Age-Related Differences in Discounting Depend on Income Result

Matt Wan

March 24, 2024

1 Preliminaries

1.1 Clear the Console Panes

```
options(replace.assign = TRUE, width = 65, digits = 4, scipen = 4, fig.width = 4,
    fig.height = 4)
# Clear the workspace and console.
rm(list = ls(all.names = TRUE))
cat("\f")
```

```
how_long <- Sys.time()
set.seed(1222023)
library(knitr)
```

1.2 Packages

```
library(psych)
library(ltm)
library(readr)
library(tidyr)
library(minpack.lm)
library(MuMIn)
library(janitor)
library(data.table)
library(tidyverse)
library(corrr)
library(bayestestR)
library(posterior)
```

```r
library(ggpubr)
library(apaTables)
library(ggdist)
library(lmtest)
library(betareg)
library(emmeans)
library(merTools)
library(StepBeta)
library(multcomp)
library(modelr)
library(scales)
library(lemon)
library(broom)
library(ggpattern)
library(fastDummies)
library(glmmTMB)
library(here)
```

1.3 Get the Data

```r
# Get the data from the working directory.
setwd(here("r code"))
source("Data.R")
source("Function.R")
```

2 Group-Level Analyses

The following analyses were conducted to establish the representativeness of the current discounting data: For each discounting procedure (Adj-Amt and MCQ), we examined whether the results reflected the systematic changes in preference usually observed as the amount of reward and/or the delay to a reward increase.

2.1 Adjusting-Amount

```
# Hyperboloid Model Fit
r2_aa_grp <- matrix(NA, nrow = 3, ncol = 2) |>
    `colnames<-`(c("Prolific", "MTurk")) |>
    `rownames<-`(c("$30", "$80", "$500"))
for (x in 1:3) {
    r2_aa_grp[x, 1] <- rsquare(nlsLM(med_sv ~ 1/(1 + exp(k) * iv)^(b),
        data = subset(disc_grp_df, amt == x & procedure == "aa" &
            provider == "Prolific"), start = list(k = -4, b = 1),
        control = list(maxiter = 1000)), data = subset(disc_grp_df,
        amt == x & procedure == "aa" & provider == "Prolific"))
    r2_aa_grp[x, 2] <- rsquare(nlsLM(med_sv ~ 1/(1 + exp(k) * iv)^(b),
        data = subset(disc_grp_df, amt == x & procedure == "aa" &
            provider == "MTurk"), start = list(k = -4, b = 1), control = list(maxiter = 1000)),
        data = subset(disc_grp_df, amt == x & procedure == "aa" &
            provider == "MTurk"))
}
print(r2_aa_grp, digits = 3)
```

```
##      Prolific MTurk
## $30     0.980 0.882
## $80     0.983 0.996
## $500    0.979 0.987
```

```
# Amount Effect
group_by(filter(behav, procedure == "aa"), provider) |>
    summarise(pvalue = tidy(summary(glht(glmmTMB(atheoretical ~ -1 +
        as.factor(amt) + (1 | id), family = beta_family()), linfct = matrix(c(-1,
        0, 1), nc = 3), alternative = "two.sided", rhs = 0)))[[6]][[1]])
```

```
## # A tibble: 2 x 2
##   provider pvalue
##   <chr>     <dbl>
## 1 MTurk         0
## 2 Prolific      0
```

2.2 MCQ

```
# Logistic Growth Model Fit
r2_mcq_grp <- matrix(NA, nrow = 3, ncol = 2) |>
    `colnames<-`(c("Prolific", "MTurk")) |>
    `rownames<-`(c("$30", "$55", "$80"))
for (x in 1:3) {
    r2_mcq_grp[x, 1] <- rsquare(nlsLM(mean_sv ~ 1/(1 + exp(-(iv -
        (x)) * (r))), data = subset(disc_grp_df, amt == x & procedure ==
        "mcq" & provider == "Prolific"), start = list(x = -4, r = 1),
        control = list(maxiter = 1000)), data = subset(disc_grp_df,
        amt == x & procedure == "mcq" & provider == "Prolific"))
    r2_mcq_grp[x, 2] <- rsquare(nlsLM(mean_sv ~ 1/(1 + exp(-(iv -
        (x)) * (r))), data = subset(disc_grp_df, amt == x & procedure ==
        "mcq" & provider == "MTurk"), start = list(x = -4, r = 1),
        control = list(maxiter = 1000)), data = subset(disc_grp_df,
        amt == x & procedure == "mcq" & provider == "MTurk"))
}
print(r2_mcq_grp, digits = 3)

##     Prolific MTurk
## $30    0.991 0.992
## $55    0.984 0.983
## $80    0.997 0.968

# Amount Effect
group_by(filter(behav, procedure == "mcq"), provider) |>
    summarise(pvalue = tidy(summary(glht(glmer(cbind(atheoretical,
        9 - atheoretical) ~ -1 + as.factor(amt) + (1 | id), family = binomial()),
        linfct = matrix(c(contr.poly(3)[, 1]), nc = 3), alternative = "two.sided",
        rhs = 0)))[[6]][[1]])

## # A tibble: 2 x 2
##   provider   pvalue
##   <chr>       <dbl>
## 1 MTurk    6.60e-10
## 2 Prolific 1.06e-11
```

