

```
library(gplots)
library(patchwork)
```

```
## Warning: package 'patchwork' was built under R version 4.2.3
```

```
final_data <- read.csv(file = '../results_encoded_EXP1.csv')
```

```
model <- lm(Browse_Time ~ Prev_Length * Match_Score * Tile_Size * Prev_Type, data=final_data)
summary(model)
```

```
##
## Call:
## lm(formula = Browse_Time ~ Prev_Length * Match_Score * Tile_Size *
##     Prev_Type, data = final_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.3718 -0.6329 -0.0051  0.6431  2.8849
##
## Coefficients:
##              Estimate Std. Error t value
## (Intercept)    20.7468463   0.0240804  861.567
## Prev_Length      0.8635337   0.0240804   35.860
## Match_Score    -0.7245971   0.0240804  -30.091
## Tile_Size      -0.0325695   0.0240804   -1.353
## Prev_Type      -2.4833158   0.0240804 -103.126
## Prev_Length:Match_Score
##               1.4038212   0.0240804   58.297
## Prev_Length:Tile_Size
##             -0.0268820   0.0240804   -1.116
## Match_Score:Tile_Size
##            -0.0307261   0.0240804   -1.276
## Prev_Length:Prev_Type
##             0.0494300   0.0240804    2.053
## Match_Score:Prev_Type
##            -0.0061987   0.0240804   -0.257
## Tile_Size:Prev_Type
##             0.0027799   0.0240804    0.115
## Prev_Length:Match_Score:Tile_Size
##             