Still suspicious: The suspicious coincidence effect revisited

Supplementary Information

Molly Lewis and Michael C. Frank

2018-02-09

Contents

This document was created from an R markdown file. The manuscript itself was also produced from an R markdown file, and all analyses presented in the paper can be reproduced from that document (https://github.com/mllewis/XTMEM/blob/master/paper/xtmem.Rmd).

The SI can be viewed interactively online at https://mlewis.shinyapps.io/xtmem_SI/

View experiments

To directly view an experiment, select an experiment from the dropdown menu, and click the "View Experiment" button. The experiment will open in a new window. [see online version for this content]

Power calculation

To determine the sample size in our experiments, we conducted a power calculation using an effect size estimated from previous experiments that used the same design parameters as the original Xu and Tenenbaum studies (2007a; XT). In total, there were three studies that satisfied this criterion (two from XT and one from Spencer, Perone, Smith & Samuelson, 2011; SPSS). For these three studies, we calculate the effect size for the basic level selections in the one versus three subordinate example conditions (the critical comparision).

Presented below are means and standard deviations of proportion selections of basic level exemplars from all three experiments and their corresponding sample sizes. Note that we do not have SDs for the XT data and so we estimate these values to be the same as the SPSS experiment.

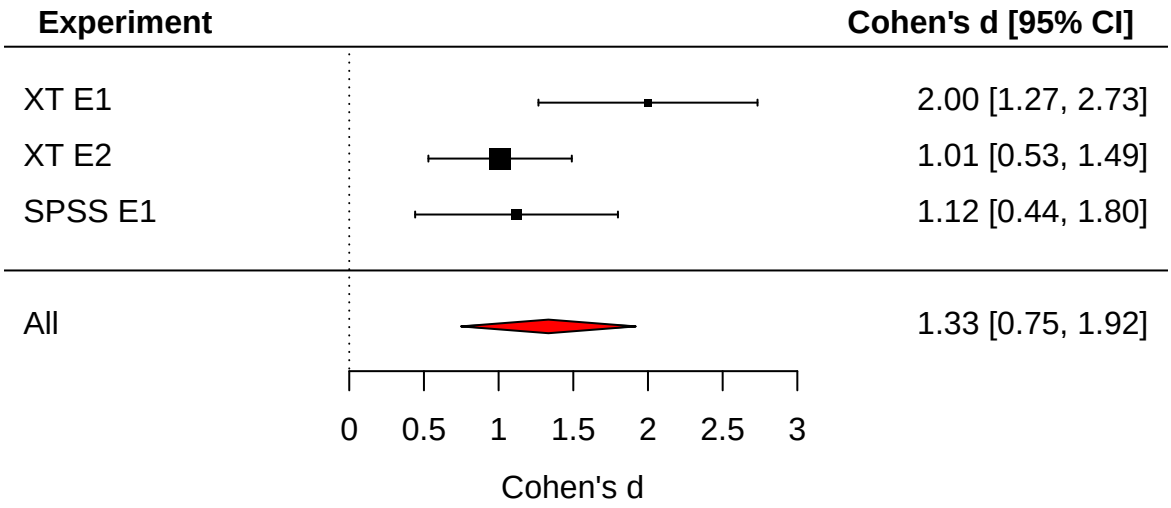
```
df <- data.frame(exp = c("XT E1", "XT E2", "SPSS E1"),
                 one_means = c( 76, 40, 48.24),
                 three_means = c(9, 6, 10.53),
                 one_sd = c(40.40, 40.40, 40.40),
                 three_sd = c(24.97, 24.97, 24.97),
                 n = c(22, 36, 19))
kable(df , col.names = c("Exp.",    "one-exmplar means",    "three-exemplar means",
                         "one-exemplar sd",     "three-exmplar sd", "sample size"))
```

Exp.	one-exmplar means	three-exemplar means	one-exemplar sd	three-exmplar sd	sample size
XT E1	76.00	9.00	40.4	24.97	22
XT E2	40.00	6.00	40.4	24.97	36
SPSS E1	48.24	10.53	40.4	24.97	19

```
df$d = mes(df$one_means, df$three_means, df$one_sd,
           df$three_sd, df$n, df$n, verbose = F)$d

df$d_var = mes(df$one_means, df$three_means, df$one_sd,
               df$three_sd, df$n, df$n, verbose = F)$var.d

model = rma(d, d_var, data = df)
```

Using a random effect meta-analytic model, we use these values to obtain a meta-analytic effect size estimate. We estimate this effect size to be 1.33.

```
par(cex = 1, font = 1)
forest(model,
       slab = df$exp,
       mlab = "All",
       xlab = "Cohen's d")
par(font = 2)
text(-1.5, 4.4, "Experiment")
text(4.3, 4.4, "Cohen's d [95% CI]")

addpoly(model, row = -1, cex = .75,
        annotate = F,  col = "red", mlab = "", efac = 2)
```



We then use this value to calculate the necessary sample size for a two-sample t-test with a power of .99 and a significance level of .05.

```
DESIRED_POWER <- .99

p.t.two <- pwr.t.test(d = model$b,
                      power = DESIRED_POWER,
                      type = "two.sample",
                      alternative = "two.sided")

p.t.two
```

```
##
##      Two-sample t test power calculation
##
##               n = 21.68692
##               d = 1.333243
##       sig.level = 0.05
##           power = 0.99
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

For nearly perfect power, we would need 22 independent observations. To be conservative, we set our sample size at N = 50.

Effect size calculation

The classical Cohen's *d* measure was originally developed for between-subject designs and, as such, researchers have adapted the measure to within-subject designs in a variety of ways (http://jakewestfall.org/blog/index.php/category/effect-size/). We calculate our effect sizes using the "classic" Cohen's *d* formula, which takes the mean difference between conditions divided by the pooled standard deviation. Note that because this method does not take into account the fact that the means are within-subject, these are conservative estimates of effect size (since within-subject designs have more power). We use the `mes` function from the `compute.es` package (AC Del Re, 2013) to calculate our effect sizes

Here is an example calculation of the effect size for Exp. 1. We first get the means and variances across participants of the proportion basic selections for the 1 subordinate and 3 subordinate conditions.