2.3 Within-Procedure Correlation

The following analyses were conducted to evaluate the correlations among Amounts within each discounting procedure.

cor1_2 represents the correlation between small and medium amounts.
cor1_3 represents the correlation between small and large amounts.
cor2_3 represents the correlation between medium and large amounts.

It should be noted that the log k was calculated using Stan in the published article, so there might be minor differences in p-values between the following outputs and the ones in the published article.

```
# Correlation within each discounting measure from each
# procedure
behav %>%
    pivot_longer(names_to = "measure", values_to = "value", cols = c(atheoretical,
        theoretical)) %>%
    mutate(amt = ifelse(amt == 1, "small", ifelse(amt == 2, "medium",
        "large"))) %>%
    pivot_wider(names_from = amt, values_from = value) %>%
    group_by(measure, procedure, provider) %>%
    summarise(cor1_2 = cor(small, medium), cor1_3 = cor(small, large),
        cor2_3 = cor(medium, large))

## # A tibble: 8 x 6
## # Groups:   measure, procedure [4]
##   measure      procedure provider cor1_2 cor1_3 cor2_3
##   <chr>        <chr>     <chr>     <dbl>  <dbl>  <dbl>
## 1 atheoretical aa        MTurk     0.877  0.841  0.879
## 2 atheoretical aa        Prolific  0.841  0.737  0.870
## 3 atheoretical mcq       MTurk     0.897  0.890  0.925
## 4 atheoretical mcq       Prolific  0.898  0.837  0.882
## 5 theoretical  aa        MTurk     0.874  0.840  0.865
## 6 theoretical  aa        Prolific  0.839  0.733  0.832
## 7 theoretical  mcq       MTurk     0.887  0.841  0.895
## 8 theoretical  mcq       Prolific  0.902  0.834  0.874

# Correlation between discounting measures (theoretical and
# atheoretical) from each procedure
behav %>%
    group_by(provider, procedure, amt) %>%
    summarise(cor = cor(atheoretical, theoretical)) %>%
    mutate(amt = ifelse(amt == 1, "small", ifelse(amt == 2, "medium",
        "large"))) %>%
    pivot_wider(names_from = amt, values_from = cor)

## # A tibble: 4 x 5
```

```
## # Groups:   provider, procedure [4]
##   provider procedure  small medium  large
##   <chr>    <chr>      <dbl>  <dbl>  <dbl>
## 1 MTurk    aa        -0.973 -0.979 -0.982
## 2 MTurk    mcq       -0.967 -0.983 -0.975
## 3 Prolific aa        -0.963 -0.958 -0.956
## 4 Prolific mcq       -0.995 -1.00  -0.995
```

2.4 Between-Procedure Correlation

The following analyses were conducted to evaluate the intercorrelations among Amounts and the two discounting procedures.

cor_atheoretical represents the correlation between AuC (Adj-Amt) and delayed-choice proportion (MCQ).
cor_theoretical represents the correlation between log k measures from both procedures.

