The Influence of Fatigue on Usage of Model-Based vs Model-Free Reinforcement Learning Strategies

Colton Loftus, Maya Rozenshteyn

2022-12-15

Contents

```
setwd("./")

library(purrr)
library(readr)
library(jsonlite)

Attaching package: 'jsonlite'
The following object is masked from 'package:purrr':

    flatten
library("data.table")

Attaching package: 'data.table'
The following object is masked from 'package:purrr':

    transpose
The following objects are masked from 'package:dplyr':

    between, first, last
library(dplyr)
library(rlist)
library(magrittr)

Attaching package: 'magrittr'
The following object is masked from 'package:purrr':

    set_names
library(lme4)
Loading required package: Matrix
library(plotrix)
library(ggplot2)
```

```
# Preprocess experiment data to extract relevant information: reward,
# transition, first state choice, tiredness, experimental condition (morning
# vs evening), and task run (first or second time conducting the experiment).
```

```r
# Note 1: reward is coded as -1 (no reward) and 1 (reward), transition is coded as
# --~0.7 (rare transition) and ~0.3 (common transition), tiredness is code on a
# scale from -3 (least tired) to 3 (most tired), experimental condition is coded
# as 1 (morning) and -1 (evening) and task run as -1 (first run) and 1
# (second run).
# Note 2: "stay" is true when the participant selected the same first-stage
# choice in successive trials.
# Note 3: all predictors are zero-centered.
survey_response_to_val <- jsonlite::fromJSON("data/sleepiness.json")$meta
preprocess <- function(csv, survey_string, is_morning, morning_first, is_more_tired_condition, person_id
    relevant_data <- csv %>%
        dplyr::filter(trial_type == "two-step-trial") %>%
        select(state_1_choice, transition, outcome) %>%
        dplyr::filter(!grepl("null", state_1_choice)) %>%
        dplyr::filter(!grepl("null", transition)) %>%
        dplyr::filter(!grepl("null", outcome)) %>%
        mutate(state_1_choice = as.numeric(state_1_choice)) %>%
        mutate(outcome = as.numeric(outcome)) %>%
        mutate(next_state1_choice = lead(state_1_choice, 1)) %>%
        mutate(stay = state_1_choice == next_state1_choice) %>%
        mutate(tiredness = survey_response_to_val[survey_string][[1]]) %>%
        mutate(is_morning = is_morning) %>%
        mutate(task_run = ((is_morning && morning_first) || (is_morning && morning_first))) %>%
        mutate(is_morning = ifelse(is_morning == TRUE, 1, -1)) %>%
        mutate(task_run = ifelse(task_run == TRUE, -1, 1)) %>%
        mutate(subject = person_id) %>%
        dplyr::filter(!is.na(next_state1_choice)) %>%
        mutate(outcome = ifelse(outcome == 0, -1, outcome)) %>%
        as.data.frame()

        attr(relevant_data, "is_morning") <- is_morning
        attr(relevant_data, "task_run") <- (is_morning && morning_first) || (is_morning && morning_first
        attr(relevant_data, "tiredness") <- survey_response_to_val[survey_string][[1]]
        attr(relevant_data, "subject") <- person_id
        attr(relevant_data, "is_more_tired_condition") <- is_more_tired_condition

    return(relevant_data)
}
```

```r
all_data <- list()
# Read in and preprocess experimental data.
for (person in list.files("data/")) {
    if (!grepl("json", person)) {
        survey_response_map <- jsonlite::fromJSON(paste0("data/", person, "/meta.json"))$meta
        morning_first <- survey_response_map$first == "morning"
        for (csv in list.files(paste0("data/", person, "/"))) {
            if (!grepl("json", csv)) {
                csv_is_morning <- grepl("day", csv)
                if (csv_is_morning && morning_first | !csv_is_morning && !morning_first) {
                    survey_response <- survey_response_map["form1"]
                    is_more_tired_condition <- (survey_response_map["form1"][[1]] >= survey_response_map
                } else {
                    survey_response <- survey_response_map["form2"]
                    is_more_tired_condition <- (survey_response_map["form2"][[1]] > survey_response_map
                }
                read_in_csv <- read_csv(paste0("data/", person, "/", csv))
                processed_csv <- preprocess(read_in_csv, survey_response[[1]], csv_is_morning, morning_i
                all_data[[length(all_data) + 1]] <- processed_csv
            }
        }
    }
}

all_data_concat <- bind_rows(all_data) %>% mutate(transition = transition - mean(transition))
```

```
# Mixed effect modeling illustrating random and fixed effects of outcome and
# transition on stay.
base_reg <- glmer(stay ~ outcome*transition + (1 | subject), data = all_data_concat, family = binomial)
summary(base_reg)
Generalized linear mixed model fit by maximum likelihood
  (Laplace Approximation) [glmerMod]
 Family: binomial  ( logit )
Formula: stay ~ outcome * transition + (1 | subject)
   Data: all_data_concat

     AIC      BIC   logLik deviance df.resid
  4236.1   4268.6  -2113.0   4226.1     4973

Scaled residuals:
    Min      1Q  Median      3Q     Max
-7.8028  0.1570  0.3464  0.4352  1.2803

Random effects:
 Groups  Name        Variance Std.Dev.
 subject (Intercept) 1.052    1.026
Number of obs: 4978, groups:  subject, 16

Fixed effects:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)         1.73443    0.26089   6.648 2.97e-11 ***
outcome             0.12957    0.03979   3.257  0.00113 **
transition         -0.11357    0.08795  -1.291  0.19663
outcome:transition  0.24767    0.08805   2.813  0.00491 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) outcom trnstn
outcome      0.028
transition  -0.005  0.009
otcm:trnstn  0.005 -0.021  0.153
```

```
# Mixed effect modeling illustrating random and fixed effects of outcome,
# transition, task run, and time of day on stay.
tod_reg <- glmer(stay ~ outcome*transition*(task_run + is_morning) + (1 | subject), data = all_data_conc
summary(tod_reg)
Generalized linear mixed model fit by maximum likelihood
  (Laplace Approximation) [glmerMod]
```

```
 Family: binomial  ( logit )
Formula:
stay ~ outcome * transition * (task_run + is_morning) + (1 |
    subject)
   Data: all_data_concat

     AIC      BIC   logLik deviance df.resid
  4234.7   4319.3  -2104.3   4208.7     4965

Scaled residuals:
    Min      1Q  Median      3Q     Max
-7.9662  0.1630  0.3540  0.4498  1.3296

Random effects:
 Groups  Name        Variance Std.Dev.
 subject (Intercept) 1.069    1.034
Number of obs: 4978, groups:  subject, 16

Fixed effects:
                              Estimate Std. Error z value
(Intercept)                    1.65746    0.26553   6.242
outcome                        0.19214    0.04840   3.970
transition                    -0.06438    0.10644  -0.605
task_run                       0.17380    0.07858   2.212
is_morning                     0.13738    0.05631   2.440
outcome:transition             0.20535    0.10651   1.928
outcome:task_run              -0.14010    0.05767  -2.429
outcome:is_morning            -0.04169    0.05028  -0.829
transition:task_run           -0.12928    0.12809  -1.009
transition:is_morning         -0.03410    0.11246  -0.303
outcome:transition:task_run    0.10996    0.12812   0.858
outcome:transition:is_morning -0.02721    0.11248  -0.242
                              Pr(>|z|)
(Intercept)                   4.32e-10 ***
outcome                       7.19e-05 ***
transition                      0.5453
task_run                        0.0270 *
is_morning                      0.0147 *
outcome:transition              0.0538 .
outcome:task_run                0.0151 *
outcome:is_morning              0.4070
transition:task_run             0.3128
transition:is_morning           0.7617
outcome:transition:task_run     0.3908
outcome:transition:is_morning   0.8089
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
               (Intr) outcom trnstn tsk_rn is_mrn otcm:t
outcome         0.045
transition     -0.003 -0.007
task_run       -0.139 -0.114 -0.006
```

```
is_morning     -0.094 -0.048  0.006  0.706
otcm:trnstn     0.002 -0.012  0.217  0.007 -0.006
otcm:tsk_rn    -0.029 -0.562  0.010  0.123  0.049  0.003
otcm:s_mrnn    -0.012 -0.317 -0.003  0.049  0.103  0.009
trnstn:tsk_     0.001  0.010 -0.554 -0.027 -0.035 -0.147
trnstn:s_mr     0.001 -0.005 -0.316 -0.021 -0.045 -0.039
otcm:trnstn:t_  0.001  0.001 -0.147  0.019  0.035 -0.553
otcm:trnstn:s_ -0.003  0.010 -0.038  0.028  0.044 -0.316
               otcm:t_ otcm:s_ trnstn:t_ trnstn:s_
outcome
transition
task_run
is_morning
otcm:trnstn
otcm:tsk_rn
otcm:s_mrnn     0.607
trnstn:tsk_     0.025   0.036
trnstn:s_mr     0.036   0.047   0.616
otcm:trnstn:t_ -0.035  -0.036   0.138     0.017
otcm:trnstn:s_ -0.037  -0.054   0.018     0.057
               otcm:trnstn:t_
outcome
transition
task_run
is_morning
otcm:trnstn
otcm:tsk_rn
otcm:s_mrnn
trnstn:tsk_
trnstn:s_mr
otcm:trnstn:t_
otcm:trnstn:s_  0.616
```

```
# Mixed effect modeling illustrating random and fixed effects of outcome,
# transition, task run, and tiredness on stay.
tiredness_reg <- glmer(stay ~ outcome*transition*(task_run + tiredness) + (1 | subject), data = all_data
summary(tiredness_reg)
Generalized linear mixed model fit by maximum likelihood
  (Laplace Approximation) [glmerMod]
 Family: binomial  ( logit )
Formula:
stay ~ outcome * transition * (task_run + tiredness) + (1 | subject)
   Data: all_data_concat

     AIC      BIC   logLik deviance df.resid
  4207.0   4291.6  -2090.5   4181.0     4965

Scaled residuals:
    Min      1Q  Median      3Q     Max
-8.6816  0.1629  0.3387  0.4576  1.2465

Random effects:
 Groups  Name        Variance Std.Dev.
 subject (Intercept) 1.021    1.01
```