```r
# this data has been pre-processed with analysis/munge_anonymize_data.R script
all_d <- read_csv("data/anonymized_data/all_data_munged_A.csv") %>%
  mutate(condition = fct_recode(condition,
                                "1 sub." = "one",
                                "3 basic"= "three_basic",
                                "3 sub." = "three_subordinate",
                                "3 super." = "three_superordinate",
                                "3 basic" = "3bas",
                                "3 super." = "3sup",
                                "3 sub." = "3sub"),
         condition = fct_relevel(condition, "1 sub.", "3 sub.", "3 basic", "3 super."))
# there were 28 trials across all 12 experiments (.4%) in which there was an error in data recording su

## key to experiment factors
exp_key <- read_csv("data/experiment_key.csv") %>%
  mutate(order = gsub("\"", "", order),
         exp = as.integer(exp)
  ) %>%
  select(-preregistered)
```

```r
es_1_calc <- all_d %>%
  left_join(exp_key %>% select(exp, exp_recoded)) %>%
  filter(exp_recoded == 1) %>% # we only want exp 1
  filter(condition == "1 sub." | condition == "3 sub.") %>%
  # we only care about these conds. for calculating d
  gather(variable, value, c(prop_sub, prop_bas, prop_sup)) %>%
  filter(variable == "prop_bas") %>%
  # we only care about this DV for calculating d
  group_by(condition, subids) %>%
  summarize(value = mean(value)) %>%
  # get the mean for each subjects across trials
  group_by(condition) %>%
  summarize(mean_prop_bas = mean(value),
            var_prop_bas = var(value))
# get the mean for each condition acros subjects


kable(es_1_calc, digits = 2, col.names = c("Condition", "Mean", "Var"))
```

Condition	Mean	Var
1 sub.	0.62	0.16
3 sub.	0.17	0.09

We then calculate Cohen's d as follows:

$$\begin{aligned} d &= \frac{M_1 - M_2}{\sigma_{pooled}} \\ &= \frac{M_{1sub} - M_{3sub}}{\sqrt{(\frac{var_{1sub} + var_{3sub}}{2})}} \\ &= \frac{.64 - .18}{\sqrt{(\frac{.15 + .09}{2})}} \\ &\approx 1.32 \end{aligned}$$

For Exp. 1, we calculate Cohen's $d = 1.32$.

Results for all conditions and measures

In the Main Text, we report the proportion basic level selections for two training conditions, one-subordinate and three-subordinate. Here we report the data for all four conditions and all three dependent measures (proportion basic level, subordinate level, and superordinate level selections). [see online version for this content]

By category analyses

In the Main Text, we report our analyses collapsed across all three stimulus categories (animals, vehicles and vegetables). Here we present the effect sizes for each experiment separately for the different stimulus categories. While there is some variability in effect size by category (the effect is generally larger for animals), this variability is small relative to the effect of condition order.

```
# remap condition values and select relevant conditions
all_d_clean <- all_d %>%
  mutate(condition = as.factor(condition)) %>%
  filter(condition == "1 sub." | condition == "3 sub.") %>%
  select(exp, everything())

all_ms_subj_cat <- all_d_clean %>%
  left_join(exp_key %>% select(exp, exp_recoded)) %>%
  select(-exp) %>%
  gather(variable, value, c(prop_sub, prop_bas, prop_sup))  %>%
  group_by(condition, category, variable, exp_recoded, subids) %>%
  mutate(value = as.numeric(value)) %>%
  summarize(value = mean(value)) %>%
  filter(variable == 'prop_bas') %>%
  spread(condition, value) %>%
  ungroup()  %>%
  select(-variable)

LF_means_cat <- all_ms_subj_cat %>%
  group_by(exp_recoded, category) %>%
  summarize(m_one = mean(`1 sub.`),
            sd_one = sd(`1 sub.`),
            m_3sub = mean(`3 sub.`),
            sd_3sub = sd(`3 sub.`),
```

```
            n = n())

