

# combined\_analysis

Avery Katko

August 12, 2014

```
rm(list=ls())
library(ggplot2)
library(reshape2)
library(plyr)
library(bootstrap)
setwd("~/Documents/pragmods")
#source("useful.R")
d.seq <- read.csv("seq_data/pragmods_seq.anondata.csv")
d.wx3 <- read.csv("seq_data/pragmods_wx3.anondata.csv")
d.bx3 <- read.csv("seq_data/pragmods_bx3.anondata.csv")
d.seq2 <- read.csv("seq_data/pragmods_seq2.anondata.csv")
d.L2second <- read.csv("seq_data/pragmods_L2second.anondata.tsv", sep="\t")
raw.list <- list(d.seq, d.wx3, d.bx3, d.seq2, d.L2second)
```

Exclude participants that either failed manipulation check or were rejected.

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

exclude.wx3 <- d.wx3$assignmentstatus == "Rejected" |
  d.wx3$Answer.name_check_correct == "\"FALSE\""
d.wx3 <- subset(d.wx3, exclude.wx3 == FALSE)

exclude.bx3 <- d.bx3$assignmentstatus == "Rejected" |
  d.bx3$Answer.name_check_correct == "\"FALSE\""
d.bx3 <- subset(d.bx3, exclude.bx3 == FALSE)

exclude.seq2 <- d.seq2$assignmentstatus == "Rejected" |
  d.seq2$Answer.name_check_correct == "\"FALSE\""
d.seq2 <- subset(d.seq2, exclude.seq2 == FALSE)

exclude.L2second <- d.L2second$assignmentstatus == "Rejected" |
  d.L2second$Answer.name_check_correct == "\"FALSE\""
d.L2second <- subset(d.L2second, exclude.L2second == FALSE)
```

Choice correct as logical factors

```
d.seq$Answer.choice_correct_1 <- factor(as.logical(d.seq$Answer.choice_correct_1))
d.seq$Answer.choice_correct_2 <- factor(as.logical(d.seq$Answer.choice_correct_2))
d.seq$Answer.choice_correct_3 <- factor(as.logical(d.seq$Answer.choice_correct_3))

d.wx3$Answer.choice_correct_1 <- factor(as.logical(d.wx3$Answer.choice_correct_1))
d.wx3$Answer.choice_correct_2 <- factor(as.logical(d.wx3$Answer.choice_correct_2))
d.wx3$Answer.choice_correct_3 <- factor(as.logical(d.wx3$Answer.choice_correct_3))
d.wx3$Answer.choice_correct_4 <- factor(as.logical(d.wx3$Answer.choice_correct_4))
d.wx3$Answer.choice_correct_5 <- factor(as.logical(d.wx3$Answer.choice_correct_5))
```

```

d.wx3$Answer.choice_correct_6 <- factor(as.logical(d.wx3$Answer.choice_correct_6))

d.bx3$Answer.choice_correct_1 <- factor(as.logical(d.bx3$Answer.choice_correct_1))
d.bx3$Answer.choice_correct_2 <- factor(as.logical(d.bx3$Answer.choice_correct_2))
d.bx3$Answer.choice_correct_3 <- factor(as.logical(d.bx3$Answer.choice_correct_3))
d.bx3$Answer.choice_correct_4 <- factor(as.logical(d.bx3$Answer.choice_correct_4))
d.bx3$Answer.choice_correct_5 <- factor(as.logical(d.bx3$Answer.choice_correct_5))
d.bx3$Answer.choice_correct_6 <- factor(as.logical(d.bx3$Answer.choice_correct_6))

d.seq2$Answer.choice_correct_1 <- factor(as.logical(d.seq2$Answer.choice_correct_1))
d.seq2$Answer.choice_correct_2 <- factor(as.logical(d.seq2$Answer.choice_correct_2))
d.seq2$Answer.choice_correct_3 <- factor(as.logical(d.seq2$Answer.choice_correct_3))

d.L2second$Answer.choice_correct_1 <- factor(as.logical(d.L2second$Answer.choice_correct_1))
d.L2second$Answer.choice_correct_2 <- factor(as.logical(d.L2second$Answer.choice_correct_2))
d.L2second$Answer.choice_correct_3 <- factor(as.logical(d.L2second$Answer.choice_correct_3))

```

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