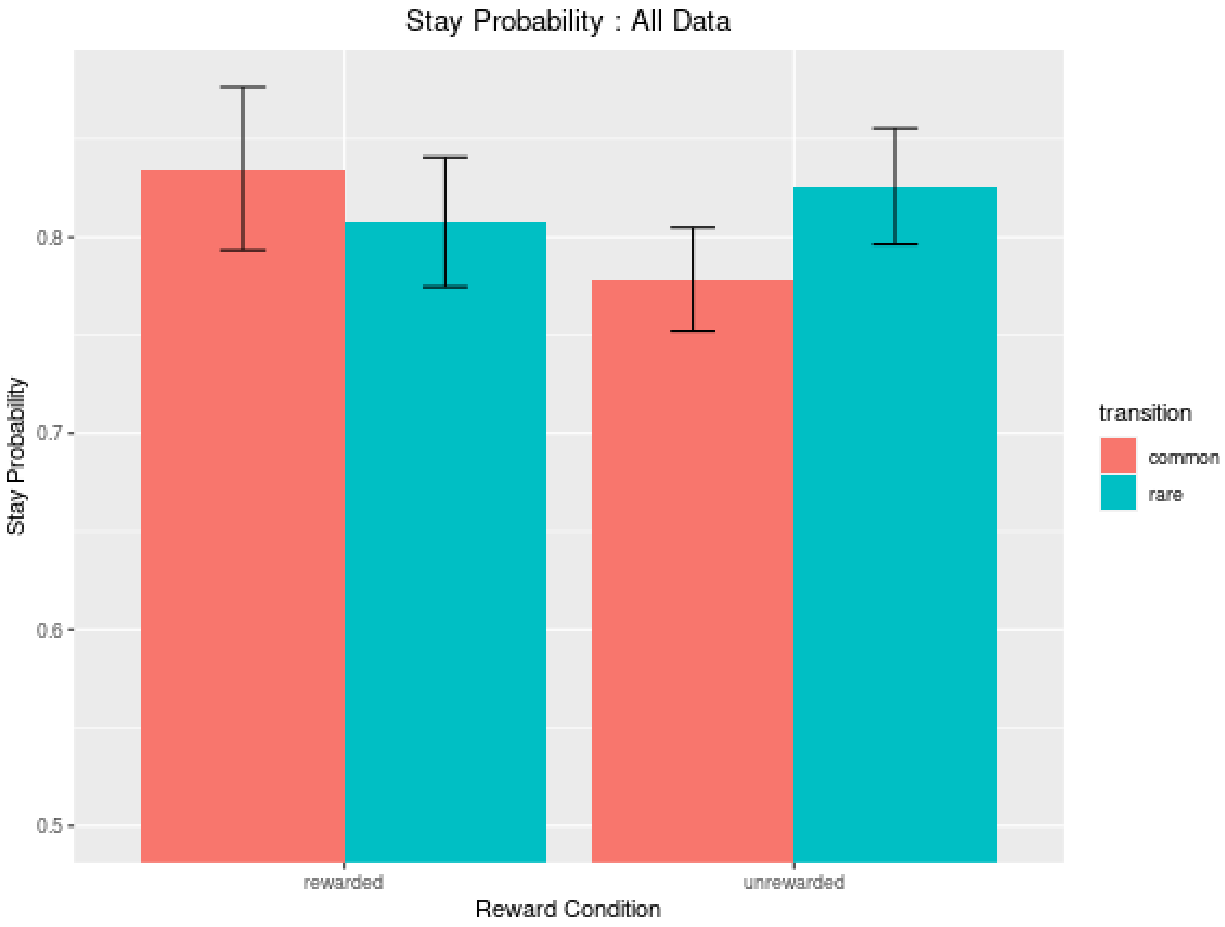
```
Number of obs: 4978, groups:  subject, 16

Fixed effects:
                             Estimate Std. Error z value
(Intercept)                   1.90075    0.27577   6.893
outcome                       0.44841    0.07400   6.060
transition                    0.23751    0.15229   1.560
task_run                      0.02491    0.05631   0.442
tiredness                     0.09960    0.06194   1.608
outcome:transition            0.57259    0.15258   3.753
outcome:task_run             -0.16694    0.04775  -3.496
outcome:tiredness             0.14152    0.02996   4.723
transition:task_run          -0.16360    0.10405  -1.572
transition:tiredness          0.15217    0.06218   2.447
outcome:transition:task_run   0.04690    0.10406   0.451
outcome:transition:tiredness  0.19969    0.06244   3.198
                             Pr(>|z|)    
(Intercept)                  5.48e-12 ***
outcome                      1.36e-09 ***
transition                   0.118862    
task_run                     0.658304    
tiredness                    0.107838    
outcome:transition           0.000175 ***
outcome:task_run             0.000472 ***
outcome:tiredness            2.32e-06 ***
transition:task_run          0.115871    
transition:tiredness         0.014391 *  
outcome:transition:task_run  0.652217    
outcome:transition:tiredness 0.001384 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) outcom trnstn tsk_rn trdnss otcm:trn otcm:_
outcome      0.107                                            
transition   0.037  0.208                                     
task_run    -0.136 -0.113 -0.026                              
tiredness    0.345  0.162  0.063 -0.110                       
otcm:trnstn  0.057  0.141  0.257 -0.015  0.107                
otcm:tsk_rn -0.044 -0.491 -0.036  0.191 -0.027 -0.031         
otcm:trdnss  0.090  0.777  0.205 -0.024  0.180  0.135   -0.256
trnstn:tsk_ -0.012 -0.035 -0.494  0.002 -0.021 -0.144    0.024
trnstn:trdn  0.032  0.201  0.747 -0.012  0.044  0.169   -0.034
otcm:trns:_ -0.010 -0.031 -0.143  0.011 -0.026 -0.490    0.006
otcm:trnst:  0.055  0.131  0.169 -0.017  0.098  0.748   -0.014
            otcm:trd trns:_ trnst: otc::_
outcome                                  
transition                               
task_run                                 
tiredness                                
otcm:trnstn                              
otcm:tsk_rn                              
otcm:trdnss                              
```

```
trnstn:tsk_ -0.035
trnstn:trdn  0.188   -0.255
otcm:trns:_ -0.015    0.225 -0.029
otcm:trnst:  0.119   -0.029  0.179 -0.251
```

