

# wx3\_analysis

Avery Katko

August 3, 2014

```
rm(list=ls())
library(ggplot2)
library(reshape2)
library(plyr)
library(bootstrap)
setwd("~/github/local/pragmods")
#source("useful.R")
d <- read.csv("seq_data/pragmods_wx3.anondata.csv")
#head(d)
```

Exclude participants that either failed manipulation check or were rejected.

```
exclude <- d$assignmentstatus == "Rejected" |
  d$Answer.name_check_correct == "\"FALSE\""
sum(exclude)
```

```
## [1] 3
```

```
mean(exclude)
```

```
## [1] 0.06
```

```
d <- subset(d, exclude == FALSE)

d$Answer.choice_correct_1 <- factor(as.logical(d$Answer.choice_correct_1))
d$Answer.choice_correct_2 <- factor(as.logical(d$Answer.choice_correct_2))
d$Answer.choice_correct_3 <- factor(as.logical(d$Answer.choice_correct_3))
d$Answer.choice_correct_4 <- factor(as.logical(d$Answer.choice_correct_4))
d$Answer.choice_correct_5 <- factor(as.logical(d$Answer.choice_correct_5))
d$Answer.choice_correct_6 <- factor(as.logical(d$Answer.choice_correct_6))
```

Restructure data to have trial variable. (working around mturk data submission limitations; can't safely submit arrays, so we have to use separate variables for each trial)

```
d$level_1 <- factor(rep(0,nrow(d)))
d$level_2 <- factor(rep(1,nrow(d)))
d$level_3 <- factor(rep(0,nrow(d)))
d$level_4 <- factor(rep(1,nrow(d)))
d$level_5 <- factor(rep(0,nrow(d)))
d$level_6 <- factor(rep(1,nrow(d)))

trial.df <- function(tn){
  df <- data.frame(id = d$workerid,
    seqCond = d$Answer.sequence_condition,
    trial = factor(rep(tn,nrow(d))),
```

```

    level = d[sprintf("level_%d",tn)],
    item = d[sprintf("Answer.item_%d",tn)],
    targetProp = d[sprintf("Answer.target_prop_%d",tn)],
    distractorProp = d[sprintf("Answer.distractor_prop_%d",tn)],
    targetPosition = d[sprintf("Answer.target_position_%d",tn)],
    distractorPosition = d[sprintf("Answer.distractor_position_%d",tn)],
    choice = d[sprintf("Answer.choice_%d",tn)],
    choiceCorrect = d[sprintf("Answer.choice_correct_%d",tn)])
names(df) <- c("id","seqCond","trial","level","item","targetProp","distractorProp",
              "targetPosition","distractorPosition","choice","choiceCorrect")
return(df)
}

d2 <- rbind(trial.df(1),trial.df(2),trial.df(3),trial.df(4),trial.df(5),trial.df(6))
summary(d2)

```

```

##          id          seqCond    trial  level          item
##  Min.   : 1.0    "(0w1w)x3":282  1:47   0:141    "boat"          :66
##  1st Qu.:12.0                                2:47   1:141    "Christmas tree":54
##  Median :24.0                                3:47                                "friend"        :30
##  Mean   :25.2                                4:47                                "pizza"         :54
##  3rd Qu.:39.0                                5:47                                "snowman"       :48
##  Max.   :50.0                                6:47                                "sundae"        :30
##
##          targetProp    distractorProp targetPosition distractorPosition
##  "motor"      : 24    "motor"      : 24    "left"      : 78    "left"      : 78
##  "mushrooms": 24    "mushrooms": 24    "middle":108    "middle":108
##  "sail"       : 24    "sail"       : 24    "right"     : 96    "right"     : 96
##  "hat"        : 21    "hat"        : 21
##  "cabin"      : 18    "cabin"      : 18
##  "lights"     : 18    "lights"     : 18
##  (Other)      :153    (Other)      :153
##          choice    choiceCorrect
##  "logical": 25    FALSE: 31
##  "target"  :251    TRUE :251
##  "foil"    : 6
##
##
##
##
##

