PS931 Bayesian Approaches in Behavioural Science Problem Set

2138473

21/02/2022

```r
# load the packages

library("tidyverse")
library("afex")
library("BayesFactor")
library("emmeans")
library('bayestestR')
```

Task 1

Section 1

Decision making is, essentially, a complicated process which is often noisy. For example, participants tend to behave inconsistently when making decisions between equivalent pairs of risky choices under the laboratory settings. Thus, an important question to be ask is that *"Why does this choice inconsistency emerge?"*. One assumption is that being inconsistent is innate, thus decisions are made under equal amount of noise irregardless of the context of choices. Or otherwise the level of inconsistency might depend on different types of choices that people encounter. The current investigation was developed upon these assumptions.

In simulation, a total of 20 participants undertook a task in which they were requested to respond to a series of gamble pairs. These pairs were manipulated into 3 types. One of which were pairs of choices with same number of outcomes and one of them were dominant alternatives (DA), in other words, indicating one alternative out-competes the other (e.g., A: 10% of chance of winning £8 otherwise £0 *vs* DA: 10% of winning £10 otherwise £0). The second type, by contrast, were those with no dominant alternatives (NDA; e.g., A: 10% of chance of winning £8 otherwise £0 *vs* NDA: 8% of winning £12 otherwise £0). The third type, being slightly different from the above ones, are gamble pairs with different number of possible outcomes (DNO; e.g., e.g., A: 10% of chance of winning £8 otherwise £0 *vs* DNO: 8% of winning £5 or 8% of winning £10 otherwise £0). All simulated participants were exposed to the same number of gamble pairs of each type twice by random and their choice consistency (i.e., proportion of identical choices made across the two repeated exposures) were recorded for further analyses.

We firstly analysed the data with *Frequentist* methods using a Greenhouse-Geissor corrected ANOVA with a single within-subject factor of *Types* of gamble pairs (DA *vs* NDA *vs* DNO). Meanwhile, *Bayesian* ANOVA was performed using a `medium` prior on the effect size built in the `BayesFactor` package. Results demonstrates a significant main effect of *Type* on consistency, $F(1.97, 37.36) = 128.56$, $p < .001$, $BF_{10} > 6.6\text{x}10^6$. The results were qualitative the same for *Bayesian* ANOVA after using `wide` ($BF_{10} > 8.3\text{x}10^6$), `ultrawide` ($BF_{10} > 9.3\text{x}10^6$) priors on the effect size, and AIC ($AIC = 1284971537$), BIC ($BIC = 158245446$) approximation. *Figure 1* shows the means and distribution of consistency of different types of choice pairs, and illustrates that the consistency of DA type group is higher than those of NDA and DNO groups, yet the difference between the last 2 groups is unclear.

To further explore on the main effect of *Type* and differences between groups, *Frequentist* pairwise comparisons with Bonferroni-Holm correction and *Bayesian* t-tests with `medium` prior on consistency were performed.

Post hoc results validate that the level of consistency for DA group is significantly higher than NDA group (difference = 23.5, SE = 3.65), $t(1, 19) = 6.443$, $p < .0001$, $BF_{10} > 1.1\text{x}10^4$, and the bayes factors for `wide` ($BF_{10} > 1.2\text{x}10^4$) and `ultrawide` ($BF_{10} > 1.3\text{x}10^4$) priors were also decisive. Also, a significant difference was also found between DA and DNO group (difference = 21.0, SE = 3.32), $t(1, 19) = 6.332$, $p < .0001$, $BF_{10} > 9.1\text{x}10^3$. In line with this, the evidence provided by *Bayes factors* for `wide` and `ultrawide` priors on the effect size was decisive as well ($BF_{10} > 10^4$). However, no significant difference was observed between NDA and DNO group (difference = 2.5, SE = 3.69), $t(1, 19) = 0.677$, $p = .506$, $BF_{10} = 0.42$, and this finding was approximately the same for `wide` and `ultrawide` priors on the effect size.

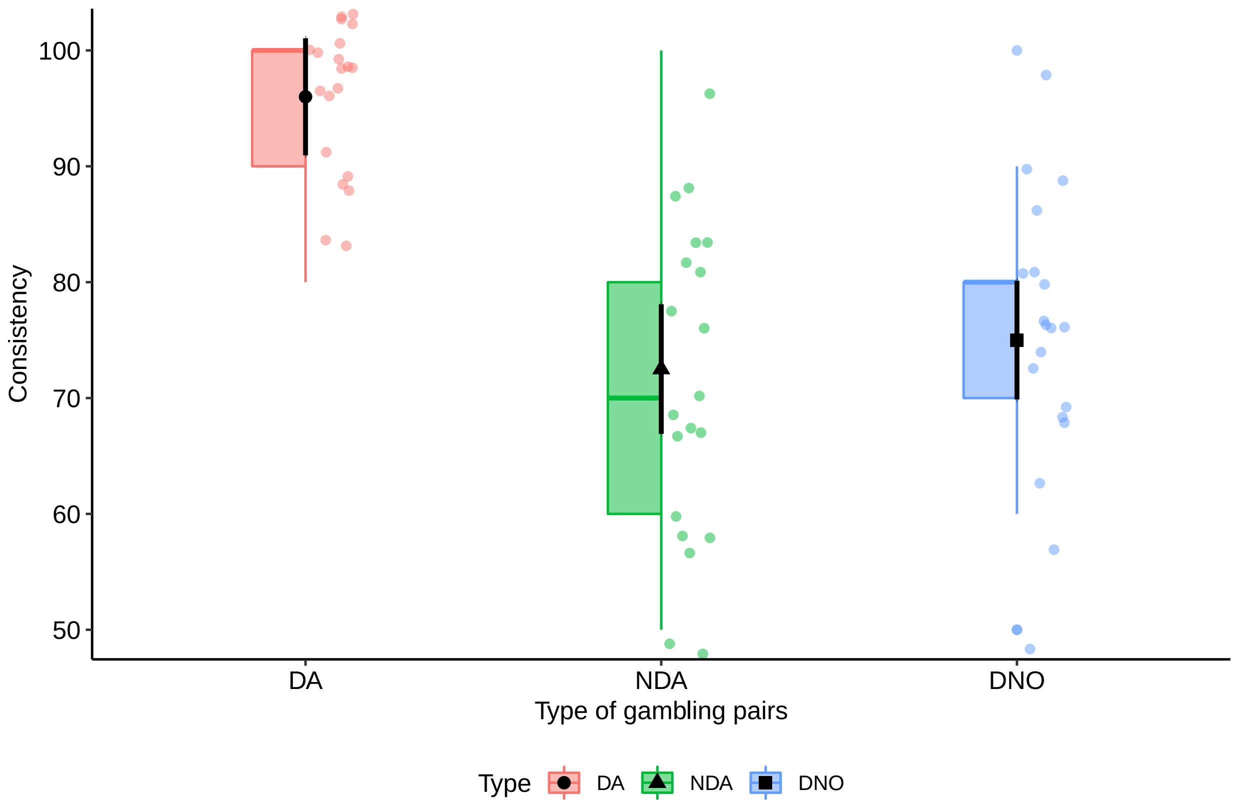


Figure 1. Consistency of choices as a function of types of gamble pairs. Specifically, DA: pairs of dominant alternatives; NDA: pairs with no dominant alternatives; DNO: pairs with different number of possible outcomes. Coloured points in the background show the raw data, coloured boxplots demonstrate the median and quantiles, black points in the foreground show the mean, error bars show 95% within-subjects confidence intervals.