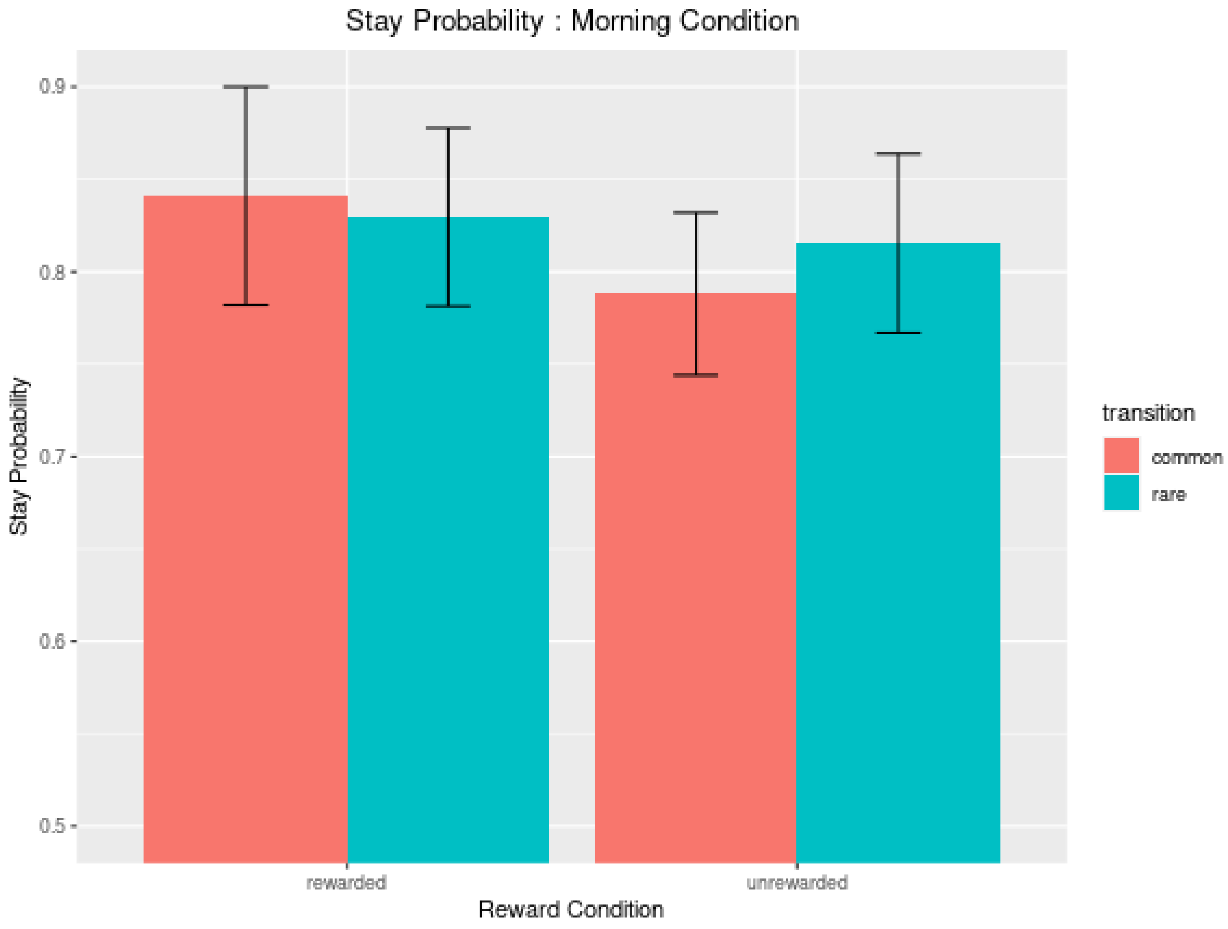
```
sum_decision_type <- function(df, stay, outcome, transition) {
    return(sum(df$stay == stay & df$outcome == outcome & df$transition == transition))
}

stay_probability_computer <- function(df) {
  # Common Transition, Rewarded, Stay
  ct_r_s <- sum_decision_type(df, stay = TRUE, outcome = 1, transition = 1)

  # Common Transition, Rewarded, Leave
  ct_r_l <- sum_decision_type(df, stay = FALSE, outcome = 1, transition = 1)

  # Rare Transition, Rewarded, Stay
  rt_r_s <- sum_decision_type(df, stay = TRUE, outcome = 1, transition = 0)

  # Rare Transition, Rewarded, Leave
  rt_r_l <- sum_decision_type(df, stay = FALSE, outcome = 1, transition = 0)

  # Common Transition, Unrewarded, Stay
  ct_ur_s <- sum_decision_type(df, stay = TRUE, outcome = -1, transition = 1)

  # Common Transition, Unrewarded, Leave
  ct_ur_l <- sum_decision_type(df, stay = FALSE, outcome = -1, transition = 1)

  # Rare Transition, Unrewarded, Stay
  rt_ur_s <- sum_decision_type(df, stay = TRUE, outcome = -1, transition = 0)

  # Rare Transition, Unrewarded, Leave
  rt_ur_l <- sum_decision_type(df, stay = FALSE, outcome = -1, transition = 0)

  # Common Transition, Rewarded
  b1 <- ct_r_s / (ct_r_s + ct_r_l)
  # Rare Transition, Rewarded
  b2 <- rt_r_s / (rt_r_s + rt_r_l)
  # Common Transition, Unrewarded
  b3 <- ct_ur_s / (ct_ur_s + ct_ur_l)
  # Rare Transition, Unrewarded
  b4 <- rt_ur_s / (rt_ur_s + rt_ur_l)

  res = c(b1, b2, b3, b4)
  names(res) <- c("b1", "b2", "b3", "b4")
  return(res)
}

bp_val_averger <- function(bp_vals) {
  b1s <- c()
  b2s <- c()
  b3s <- c()
  b4s <- c()
  for (i in 1:length(bp_vals)) {
```

```r
    b1s <- c(b1s, bp_vals[[i]][["b1"]])
    b2s <- c(b2s, bp_vals[[i]][["b2"]])
    b3s <- c(b3s, bp_vals[[i]][["b3"]])
    b4s <- c(b4s, bp_vals[[i]][["b4"]])
  }

  b1_av <- mean(b1s)
  b2_av <- mean(b2s)
  b3_av <- mean(b3s)
  b4_av <- mean(b4s)

  b1_stderr <- std.error(b1s)
  b2_stderr <- std.error(b2s)
  b3_stderr <- std.error(b3s)
  b4_stderr <- std.error(b4s)

  return(c(b1_av, b2_av, b3_av, b4_av, b1_stderr, b2_stderr, b3_stderr, b4_stderr))
}

stay_bar_plot_creator <- function(bars_to_plot, title) {
  reward_type <- c(rep("rewarded", 2), rep("unrewarded", 2))
  transition <- rep(c("common", "rare"), 2)
  value <- bars_to_plot[1:4]
  se <- bars_to_plot[5:8]
  data <- data.frame(reward_type, transition, value)
  ggplot(data, aes(fill = transition, y = value, x = reward_type)) +
      geom_bar(position = "dodge", stat = "identity") +
      labs(y= "Stay Probability", x = "Reward Condition", title = title) +
      coord_cartesian(ylim = c(.5, max(value) + max(se))) +
      theme(plot.title = element_text(hjust = 0.5)) +
      geom_errorbar(aes(ymin=value-se, ymax=value+se),
                    width=.2,
                    position=position_dodge(.9))
}
```

