

```
library(readxl)
library(lme4)
```

```
## Loading required package: Matrix
```

```
library(lmerTest)
```

```
##
```

```
## Attaching package: 'lmerTest'
```

```
## The following object is masked from 'package:lme4':
```

```
##
```

```
##      lmer
```

```
## The following object is masked from 'package:stats':
```

```
##
```

```
##      step
```

```
library(car)
```

```
## Loading required package: carData
```

```
library(MuMIn)
```

```
library(afex)
```

```
## *****
```

```
## Welcome to afex. For support visit: http://afex.singmann.science/
```

```
## - Functions for ANOVAs: aov_car(), aov_ez(), and aov_4()
```

```
## - Methods for calculating p-values with mixed(): 'S', 'KR', 'LRT', and 'PB'
```

```
## - 'afex_aov' and 'mixed' objects can be passed to emmeans() for follow-up tests
```

```
## - NEWS: emmeans() for ANOVA models now uses model = 'multivariate' as default.
```

```
## - Get and set global package options with: afex_options()
```

```
## - Set orthogonal sum-to-zero contrasts globally: set_sum_contrasts()
```

```
## - For example analyses see: browseVignettes("afex")
```

```
## *****
```

```
##
```

```
## Attaching package: 'afex'
```

```
## The following object is masked from 'package:lme4':
```

```
##
```

```
##      lmer
```

```
data <- read_excel("../Data/PredictingOutcomes_ParticipantPredictions.xlsx", sheet = "Study 2A")
```

```
# divide the data based on the generator
```

```
data1 <- data[data$generator == "analyst",]
```

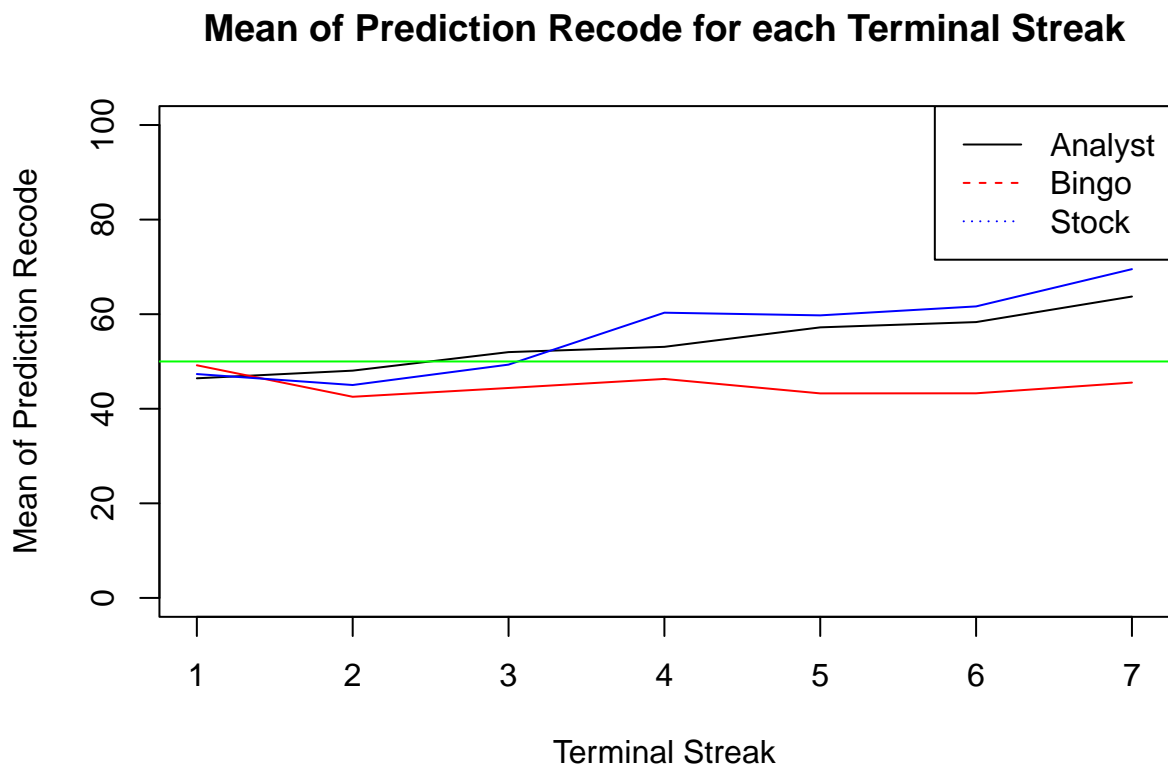
```
data2 <- data[data$generator == "bingo",]
```

```
data3 <- data[data$generator == "stock",]
```

calculate the mean of prediction_recode for each terminal streak from 1 to 7

```
mean1 <- aggregate(data1$prediction_recode, by = list(data1$terminal_streak_length), FUN = mean)
mean2 <- aggregate(data2$prediction_recode, by = list(data2$terminal_streak_length), FUN = mean)