Section 2

```r
#load the data for Task 1

df1 <- read.csv("consistency_by_type.csv")
df1 <- mutate_at(df1, vars(X, id, Type), as.factor)

head(df1)
```

```
##   X id Type Consistency
## 1 1  1   DA         100
## 2 2  1  NDA          70
## 3 3  1  DNO          80
```

```
## 4 4  2   DA         100
## 5 5  2  NDA          60
## 6 6  2  DNO          80
```

```r
# checking NAs

df1 %>%
  select_if(function(x) any(is.na(x))) %>%
  summarise_each(funs(sum(is.na(.))))
```

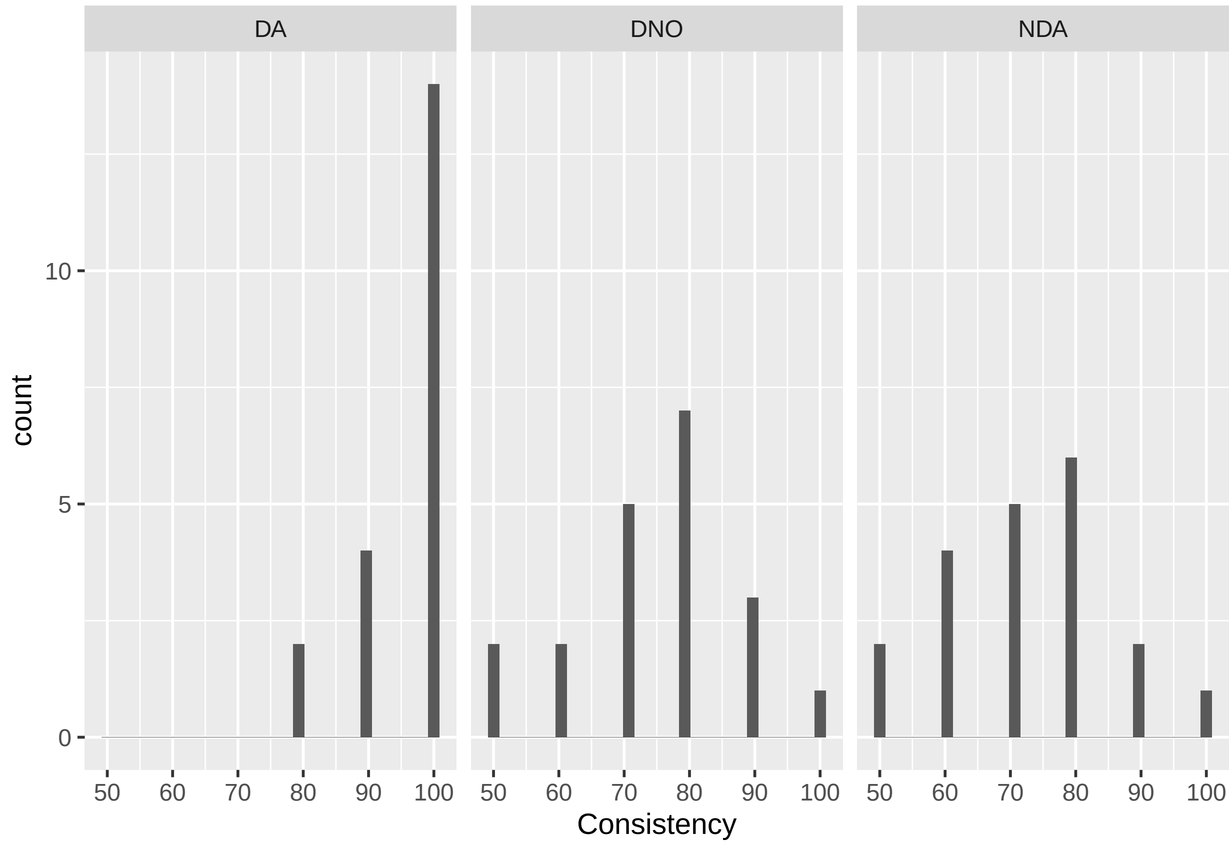
```
## data frame with 0 columns and 1 row
```

```r
# great no missing values
```

```r
# check the sample size per group

df1 %>% group_by(Type) %>% count(Type) # n = 20
```

```
## # A tibble: 3 x 2
## # Groups:   Type [3]
##   Type      n
##   <fct> <int>
## 1 DA       20
## 2 DNO      20
## 3 NDA      20
```

```r
# checking data distribution

dis <- ggplot(df1, aes(Consistency)) +
  geom_histogram() +
  facet_wrap(~Type)

dis
```




```r
# Frequentist: Repeated measures ANOVA

a1 <- aov_ez(id = "id", dv = "Consistency", df1, within = "Type")

a1
```