LF_effect_sizes_cat <- LF_means_cat %>%
  ungroup() %>%
  do(data.frame(d = mes(.$m_one, .$m_3sub, .$sd_one,
                        .$sd_3sub, .$n, .$n, verbose = F)$d,
                d_var = mes(.$m_one, .$sd_3sub, .$sd_one,
                            .$sd_3sub, .$n, .$n, verbose = F)$var.d)) %>%
  mutate(high = d + (1.96*d_var),
         low = d - (1.96*d_var),
         es_type = "nonpaired",
         exp_recoded = LF_means_cat$exp_recoded,
         category = LF_means_cat$category)
```

Repeat participants excluded

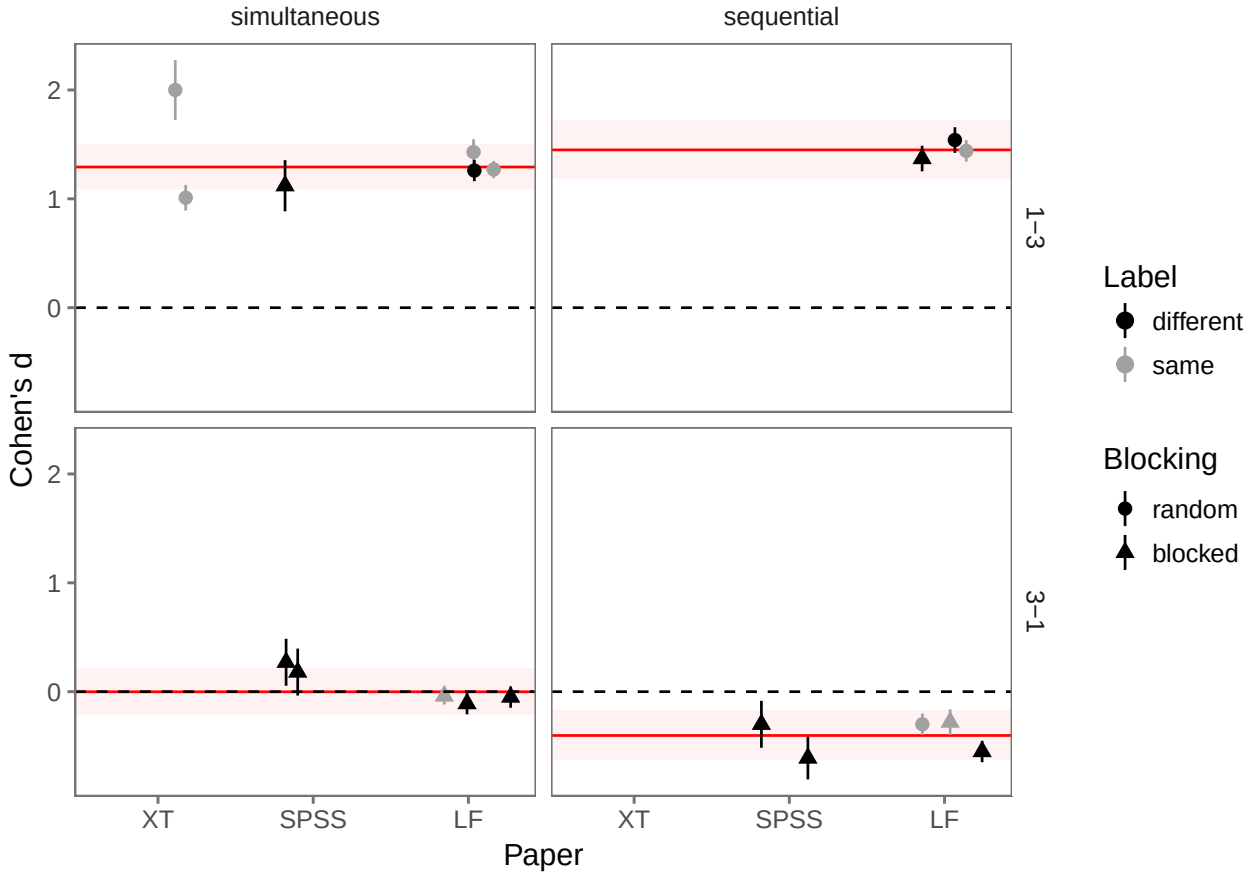
```
all_d_filtered <- read_csv("data/anonymized_data/no_dups_data_munged_A.csv")  %>%
  mutate(condition = as.factor(condition),
         condition = fct_recode(condition,
                                three_basic = "3bas",
                                three_subordinate = "3sub",
                                three_superordinate = "3sup")) %>%
  filter(condition == "one" | condition == "three_subordinate") %>%
  select(exp, everything())
# this file has been pre-processed with analysis/munge_anonymize_data_no_dups.R script

n_unique <- all_d_filtered %>%
  distinct(exp, subids) %>%
  summarize(n = n())

n_total <- all_d %>%
  distinct(exp, subids) %>%
  summarize(n = n())

percent_duplicates <- round((n_total-n_unique)/n_total, 2) * 100
```

13% of all participants (N = 600) completed more than one experiment. The data reported in the Main Text include all participants. Below we plot the effect sizes with participants excluded who had already participanted in a prior experiment (effect size estimates from XT and SPSS are also included for reference). The overall pattern looks the same as with all participants.