0.0079499   0.0240804    0.330
## Prev_Length:Match_Score:Prev_Type
##             0.0068455   0.0240804    0.284
## Prev_Length:Tile_Size:Prev_Type
##             0.0001886   0.0240804    0.008
## Match_Score:Tile_Size:Prev_Type
##             0.0342813   0.0240804    1.424
## Prev_Length:Match_Score:Tile_Size:Prev_Type
##             0.0240772   0.0240804    1.000
##
##              Pr(>|t|)
## (Intercept)    <2e-16 ***
## Prev_Length    <2e-16 ***
## Match_Score    <2e-16 ***
## Tile_Size      0.1764
## Prev_Type      <2e-16 ***
## Prev_Length:Match_Score
##             <2e-16 ***
## Prev_Length:Tile_Size
##             0.2644
## Match_Score:Tile_Size
##             0.2021
## Prev_Length:Prev_Type
##             0.0403 *
## Match_Score:Prev_Type
##             0.7969
## Tile_Size:Prev_Type
##             0.9081
## Prev_Length:Match_Score:Tile_Size
##             0.7413
## Prev_Length:Match_Score:Prev_Type
##             0.7762
## Prev_Length:Tile_Size:Prev_Type
##             0.9938
```

```
## Match_Score:Tile_Size:Prev_Type          0.1548
## Prev_Length:Match_Score:Tile_Size:Prev_Type 0.3175
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9632 on 1584 degrees of freedom
## Multiple R-squared:  0.9111, Adjusted R-squared:  0.9103
## F-statistic: 1082 on 15 and 1584 DF, p-value: < 2.2e-16
```

