Final_BaselineBlocking

DI_LC_AZT

17/11/2019

```
library(blockTools)
library(data.table)
library(knitr)
library(kableExtra)
```

Import Data and Clean / Check

```
rawdata <- read.csv('FinalStudy-AllBaselineOutcomeMeasures.csv')
surveydata <- read.csv('FinalStudy-EnrollmentSurvey2-clean.csv')
#dim(rawdata)
```

Save data as data.tables

```
rawdata <- data.table(rawdata)
surveydata <- data.table(surveydata)
```

Create Baseline data

Average 3 days of baseline for each participant before merging with survey for blocking

```
# Calculate fractional meat consumption per day for each participant for each day of baseline
baseline <- rawdata[,fracmeatperday := NumMeatEO/TotalNumEO]

# Take mean over the 3 days of baseline - if any na's then average is only over 1 or 2 days
baseline[,meanmeat := mean(fracmeatperday,na.rm=TRUE),by=ParticipantId]

# Take only one line for each participant from their baseline data,
# to represent the average meat eaten in the 3 days of baseline.
baseline <- baseline[MeasureNum==1,c("ParticipantId","Name","PhoneNumber","meanmeat")]
```

- Merge dataframes on PhoneNumber as that is what we are assigning ID by

```
workingdt <- merge(baseline,surveydata,by='PhoneNumber')
```

Create Blocking and Treatment Variables

Add columns to represent the following blocking variables as integers (this is how blockTools likes the variables, not as factors):

- Meal Planning Status: (0:No, I do not meal plan for my household,1: Not Relevant, 2: Shared responsibility, 3: Yes Main Planner)
- Food Allergies present (1) or None (0)
- Proportion of meat meals (1: >0-0.25, 2: >0.25-0.5, 3: >0.5-0.75, 4: 0.75-1) from baseline (not including enrollment survey)

- Ketogenic diet (1 - yes, 0 all other)
- Vegetarian or Pescatarian Eater (1 - yes, 0 all other)
- Finish eating time (0: Empty, 1: 7 PM, 2: 8 PM, 3: 9 PM, 4: 10PM or later)

also create the treatment and control columns, and set all to 0

```
# Meal Planner responsibility integer variable
workingdt[mealplanner=="Shared responsibility for meal planning in my household",
          mealplan := 2]
workingdt[mealplanner=="Not relevant to me",mealplan := 1]
workingdt[mealplanner=="Yes, I am the main meal planner",mealplan := 3]
workingdt[mealplanner=="No, I do not meal plan for my household",mealplan := 0]
```

```
# Allergy
workingdt[,allergybinary:=0]
workingdt[allergy != 'None', allergybinary := 1]

# Proportion of meat from baseline
workingdt[,propmeatrange := 1] # set default value
workingdt[meanmeat > 0.25 & meanmeat <= 0.5, propmeatrange := 2]
workingdt[meanmeat > 0.5 & meanmeat <= 0.75, propmeatrange := 3]
workingdt[meanmeat > 0.75 & meanmeat <= 1.0, propmeatrange := 4]

# create keto and vegetarian columns
workingdt[,c('keto','vegetarian'):=0]

# binarize the Keto folks
workingdt[diet=="Ketogenic (Dukan, Atkins, Generalized Keto)",keto :=1]

# Binarize non-meat eaters
workingdt[eatertype=="Vegetarian" | eatertype == "Pescatarian", vegetarian:=1]

# Make time person is done eating into an integer variable
workingdt[,endtime:=0]
workingdt[doneeating=="10 PM or Later",endtime:=4]
workingdt[doneeating=="9:00 PM",endtime:=3]
workingdt[doneeating=="8:00 PM",endtime:=2]
workingdt[doneeating=="7:00 PM",endtime:=1]

# Add Treatment and Control columns
workingdt[,c('treatment','control'):= 0]

#kable(head(workingdt))
```

Create Blocking

using the 6 variables listed above and 2 treatment conditions; adding `PhoneNumber`, name from Baseline (`Name.x`) and `ParticipantID` as identifiers.

```
out <- block(workingdt,
             n.tr=2,
             id.vars=c('PhoneNumber','Name.x','ParticipantId'),
             block.vars=c("propmeatrange","allergybinary","mealplan",
```

```
                          "keto","vegetarian","endtime"),
            verbose=FALSE)
```

Resultant Blocks

From the above we can see the 2 Treatment conditions: 'Unit 1' and 'Unit 2' 41 blocks - the rows. So this ends up being a Matched Pair Design

Used the `blockTools` library, so citing the software:
Moore, Ryan T. and Keith Schnakenberg. "blockTools: Blocking, Assignment, and Diagnosing Interference in Randomized Experiments", Version 0.6-3, December 2016.

and the papers:

Moore, Ryan T. "Multivariate Continuous Blocking to Improve Political Science Experiments". Political Analysis, 20(4):460–479, Autumn 2012.

Moore, Ryan T. and Sally A. Moore. "Blocking for Sequential Political Experiments". Political Analysis, 21(4):507–523, 2013.

Now assign these individuals to the 2 treatment conditions in the workingdata table at random.

```
assigned <- assignment(out, namesCol=c('control','treatment'),seed=42)
outCSV(assigned)
```

Set this data back into the working data table - can't figure out how to get it out of the blocktools assignment object, so just sent to CSV and remiported it.

```
randomized <- data.table(read.csv('Group1.csv'))
```

```
# Set the Control or Treatment indicator based on the randomized
# assignment within each block
workingdt[PhoneNumber %in% randomized[,control], control:=1]
workingdt[PhoneNumber %in% randomized[,treatment], treatment:=1]

# Create a column to carry the BlockNo with each individual
workingdt[,BlockNo:=0]

# Add the subject's block number to their data
blocks <- dim(randomized)[1]
for (i in 1:blocks){
  workingdt[PhoneNumber %in% randomized[X==i],BlockNo := i]
}
```

Treatment Assignments for Experiment

Extract to send to Diana for administration of treatment

Only pull out columns that are useful for treatment application

```
assignments <- workingdt[,c("Name.x","PhoneNumber","ParticipantId",
                            "treatment","control","BlockNo"),with=FALSE]
#kable(head(assignments))
```

Export to CSV

```
write.csv(assignments,file='FinalTreatAssignments.csv', row.names = FALSE)
```