Matched Cohort Tables

Shawn Garbett, Kyle Rove

2020-05-07

Matched Cohort

A matched cohort study utilizes specific summary statistics across the group. Recommendations are taken from the following references:

- Austin PC. A critical appraisal of propensity-score matching in the medical literature between 1996 and 2003. Statist Med 2008; 27: 2037–49. doi:10.1002/sim.3150.
- Fleiss JL, Levin B, Paik MC. Statistical Methods for Rates and Proportions (3rd edn). Wiley: New York, NY, 2003.

The goal is to be able to use tangram defaults as much as possible, while using the following tests.

For 1:1 matching

variables	statistical test	Notes
Numeric x Cat	paired Student's t	`t.test(x=covariate, y=arm, paired=TRUE)`
	Wilcoxon *(preferred)*	`wilcox.test(x=covariate, y=arm, paired=TRUE)`
	Cox proportional hazards models stratifying on matched groups	`survival::coxph(outcome ~ covariate + strata(block), data = m1.final)` useful for time to event analysis
Cat X Cat	McNemar's	`mcnemar.test(x=covariate, y=arm)` 2 x 2 cases only. Expects factors
	Stuart Maxwell chi-squared test	`DescTools::StuartMaxwellTest(x=covariate, y=arm)` for 2 x k polytomous covariates, where k >= 2 expects factors

For 1:many matching

variables	statistical test	Notes
Numeric x Cat	logistic regression with generalized estimated equations	`geepack::geeglm(formula = outcome ~ covariate, family = binomial("logit"), data = m2.final, id = block, corstr = "independence", zcor = "zcor")` outcome must be binary numeric (not a factor) covariate must be numeric block must be numeric (not a factor)
Cat x Cat	conditional logistic regression	`survival::clogit(outcome ~ covariate + strata(block), data = m2.final)`

variables	statistical test	Notes
		strata with only 1 occurence cause errors, there should be check for this covers 2 x 2 and 2 x >2 polytomous covariates