```

```

#statistics for boolean factors; copied from useful.R, with a slightly different mean function to work
l.mean <- function(...){mean(as.logical(...))}
l.theta <- function(x,xdata,na.rm=T) {l.mean(xdata[x],na.rm=na.rm)}
l.ci.low <- function(x,na.rm=T) {
  l.mean(x,na.rm=na.rm) - quantile(bootstrap(1:length(x),1000,l.theta,x,na.rm=na.rm)$thetastar,.025,na.rm=na.rm)
l.ci.high <- function(x,na.rm=T) {
  quantile(bootstrap(1:length(x),1000,l.theta,x,na.rm=na.rm)$thetastar,.975,na.rm=na.rm) - l.mean(x,na.rm=na.rm)
}

ms <- aggregate(choiceCorrect ~ trial + level,data = d2,l.mean)
ms$cil <- aggregate(choiceCorrect ~ trial + level, data = d2, l.ci.low)$choiceCorrect
ms$cih <- aggregate(choiceCorrect ~ trial + level, data = d2, l.ci.high)$choiceCorrect

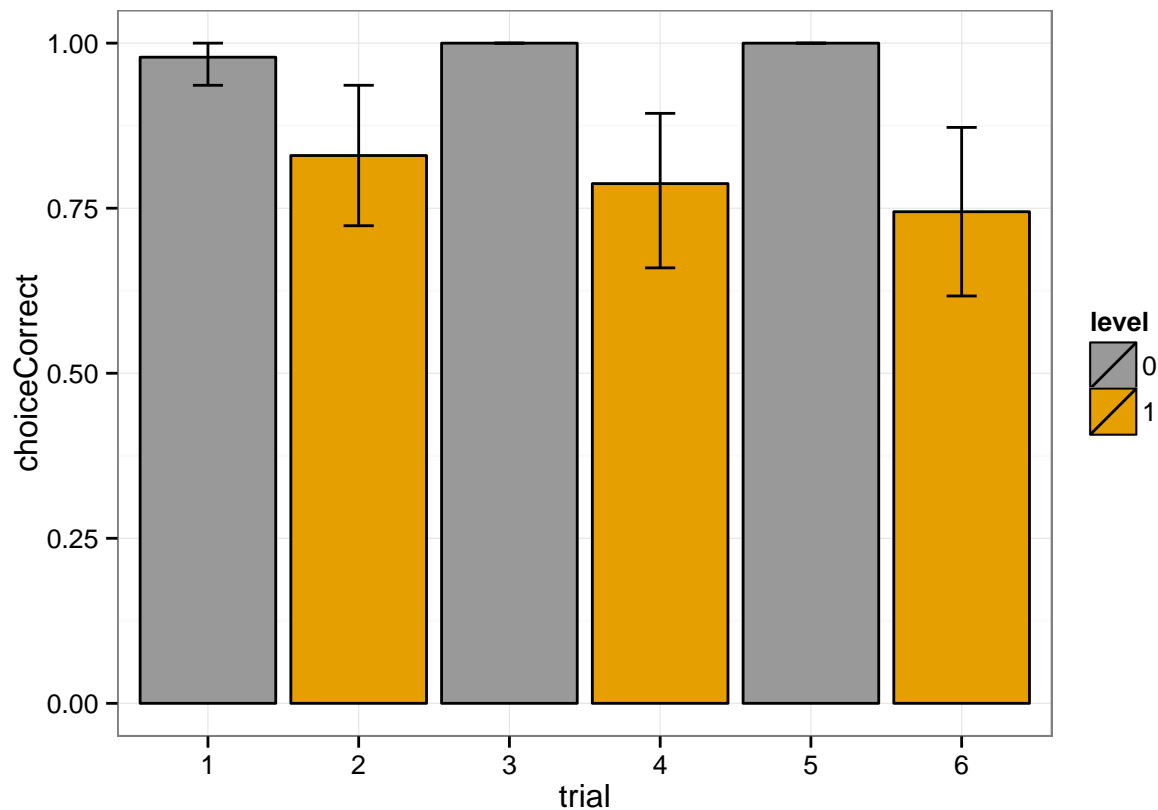
```

```
#colorblind-friendly color palettes
```

```
cbPalette <- c("#999999", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7")
cbbPalette <- c("#000000", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7")
```