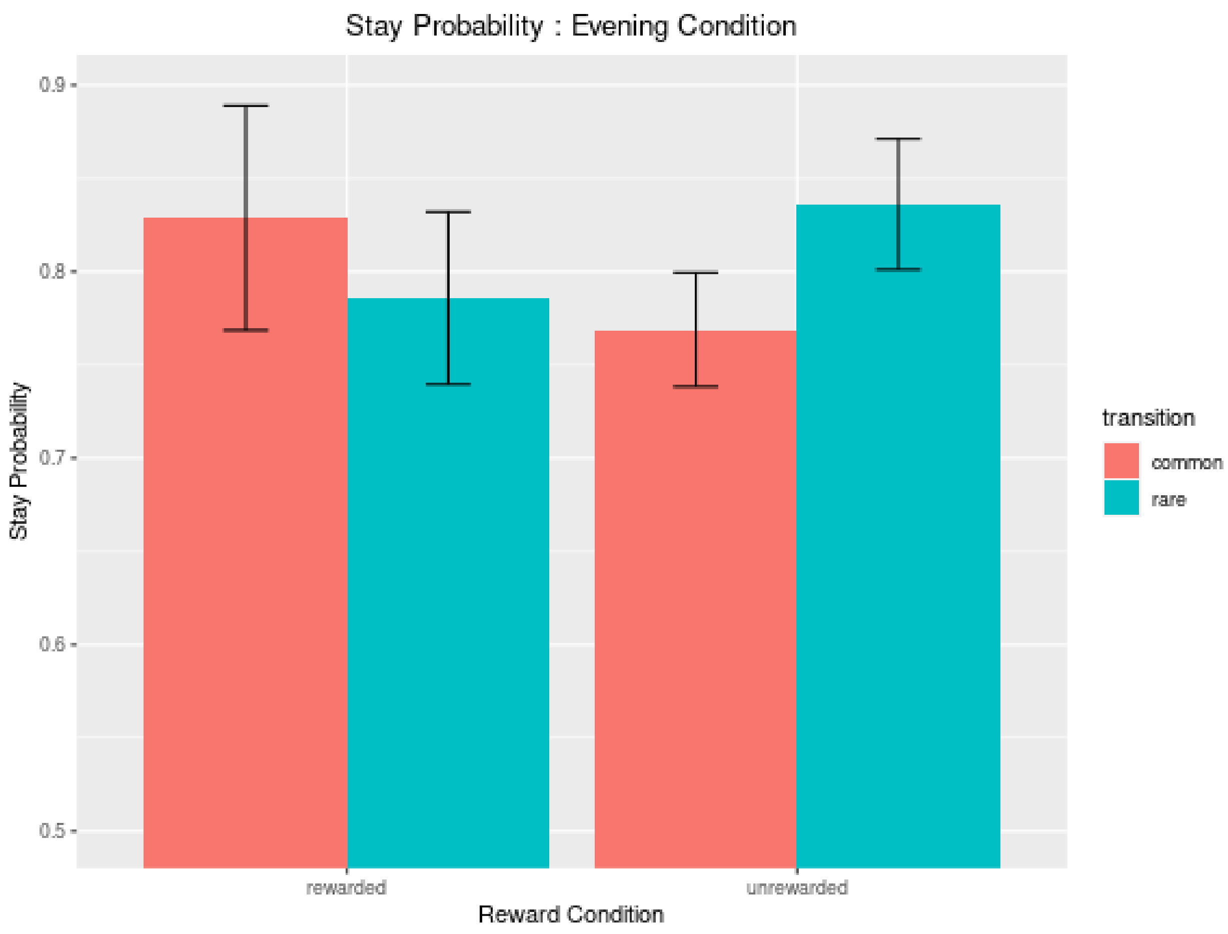
```r
all_data_processed <- all_data %>%
    map(stay_probability_computer)

stay_bar_plot_creator(bp_val_averger(all_data_processed), "Stay Probability : All Data")
```



