
data(Vocab_Post_WSC)
```

Format

Vocab_Post_WSC is a data frame with 2200 cases (rows) and 24 variables (columns). The variables are named item_1, item_2, ... , item_24.

Examples

```
Vocab_Post_WSC
```

Vocab_Pre_WSC *Vocabulary Pre-test Dataset*

Description

This dataset contains vocabulary pretest data for 2,200 Amazon Mechanical Turk workers. The test consists of 24 items that ask participants to match words with their definitions.

Usage

```
data(Vocab_Pre_WSC)
```

Format

Vocab_Pre_WSC is a data frame with 2200 cases (rows) and 24 variables (columns). The variables are named item_1, item_2, ... , item_24.

Examples

```
Vocab_Pre_WSC
```

WSCdata *A new four-arm within-study comparison data on math and vocabulary training*

Description

This dataset was collected using a new four-arm within-study comparison design. The study aimed to examine the impact of a mathematics training intervention and a vocabulary study session on posttest scores in mathematics and vocabulary, respectively. The innovative four-arm within-study comparison design facilitates both experimental and quasi-experimental identification of average causal effects.

Usage

```
data(WSCdata)
```

Format

A data frame with 2200 observations. There are 33 variables recorded for each individual.

- "female" is the indicator variable for female (0 = Non-female, 1 = Female).
- "white" is the indicator variable for White ethnicity (0 = No, 1 = Yes).
- "black" is the indicator variable for Black ethnicity (0 = No, 1 = Yes).
- "hisp" is the indicator variable for Hispanic ethnicity (0 = No, 1 = Yes).

- "asian" is the indicator variable for Asian ethnicity (0 = No, 1 = Yes).
- "married" is the indicator variable for marital status (0 = Not Married, 1 = Married).
- "age" is age in years of the individual.
- "income" is the indicator variable for household income equal to or greater than $40k (0 = Less than $40k, 1 = Equal to or greater than $40k).
- "collegeS" is the indicator variable for college education for self (0 = No, 1 = Yes).
- "collegeM" is the indicator variable for college education for mother (0 = No, 1 = Yes).
- "collegeD" is the indicator variable for college education for father (0 = No, 1 = Yes).
- "calc" is the indicator variable for having taken a calculus course (0 = No, 1 = Yes).
- "books" is the number of books read in past year (ranging from 0 to 60).
- "mathLike" is the indicator variable for (0 = No, 1 = Yes).
- "Big5O" is Big Five Personality Inventory Openness Subscale score.
- "Big5C" is Big Five Personality Inventory Conscientiousness Subscale score.
- "Big5E" is Big Five Personality Inventory Extraversion Subscale score.
- "Big5A" is Big Five Personality Inventory Agreeableness Subscale score.
- "Big5N" is Big Five Personality Inventory Neuroticism Subscale score.
- "AMAS" is Abbreviated Math Anxiety Scale (AMAS) score.
- "BDI" is Beck Depression Inventory (BDI) score.
- "MCS" is Math Confidence Scale (MCS) score.
- "GSES" is General Self-Efficacy Scale (GSES) score.
- "vocabPre" is pre-test vocabulary score.
- "mathPre" is pre-test math score.
- "vocabPost" is post-test vocabulary score.
- "mathPost" is post-test math score.
- "MCSpost" is post-test MCS score.
- "mathGrp" is the indicator variable for group where participants were randomly assigned to math training (0 = No, 1 = Yes).
- "mathSel" is the indicator variable for group where participants selected math training (0 = No, 1 = Yes).
- "logAge" is log-transformed age.
- "logBooks" is log-transformed number of books.
- "logBDI" is log-transformed BDI (Beck Depression Inventory) score.

"mathGrp" and "mathSel" can derive different treatment and selection indicators for either math training or vocabulary training. The four arms in the new WSC design can be represented by these two variables. See paper for more details.

Source

Keller et al., ...

References

Keller et al., ...

Examples

```
if (requireNamespace("tidyverse", quietly = TRUE)) {
# Load functions ------------------------------------------------------------
library(tidyverse)

# Load data -----------------------------------------------------------------
head(WSCdata)

# Indicators for group comparisons (math and vocabulary)  -------------------
# Group indicators for ATE, ATT, and ATU are generated from random assignment and
# self-selection group indicators. For example, the math training group indicator
# among the treated (used to estimate the average treatment effect among the treated)
# is determined based on whether participants were randomized and self-selected for
# training. Specifically, the treatment group includes participants who were both
# randomized and self-selected for training. The control group includes participants
# who were self-selected to be trained in math but was not randomized to be.