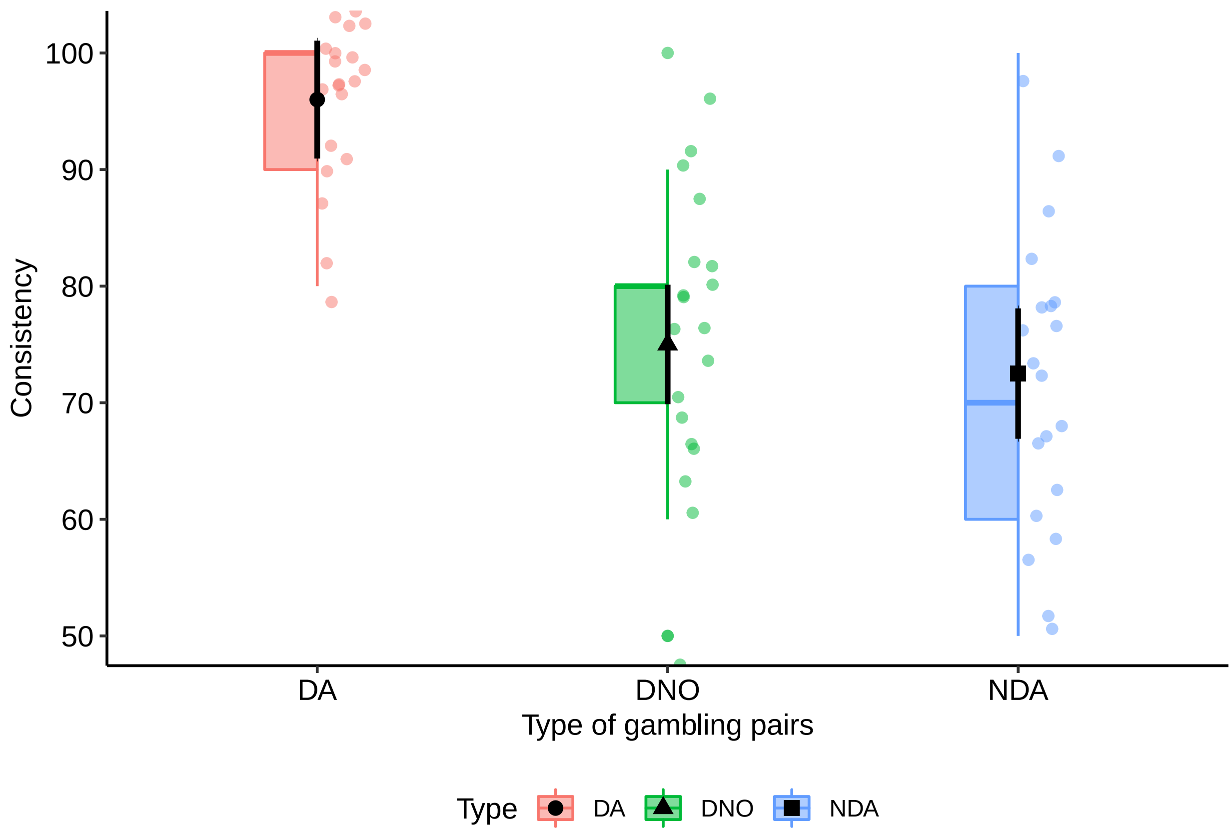
```
## Anova Table (Type 3 tests)
##
## Response: Consistency
##   Effect           df    MSE         F  ges p.value
## 1   Type 1.97, 37.36 128.56 26.36 *** .469  <.001
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
##
## Sphericity correction method: GG
```

```r
# Frequentist post hocs

em1 <- emmeans(a1, "Type")

em1 %>% pairs() %>% update(by = NULL) %>% summary(adjust = "holm")
```

```
##  contrast  estimate   SE df t.ratio p.value
##  DA - DNO      21.0 3.32 19   6.332  <.0001
##  DA - NDA      23.5 3.65 19   6.443  <.0001
##  DNO - NDA      2.5 3.69 19   0.677  0.5063
##
```

```
## P value adjustment: holm method for 3 tests
```

```r
# plot the difference

p1 <- afex_plot(a1, x ="Type", error_level = 0.95, dodge = 0.5,
                error = "within",
                mapping = c("shape", "fill", 'color'),
                data_geom = ggpol::geom_boxjitter,
                data_arg = list(width = 0.3),
                point_arg = list(size = 2.4, color = "black"),
                error_arg = list(size = 1, width = 0, color ="black")) +
  ylab("Consistency") + xlab("Type of gambling pairs") +
  ggpubr::theme_pubr(base_size = 11) +
  theme(legend.position="bottom", panel.grid.major.x = element_blank())

p1
```




```r
# Bayesian: try different priors

a2_m <- anovaBF(Consistency ~ Type + id, data = data.frame(df1), whichRandom = "id",
                rscaleFixed = "medium")
a2_m
```

```
## Bayes factor analysis
## --------------
## [1] Type + id : 6714554 ±1.55%
##
```

```
## Against denominator:
##   Consistency ~ id
## ---
## Bayes factor type: BFlinearModel, JZS
```

```r
a2_w <- anovaBF(Consistency ~ Type + id, data = data.frame(df1), whichRandom = "id",
                rscaleFixed = "wide")
a2_w
```

```
## Bayes factor analysis
## --------------
## [1] Type + id : 8294346 ±0.83%
##
## Against denominator:
##   Consistency ~ id
## ---
## Bayes factor type: BFlinearModel, JZS
```

```r
a2_uw <- anovaBF(Consistency ~ Type + id, data = data.frame(df1), whichRandom = "id",
                 rscaleFixed = "ultrawide")
a2_uw
```

```
## Bayes factor analysis
## --------------
## [1] Type + id : 9341166 ±1.11%
##
## Against denominator:
##   Consistency ~ id
## ---
## Bayes factor type: BFlinearModel, JZS
```

```r
# try using AIC and BIC weights

a2 <- lmer(Consistency ~ Type + (1|id), data = df1)
a3 <- lmer(Consistency ~ 1 + (1|id), data = df1)

aics <- AIC(a2,a3)
bics <- BIC(a2,a3)

ic_as_bf <- function(x) {exp(-x[1]/2)/exp(-x[2]/2)}
ic_as_bf(bics$BIC)
```

```
## [1] 158245446
```

```r
ic_as_bf(aics$AIC)
```

```
## [1] 1284971537
```

```r
# Pivot the data frame for Bayesian t-tests

dft <- df1 %>% select(id:Consistency) %>%  pivot_wider(names_from = Type, values_from = Consistency)
head(dft)
```

```
## # A tibble: 6 x 4
##   id       DA   NDA   DNO
##   <fct> <int> <int> <int>
## 1 1       100    70    80
## 2 2       100    60    80
```

```
## 3 3       100   100    80
## 4 4        80    80    60
## 5 5       100    80    70
## 6 6        90    60    70
```

```r
# Bayesian t-tests on DNO vs NDA

a.ttestBF.m <- ttestBF(x = dft$DNO, y = dft$NDA, paired=T, nullInterval=c(0,Inf), rscale="medium")
a.ttestBF.m
```

```
## Bayes factor analysis
## --------------
## [1] Alt., r=0.707 0<d<Inf    : 0.4200382 ±0%
## [2] Alt., r=0.707 !(0<d<Inf) : 0.1505489 ±0%
##
## Against denominator:
##   Null, mu = 0
## ---
## Bayes factor type: BFoneSample, JZS
```

```r
a.ttestBF.w <- ttestBF(x = dft$DNO, y = dft$NDA, paired=T, nullInterval=c(0,Inf), rscale="wide")
a.ttestBF.w
```

```
## Bayes factor analysis
## --------------
## [1] Alt., r=1 0<d<Inf    : 0.3157032 ±0%
## [2] Alt., r=1 !(0<d<Inf) : 0.1090593 ±0.01%
##
## Against denominator:
##   Null, mu = 0
## ---
## Bayes factor type: BFoneSample, JZS
```

```r
a.ttestBF.uw <- ttestBF(x = dft$DNO, y = dft$NDA, paired=T, nullInterval=c(0,Inf), rscale="ultrawide")
a.ttestBF.uw
```

```
## Bayes factor analysis
## --------------
## [1] Alt., r=1.414 0<d<Inf    : 0.2313646  ±0%
## [2] Alt., r=1.414 !(0<d<Inf) : 0.07815114 ±0%
##
## Against denominator:
##   Null, mu = 0
## ---
## Bayes factor type: BFoneSample, JZS
```

```r
# DA vs NDA

a.ttestBF.m <- ttestBF(x = dft$DA, y = dft$NDA, paired=T, nullInterval=c(0,Inf), rscale="medium")
a.ttestBF.m
```

```
## Bayes factor analysis
## --------------
## [1] Alt., r=0.707 0<d<Inf    : 11279.28 ±NA%
## [2] Alt., r=0.707 !(0<d<Inf) : 0.020017 ±NA%
##
## Against denominator:
##   Null, mu = 0
```

```
## ---
## Bayes factor type: BFoneSample, JZS
```

```r
a.ttestBF.w <- ttestBF(x = dft$DA, y = dft$NDA, paired=T, nullInterval=c(0,Inf), rscale="wide")
a.ttestBF.w
```

```
## Bayes factor analysis
## --------------
## [1] Alt., r=1 0<d<Inf    : 12935.26   ±NA%
## [2] Alt., r=1 !(0<d<Inf) : 0.02295583 ±NA%
##
## Against denominator:
##   Null, mu = 0
## ---
## Bayes factor type: BFoneSample, JZS
```

```r
a.ttestBF.uw <- ttestBF(x = dft$DA, y = dft$NDA, paired=T, nullInterval=c(0,Inf), rscale="ultrawide")
a.ttestBF.uw
```

```
## Bayes factor analysis
## --------------
## [1] Alt., r=1.414 0<d<Inf    : 13446.35   ±NA%
## [2] Alt., r=1.414 !(0<d<Inf) : 0.02386285 ±NA%
##
## Against denominator:
##   Null, mu = 0
## ---
## Bayes factor type: BFoneSample, JZS
```

```r
# DNO vs DA

a.ttestBF.m <- ttestBF(x = dft$DA, y = dft$DNO, paired=T, nullInterval=c(0,Inf), rscale="medium")
a.ttestBF.m
```

```
## Bayes factor analysis
## --------------
## [1] Alt., r=0.707 0<d<Inf    : 9142.17    ±NA%
## [2] Alt., r=0.707 !(0<d<Inf) : 0.02043975 ±NA%
##
## Against denominator:
##   Null, mu = 0
## ---
## Bayes factor type: BFoneSample, JZS
```

```r
a.ttestBF.w <- ttestBF(x = dft$DA, y = dft$DNO, paired=T, nullInterval=c(0,Inf), rscale="wide")
a.ttestBF.w
```

```
## Bayes factor analysis
## --------------
## [1] Alt., r=1 0<d<Inf    : 10429.83   ±NA%
## [2] Alt., r=1 !(0<d<Inf) : 0.02331865 ±NA%
##
## Against denominator:
##   Null, mu = 0
## ---
## Bayes factor type: BFoneSample, JZS
```

```r
a.ttestBF.uw <- ttestBF(x = dft$DA, y = dft$DNO, paired=T, nullInterval=c(0,Inf), rscale="ultrawide")
a.ttestBF.uw
```

```
## Bayes factor analysis
## --------------
## [1] Alt., r=1.414 0<d<Inf    : 10779.19   ±NA%
## [2] Alt., r=1.414 !(0<d<Inf) : 0.02409975 ±NA%
##
## Against denominator:
##   Null, mu = 0
## ---
## Bayes factor type: BFoneSample, JZS
```