Create functions for matching testing

```
psm <- hmisc

psm[['Cell']][['fraction']] <- function(numerator, denominator, format=3, ...) {
  paste0(numerator,' (',render_f(100*numerator/denominator, format),'%)')
}

psm[['Footnote']] = paste("N is the number of non-missing value.",
  "^1^*t*-test.",
  "^2^Wilcoxon signed rank test.",
  "^3^Cox proportional hazards.",
  "^4^Logistic regression with GEE.",
  "^5^Conditional logistic regression.",
  "^6^McNemar's test.",
  "^7^Stuart Maxwell \u03a7^2^ test.",
  "^8^Cochran-Mantel-Haenszel \u03a7^2^ test."
)

mctest.numxcat <- function(rdata, cdata, cell_style, block, pref_test="default", ...)
{
  # get data
  covariate <- rdata
  outcome   <- cdata
  n_matched <- length(block) / length(levels(as.categorical(block)))

  # make the df and sort it
  df <- data.frame(
    covariate=as.numeric(covariate),
    outcome=as.numeric(levels(factor(outcome,levels=c(0,1))))[outcome],
                              block=as.numeric(block)
  ) %>% arrange(block,outcome)

  p_val <- NA
  ref   <- " "

  # first branch point is whether data is matched 1:1 or 1:many
  if (n_matched == 2) {

    # paired Student's t-test
    if (pref_test == "t.test") {
      # run test
      stat <- t.test(x = df$covariate[df$outcome == 0],
                     y = df$covariate[df$outcome == 1],
                     paired = TRUE)
      ref  <- "1"
```

```
      if (length(stat) > 1) p_val <- broom::tidy(stat)$p.value

    # Wilcoxon signed rank test
    } else if (pref_test == "default" || pref_test == "wilcox.test") {
      # run test
      stat <- wilcox.test(x = df$covariate[df$outcome == 0],
                          y = df$covariate[df$outcome == 1],
                          paired=TRUE)
      ref  <- "2"

      if (length(stat) > 1) p_val <- broom::tidy(stat)$p.value

    # Cox proportional hazards model stratefied on matched pairs
    } else if (pref_test == "coxph") {
      # run regression
      stat <- survival::coxph(outcome ~ covariate + strata(block), data = df)
      ref  <- "3"

      if (length(stat) > 1) p_val <- broom::tidy(stat)$p.value
    }

  } else if (n_matched > 2) {

    # logistic regression with generalized estimating equations
    if (pref_test == "default" || pref_test == "geeglm") {
      # run regression
      stat <- suppressWarnings(
        geepack::geeglm(formula = outcome ~ covariate,
                        family  = binomial("logit"),
                        data    = df,
                        id      = block,
                        corstr  = "independence",
                        zcor    = "zcor")
      )
      ref <- "4"

      if (length(stat) > 1) p_val <- broom::tidy(stat)$p.value[[1]]

    # conditional logistic regression
    } else if (pref_test == "clogit") {
      # run regression
      stat <- survival::clogit(outcome ~ covariate + strata(block), data = df)
      ref  <- "5"

      if (length(stat) > 1) p_val <- broom::tidy(stat)$p.value[[1]]
    }
  }

  paste0(cell_style[['p']](p = p_val), "^", ref, "^")
}

mctest.catxcat <- function(rdata, cdata, cell_style, block, ...)
{
```

```
covariate <- as.categorical(rdata)
outcome   <- as.categorical(cdata)
block     <- as.categorical(block)
n_matched <- length(block) / length(levels(as.categorical(block)))
grid          <- table(covariate, outcome, block, useNA="no")
validrow      <- which(!apply(grid,1,FUN = function(x){all(x == 0)}))
validcol      <- which(!apply(grid,2,FUN = function(x){all(x == 0)}))
validblocks   <- which(!apply(grid,3,FUN = function(x){all(x == 0)}))
invalidstatum <- which(apply(grid,1,FUN = function(x){sum(ifelse(x==1,1,0))})==1)

# make the df and sort it
df <- data.frame(var=as.numeric(covariate),
                 outcome=as.numeric(levels(factor(outcome,levels=c(0,1))))[outcome],
                 block=as.numeric(block) )
df <- df %>% arrange(block,outcome)

p_val <- NA
ref   <- " "

if (n_matched == 2 && length(levels(covariate)) == 2) {
  # McNemar's test

  # x and y must be equal length vectors and have same levels
  # also removed hard-coded outcome levels in cases of other 2-level variables are used
  # (e.g., "M","F" or 1,2 or "No","Yes")
  # code above was somehow stripping a level if the responses were only one level
  # each stats test could have exclusions based on expectations of the underlying
  # function with smarter error handling messages
  stat <- exact2x2::mcnemar.exact(
    x = factor(df$var[df$outcome == levels(outcome)[[1]]], levels = levels(covariate)),
    y = factor(df$var[df$outcome == levels(outcome)[[2]]], levels = levels(covariate)),
    conf.level=.95)
  ref  <- "6"

  # get p value
  if (length(stat) > 1) p_val <- stat$p.value

} else if (n_matched == 2 && length(levels(covariate)) > 2) {

  # Stuart Maxwell chi-squared test
  stat <- StuartMaxwellTest(x=df$var[df$outcome == 0], y=df$var[df$outcome == 1])
  ref  <- "7"

  # get p value
  if (length(stat) > 1) p_val <- broom::tidy(stat)$p.value[[1]]

} else if (n_matched > 2) {

  # Cochran-Mantel-Haenszel chi-squared test
  stat <- if(length(validrow)      < 2 ||
             length(validcol)      < 2 ||
             length(validblocks)   < 1 ||
             length(invalidstatum) > 0) NA else
```

```
              mantelhaen.test(covariate,outcome,block)
    ref  <- "8"

    # get p value
    if (length(stat) > 1) p_val <- broom::tidy(stat)$p.value
  }

  paste0(cell_style[['p']](p = p_val), "^", ref, "^")
}

mctest <- function(row, col, cell_style, block=NULL, ...)
{
  if(is.null(block)) stop("Block must be specified for matched cohort testing")

  if(is.numeric(row$data)  && is.categorical(col$data))
    return(mctest.numxcat(row$data, col$data, cell_style, block, ...))
  if(is.categorical(row$data) && is.categorical(col$data))
    return(mctest.catxcat(row$data, col$data, cell_style, block, ...))

  stop(paste("Unsupported comparison for", row$name, "x", col$name, "\nAppears to be",
    hmisc_data_type(row$data), "X", hmisc_data_type(column$data)))
}
```