```
all_ms_subj <- all_d_filtered %>%
  left_join(exp_key %>% select(exp, exp_recoded)) %>%
  select(-exp) %>%
  gather(variable, value, c(prop_sub, prop_bas, prop_sup)) %>%
  group_by(condition,variable, exp_recoded, subids) %>%
  mutate(value = as.numeric(value)) %>%
  summarize(value = mean(value)) %>%
  filter(condition == "one" | condition == "three_subordinate",
         variable == 'prop_bas') %>%
  spread(condition, value)
```

```r
# means across participants (condition means)
LF_means_wide <- all_ms_subj %>%
  group_by(exp_recoded) %>%
  summarize(m_one = mean(one),
            sd_one = sd(one),
            m_3sub = mean(three_subordinate),
            sd_3sub = sd(three_subordinate),
            n = n())

LF_effect_sizes <- LF_means_wide %>%
  do(data.frame(d = compute.es::mes(.$m_one, .$m_3sub, .$sd_one,
                                    .$sd_3sub, .$n, .$n, verbose = F)$d,
                d_var = compute.es::mes(.$m_one, .$sd_3sub, .$sd_one,
                                        .$sd_3sub, .$n, .$n, verbose = F)$var.d)) %>%
  mutate(high = d + (1.96*d_var),
         low = d - (1.96*d_var),
         es_type = "nonpaired",
         exp_recoded = LF_means_wide$exp_recoded) %>%
  left_join(LF_means_wide %>% select(exp_recoded, n)) %>%
  select(exp_recoded, n, everything())

literature_effect_sizes <- read_csv("data/literature_ES.csv")
# see ../../analysis/get_literature_ES.R

all_es <- literature_effect_sizes %>%
  bind_rows(LF_effect_sizes) %>%
  left_join(exp_key) %>%
  mutate(source = ifelse(str_detect(exp_recoded, "XT"), "XT2007a",
                         ifelse(str_detect(exp_recoded, "SPSS"), "SPSS2011", "LF")),
         source = fct_relevel(source, "XT2007a","SPSS2011","LF"),
         source = as.numeric(source),
         timing = fct_relevel(timing,"simultaneous", "sequential"))

seq13 <- rma(d, d_var, dat = filter(all_es, timing == "sequential", order == "1-3"))
seq31 <- rma(d, d_var, dat = filter(all_es, timing == "sequential", order == "3-1"))
sim13 <- rma(d, d_var, dat = filter(all_es, timing == "simultaneous", order == "1-3"))
sim31 <- rma(d, d_var, dat = filter(all_es, timing == "simultaneous", order == "3-1"))

ma_es <- data.frame(order = c("1-3", "3-1", "1-3", "3-1"),
                    timing = c("sequential", "sequential", "simultaneous", "simultaneous"),
                    d = c(seq13$b[[1]], seq31$b[[1]],
                          sim13$b[[1]], sim31$b[[1]]),
                    d_low = c(seq13$ci.lb[[1]], seq31$ci.lb[[1]],
                              sim13$ci.lb[[1]], sim31$ci.lb[[1]]),
                    d_high = c(seq13$ci.ub[[1]], seq31$ci.ub[[1]],
                               sim13$ci.ub[[1]], sim31$ci.ub[[1]]))

ggplot(all_es) +
  geom_rect(aes(xmin = -Inf, xmax = Inf, ymin = d_low, ymax = d_high),
            fill = "red", alpha = 0.05, inherit.aes = FALSE, data = ma_es) +
  geom_hline(aes(yintercept = d), data = ma_es, color = "red") +
  scale_color_manual(values = c("black", "grey63")) +
  geom_pointrange(aes(x = jitter(source, 1.4),  y = d, ymax = high,
```

```
                    ymin = low, color = one_3sub_label, shape = fct_rev(blocking)),
                size = .5) +
  geom_hline(yintercept = 0, linetype = 2, color = "black") +
  facet_grid(order ~ timing) +
  scale_x_continuous(breaks = c(1:3), limits = c(.6, 3.3),
                     labels = c("XT","SPSS","LF")) +
  ylab("Cohen's d") +
  xlab("Paper") +
  guides(color = guide_legend("Label"),
         shape = guide_legend("Blocking"))  +
  ggthemes::theme_few()
```



Below is the meta-analytical model presented in the Main Text for the sample with repeat-participants excluded. The pattern is the same as for the full sample.

```
mod <- metafor::rma(d ~ timing + order + one_3sub_label + blocking, d_var, dat = all_es)

mod_df <- data.frame(fixed_effect = c("Intercept",
                                      "Simultaneous vs. sequential timing",
                                      "1-3 vs. 3-1 trial order",
                                      "Different vs. same label",
                                      "Blocked vs. pseudo-random trial structure"),
                     beta_string = paste0(round(mod$beta, 2),
                                          " [", round(mod$ci.lb, 2), ", "
                                          , round(mod$ci.ub,2) , "]"),
                     zval = mod$zval,
```

```
                      pval_string = round(mod$pval, 2)) %>%
  mutate(pval_string = ifelse(pval_string == 0, "<.0001", pval_string))