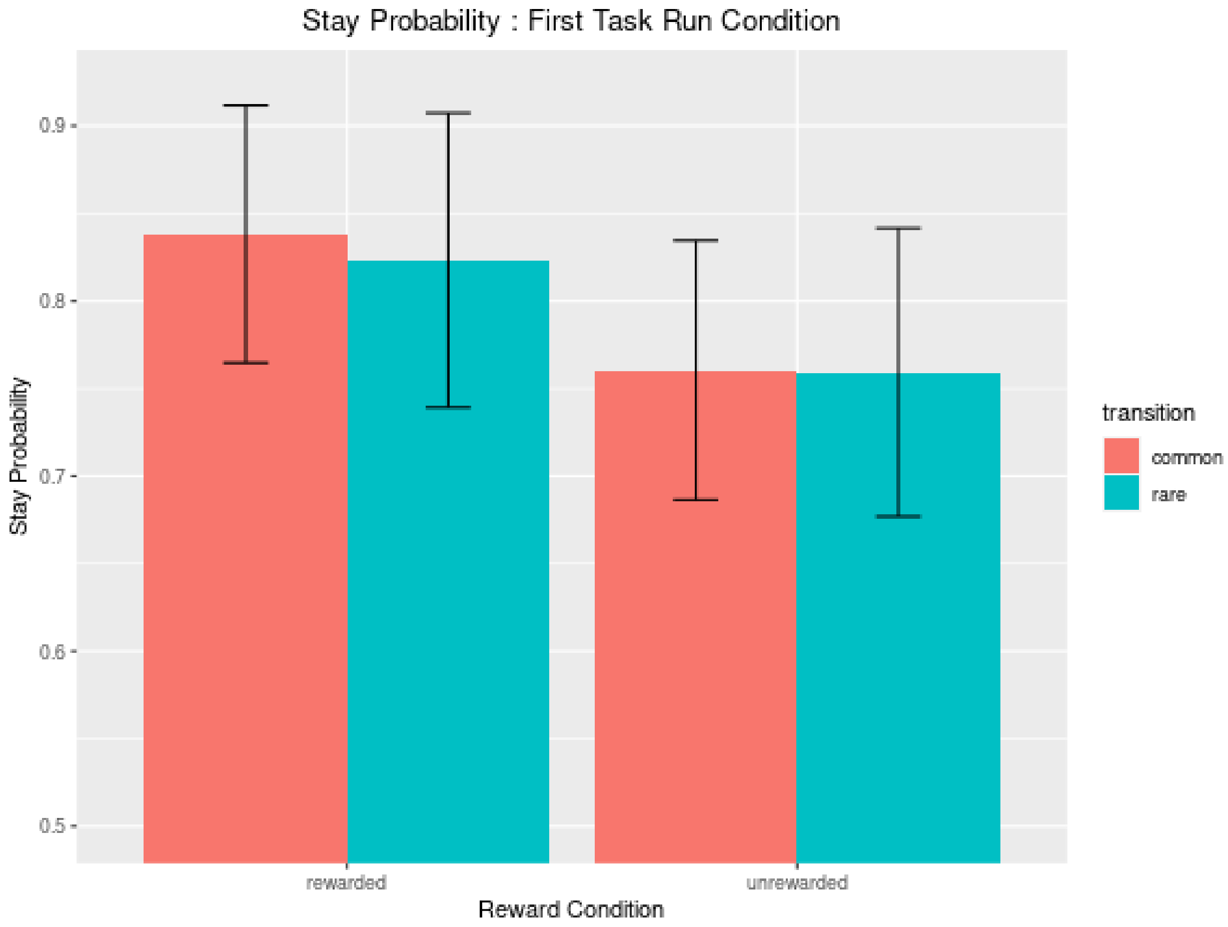
```
morning_data <- all_data %>%
    list.filter(attr(., "is_morning") == TRUE) %>%
    map(stay_probability_computer)

stay_bar_plot_creator(bp_val_averger(morning_data), "Stay Probability : Morning Condition")
```