mean3 <- aggregate(data3$prediction_recode, by = list(data3$terminal_streak_length), FUN = mean)

plot(mean1$Group.1, mean1$x, type = "l", ylim=c(0,100), xlab = "Terminal Streak", ylab = "Mean of Prediction Recode")
lines(mean2$Group.1, mean2$x, col = "red")
lines(mean3$Group.1, mean3$x, col = "blue")
abline(h = 50, col = "green")
legend("topright", legend = c("Analyst", "Bingo", "Stock"), col = c("black", "red", "blue"), lty = 1:3)
```



```
aov1 <- aov_ez('participant_id', 'prediction_recode', data, between=c('generator'), within=c('terminal_streak'))
```

```
## Converting to factor: generator
```

```
## Warning: More than one observation per design cell, aggregating data using 'fun_aggregate = mean'.
## To turn off this warning, pass 'fun_aggregate = mean' explicitly.
```

```
## Contrasts set to contr.sum for the following variables: generator
```

```
aov1
```

```
## Anova Table (Type 3 tests)
##
## Response: prediction_recode
##               Effect              df      MSE        F    ges p.value
## 1               generator           2, 153 1915.52   6.96 **  .040   .001
## 2      terminal_streak_length 4.08, 624.73  555.43  9.82 ***  .034   <.001
## 3 generator:terminal_streak_length 8.17, 624.73  555.43  3.61 ***  .025   <.001
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
##
## Sphericity correction method: GG
```

```
pairwise.t.test(data$prediction_recode, data$generator, p.adjust.method = "bonferroni")
```

```
##
## Pairwise comparisons using t tests with pooled SD
##
## data: data$prediction_recode and data$generator
##
##      analyst bingo
## bingo 0.249    -
## stock 0.722    0.012
##
## P value adjustment method: bonferroni
```

```
aov1<-aov_ez('participant_id','prediction_recode',data, between=c('generator'),within=c('terminal_streak_length'))
```

```
## Converting to factor: generator
```