# MA model table

kable(mod_df, caption = "Meta-analytic model with manipulations as fixed effects.",
      align = c('l', 'r', 'r', 'r'), digits = 2,
      col.names = c("Fixed effect", "beta", "z-value", "p-value")) %>%
  kable_styling(font_size = 12)
```

Table 3: Meta-analytic model with manipulations as fixed effects.

Fixed effect	beta	z-value	p-value
Intercept	1.41 [1.09, 1.72]	8.70	<.0001
Simultaneous vs. sequential timing	-0.14 [-0.37, 0.09]	-1.22	0.22
1-3 vs. 3-1 trial order	-1.53 [-1.86, -1.2]	-9.10	<.0001
Different vs. same label	0 [-0.27, 0.27]	0.00	1
Blocked vs. pseudo-random trial structure	0 [-0.37, 0.36]	-0.02	0.98

Cross-category generalization trials excluded

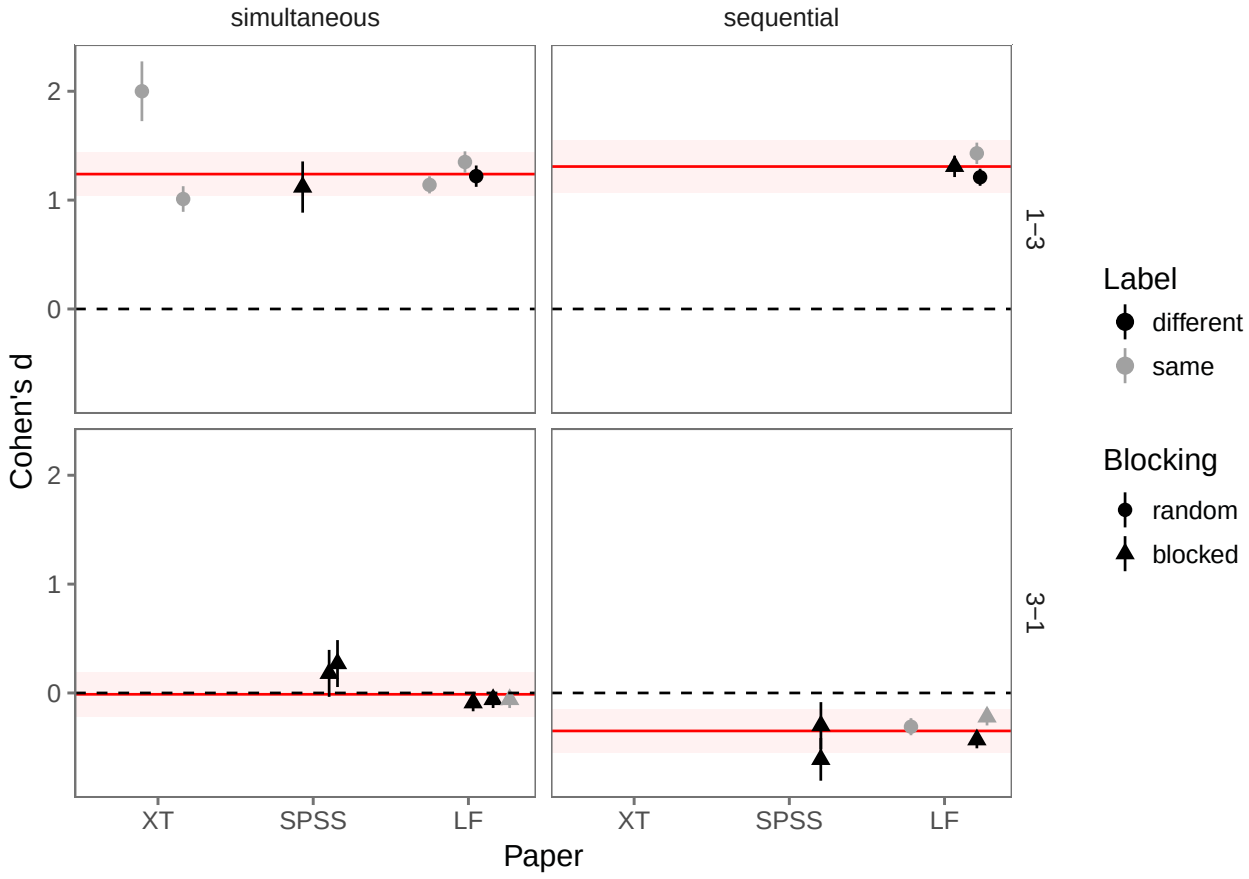
```
all_d <- read_csv("data/anonymized_data/all_data_munged_A.csv")  %>%
  mutate(condition = as.factor(condition),
         condition = fct_recode(condition,
                                three_basic = "3bas",
                                three_subordinate = "3sub",
                                three_superordinate = "3sup")) %>%
  filter(condition == "one" | condition == "three_subordinate") %>%
  select(exp, everything())

n_cross_category_generalizers <- all_d %>%
  filter(only_responded_with_target_category == "other") %>%
  nrow()

n_total <- nrow(all_d)

percent_cross_category <- round(n_cross_category_generalizers/n_total, 2) * 100
```

In 3% of trials, participants selected at least one exemplar from the non-target category (e.g., selected pepper after being trained on dogs). The data reported in the Main Text include all trials. Below we plot the effect sizes excluding trials with cross-category selections (effect size estimates from XT and SPSS are also included for reference). The overall pattern looks the same as with all trials

```
all_ms_subj <- all_d %>%
  filter(only_responded_with_target_category == "only_target") %>%
  left_join(exp_key %>% select(exp, exp_recoded)) %>%
  select(-exp) %>%
  gather(variable, value, c(prop_sub, prop_bas, prop_sup)) %>%
  group_by(condition,variable, exp_recoded, subids) %>%
  mutate(value = as.numeric(value)) %>%
  summarize(value = mean(value)) %>%
```

```
  filter(condition == "one" | condition == "three_subordinate",
         variable == 'prop_bas') %>%
  spread(condition, value) %>%
  filter(!is.na(one) & !is.na(three_subordinate))
# exclude participants where one of the two condition is missing

# means across participants (condition means)
LF_means_wide <- all_ms_subj %>%
  group_by(exp_recoded) %>%
  summarize(m_one = mean(one),
            sd_one = sd(one),
            m_3sub = mean(three_subordinate),
            sd_3sub = sd(three_subordinate),
            n = n())