# RCT math
WSCdata <- WSCdata %>%
 mutate(ind_rct_math = mathGrp)

# ATT math
WSCdata <- WSCdata %>%
 mutate(ind_att_math = case_when(mathGrp == 1 & mathSel == 1 ~ 1,
                                 mathGrp == 0 & mathSel == 1 ~ 0))
table(WSCdata$ind_att_math, useNA = "always")

# ATU math
WSCdata <- WSCdata %>%
mutate(ind_atu_math = case_when(mathGrp == 1 & mathSel == 0 ~ 1,
                                 mathGrp == 0 & mathSel == 0 ~ 0))
table(WSCdata$ind_atu_math, useNA = "always")

# QED math
WSCdata <- WSCdata %>%
mutate(ind_QED_math = case_when(mathGrp == 1 & mathSel == 1 ~ 1,
                                mathGrp == 0 & mathSel == 0 ~ 0))
table(WSCdata$ind_QED_math, useNA = "always")

# RCT vocab
WSCdata <- WSCdata %>%
 mutate(ind_rct_vocab = 1-ind_rct_math)

# ATT vocab
WSCdata <- WSCdata %>%
 mutate(ind_att_vocab = case_when(mathGrp == 0 & mathSel == 0 ~ 1,
                                  mathGrp == 1 & mathSel == 0 ~ 0))
table(WSCdata$ind_att_vocab, useNA = "always")
```

```
# ATU vocab
WSCdata <- WSCdata %>%
 mutate(ind_atu_vocab = case_when(mathGrp == 0 & mathSel == 1 ~ 1,
                                  mathGrp == 1 & mathSel == 1 ~ 0))
table(WSCdata$ind_atu_vocab, useNA = "always")

# QED vocab
WSCdata <- WSCdata %>%
 mutate(ind_QED_vocab = case_when(mathGrp == 0 & mathSel == 0 ~ 1,
                                  mathGrp == 1 & mathSel == 1 ~ 0))
table(WSCdata$ind_QED_vocab, useNA = "always")

# Baseline covariates list ----------------------------------------------------------------
# A list of covariates that will be used for further adjustment
cov_nms <- c(
"female",
"white",
"black",
"asian",
"hisp",
"married",
"logAge",
"income",
"collegeS",
"collegeM",
"collegeD",
"calc",
"logBooks",
"mathLike",
"big5O",
"big5C",
"big5E",
"big5A",
"big5N",
"AMAS",
"logBDI",
"MCS",
"GSES",
"vocabPre",
"mathPre"
)

# Differences in means ------------------------------------------------
# Experimentally identified estimates for ATE for math
fmla_unadj_ate_math <- as.formula(paste("mathPost ~ ind_rct_math"))
summary(lm(fmla_unadj_ate_math, data = WSCdata))

# Experimentally identified estimates for ATT for math
fmla_unadj_att_math <- as.formula(paste("mathPost ~ ind_att_math"))
summary(lm(fmla_unadj_att_math, data = WSCdata))

# Experimentally identified estimates for ATU for math
```

```
fmla_unadj_atu_math <- as.formula(paste("mathPost ~ ind_atu_math"))
summary(lm(fmla_unadj_atu_math, data = WSCdata))

# Experimentally identified estimates for ATE for vocabulary
fmla_unadj_ate_vocab <- as.formula(paste("vocabPost ~ ind_rct_vocab"))
summary(lm(fmla_unadj_ate_vocab, data = WSCdata))

# Experimentally identified estimates for ATT for vocabulary
fmla_unadj_att_vocab <- as.formula(paste("vocabPost ~ ind_att_vocab"))
summary(lm(fmla_unadj_att_vocab, data = WSCdata))

# Experimentally identified estimates for ATU for vocabulary
fmla_unadj_atu_vocab <- as.formula(paste("vocabPost ~ ind_atu_vocab"))
summary(lm(fmla_unadj_atu_vocab, data = WSCdata))

# ANCOVA ------------------------------------------------------------------
# Covariates adjusted estimates for ATE for math
fmla_ancova_ate_math <-
 as.formula(paste("mathPost ~ ind_rct_math + ", paste(cov_nms, collapse = " + ")))
summary(lm(fmla_ancova_ate_math, data = WSCdata))

# Covariates adjusted estimates for ATT for math
fmla_ancova_att_math <-
 as.formula(paste("mathPost ~ ind_att_math + ", paste(cov_nms, collapse = " + ")))
summary(lm(fmla_ancova_att_math, data = WSCdata))