Task 2

Section 1

Investigations upon inconsistency after repetitive exposures to the same choice have attracted considerable academic interests. Aside from questioning the mechanisms that mediate this process, the factors that moderate the extent of choice inconsistency are also worthy to study in understanding human decision making. People might be inconsistent when encounter a decision scenario that they have experienced previously. Yet it is still unclear that how this inconsistency will shift after increasing the repetitions of certain situation. Furthermore, some other exogenous factors, such as the random presentation of the gamble pairs in the laboratory setting was assumed to result in various level of choice inconsistency as well. Specifically, it has been suggested that participants' decision inconsistency might be, to some extent, a strategy to adapt to the random presence of gamble pairs. Thus, the current inquiry was developed upon these assumptions to investigate whether, or to what extent, repeated experiences alongside with randomness of order of gamble pairs might influence participants choices.

A total of 28 participants were recruited in the experiment in which their main task is to respond to 12 blocks of 75 gambling scenarios. Each gambling trial requires participants to choose between one option and an alternative, each with different probabilities of two possible rewards, and all 12 blocks contained the same 75 gambling pairs and was presented in a series of pre-arranged orders. Participants were requested to complete all 12 blocks within a few days, the *order* of gambling pairs of the *early* two *blocks* (i.e, the 1^{st} and 2^{nd} block) and the *late* two *blocks* (i.e., the 11^{th} and 12^{th} block) were manipulated to be either the *same* or *different* from one another. The choice consistency of the early and late four blocks were taken to further analyses.

Figure 2 shows the means and distributions of consistency and indicates that participants were being more consistent after repeated exposures to the same set of gamble pairs, while the *order* seemingly does not have a significant impact. After the inspection of plot, *Frequestist* mixed-effect model and *Bayesian* ANOVA with a `medium` prior on the effect size in `BayesFactor` package and summarised using a *Bayes Factor inclusion* for matching models were performed. In line with the information illustrated by *Figure 2*, results indicate a significant main effect of *blocks*, $F(1, 29.53) = 18.32$, $p < .001$, $BF_{\text{inclusion}} = 923.34$, and the *Bayes Factor* for `wide` and `ultrawide` ($BF_{\text{inclusion}} > 10^3$) priors on the effect size were also being decisive. Furthermore, no significant main effect of *order* [$F(1, 34.77) = 0.00$, $p = .949$, $BF_{\text{inclusion}} = 0.282$], nor a interaction of *order* by *blocks* effect [$F(1, 43.22) = 0.14$, $p = .014$, $BF_{\text{inclusion}} = 0.444$] were observed, indicating the order of either being different or the same does not play a significant role in moderating choice consistency. In alignment with this, the *Bayes Factor* for `wide` and `ultrawide` priors on the effect size demonstrate poor evidence as well.