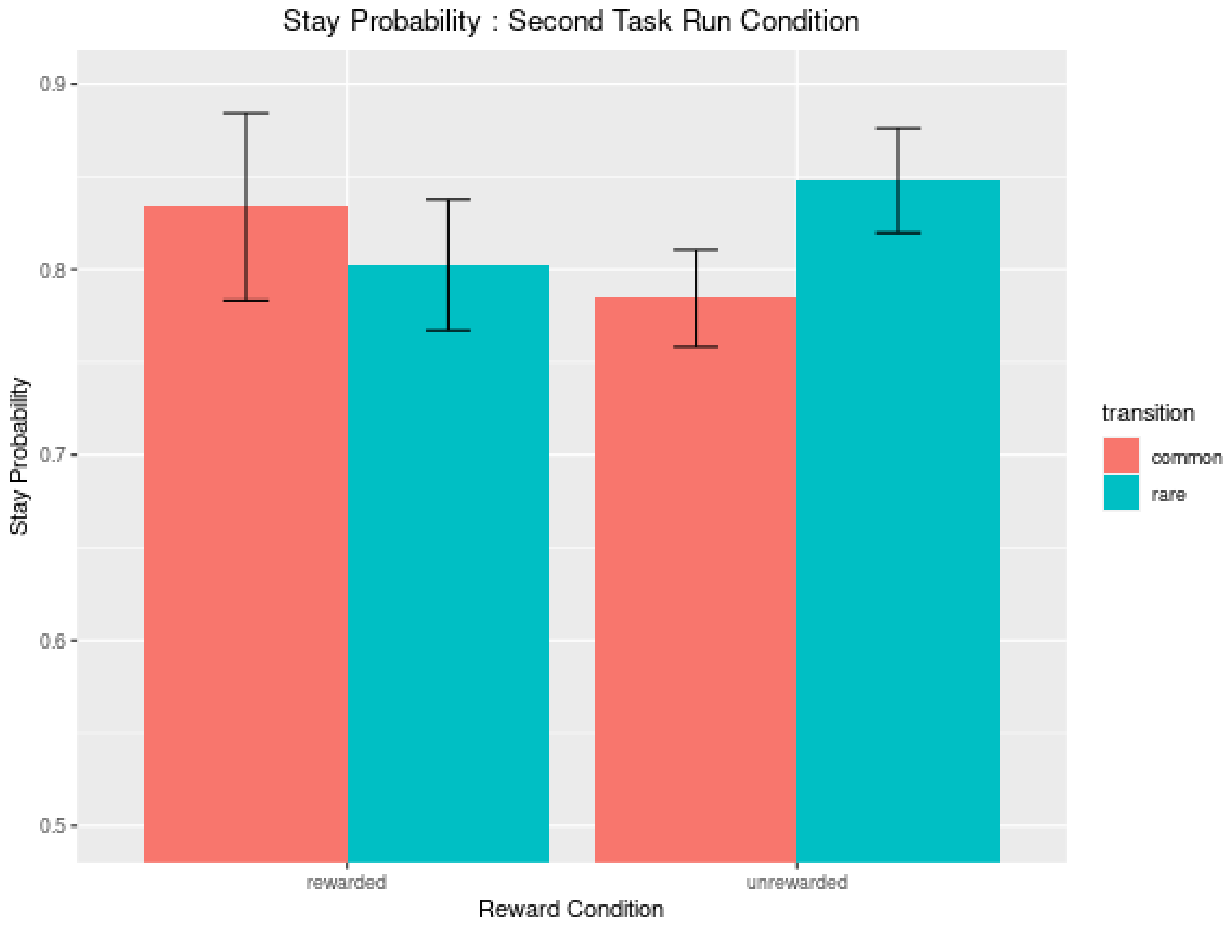
```
evening_data <- all_data %>%
    list.filter(attr(., "is_morning") == FALSE) %>%
    map(stay_probability_computer)

stay_bar_plot_creator(bp_val_averger(evening_data), "Stay Probability : Evening Condition")
```



```
first_run_data <- all_data %>%
    list.filter(attr(., "task_run") == TRUE) %>%
    map(stay_probability_computer)

stay_bar_plot_creator(bp_val_averger(first_run_data), "Stay Probability : First Task Run Condition")
```

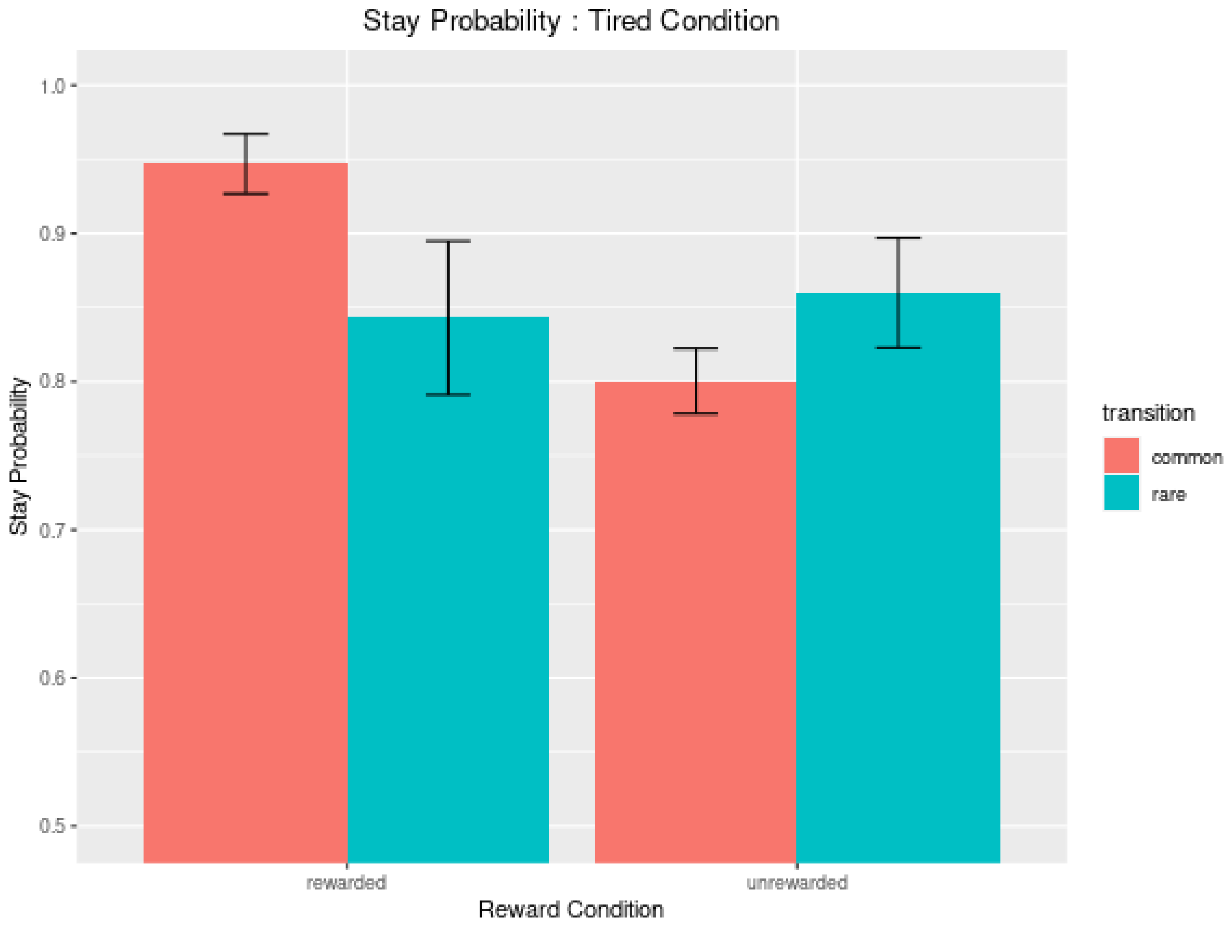


```
second_run_data <- all_data %>%
    list.filter(attr(., "task_run") == FALSE) %>%
    map(stay_probability_computer)

stay_bar_plot_creator(bp_val_averger(second_run_data), "Stay Probability : Second Task Run Condition")
```