Results

```
# 2-level categories are typically dependent variable in propensity score matching
tangram ( group
            ~ age[1]     # numeric
            + sex        # binary categorical
            + lang       # multi-level categorical
            + opioids[1] # numeric
            + los[1]     # numeric
            + nsaid      # binary categorical
            + neuro,     # multi-level categorical
        data      = m1.final,
        block     = m1.final$block,
        test      = mctest,
        transform = psm,
        digits    = 2,
        style     = "nejm",
        caption   = "Example 1:1 Matching")
```

Table 3: Example 1:1 Matching

Table 3: Example 1:1 Matching

	N	Historical	Prospective	Test Statistic
		(N=30)	(N=30)	
Age	60	8.8 **10.4** 11.7	8.9 **10.1** 11.2	P=0.245[2]
Gender : Female	60	22 (73.33%)	22 (73.33%)	P=1.000[6]
Language	60			P=0.630[7]
English		12 (40.00%)	8 (26.67%)	
Spanish		6 (20.00%)	7 (23.33%)	
French		6 (20.00%)	10 (33.33%)	
Other		6 (20.00%)	5 (16.67%)	
Opioids *mg/kg*	60	1.3 **2.4** 3.8	1.6 **2.8** 4.5	P=0.715[2]
Length of Stay *days*	60	21.0 **28.4** 47.2	18.5 **30.3** 50.2	P=0.839[2]
NSAID Given : Yes	60	23 (76.67%)	30 (100.00%)	P=1.000[6]
Neurological	60			P=0.635[7]
Cervical		7 (23.33%)	10 (33.33%)	
Thoracic		7 (23.33%)	6 (20.00%)	
Lumbar		2 (6.67%)	5 (16.67%)	
Sacral		9 (30.00%)	6 (20.00%)	
Other		5 (16.67%)	3 (10.00%)	

N is the number of non-missing value. [1]*t*-test. [2]Wilcoxon signed rank test. [3]Cox proportional hazards. [4]Logistic regression with GEE. [5]Conditional logistic regression. [6]McNemar's test. [7]Stuart Maxwell χ^2 test. [8]Cochran-Mantel-Haenszel χ^2 test.

```
# 2-level categories are typically dependent variable in propensity score matching
tangram ( group
            ~ age[1]     # numeric
            + sex        # binary categorical
            + lang       # multi-level categorical
            + opioids[1] # numeric
            + los[1]     # numeric
            + nsaid      # binary categorical
            + neuro,     # multi-level categorical
        data      = m2.final,
        block     = m2.final$block,
        test      = mctest,
        transform = psm,
        digits    = 2,
        style     = "nejm",
        caption   = "Match 1:k Example")
```

Table 4: Match 1:k Example

Table 4: Match 1:k Example

	N	Historical (N=60)	Prospective (N=30)	Test Statistic
Age	90	8.9 **10.2** 11.5	8.9 **10.1** 11.2	P=0.974[4]
Gender : Female	90	34 (56.67%)	22 (73.33%)	P=0.057[8]
Language	90			P=0.713[8]
English		19 (31.67%)	8 (26.67%)	
Spanish		13 (21.67%)	7 (23.33%)	
French		14 (23.33%)	10 (33.33%)	
Other		14 (23.33%)	5 (16.67%)	
Opioids *mg/kg*	90	1.6 **2.3** 3.5	1.6 **2.8** 4.5	P=0.002[4]
Length of Stay *days*	90	20.7 **27.3** 48.2	18.5 **30.3** 50.2	P=0.070[4]
NSAID Given : Yes	90	42 (70.00%)	30 (100.00%)	P=0.004[8]
Neurological	90			P=0.526[8]
Cervical		12 (20.00%)	10 (33.33%)	
Thoracic		11 (18.33%)	6 (20.00%)	
Lumbar		7 (11.67%)	5 (16.67%)	
Sacral		19 (31.67%)	6 (20.00%)	
Other		11 (18.33%)	3 (10.00%)	

N is the number of non-missing value. [1]*t*-test. [2]Wilcoxon signed rank test. [3]Cox proportional hazards. [4]Logistic regression with GEE. [5]Conditional logistic regression. [6]McNemar's test. [7]Stuart Maxwell χ^2 test. [8]Cochran-Mantel-Haenszel χ^2 test.