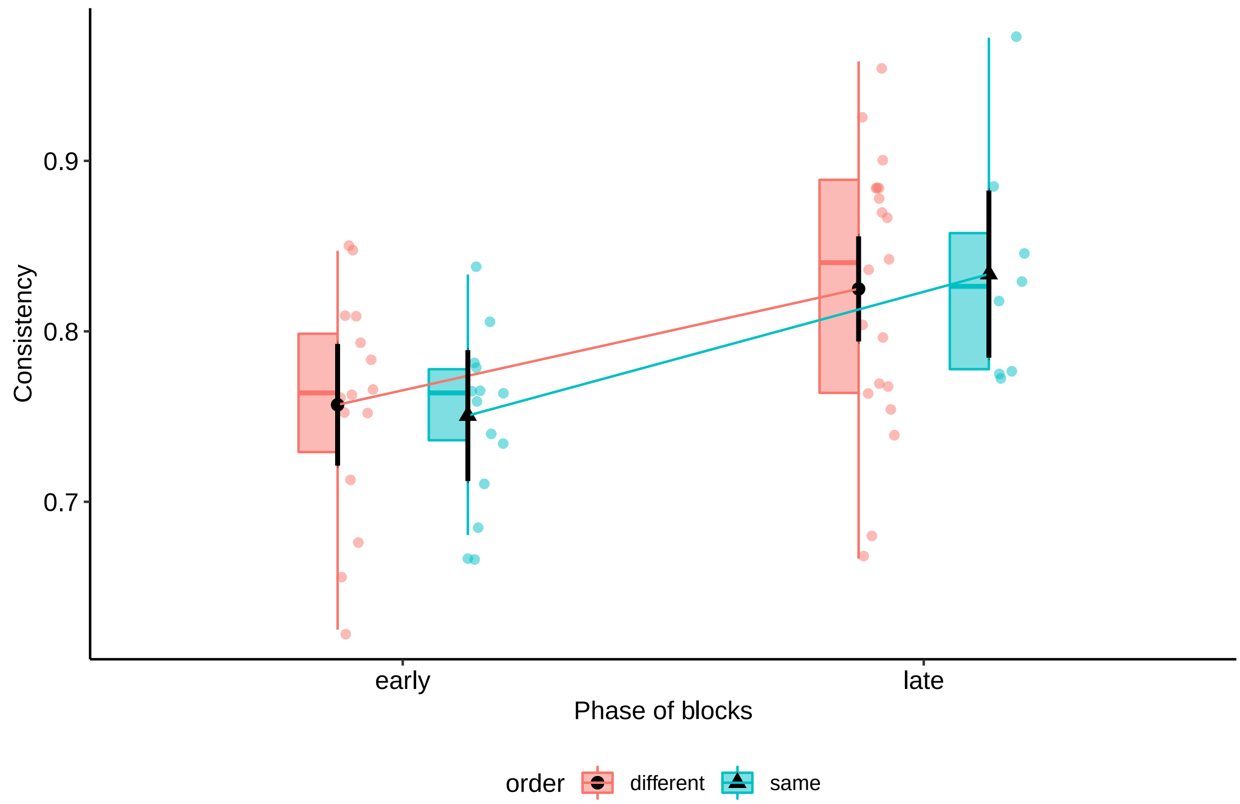


Figure 2. Consistency of choices as a function of block phases and order of gamble pairs. Coloured points in the background show the raw data, coloured boxplots demonstrate the median and quantiles, black points in the foreground show the mean, error bars show 95% confidence intervals for the mixed-effects model.

Section 2

```
#load the data

df2 <- read.csv("block_and_order_consistency.csv")
df2 <- mutate_at(df2, vars(X, id, blocks, order), as.factor)

head(df2)
```

```
##   X id blocks     order consistency
## 1 1 10  early      same   0.7083333
## 2 2 10   late different   0.9583333
## 3 3 11  early different   0.6527778
## 4 4 11   late different   0.8333333
## 5 5 12  early different   0.7638889
## 6 6 12   late      same   0.7777778
```

```
# within-subject factor - blocks

df2 %>% group_by(blocks) %>% count(blocks) # n = 28
```

```
## # A tibble: 2 x 2
## # Groups:   blocks [2]
##   blocks     n
```

```
##   <fct>  <int>
## 1 early     28
## 2 late      28
```

```r
# check with factor level of orders

df2 %>% group_by(order) %>% count(order) # class imbalance
```

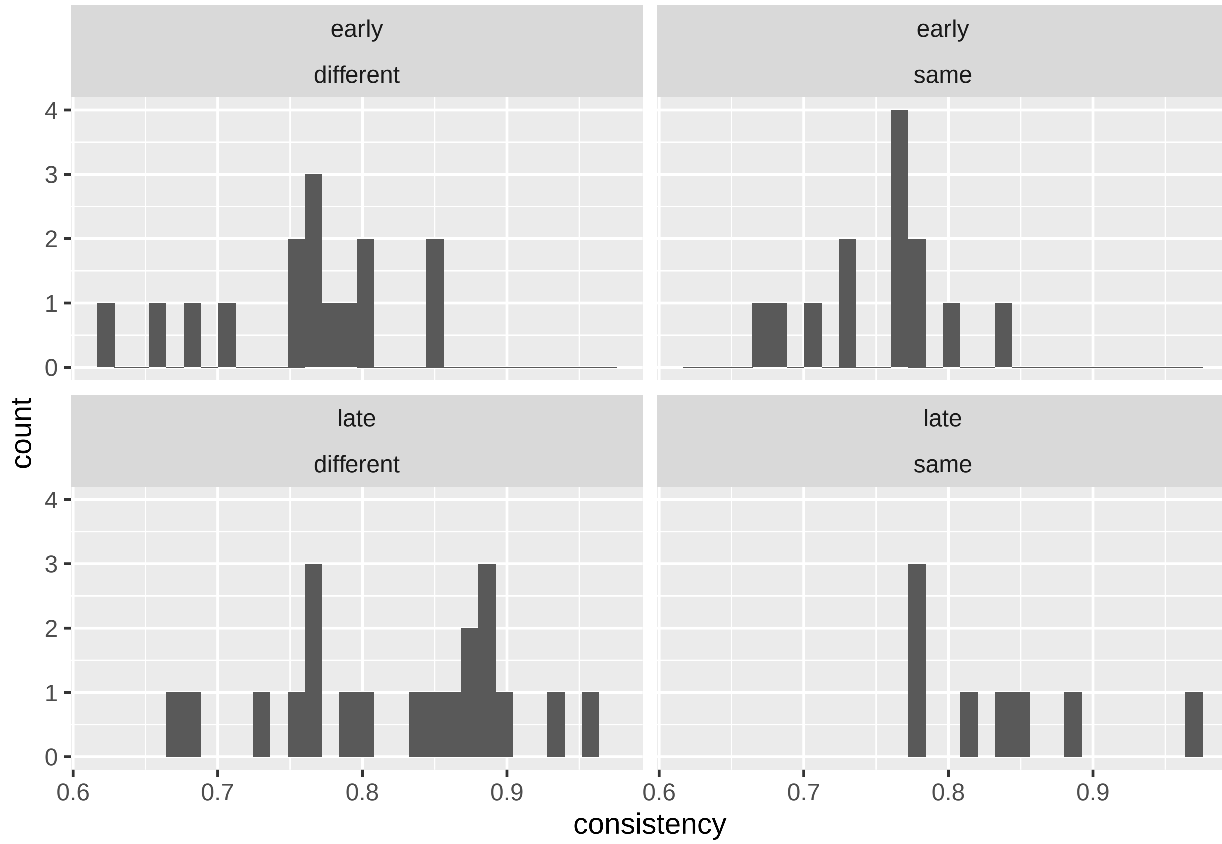
```
## # A tibble: 2 x 2
## # Groups:   order [2]
##   order         n
##   <fct>     <int>
## 1 different    35
## 2 same         21
```

```r
# checking NAs

df2 %>%
  select_if(function(x) any(is.na(x))) %>%
  summarise_each(funs(sum(is.na(.))))
```

```
## data frame with 0 columns and 1 row
```