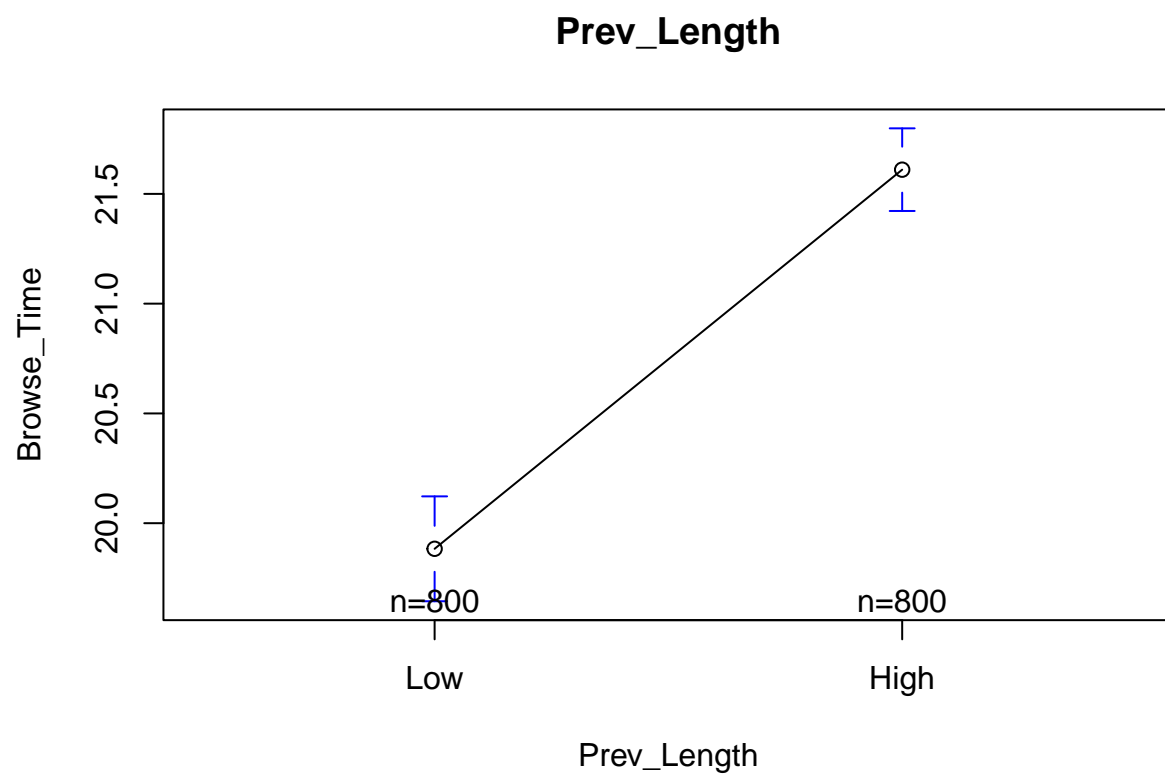
Reduced model

```
# formally test the reduced model
# insignificant terms
model_reduced <- lm(Browse_Time ~ Tile_Size +
                    Prev_Length:Tile_Size +
                    Match_Score:Tile_Size +
                    Prev_Length:Prev_Type +
                    Match_Score:Prev_Type +
                    Tile_Size:Prev_Type +
                    Prev_Length:Match_Score:Tile_Size +
                    Prev_Length:Match_Score:Prev_Type +
                    Prev_Length:Tile_Size:Prev_Type +
                    Match_Score:Tile_Size:Prev_Type +
                    Prev_Length:Match_Score:Tile_Size:Prev_Type,
                    data = final_data
                    )

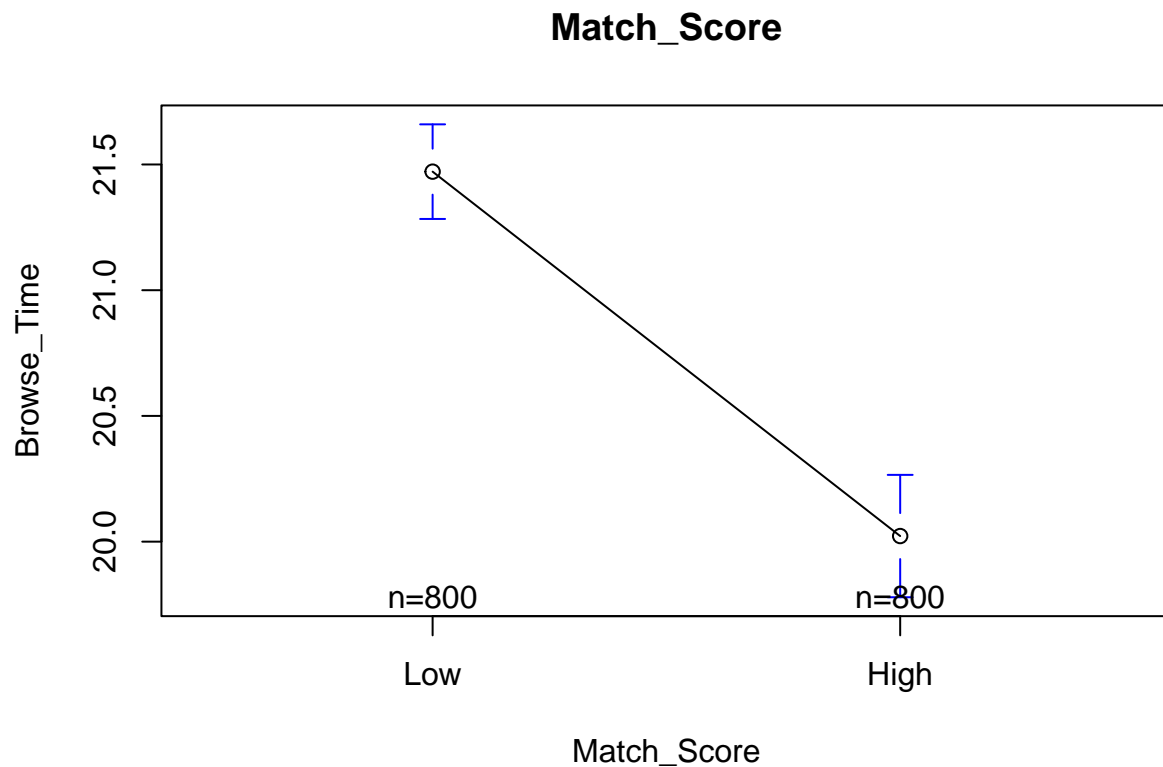
# summary(model_reduced)
```

Huge p-value for reduced model so fail to reject that the H0 (insignificant) is zero <- weird

```
plotmeans(Browse_Time ~ Prev_Length,
          data = final_data,
          # ylim = c(0.4, 0.7),
          main = "Prev_Length",
          legends = c("Low", "High"))
```



```
plotmeans(Browse_Time ~ Match_Score,  
  data = final_data,  
  #   ylim = c(0.4, 0.7),  
  main = "Match_Score",  
  legends = c("Low", "High"))
```



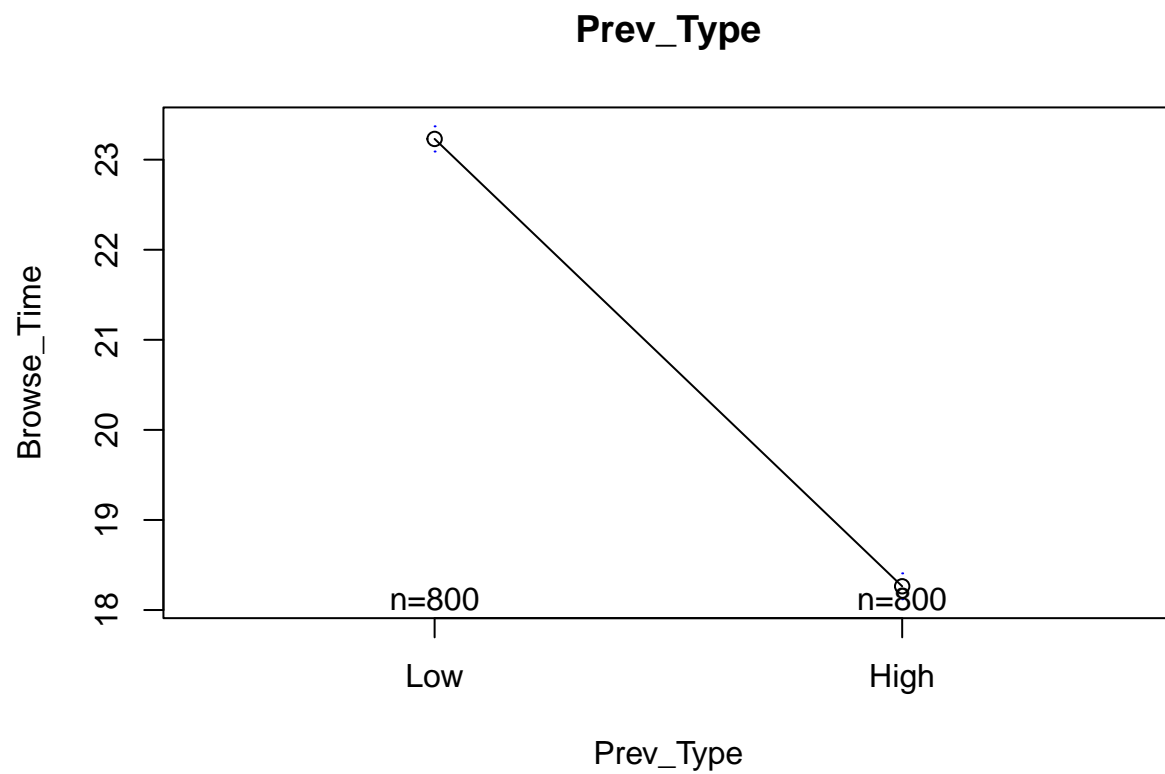
```
plotmeans(Browse_Time ~ Prev_Type,