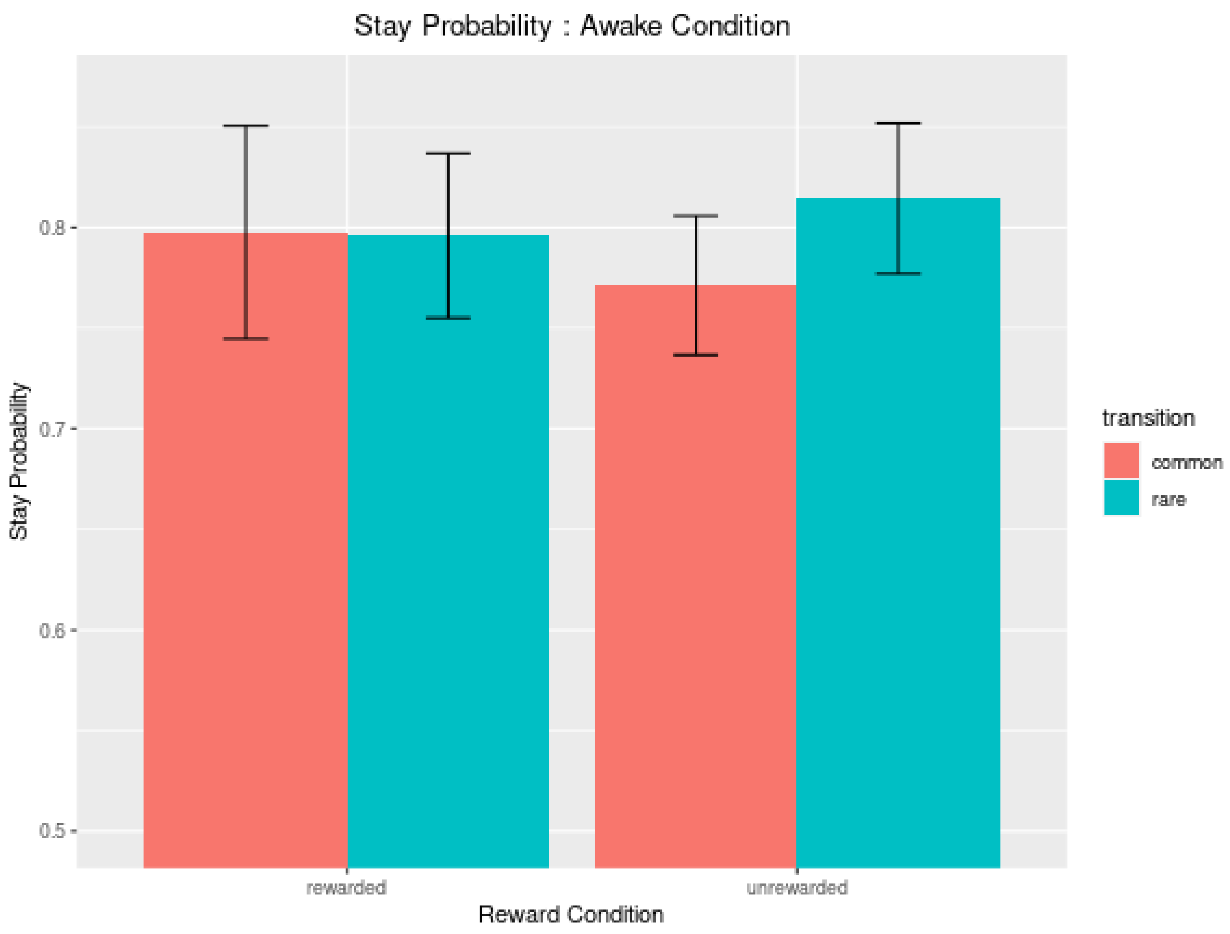


Tiredness Plotting Method 1: Tiredness Scores

```
tired_data <- all_data %>%
    list.filter(attr(., "tiredness") >= 0) %>%
    map(stay_probability_computer)

stay_bar_plot_creator(bp_val_averger(tired_data), "Stay Probability : Tired Condition")
```