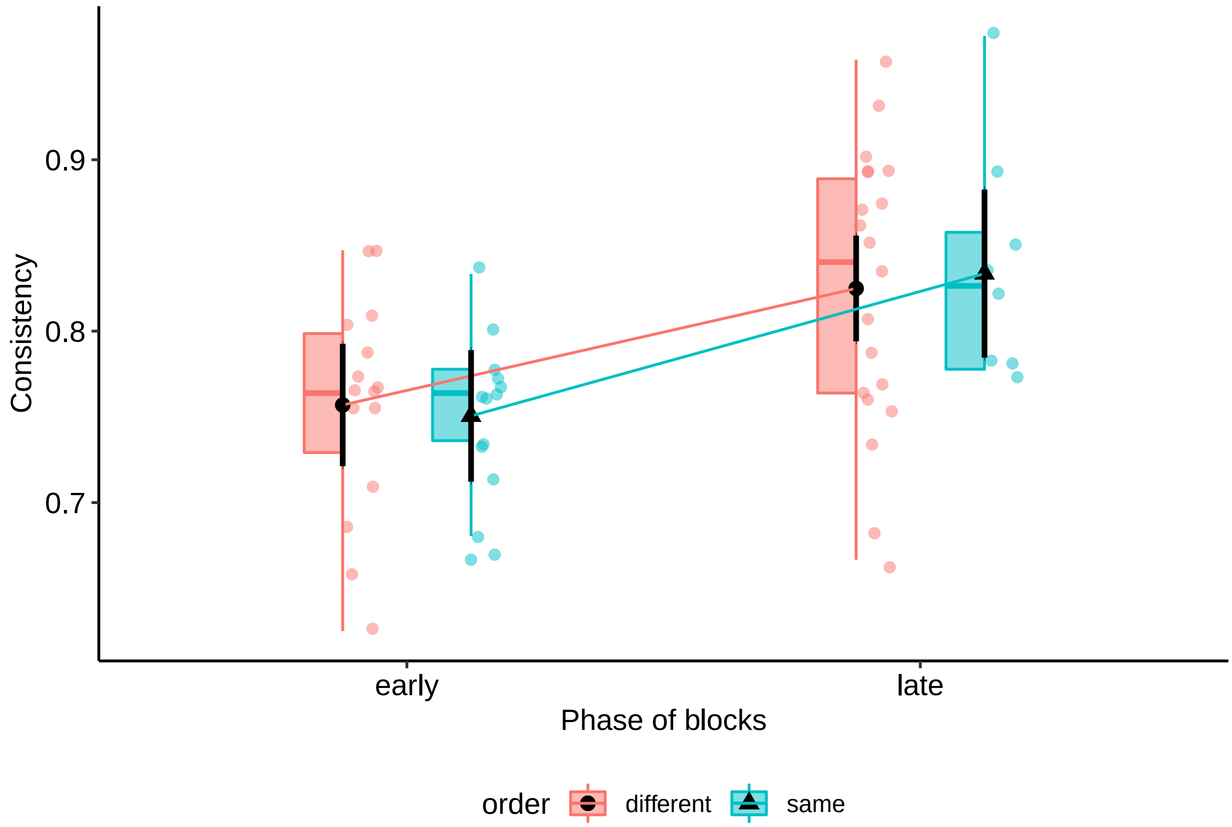
```r
# great no missing values
```

```r
# checking data distribution

dis <- ggplot(df2, aes(consistency)) +
  geom_histogram() +
  facet_wrap(blocks~order)

dis
```




```
# Frequentist: mixed effect ANOVA, treating participants as random effect

m1 <- mixed(consistency ~ blocks * order + (1|id), data=df2)

## Fitting one lmer() model. [DONE]
## Calculating p-values. [DONE]

m1

## Mixed Model Anova Table (Type 3 tests, S-method)
##
## Model: consistency ~ blocks * order + (1 | id)
## Data: df2
##          Effect       df         F p.value
## 1        blocks 1, 29.53 18.32 ***   <.001
## 2         order 1, 34.77      0.00    .949
## 3 blocks:order 1, 43.32      0.14    .711
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1

# plot the difference

p2 <- afex_plot(m1, x ="blocks", trace = "order", error_level = 0.95, dodge = 0.5,
                        mapping = c("shape", "fill", 'color'),
                        data_geom = ggpol::geom_boxjitter,
                        data_arg = list(width = 0.3),
                        point_arg = list(size = 2.4, color = "black"),
                        error_arg = list(size = 1, width = 0, color ="black")) +
```

```
  ylab("Consistency") + xlab("Phase of blocks") +
  ggpubr::theme_pubr(base_size = 11) +
  theme(legend.position="bottom", panel.grid.major.x = element_blank())

p2
```




```
# Bayesain ANOVA with different priors

a3_m <- anovaBF(consistency ~ blocks * order + id, data = data.frame(df2), whichRandom = "id",
                rscaleFixed = "medium")
a3_m
```

```
## Bayes factor analysis
## --------------
## [1] blocks + id                          : 981.5197  ±1.11%
## [2] order + id                           : 0.3688183 ±2.74%
## [3] blocks + order + id                  : 283.4104  ±1.63%
## [4] blocks + order + blocks:order + id : 130.4956  ±2.59%
##
## Against denominator:
##   consistency ~ id
## ---
## Bayes factor type: BFlinearModel, JZS
```

```
a3_w <- anovaBF(consistency ~ blocks * order + id, data = data.frame(df2), whichRandom = "id",
                rscaleFixed = "wide")
a3_w
```

```
## Bayes factor analysis
## --------------
## [1] blocks + id                         : 1331.344 ±18.43%
## [2] order + id                          : 0.273146 ±1.23%
## [3] blocks + order + id                 : 231.8719 ±1.57%
## [4] blocks + order + blocks:order + id : 87.07502 ±4.83%
##
## Against denominator:
##   consistency ~ id
## ---
## Bayes factor type: BFlinearModel, JZS
```