```

d.seq$level_1 <- factor(ifelse(d.seq$Answer.sequence_condition == "\"1w0w1b\"",1,0))
d.seq$level_2 <- factor(ifelse(d.seq$Answer.sequence_condition == "\"1w0w1b\"",0,1))
d.seq$level_3 <- factor(rep(1,nrow(d.seq)))

d.wx3$level_1 <- factor(rep(0,nrow(d.wx3)))
d.wx3$level_2 <- factor(rep(1,nrow(d.wx3)))
d.wx3$level_3 <- factor(rep(0,nrow(d.wx3)))
d.wx3$level_4 <- factor(rep(1,nrow(d.wx3)))
d.wx3$level_5 <- factor(rep(0,nrow(d.wx3)))
d.wx3$level_6 <- factor(rep(1,nrow(d.wx3)))

d.bx3$level_1 <- factor(rep(0,nrow(d.bx3)))
d.bx3$level_2 <- factor(rep(1,nrow(d.bx3)))
d.bx3$level_3 <- factor(rep(0,nrow(d.bx3)))
d.bx3$level_4 <- factor(rep(1,nrow(d.bx3)))
d.bx3$level_5 <- factor(rep(0,nrow(d.bx3)))
d.bx3$level_6 <- factor(rep(1,nrow(d.bx3)))

d.seq2$level_1 <- factor(ifelse(d.seq2$Answer.sequence_condition == "\"0w1w2w\"",0,2))
d.seq2$level_2 <- factor(rep(1,nrow(d.seq2)))
d.seq2$level_3 <- factor(ifelse(d.seq2$Answer.sequence_condition == "\"0w1w2w\"",2,0))

d.L2second$level_1 <- factor(ifelse(d.L2second$Answer.sequence_condition == "\"0w2w1w\"",0,1))
d.L2second$level_2 <- factor(rep(2,nrow(d.L2second)))
d.L2second$level_3 <- factor(ifelse(d.L2second$Answer.sequence_condition == "\"0w2w1w\"",1,0))

trial.df <- function(d,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)
}

df <- rbind(trial.df(d.seq,1),trial.df(d.seq,2),trial.df(d.seq,3),
            trial.df(d.wx3,1),trial.df(d.wx3,2),trial.df(d.wx3,3),trial.df(d.wx3,4),trial.df(d.wx3,5),t
            trial.df(d.bx3,1),trial.df(d.bx3,2),trial.df(d.bx3,3),trial.df(d.bx3,4),trial.df(d.bx3,5),t
            trial.df(d.seq2,1),trial.df(d.seq2,2),trial.df(d.seq2,3),
            trial.df(d.L2second,1),trial.df(d.L2second,2),trial.df(d.L2second,3))

df$seqCondAndTrial <- sprintf("%s[%s]",df$seqCond,df$trial)

summary(df)

```

```

##          id          seqCond   trial   level          item
## Min.      : 1.0    "1w0w1b" :333   1:477   0:663    "boat"          :343
## 1st Qu.: 21.0    "(0w1w)x3":282   2:477   1:854   "Christmas tree":283
## Median : 44.0    "(0b1b)x3":276   3:477   2:193   "friend"         :292
## Mean      : 58.9  "0w1w1b" :240    4: 93    "pizza"          :283
## 3rd Qu.: 85.0    "2w1w0w" :159    5: 93    "snowman"        :261
## Max.      :200.0  "1w2w0w" :156    6: 93    "sundae"         :248
##              (Other) :264
##          targetProp    distractorProp targetPosition distractorPosition
## "hat"      :183    "hat"      :185    "left" :556    "left" :554
## "sail"      :118    "motor"   :115    "middle":593   "middle":583
## "cabin"     :115    "cabin"   :114    "right" :561   "right" :573
## "motor"     :110    "sail"     :114
## "glasses"   :101    "glasses" :102
## "ornaments":100    "mushrooms": 97
## (Other)     :983    (Other)    :983
##          choice    choiceCorrect seqCondAndTrial
## "foil"      : 54    FALSE: 282    Length:1710
## "logical": 228    TRUE :1428    Class :character
## "target" :1428      Mode :character
##
##
##
##

```

```

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

ms2 <- aggregate(choiceCorrect ~ seqCondAndTrial + seqCond + level, data = df, l.mean)
ms2$cil <- aggregate(choiceCorrect ~ seqCondAndTrial + level, data = df, l.ci.low)$choiceCorrect
ms2$cih <- aggregate(choiceCorrect ~ seqCondAndTrial + level, data = df, l.ci.high)$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")