# Covariates adjusted estimates for ATU for math
fmla_ancova_atu_math <-
 as.formula(paste("mathPost ~ ind_atu_math + ", paste(cov_nms, collapse = " + ")))
summary(lm(fmla_ancova_atu_math, data = WSCdata))

# Covariates adjusted estimates for ATE for vocabulary
fmla_ancova_ate_vocab <-
 as.formula(paste("vocabPost ~ ind_rct_vocab + ", paste(cov_nms, collapse = " + ")))
summary(lm(fmla_ancova_ate_vocab, data = WSCdata))

# Covariates adjusted estimates for ATT for math
fmla_ancova_att_vocab <-
 as.formula(paste("mathPost ~ ind_att_vocab + ", paste(cov_nms, collapse = " + ")))
summary(lm(fmla_ancova_att_vocab, data = WSCdata))

# Covariates adjusted estimates for ATU for math
fmla_ancova_atu_vocab <-
 as.formula(paste("mathPost ~ ind_atu_vocab + ", paste(cov_nms, collapse = " + ")))
summary(lm(fmla_ancova_atu_vocab, data = WSCdata))

# Propensity Score Overlaps -----------------------------------------------------------
# Propensity Score overlap plot for RCT math training propensity
## Define the formula for the propensity score model for math training propensity
fmla_ps_rct_math <-
  as.formula(paste("ind_rct_math ~ ", paste(cov_nms, collapse = " + ")))

## Fit a logistic regression model to predict propensity scores
```

```
ps_rct_math <- predict(glm(
  formula = fmla_ps_rct_math,
  family = "binomial",
  data = WSCdata
), type = "response")

## Merge propensity scores to the original dataset
lps_rct_math <- data.frame(cbind(lps = log(ps_rct_math), ind_rct_math = WSCdata$ind_rct_math))

## Create an overlap density plot based on log transformed propensity scores for
## treatment and control group
lps_rct_math |>
  mutate(ind_rct_math_fct = case_when(ind_rct_math == 1 ~ "treatment",
                                ind_rct_math == 0 ~ "control")) |>
  ggplot(aes(x = lps, fill = ind_rct_math_fct)) + geom_density(alpha = 0.25) +
  xlab("Log Propensity Score") +
  ylab("Density") +
  ggtitle("Propensity score overlap for math training in RCT groups") +
  guides(fill=guide_legend(title="RCT group"))

# Propensity Score overlap plot for math training propensity among the treated group
## Define the formula for the propensity score model for math training propensity
## among the treated group
fmla_ps_att_math <-
  as.formula(paste("ind_att_math ~ ", paste(cov_nms, collapse = " + ")))

## Fit a logistic regression model to predict propensity scores
ps_att_math <- predict(glm(
  formula = fmla_ps_att_math,
  family = "binomial",
  data = WSCdata
), type = "response")

## Merge propensity scores to the original dataset
lps_att_math <- data.frame(cbind(lps = log(ps_att_math), ind_att_math = WSCdata$ind_att_math))

## Create an overlap density plot based on log transformed propensity scores for
## treatment and control group
lps_att_math |>
  filter(!is.na(ind_att_math)) |>
  mutate(ind_att_math_fct = case_when(ind_att_math == 1 ~ "treatment",
                                ind_att_math == 0 ~ "control")) |>
  ggplot(aes(x = lps, fill = ind_att_math_fct)) + geom_density(alpha = 0.25) +
  xlab("Log Propensity Score") +
  ylab("Density") +
 ggtitle("Propensity score overlap for math training in treatment groups among the treated") +
  guides(fill=guide_legend(title="Att group"))

# Propensity Score overlap plot for math training propensity among the control group
## Define the formula for the propensity score model for math training
## propensity among the control group
fmla_ps_atu_math <-
  as.formula(paste("ind_atu_math ~ ", paste(cov_nms, collapse = " + ")))
```

```
## Fit a logistic regression model to predict propensity scores
ps_atu_math <- predict(glm(
  formula = fmla_ps_atu_math,
  family = "binomial",
  data = WSCdata
), type = "response")

## Merge propensity scores to the original dataset
lps_atu_math <- data.frame(cbind(lps = log(ps_atu_math), ind_atu_math = WSCdata$ind_atu_math))