LF_effect_sizes <- LF_means_wide %>%
  do(data.frame(d = compute.es::mes(.$m_one, .$m_3sub, .$sd_one,
                                    .$sd_3sub, .$n, .$n, verbose = F)$d,
                d_var = compute.es::mes(.$m_one, .$sd_3sub, .$sd_one,
                                        .$sd_3sub, .$n, .$n, verbose = F)$var.d)) %>%
  mutate(high = d + (1.96*d_var),
         low = d - (1.96*d_var),
         es_type = "nonpaired",
         exp_recoded = LF_means_wide$exp_recoded) %>%
  left_join(LF_means_wide %>% select(exp_recoded, n)) %>%
  select(exp_recoded, n, everything())

literature_effect_sizes <- read_csv("data/literature_ES.csv")
# see ../../analysis/get_literature_ES.R

all_es <- literature_effect_sizes %>%
  bind_rows(LF_effect_sizes) %>%
  left_join(exp_key) %>%
  mutate(source = ifelse(str_detect(exp_recoded, "XT"), "XT2007a",
                         ifelse(str_detect(exp_recoded, "SPSS"), "SPSS2011", "LF")),
         source = fct_relevel(source, "XT2007a","SPSS2011","LF"),
         source = as.numeric(source),
         timing = fct_relevel(timing,"simultaneous", "sequential"))

seq13 <- rma(d, d_var, dat = filter(all_es, timing == "sequential", order == "1-3"))
seq31 <- rma(d, d_var, dat = filter(all_es, timing == "sequential", order == "3-1"))
sim13 <- rma(d, d_var, dat = filter(all_es, timing == "simultaneous", order == "1-3"))
sim31 <- rma(d, d_var, dat = filter(all_es, timing == "simultaneous", order == "3-1"))

ma_es <- data.frame(order = c("1-3", "3-1", "1-3", "3-1"),
                    timing = c("sequential", "sequential", "simultaneous", "simultaneous"),
                    d = c(seq13$b[[1]], seq31$b[[1]],
                          sim13$b[[1]], sim31$b[[1]]),
                    d_low = c(seq13$ci.lb[[1]], seq31$ci.lb[[1]],
                              sim13$ci.lb[[1]], sim31$ci.lb[[1]]),
                    d_high = c(seq13$ci.ub[[1]], seq31$ci.ub[[1]],
                               sim13$ci.ub[[1]], sim31$ci.ub[[1]]))
```

```
ggplot(all_es) +
  geom_rect(aes(xmin = -Inf, xmax = Inf, ymin = d_low, ymax = d_high),
            fill = "red", alpha = 0.05, inherit.aes = FALSE, data = ma_es) +
  geom_hline(aes(yintercept = d), data = ma_es, color = "red") +
  scale_color_manual(values = c("black", "grey63")) +
  geom_pointrange(aes(x = jitter(source, 1.4),  y = d, ymax = high,
                      ymin = low, color = one_3sub_label, shape = fct_rev(blocking)),
                  size = .5) +
  geom_hline(yintercept = 0, linetype = 2, color = "black") +
  facet_grid(order ~ timing) +
  scale_x_continuous(breaks = c(1:3), limits = c(.6, 3.3),
                     labels = c("XT","SPSS","LF")) +
  ylab("Cohen's d") +
  xlab("Paper") +
  guides(color = guide_legend("Label"),
         shape = guide_legend("Blocking"))  +
  ggthemes::theme_few()
```



Below is the meta-analytical model presented in the Main Text for the sample with repeat-participants excluded. The pattern is the same as for the full sample.

```
mod <- metafor::rma(d ~ timing + order + one_3sub_label + blocking, d_var, dat = all_es)

mod_df <- data.frame(fixed_effect = c("Intercept",
                                      "Simultaneous vs. sequential timing",
                                      "1-3 vs. 3-1 trial order",
```

```
                                    "Different vs. same label",
                                    "Blocked vs. pseudo-random trial structure"),
                    beta_string = paste0(round(mod$beta, 2),
                                         " [", round(mod$ci.lb, 2), ", "
                                         , round(mod$ci.ub,2) , "]"),
                    zval = mod$zval,
                    pval_string = round(mod$pval, 2)) %>%
  mutate(pval_string = ifelse(pval_string == 0, "<.0001", pval_string))

# MA model table

kable(mod_df, caption = "Meta-analytic model with manipulations as fixed effects.",
      align = c('l', 'r', 'r', 'r'), digits = 2,
      col.names = c("Fixed effect", "beta", "z-value", "p-value")) %>%
  kable_styling(font_size = 12)
```

Table 4: Meta-analytic model with manipulations as fixed effects.

Fixed effect	beta	z-value	p-value
Intercept	1.37 [1.07, 1.66]	9.11	<.0001
Simultaneous vs. sequential timing	-0.15 [-0.36, 0.07]	-1.35	0.18
1-3 vs. 3-1 trial order	-1.48 [-1.79, -1.18]	-9.53	<.0001
Different vs. same label	0.04 [-0.2, 0.29]	0.33	0.74
Blocked vs. pseudo-random trial structure	-0.08 [-0.42, 0.26]	-0.46	0.65

Demographics

Below we report the demographic characteristics (education, language, gender, and age) of our full sample (N = 600).