It should be noted that the log k was calculated using Stan in the published article, so there might be minor differences in p-values between the following outputs and the ones in the published article.

```
filter(behav, (procedure == "aa" & amt != 3) | (procedure == "mcq" &
    amt != 2)) |>
    mutate(amt = ifelse(amt == 1, "$30", "$80")) |>
    pivot_wider(names_from = procedure, values_from = c(atheoretical,
        theoretical)) |>
    group_by(provider, amt) |>
    summarise(cor_atheoretical = cor(atheoretical_aa, atheoretical_mcq),
        cor_theoretical = cor(theoretical_aa, theoretical_mcq))
```

```
## # A tibble: 4 x 4
## # Groups:   provider [2]
##   provider amt   cor_atheoretical cor_theoretical
##   <chr>    <chr>            <dbl>           <dbl>
## 1 MTurk    $30              0.790           0.769
## 2 MTurk    $80              0.820           0.785
## 3 Prolific $30              0.809           0.759
## 4 Prolific $80              0.821           0.799
```

3 log k Comparison

The following analyses were conducted to compare the absolute degree of discounting between the two discounting procedure using log k based on simple hyperbolic model.

It should be noted that the log k was calculated using Stan in the published article, so there might be minor differences in p-values between the following outputs and the ones in the published article.

```
tem_dat <- filter(behav, (procedure == "aa" & amt != 3) | (procedure ==
    "mcq" & amt != 2)) |>
    mutate(amt = ifelse(amt == 1, "$30", "$80"))
logk_mod <- aov(theoretical ~ (provider + amt + procedure)^2, data = tem_dat)
summary(logk_mod)

##                       Df Sum Sq Mean Sq F value  Pr(>F)
## provider               1    454     454  131.32 < 2e-16 ***
## amt                    1    141     141   40.78 2.2e-10 ***
## procedure              1      9       9    2.47   0.117
## provider:amt           1      4       4    1.08   0.298
## provider:procedure     1      4       4    1.16   0.283
## amt:procedure          1     20      20    5.71   0.017 *
## Residuals           1565   5413       3
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# Contrast
AmtProcedure_emm <- emmeans(logk_mod, c("amt", "procedure"))
AmtProvider_emm <- emmeans(logk_mod, c("amt", "provider"))
ProcedureProvider_emm <- emmeans(logk_mod, c("procedure", "provider"))
cld(AmtProcedure_emm, alpha = 0.05, adjust = "holm", details = TRUE)

## $emmeans
##  amt procedure emmean     SE   df lower.CL upper.CL .group
##  $80 mcq        -4.67 0.0959 1565    -4.91    -4.43  1
##  $80 aa         -4.62 0.0959 1565    -4.86    -4.38  1
##  $30 aa         -4.22 0.0959 1565    -4.46    -3.98   2
##  $30 mcq        -3.83 0.0959 1565    -4.07    -3.59    3
##
## Results are averaged over the levels of: provider
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## P value adjustment: holm method for 6 tests
```

```
## significance level used: alpha = 0.05
## NOTE: If two or more means share the same grouping symbol,
##       then we cannot show them to be different.
##       But we also did not show them to be the same.
##
## $comparisons
##  contrast          estimate    SE   df t.ratio p.value
##  $80 aa - $80 mcq    0.0524 0.135 1565   0.389  0.6974
##  $30 aa - $80 mcq    0.4510 0.137 1565   3.303  0.0039
##  $30 aa - $80 aa     0.3986 0.135 1565   2.961  0.0093
##  $30 mcq - $80 mcq   0.8471 0.135 1565   6.292  <.0001
##  $30 mcq - $80 aa    0.7947 0.137 1565   5.820  <.0001
##  $30 mcq - $30 aa    0.3961 0.135 1565   2.942  0.0093
##
## Results are averaged over the levels of: provider
## P value adjustment: holm method for 6 tests

cld(AmtProvider_emm, alpha = 0.05, adjust = "holm", details = TRUE)

## $emmeans
##  amt provider emmean     SE   df lower.CL upper.CL .group
##  $80 Prolific  -5.25 0.1074 1565    -5.52    -4.98  1
##  $30 Prolific  -4.53 0.1074 1565    -4.79    -4.26   2
##  $80 MTurk     -4.04 0.0844 1565    -4.25    -3.83    3
##  $30 MTurk     -3.52 0.0844 1565    -3.73    -3.31     4
##
## Results are averaged over the levels of: procedure
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## P value adjustment: holm method for 6 tests
## significance level used: alpha = 0.05
## NOTE: If two or more means share the same grouping symbol,
##       then we cannot show them to be different.
##       But we also did not show them to be the same.
##
## $comparisons
##  contrast                    estimate    SE   df t.ratio p.value
##  $30 Prolific - $80 Prolific    0.723 0.152 1565   4.763  <.0001
##  $80 MTurk - $80 Prolific       1.207 0.137 1565   8.839  <.0001
##  $80 MTurk - $30 Prolific       0.484 0.137 1565   3.542  0.0004
##  $30 MTurk - $80 Prolific       1.729 0.137 1565  12.665  <.0001
##  $30 MTurk - $30 Prolific       1.006 0.137 1565   7.368  <.0001
```

```
##  $30 MTurk - $80 MTurk          0.522 0.119 1565   4.379  <.0001
##
## Results are averaged over the levels of: procedure
## P value adjustment: holm method for 6 tests
```

```
cld(ProcedureProvider_emm, alpha = 0.05, adjust = "holm", details = TRUE)
```

```
## $emmeans
##  procedure provider emmean     SE   df lower.CL upper.CL .group
##  aa        Prolific  -5.03 0.1074 1565    -5.29    -4.76  1
##  mcq       Prolific  -4.75 0.1074 1565    -5.02    -4.48  1
##  aa        MTurk     -3.82 0.0844 1565    -4.03    -3.60   2
##  mcq       MTurk     -3.75 0.0844 1565    -3.96    -3.54   2
##
## Results are averaged over the levels of: amt
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## P value adjustment: holm method for 6 tests
## significance level used: alpha = 0.05
## NOTE: If two or more means share the same grouping symbol,
##       then we cannot show them to be different.
##       But we also did not show them to be the same.
##
## $comparisons
##  contrast                   estimate    SE   df t.ratio p.value
##  mcq Prolific - aa Prolific   0.2757 0.152 1565   1.815  0.1393
##  aa MTurk - aa Prolific       1.2103 0.137 1565   8.863  <.0001
##  aa MTurk - mcq Prolific      0.9346 0.137 1565   6.845  <.0001
##  mcq MTurk - aa Prolific      1.2784 0.137 1565   9.362  <.0001
##  mcq MTurk - mcq Prolific     1.0027 0.137 1565   7.343  <.0001
##  mcq MTurk - aa MTurk         0.0681 0.119 1565   0.571  0.5683
##
## Results are averaged over the levels of: amt
## P value adjustment: holm method for 6 tests
```