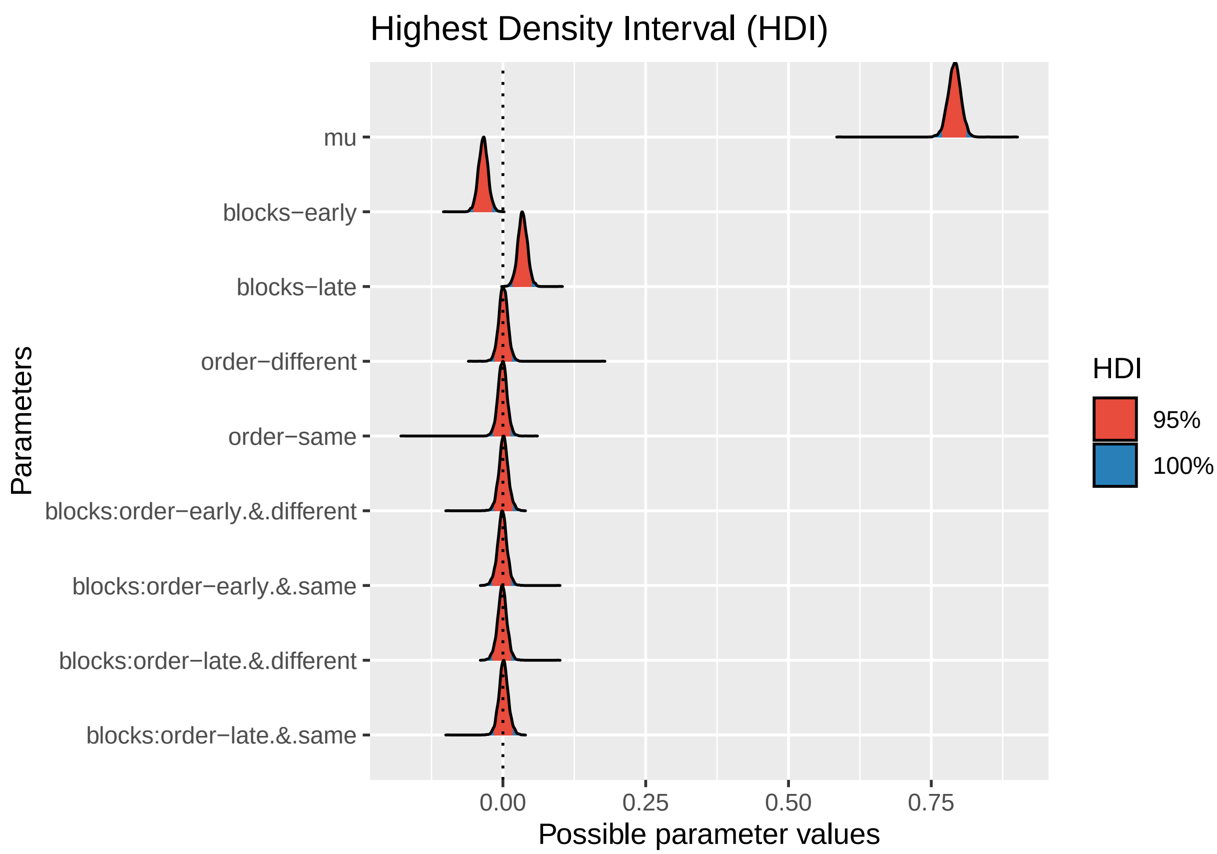
```
a3_uw <- anovaBF(consistency ~ blocks * order + id, data = data.frame(df2), whichRandom = "id",
                 rscaleFixed = "ultrawide")
a3_uw
```

```
## Bayes factor analysis
## --------------
## [1] blocks + id                         : 1074.879  ±1.23%
## [2] order + id                          : 0.2227153 ±9.33%
## [3] blocks + order + id                 : 170.4561  ±1.6%
## [4] blocks + order + blocks:order + id : 46.6901   ±5.73%
##
## Against denominator:
##   consistency ~ id
## ---
## Bayes factor type: BFlinearModel, JZS
```

```
# plotting HDI

library(insight)

a.anova.samples.m <- get_parameters(a3_m[4],effects="fixed") # prior - "medium"

parameters.to.show <- names(a.anova.samples.m)[!grepl("g_*",names(a.anova.samples.m))]
hdi(a.anova.samples.m[parameters.to.show])
```

```
## Highest Density Interval
##
## Parameter                         |        95% HDI
## ------------------------------------------------
## mu                                | [ 0.77,  0.81]
## blocks-early                      | [-0.05, -0.02]
## blocks-late                       | [ 0.02,  0.05]
## order-different                   | [-0.01,  0.02]
## order-same                        | [-0.02,  0.01]
## blocks:order-early.&.different | [-0.02,  0.02]
## blocks:order-early.&.same       | [-0.02,  0.02]
## blocks:order-late.&.different  | [-0.02,  0.02]
## blocks:order-late.&.same        | [-0.02,  0.02]
```

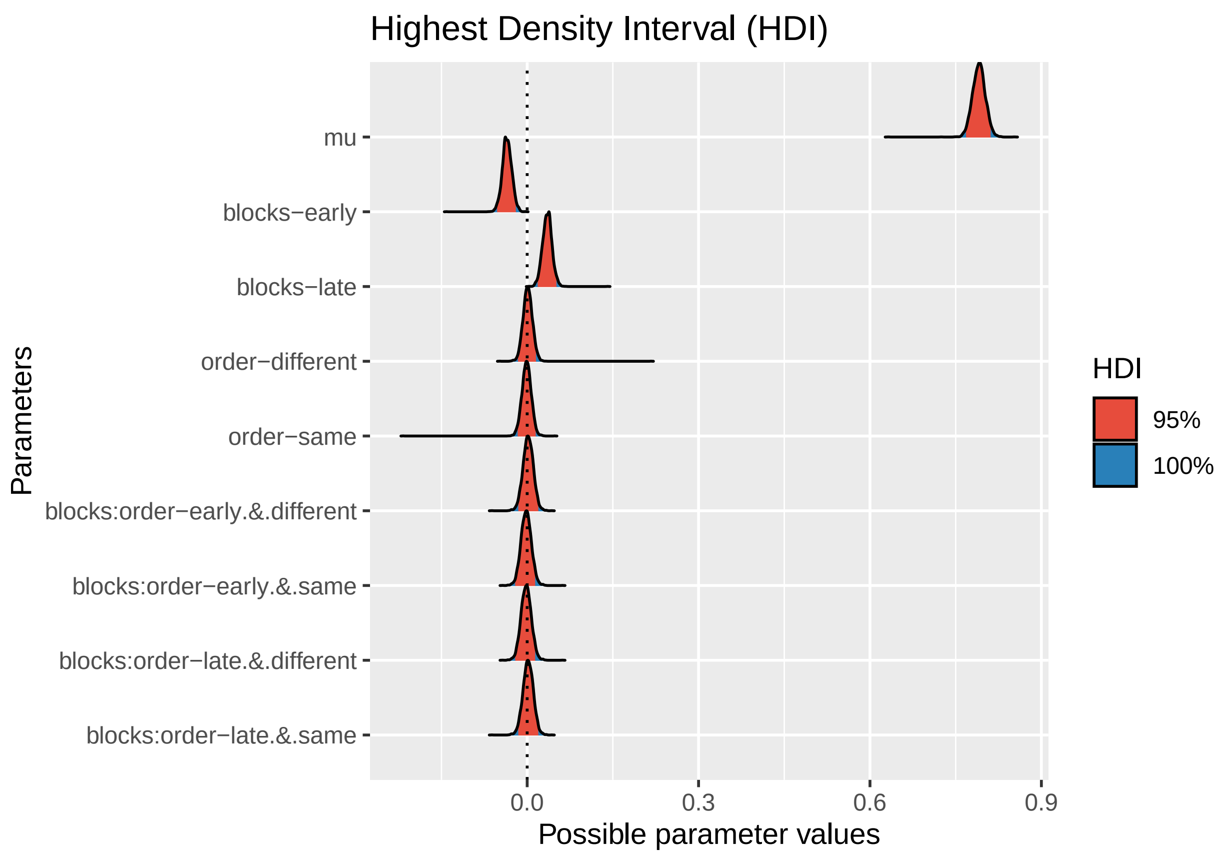
```
plot(hdi(a.anova.samples.m[parameters.to.show]), data=a3_m[4])
```




```
# HDI: prior - "wide"

a.anova.samples.w <- get_parameters(a3_w[4],effects="fixed")

parameters.to.show <- names(a.anova.samples.w)[!grepl("g_*",names(a.anova.samples.w))]
hdi(a.anova.samples.w[parameters.to.show])
```

```
## Highest Density Interval
##
## Parameter                        |        95% HDI
## ------------------------------------------------
## mu                               | [ 0.77,  0.81]
## blocks-early                     | [-0.05, -0.02]
## blocks-late                      | [ 0.02,  0.05]
## order-different                  | [-0.02,  0.02]
## order-same                       | [-0.02,  0.02]
## blocks:order-early.&.different   | [-0.01,  0.02]
## blocks:order-early.&.same        | [-0.02,  0.01]
## blocks:order-late.&.different    | [-0.02,  0.01]
## blocks:order-late.&.same         | [-0.01,  0.02]
```