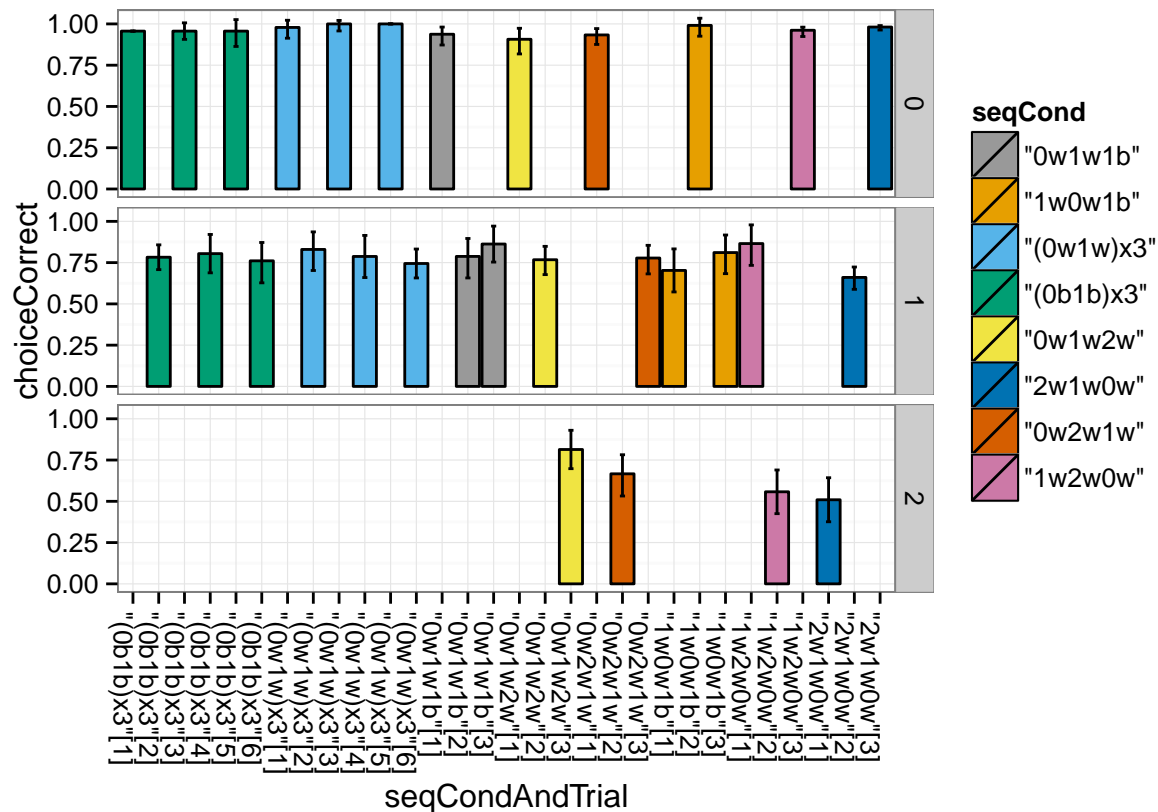
```

Compare performance in each condition for each inference level

```

ggplot(data = ms2, aes(x = seqCondAndTrial, y = choiceCorrect, fill = seqCond)) +
  geom_bar(stat = "identity", color = "black") +
  geom_errorbar(aes(ymin = choiceCorrect - cil, ymax = choiceCorrect + cih), width = 0.2) +
  facet_grid(level ~ .) +
  theme_bw() +
  scale_fill_manual(values=cbPalette) +
  theme(axis.text.x=element_text(angle = -90, hjust = 0))

```



Compare level 2 expts

```

ggplot(data = subset(ms2, grepl("2w", seqCond)), aes(x = seqCond, y = choiceCorrect, fill = seqCond)) +
  geom_bar(stat = "identity", color = "black") +
  geom_errorbar(aes(ymin = choiceCorrect - cil, ymax = choiceCorrect + cih), width = 0.2) +

```

```

facet_grid(. ~ level) +
theme_bw() +
scale_fill_manual(values=cbPalette) +
theme(axis.text.x=element_text(angle = -45, hjust = 0))

```