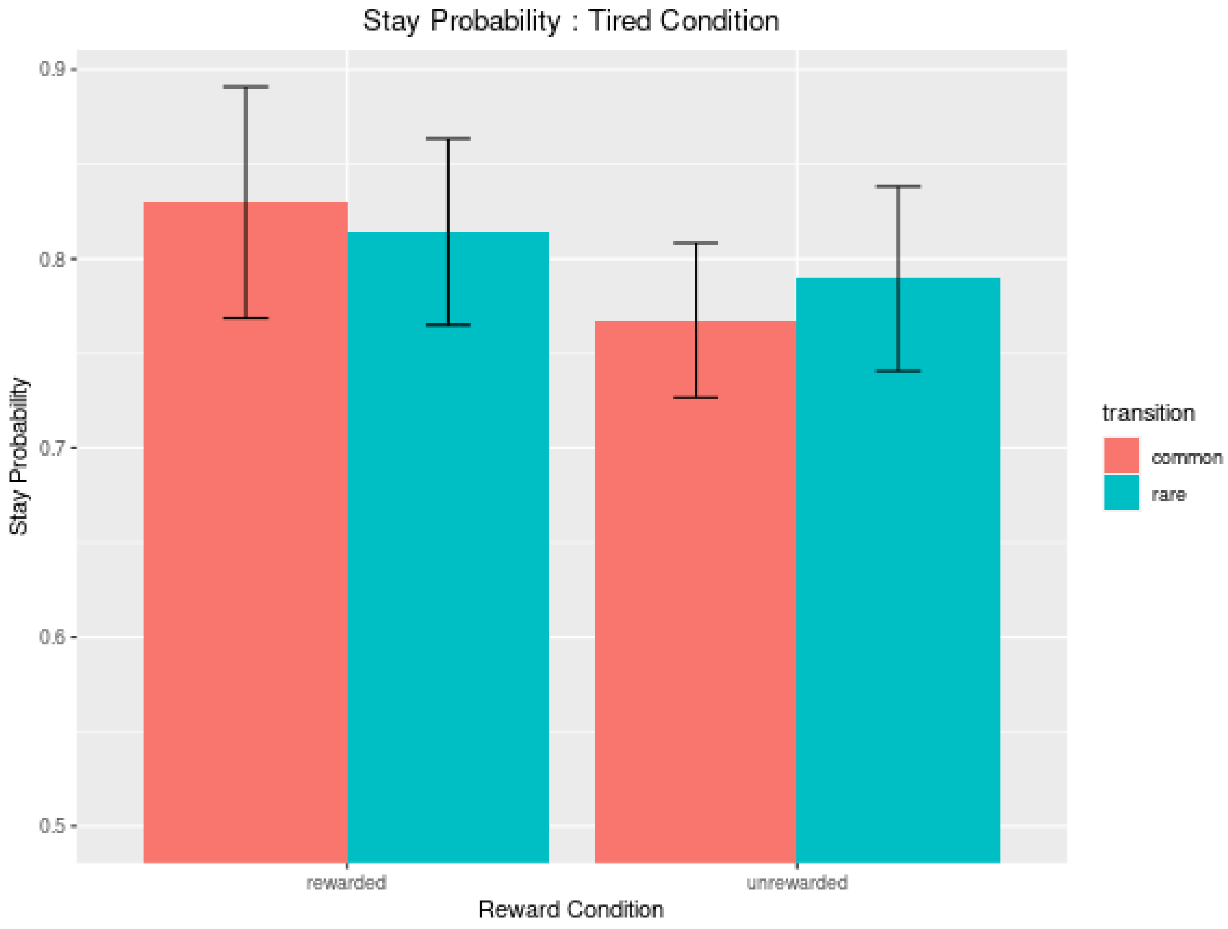


```r
awake_data <- all_data %>%
    list.filter(attr(., "tiredness") < 0) %>%
    map(stay_probability_computer)

stay_bar_plot_creator(bp_val_averger(awake_data), "Stay Probability : Awake Condition")
```

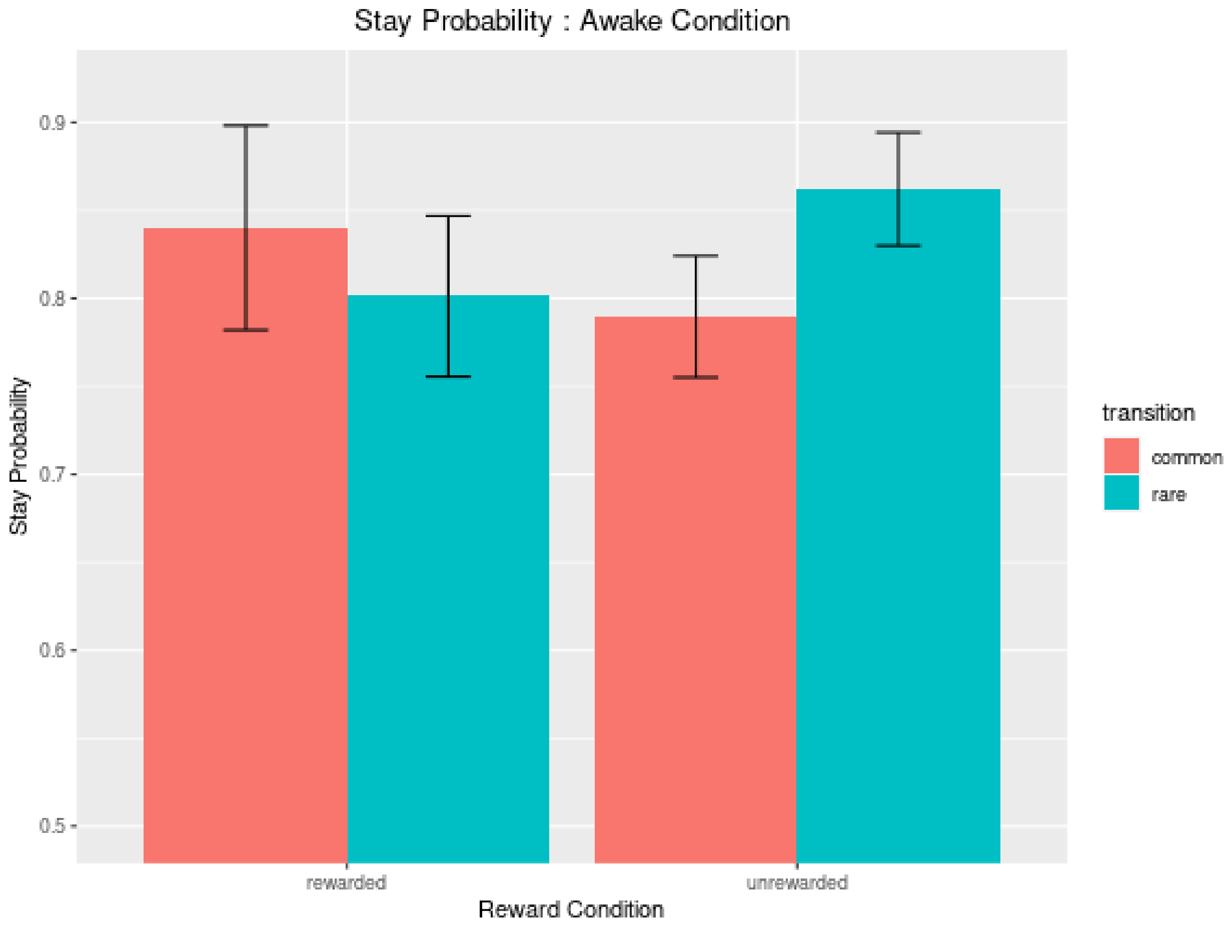


Tiredness Plotting Method 2: Relative Tiredness between Trials

```r
relatively_tired_data <- all_data %>%
    list.filter(attr(., "is_more_tired_condition") == TRUE) %>%
    map(stay_probability_computer)

stay_bar_plot_creator(bp_val_averger(relatively_tired_data), "Stay Probability : Tired Condition")
```



```
relatively_awake_data <- all_data %>%
    list.filter(attr(., "is_more_tired_condition") == FALSE) %>%
    map(stay_probability_computer)

stay_bar_plot_creator(bp_val_averger(relatively_awake_data), "Stay Probability : Awake Condition")
```


Stay Probability : Awake Condition
Stay Probability
0.9
0.8
0.7
0.6
0.5
rewarded
unrewarded
Reward Condition
transition
common
rare