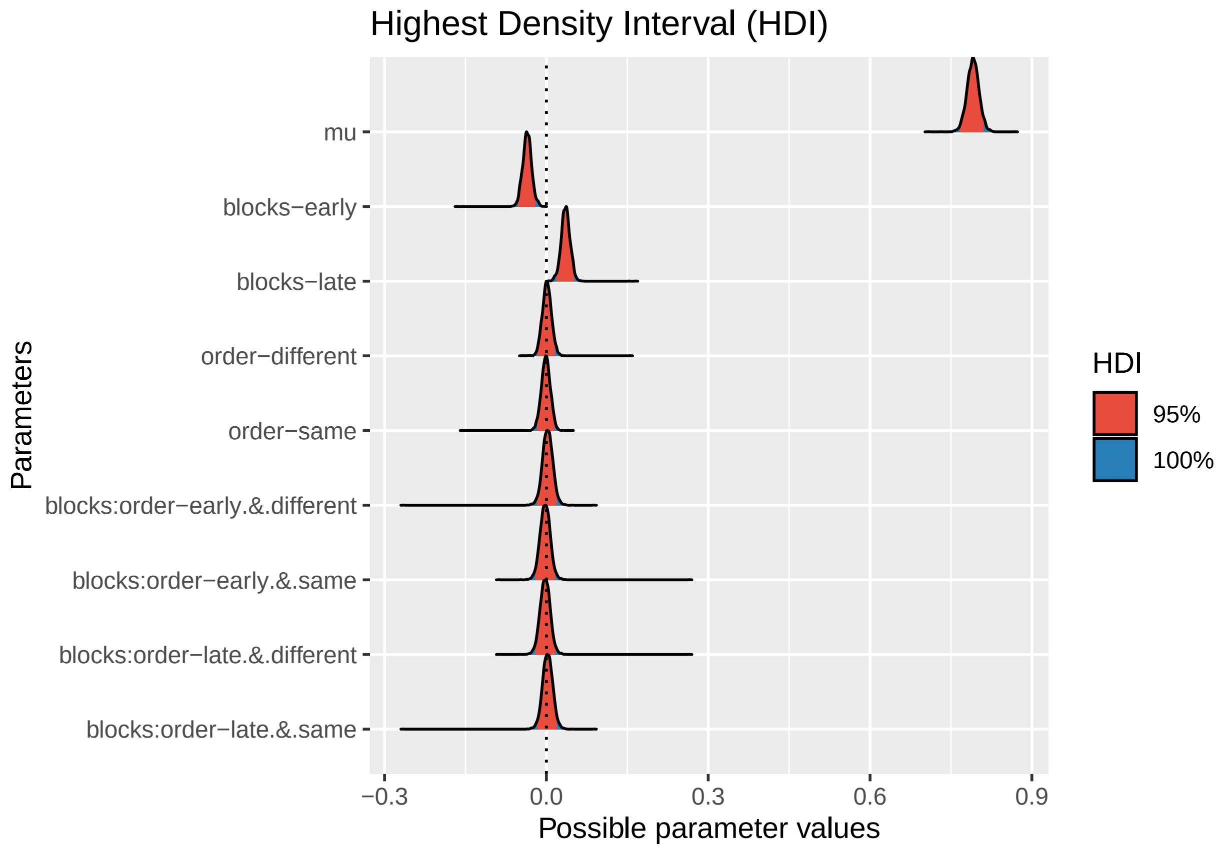
```
plot(hdi(a.anova.samples.w[parameters.to.show]), data=a3_w[4])
```




```
# HDI: prior - "ultrawide"

a.anova.samples.uw <- get_parameters(a3_uw[4],effects="fixed")

parameters.to.show <- names(a.anova.samples.uw)[!grepl("g_*",names(a.anova.samples.uw))]
hdi(a.anova.samples.uw[parameters.to.show])
```

```
## Highest Density Interval
##
## Parameter                         |        95% HDI
## -------------------------------------------------
## mu                                | [ 0.77,  0.81]
## blocks-early                      | [-0.05, -0.02]
## blocks-late                       | [ 0.02,  0.05]
## order-different                   | [-0.02,  0.02]
## order-same                        | [-0.02,  0.02]
## blocks:order-early.&.different    | [-0.02,  0.02]
## blocks:order-early.&.same         | [-0.02,  0.02]
## blocks:order-late.&.different     | [-0.02,  0.02]
## blocks:order-late.&.same          | [-0.02,  0.02]
```

```
plot(hdi(a.anova.samples.uw[parameters.to.show]), data=a3_uw[4])
```




```r
# Essentially there are not much difference for those priors
```

```r
# try model averaging using BF inclusion

bf_inclusion(a3_m, match_models = T)
```

```
## Inclusion Bayes Factors (Model Averaged)
##
##              P(prior) P(posterior) Inclusion BF
## id               1.00         1.00
## blocks           0.40         0.91       924.10
## order            0.40         0.20        0.289
## blocks:order     0.20         0.09        0.460
##
## * Compared among: matched models only
## *    Priors odds: uniform-equal
```

```r
bf_inclusion(a3_w, match_models = T)
```

```
## Inclusion Bayes Factors (Model Averaged)
##
##              P(prior) P(posterior) Inclusion BF
## id               1.00         1.00
## blocks           0.40         0.95     1.23e+03
## order            0.40         0.14        0.174
## blocks:order     0.20         0.05        0.376
##
## * Compared among: matched models only
```

```
## *    Priors odds: uniform-equal
```

```r
bf_inclusion(a3_uw, match_models = T)
```

```
## Inclusion Bayes Factors (Model Averaged)
##
##              P(prior) P(posterior) Inclusion BF
## id               1.00         1.00
## blocks           0.40         0.96     1.02e+03
## order            0.40         0.13        0.159
## blocks:order     0.20         0.04        0.274
##
## * Compared among: matched models only
## *    Priors odds: uniform-equal
```