```
## Warning: More than one observation per design cell, aggregating data using 'fun_aggregate = mean'.
## To turn off this warning, pass 'fun_aggregate = mean' explicitly.
```

```
## Contrasts set to contr.sum for the following variables: generator
```

```
aov1
```

```
## Anova Table (Type 3 tests)
##
## Response: prediction_recode
##               Effect              df      MSE        F    ges p.value
## 1               generator           2, 153 1915.52   6.96 **  .040   .001
## 2      terminal_streak_length 4.08, 624.73  555.43  9.82 ***  .034   <.001
## 3 generator:terminal_streak_length 8.17, 624.73  555.43  3.61 ***  .025   <.001
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
##
## Sphericity correction method: GG
```

```
pairwise.t.test(data$prediction_recode, data$generator, p.adjust.method = "bonferroni")
```

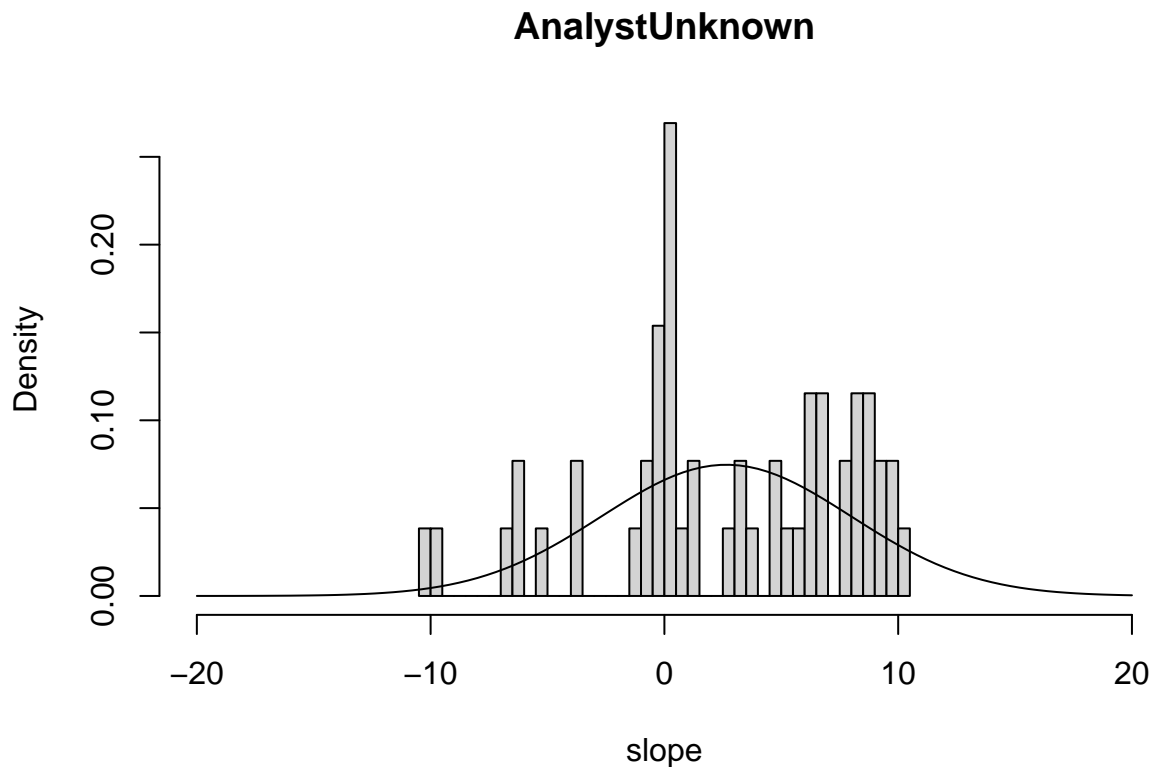
```
##
## Pairwise comparisons using t tests with pooled SD
##
## data: data$prediction_recode and data$generator
##
## analyst bingo
## bingo 0.249 -
## stock 0.722 0.012
##
## P value adjustment method: bonferroni
```

```
id <- unique(data1$participant_id)

slope <- c()

for (i in id){