           data = final_data,
           # ylim = c(0.4, 0.7),
           main = "Prev_Type",
           legends = c("Low", "High"))
```

```
## Warning in arrows(x, li, x, pmax(y - gap, li), col = barcol, lwd = lwd, :
## zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(x, li, x, pmax(y - gap, li), col = barcol, lwd = lwd, :
## zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(x, ui, x, pmin(y + gap, ui), col = barcol, lwd = lwd, :
## zero-length arrow is of indeterminate angle and so skipped
```

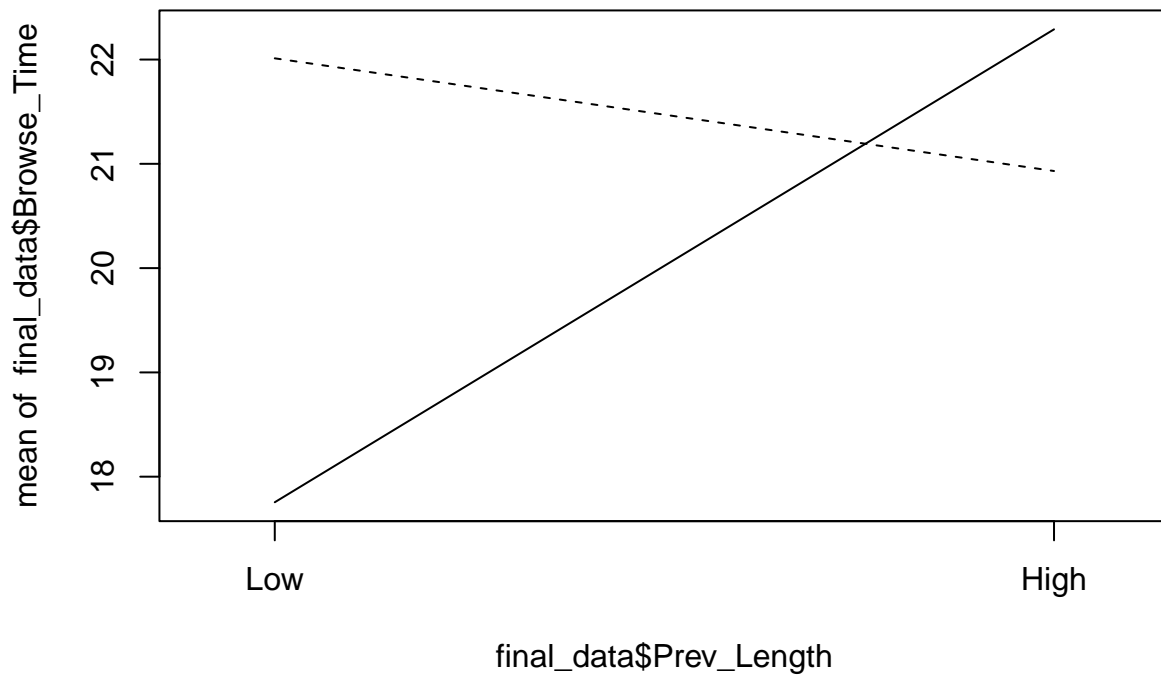
```
## Warning in arrows(x, ui, x, pmin(y + gap, ui), col = barcol, lwd = lwd, :
## zero-length arrow is of indeterminate angle and so skipped
```



```
# par(mfrow = c(1, 2))

# Preview Length and Match Score interaction effect
interaction.plot(
  final_data$Prev_Length,
  final_data$Match_Score,
  final_data$Browse_Time,
  xaxt = "n",
  legend = FALSE,
  # ylab = "Avg Session Duration (min)",
  # xlab = "Opening Feed",
  # ylim = c(2, 10)
)

axis(
  side = 1,
  at = 1:2,
  labels = c("Low", "High")
)
```



```
# axis(
#   side = 1, at = 1:2, labels = c("Home", "Popular"))
# legend("topleft", legend = c("Feed Type", "Infinite Scroll", "Pagination"),
#       lty = c(1, 1, 2), col = c("white", "black", "black"), cex = 0.5, bty = "n")
# moi.by.A.by.B <- aggregate(x = redditFD$y, by = list(A = redditFD$A, B = redditFD$B),
#   FUN = mean)
# points(x = c(1, 2, 1, 2), y = moi.by.A.by.B$x, pch = 16)
```