```
raw_d <- read_csv("data/anonymized_data/all_raw_A.csv")

raw_d_munged <- raw_d %>%
  left_join(exp_key %>% select(exp, exp_recoded, timing)) %>%
  mutate(exp_recoded = fct_relevel(exp_recoded, "10", after = 11),
         exp_recoded = fct_relevel(exp_recoded, "11", after = 11),
         exp_recoded = fct_relevel(exp_recoded, "12", after = 11)) %>%
  select(-exp) %>%
  ungroup()
```

Education

```
raw_d_munged %>%
  mutate(education = as.factor(education),
         education = fct_recode(education,
                                "No Response" = "-1",
                                "Some High School" = "0",
                                "Graduated High School" = "1",
                                "Some College"= "2",
                                "Graduated College" = "3",
```

```
                                "Hold a higher degree" = "4")) %>%
  group_by(education) %>%
  summarise(n = n()) %>%
  mutate(prop = round(n / sum(n),2)) %>%
  kable()
```

education	n	prop
No Response	2	0.00
Some High School	6	0.01
Graduated High School	83	0.14
Some College	207	0.34
Graduated College	256	0.43
Hold a higher degree	46	0.08

First language

```
raw_d_munged %>%
  mutate(language = tolower(language),
         language = ifelse(substr(language,0,1) == "e",
                           "English", "Other")) %>%
  group_by(language) %>%
  summarise(n = n()) %>%
  mutate(prop = round(n / sum(n),2)) %>%
  kable()
```

language	n	prop
English	589	0.98
Other	10	0.02
NA	1	0.00

Gender

```
raw_d_munged %>%
  group_by(gender) %>%
  summarise(n = n()) %>%
  mutate(prop = round(n / sum(n), 2)) %>%
  kable()
```

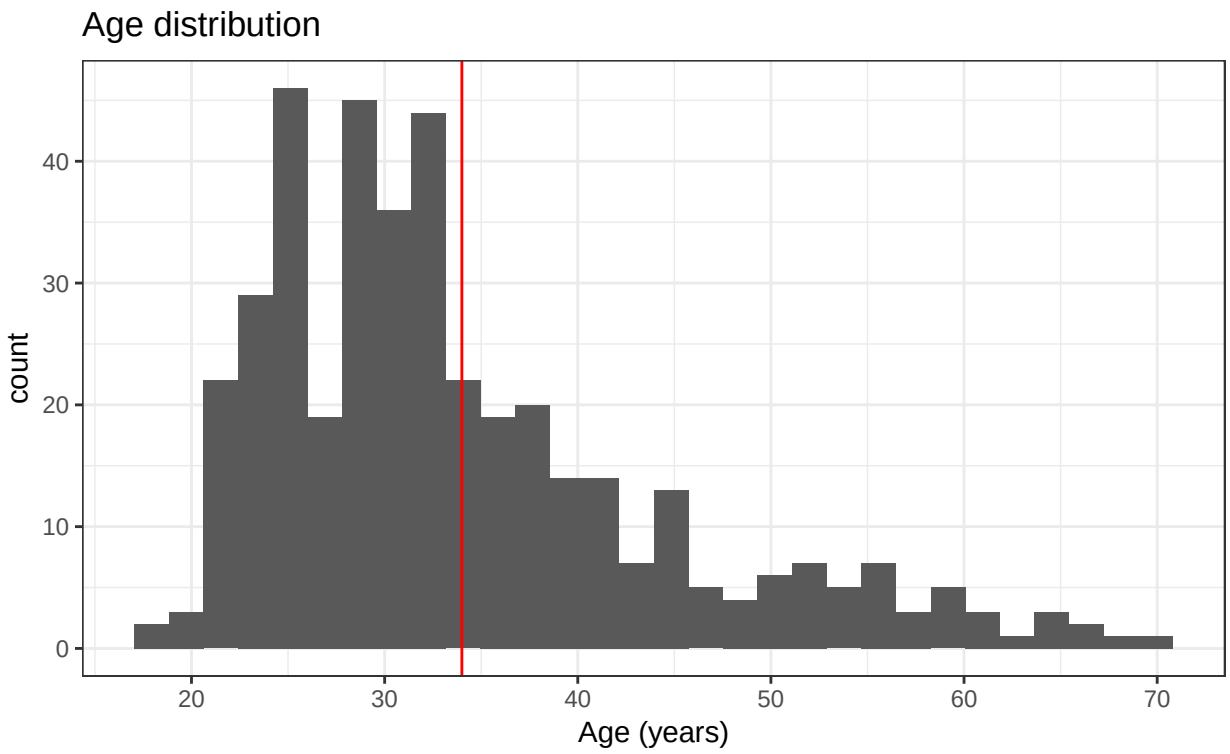
gender	n	prop
Female	179	0.30
Male	225	0.38
Other	1	0.00
NA	195	0.32

Age

```
mean_age <- round(mean(as.numeric(as.character(raw_d_munged$age)), na.rm = T),2)
```

Histogram of participant age. The red line indicates the mean ($M = 34$).