  x <- as.character(i)
  datax <- data1[data1$participant_id == x,]
  model <- lm(prediction_recode ~ terminal_streak_length, data = datax)
  slope <- c(slope, coef(model)[2])
}

hist(slope,breaks=30,xlim=c(-20,20),prob=TRUE,main="AnalystUnknown")
curve(dnorm(x, mean = mean(slope), sd = sd(slope)), add = TRUE, col = "black")
```



```

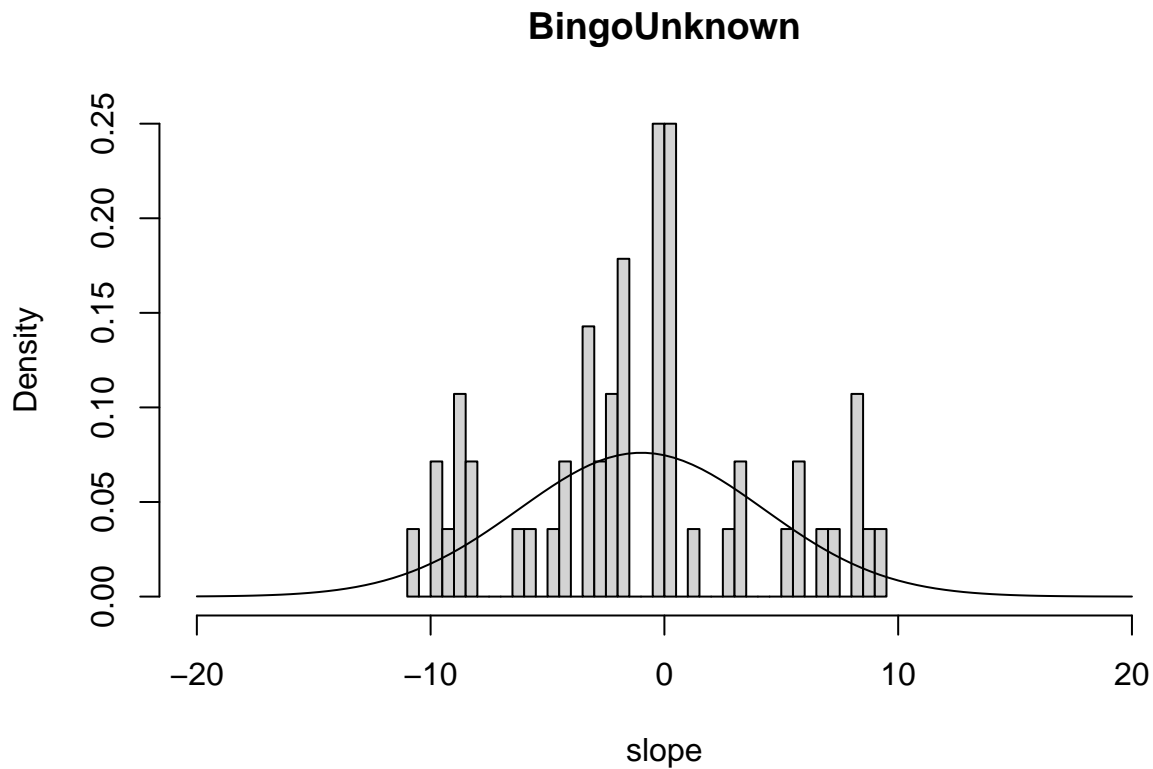
id <- unique(data2$participant_id)

slope <- c()

for (i in id){
  x <- as.character(i)
  datax <- data2[data2$participant_id == x,]
  model <- lm(prediction_recode ~ terminal_streak_length, data = datax)
  slope <- c(slope, coef(model)[2])
}

hist(slope,breaks=30,xlim=c(-20,20),prob=TRUE,main="BingoUnknown")
curve(dnorm(x, mean = mean(slope), sd = sd(slope)), add = TRUE, col = "black")

```



```

id <- unique(data3$participant_id)

slope <- c()

for (i in id){
  x <- as.character(i)
  datax <- data3[data3$participant_id == x,]
  model <- lm(prediction_recode ~ terminal_streak_length, data = datax)
  slope <- c(slope, coef(model)[2])
}

```

```
hist(slope,breaks=30,xlim=c(-20,20),prob=TRUE,main="StockUnknown")  
curve(dnorm(x, mean = mean(slope), sd = sd(slope)), add = TRUE, col = "black")
```