4 Comparsion of Choice Patterns

4.1 Steep Discounter

The following analyses were conducted to evaluate the number of participants who were steep discounter in each procedure.

```
group_by(behav, provider,procedure,id)[,-11] |> summarise_all(mean) |>
  pivot_wider(names_from = procedure,values_from = atheoretical) |>
  mutate(mcq = mcq*3,
         # AuC < .125 in the Adj-Amt
         auc_im = ifelse(aa < .125, 1, 0),
         # Delayed-choice proportion in the MCQ
         prop_im = ifelse(mcq <=3, 1, 0),
         # Steep discounter in both procedures
         im = ifelse(auc_im == 1 & prop_im == 1, 1, 0)) |>
  group_by(provider) |>
  # Calculate the number & proportion of steep discounter for each procedure
  summarise(AdjAmt_pr = sum(auc_im)/n(), AdjAmt_n = sum(auc_im), # Steep discounter in Adj-Amt
            MCQ_pr = sum(prop_im)/n(), MCQ_n = sum(prop_im),     # Steep discounter in MCQ
            Common_pr = sum(im)/n(),  Common_n = sum(im))        # Steep discounter in both proc
```

```
## # A tibble: 2 x 7
##   provider AdjAmt_pr AdjAmt_n MCQ_pr MCQ_n Common_pr Common_n
##   <chr>        <dbl>    <dbl>  <dbl> <dbl>     <dbl>    <dbl>
## 1 MTurk      0.218         53 0.239     58   0.169         41
## 2 Prolific   0.00667        1 0.0467     7   0.00667        1
```

```r
# Get system details.
S <- benchmarkme::get_sys_details()
GB <- memuse::Sys.meminfo()
```