```
raw_d_munged %>%
  ggplot(aes(x = age)) +
  geom_histogram() +
  geom_vline(aes(xintercept = mean_age),
             color = "red") +
  xlab("Age (years)") +
  theme_bw() +
  ggtitle("Age distribution")
```



Task feedback

These questions were presented to participants after the main task. Their completion was optional.

Enjoyment

```
raw_d_munged %>%
  mutate(enjoyment = as.factor(enjoyment),
         enjoyment = fct_recode(enjoyment,
                                "No Response" = "-1",
                                "Worse than the Average HIT" = "0",
                                "An Average HIT" = "1",
```

```
                                "Better than average HIT" = "2")) %>%
  rename(`Did you enjoy the hit?` = "enjoyment") %>%
  group_by(`Did you enjoy the hit?`) %>%
  summarise(n = n()) %>%
  mutate(prop = round(n / sum(n),2)) %>%
  kable()
```

Did you enjoy the hit?	n	prop
No Response	1	0.00
Worse than the Average HIT	7	0.01
An Average HIT	210	0.35
Better than average HIT	382	0.64

Understanding

```
raw_d_munged %>%
  mutate(asess = as.factor(asses)) %>%
  rename(`Did you read instructions?` = "asses") %>%
  group_by(`Did you read instructions?`) %>%
  summarise(n = n()) %>%
  mutate(prop = round(n / sum(n),2)) %>%
  kable()
```

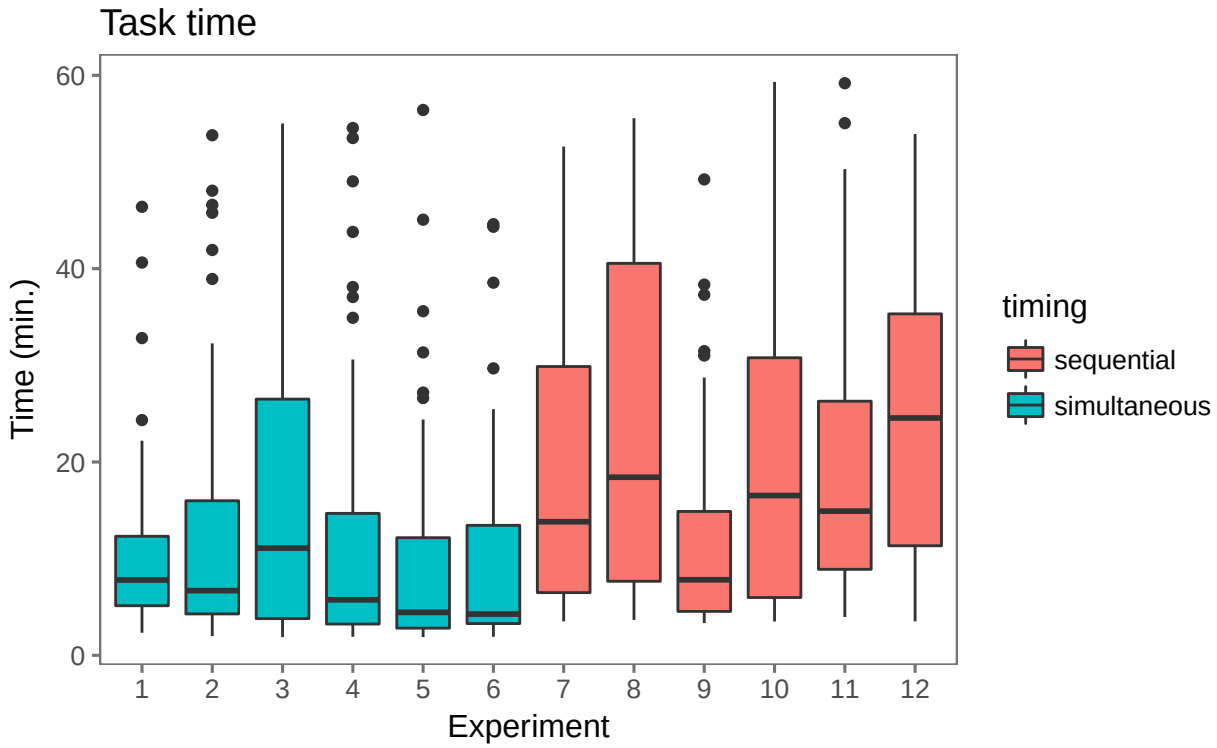
Did you read instructions?	n	prop
Confused	15	0.02
No	11	0.02
Yes	372	0.62
NA	202	0.34

Task time

Total task times were variable across experiments, but overall shorter for simultaneous timing experiments (green) compared to sequential (pink). This is due to the fact tha the sequential experiments required longer amounts of time during the training phase, whereas in the simultaneous experiments participants needed to only briefly look at the training exemplars before making their selections.

```
# dplyr doesn't play well with dates
raw_d_mungedT <- raw_d_munged

raw_d_mungedT$SubmitTime = gsub("T|Z","",raw_d_mungedT$SubmitTime)
raw_d_mungedT$AcceptTime = gsub("T|Z","",raw_d_mungedT$AcceptTime)
raw_d_mungedT$SubmitTime = strptime(raw_d_munged$SubmitTime, "%F%T")
raw_d_mungedT$AcceptTime = strptime(raw_d_mungedT$AcceptTime, "%F%T")
raw_d_mungedT$total_time = as.numeric(raw_d_mungedT$SubmitTime) -
  as.numeric(raw_d_mungedT$AcceptTime)

ggplot(raw_d_mungedT,
       aes(x = exp_recoded, y = total_time/60, fill = timing)) +
  ylab("Time (min.)") +
```

```
  xlab("Experiment") +
  ggtitle("Task time")+
  geom_boxplot() +
  ggthemes::theme_few()
```



References

AC Del Re (2013). compute.es: Compute Effect Sizes. R package version 0.2-2. URL http://cran.r-project.org/web/packages/compute.es.