## Create an overlap density plot based on log transformed propensity scores for
## treatment and control group
lps_atu_math |>
  filter(!is.na(ind_atu_math)) |>
  mutate(ind_atu_math_fct = case_when(ind_atu_math == 1 ~ "treatment",
                                ind_atu_math == 0 ~ "control")) |>
  ggplot(aes(x = lps, fill = ind_atu_math_fct)) + geom_density(alpha = 0.25) +
  xlab("Log Propensity Score") +
  ylab("Density") +
 ggtitle("Propensity score overlap for math training in treatment groups among the control") +
  guides(fill=guide_legend(title="ATU group"))

# Propensity Score overlap plot for RCT vocabulary training propensity
## Define the formula for the propensity score model for vocabulary training propensity
fmla_ps_rct_vocab <-
  as.formula(paste("ind_rct_vocab ~ ", paste(cov_nms, collapse = " + ")))

## Fit a logistic regression model to predict propensity scores
ps_rct_vocab <- predict(glm(
  formula = fmla_ps_rct_vocab,
  family = "binomial",
  data = WSCdata
), type = "response")

## Merge propensity scores to the original dataset
lps_rct_vocab <- data.frame(cbind(lps = log(ps_rct_vocab),
ind_rct_vocab = WSCdata$ind_rct_vocab))

## Create an overlap density plot based on log transformed propensity scores for
## treatment and control group
lps_rct_vocab |>
  mutate(ind_rct_vocab_fct = case_when(ind_rct_vocab == 1 ~ "treatment",
                                 ind_rct_vocab == 0 ~ "control")) |>
  ggplot(aes(x = lps, fill = ind_rct_vocab_fct)) + geom_density(alpha = 0.25) +
  xlab("Log Propensity Score") +
  ylab("Density") +
  ggtitle("Propensity score overlap for vocabulary training in RCT groups") +
  guides(fill=guide_legend(title="RCT group"))

# Propensity Score overlap plot for vocabulary training propensity among the treated group
## Define the formula for the propensity score model for vocabulary training
## propensity among the treated group
```

```
fmla_ps_att_vocab <-
  as.formula(paste("ind_att_vocab ~ ", paste(cov_nms, collapse = " + ")))

## Fit a logistic regression model to predict propensity scores
ps_att_vocab <- predict(glm(
  formula = fmla_ps_att_vocab,
  family = "binomial",
  data = WSCdata
), type = "response")

## Merge propensity scores to the original dataset
lps_att_vocab <- data.frame(cbind(lps = log(ps_att_vocab),
ind_att_vocab = WSCdata$ind_att_vocab))

## Create an overlap density plot based on log transformed propensity scores for
## treatment and control group
lps_att_vocab |>
  filter(!is.na(ind_att_vocab)) |>
  mutate(ind_att_vocab_fct = case_when(ind_att_vocab == 1 ~ "treatment",
                                ind_att_vocab == 0 ~ "control")) |>
  ggplot(aes(x = lps, fill = ind_att_vocab_fct)) + geom_density(alpha = 0.25) +
  xlab("Log Propensity Score") +
  ylab("Density") +
  ggtitle("Propensity score overlap for vocabulary training in
  treatment groups among the treated") +
  guides(fill=guide_legend(title="ATT group"))

# Propensity Score overlap plot for vocabulary training propensity among the control group
## Define the formula for the propensity score model for vocabulary training
## propensity among the control group
fmla_ps_atu_vocab <-
  as.formula(paste("ind_atu_vocab ~ ", paste(cov_nms, collapse = " + ")))

## Fit a logistic regression model to predict propensity scores
ps_atu_vocab <- predict(glm(
  formula = fmla_ps_atu_vocab,
  family = "binomial",
  data = WSCdata
), type = "response")

## Merge propensity scores to the original dataset
lps_atu_vocab <- data.frame(cbind(lps = log(ps_atu_vocab), ind_atu_vocab = WSCdata$ind_atu_vocab))

## Create an overlap density plot based on log transformed propensity scores
## for treatment and control group
lps_atu_vocab |>
  filter(!is.na(ind_atu_vocab)) |>
  mutate(ind_atu_vocab_fct = case_when(ind_atu_vocab == 1 ~ "treatment",
                                ind_atu_vocab == 0 ~ "control")) |>
  ggplot(aes(x = lps, fill = ind_atu_vocab_fct)) + geom_density(alpha = 0.25) +
  xlab("Log Propensity Score") +
  ylab("Density") +
 ggtitle("Propensity score overlap for vocabulary training in treatment groups among the contol") +
```

```
  guides(fill=guide_legend(title="ATU group"))
}
```

Index