The current machine uses the following CPU: Apple M1, with 8 cores and 16.000 GiB of RAM.

```r
sessionInfo()
```

```
## R version 4.3.3 (2024-02-29)
## Platform: aarch64-apple-darwin20 (64-bit)
## Running under: macOS Sonoma 14.3
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRlapack.dylib;  L
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## time zone: America/Chicago
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods
## [7] base
##
## other attached packages:
##  [1] benchmarkme_1.0.8 here_1.0.1        glmmTMB_1.1.9
##  [4] fastDummies_1.7.3 ggpattern_1.0.1   broom_1.0.5
##  [7] lemon_0.4.9       scales_1.3.0      modelr_0.1.11
## [10] multcomp_1.4-25   TH.data_1.1-2     survival_3.5-8
## [13] mvtnorm_1.2-4     StepBeta_2.1.0    merTools_0.6.2
## [16] arm_1.13-1        lme4_1.1-35.1     Matrix_1.6-5
## [19] emmeans_1.10.0    betareg_3.1-4     lmtest_0.9-40
## [22] zoo_1.8-12        ggdist_3.3.2      apaTables_2.0.8
## [25] ggpubr_0.6.0      posterior_1.5.0   bayestestR_0.13.2
## [28] corrr_0.4.4       lubridate_1.9.3   forcats_1.0.0
## [31] stringr_1.5.1     dplyr_1.1.4       purrr_1.0.2
## [34] tibble_3.2.1      ggplot2_3.5.0     tidyverse_2.0.0
## [37] data.table_1.15.2 janitor_2.2.0     MuMIn_1.47.5
## [40] minpack.lm_1.2-4  tidyr_1.3.1       readr_2.1.5
```

```
## [43] ltm_1.2-0          polycor_0.8-1     msm_1.7.1
## [46] MASS_7.3-60.0.1    psych_2.4.3       knitr_1.45
##
## loaded via a namespace (and not attached):
##  [1] tensorA_0.36.2.1      rstudioapi_0.15.0
##  [3] magrittr_2.0.3        estimability_1.5
##  [5] modeltools_0.2-23     nloptr_2.0.3
##  [7] vctrs_0.6.5           minqa_1.2.6
##  [9] rstatix_0.7.2         htmltools_0.5.7
## [11] distributional_0.4.0  Formula_1.2-5
## [13] parallelly_1.37.1     plyr_1.8.9
## [15] sandwich_3.1-0        TMB_1.9.10
## [17] admisc_0.35           mime_0.12
## [19] lifecycle_1.0.4       iterators_1.0.14
## [21] pkgconfig_2.0.3       R6_2.5.1
## [23] fastmap_1.1.1         future_1.33.1
## [25] shiny_1.8.0           snakecase_0.11.1
## [27] numDeriv_2016.8-1.1   digest_0.6.35
## [29] colorspace_2.1-0      furrr_0.3.1
## [31] rprojroot_2.0.4       fansi_1.0.6
## [33] timechange_0.3.0      httr_1.4.7
## [35] mgcv_1.9-1            abind_1.4-5
## [37] compiler_4.3.3        doParallel_1.0.17
## [39] bit64_4.0.5           aod_1.3.3
## [41] withr_3.0.0           backports_1.4.1
## [43] carData_3.0-5         highr_0.10
## [45] broom.mixed_0.2.9.4   ggsignif_0.6.4
## [47] tools_4.3.3           httpuv_1.6.14
## [49] nnet_7.3-19           glue_1.7.0
## [51] nlme_3.1-164          promises_1.2.1
## [53] grid_4.3.3            checkmate_2.3.1
## [55] memuse_4.2-3          generics_0.1.3
## [57] gtable_0.3.4          tzdb_0.4.0
## [59] hms_1.1.3             car_3.1-2
## [61] utf8_1.2.4            flexmix_2.3-19
## [63] foreach_1.5.2         pillar_1.9.0
## [65] vroom_1.6.5           later_1.3.2
## [67] benchmarkmeData_1.0.4 splines_4.3.3
## [69] lattice_0.22-6        bit_4.0.5
## [71] tidyselect_1.2.1      blme_1.0-5
## [73] gridExtra_2.3         stats4_4.3.3
```

```
## [75] xfun_0.42             expm_0.999-9
## [77] stringi_1.8.3         boot_1.3-30
## [79] evaluate_0.23         codetools_0.2-19
## [81] multcompView_0.1-10   cli_3.6.2
## [83] xtable_1.8-4          munsell_0.5.0
## [85] Rcpp_1.0.12           globals_0.16.3
## [87] coda_0.19-4.1         parallel_4.3.3
## [89] ellipsis_0.3.2        listenv_0.9.1
## [91] crayon_1.5.2          insight_0.19.10
## [93] combinat_0.0-8        rlang_1.1.3
## [95] formatR_1.14          mnormt_2.1.1

Sys.time() - how_long

## Time difference of 4.065 secs
```