```
final_data_exp2 <- read.csv(file = '../results_encoded_EXP2.csv')
```

```
head(final_data_exp2)
```

```
##   Prev_Length Match_Score Tile_Size Prev_Type Browse_Time
## 1          -1          -1         1         1    12.99458
## 2          -1           1         1         1    13.70756
## 3           1          -1         1         1    14.55020
## 4           1           1         1         1    18.42728
## 5          -1          -1         1         1    13.96156
## 6          -1           1         1         1    13.79851
```

```
model <- lm(Browse_Time ~ Prev_Length * Match_Score, data=final_data_exp2)
summary(model)
```

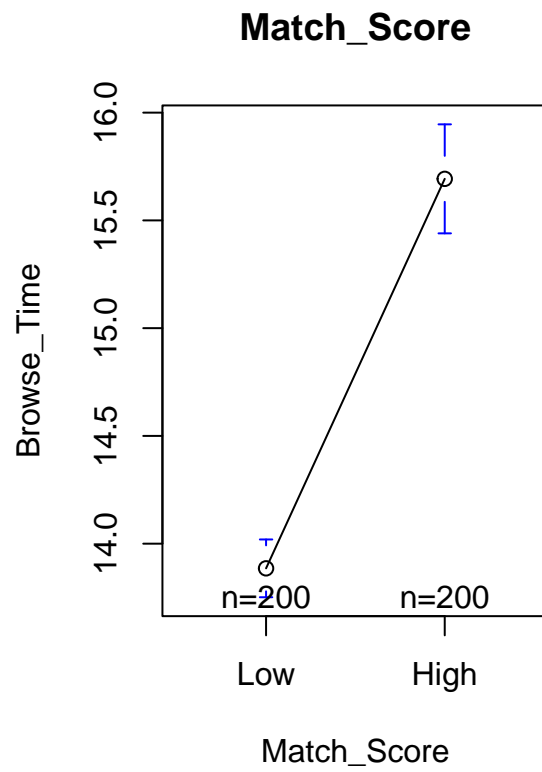
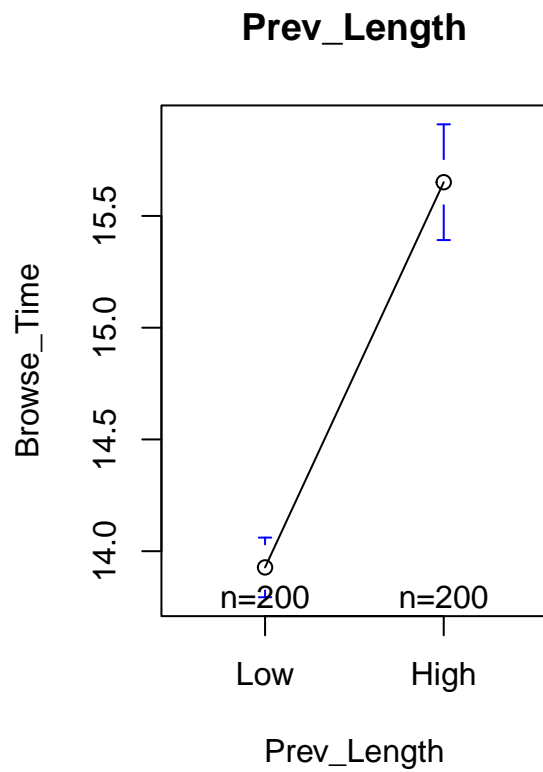
```
##
```

```
## Call:
## lm(formula = Browse_Time ~ Prev_Length * Match_Score, data = final_data_exp2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.5379 -0.6856 -0.0353  0.7599  2.8639
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      14.78918    0.04761   310.62  <2e-16 ***