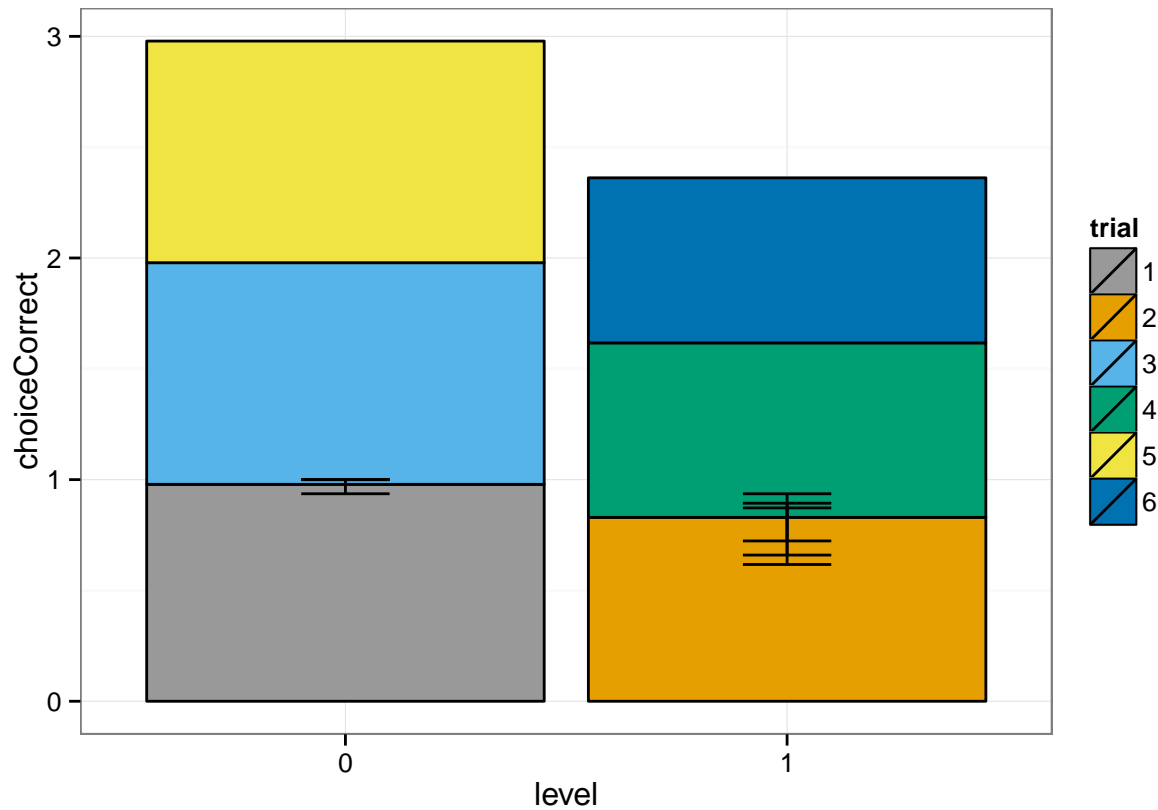
Compare performance on trials, in chronological order.

```
ggplot(data = ms, aes(x = trial, y = choiceCorrect, fill = level)) +
  geom_bar(stat = "identity", color = "black") +
  geom_errorbar(aes(ymin = choiceCorrect - cil, ymax = choiceCorrect + cih), width = 0.2) +
  #facet_grid(. ~ seqCond) +
  theme_bw() +
  scale_fill_manual(values=cbPalette)
```



Compare performance on level-0 vs level-1 trials.

```
ggplot(data = ms, aes(x = level, y = choiceCorrect, fill = trial)) +
  geom_bar(stat = "identity", color = "black") +
  geom_errorbar(aes(ymin = choiceCorrect - cil, ymax = choiceCorrect + cih), width = 0.2) +
  #facet_grid(. ~ seqCond) +
  theme_bw() +
  scale_fill_manual(values=cbPalette)
```



Chi-squared test between level-1 inferences (TODO...)

```
col_foil <- aggregate(choice ~ seqCond, data = d2[d2$level=="1",],
  function(...){sum(... == "\"foil\"")})

col_target <- aggregate(choice ~ seqCond, data = d2[d2$level=="1",],
  function(...){sum(... == "\"target\"")})

col_logical <- aggregate(choice ~ seqCond, data = d2[d2$level=="1",],
  function(...){sum(... == "\"logical\"")})

cont_table <- data.frame(cbind(col_foil,col_target[2],col_logical[2]))
names(cont_table) <- c("seqCond","foil","target","logical")

cont_table
```

```
##      seqCond foil target logical
## 1 "(0w1w)x3"   6   111      24
```

```
#chisq.test(cont_table[1:2,2:4])
```