## Prev_Length       0.86170    0.04761    18.10  <2e-16 ***
## Match_Score       0.90364    0.04761    18.98  <2e-16 ***
## Prev_Length:Match_Score 0.67521    0.04761    14.18  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9522 on 396 degrees of freedom
## Multiple R-squared:  0.6918, Adjusted R-squared:  0.6895
## F-statistic: 296.3 on 3 and 396 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(1, 2))

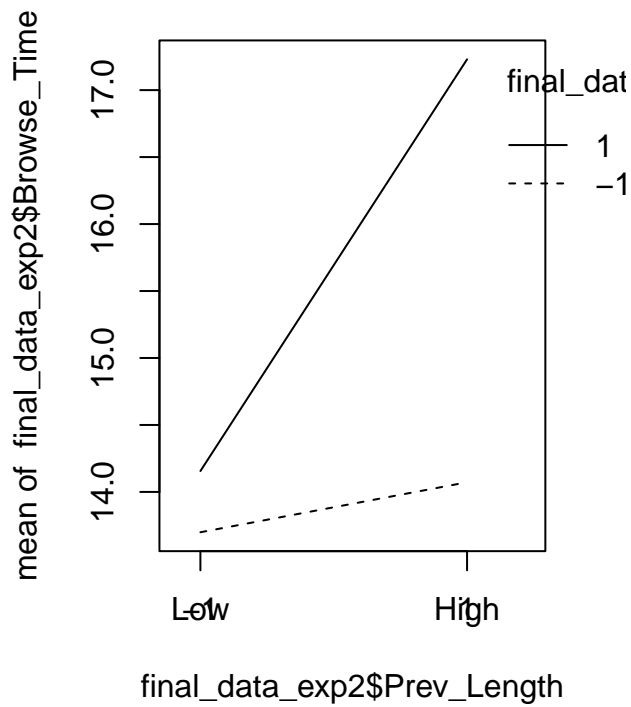
plotmeans(Browse_Time ~ Prev_Length,
  data = final_data_exp2,
  # ylim = c(0.4, 0.7),
  main = "Prev_Length",
  legends = c("Low", "High"))

plotmeans(Browse_Time ~ Match_Score,
  data = final_data_exp2,
  # ylim = c(0.4, 0.7),
  main = "Match_Score",
  legends = c("Low", "High"))
```



```
# Preview Length and Match Score interaction effect
interaction.plot(
  final_data_exp2$Prev_Length,
  final_data_exp2$Match_Score,
  final_data_exp2$Browse_Time,
  #xaxt = "n",
  legend = T,
  # ylab = "Browse_Time",
  # xlab = "Prev_Length",
  # ylim = c(2, 10)
)

axis(
  side = 1,
  at = 1:2,
  labels = c("Low", "High")
)
```

Scratch

Prev.Length : {*low* = 40, *high* = 110} Match.Score : {*low* = 30, *high* = 100} Tile.Size : {*low* = 0.1, *high* = 0.4} Prev.Type : {*A*

Next step

Prev.Length : {*low*, *medium*, *high*} Match.Score : {*low*, *medium*, *high*} Prev.Type : {*AC*, *TT*}

Prev.Length : {*low* = 40, *high* = 110} Match.Score : {*low* = 80, *high* = 100} Prev.Type : {*AC*, *TT*}