Neurotyping Analysis

Kenny Miller

5/5/2021

This file includes the code and necessary functions to complete the neurotype analysis as requested by our team. All code is proprietary and not to be shared without the written consent of the author.

Matrix Assignment

Answers Data Frame Creation

```
types <- data.frame("1.0"=rep(0,64),"0.8"=rep(0,64),"0.6"=rep(0,64),"0.4"=rep(0,64),"0.2"=rep(0,64))
colnames(types) <- c("1.0","0.8","0.6","0.4","0.2")
head(types)
```

```
##   1.0 0.8 0.6 0.4 0.2
## 1   0   0   0   0   0
## 2   0   0   0   0   0
## 3   0   0   0   0   0
## 4   0   0   0   0   0
## 5   0   0   0   0   0
## 6   0   0   0   0   0
```

Test Answers

```
m <- as.numeric(results[2,-c(1:3)])

for (i in 1:64) {
  temp <- get(paste("q",i,sep=""))
  types[i,] <- temp[,colnames(temp) == m[i]]
}
```

```
## Warning in matrix(value, n, p): data length [7] is not a sub-multiple or
## multiple of the number of columns [5]
```

```
neuro <- data.frame("player"=rep(0,1),"type1a"=rep(0,1),"type1b"=rep(0,1),"type2a"=rep(0,1),
                    "type2b"=rep(0,1),"type3"=rep(0,1))
nt <- c("1a","1b","2a","2b","3")
for (i in 1:5) {
```

```
  neuro[1,i+1] <- sum(length(which(types$`1.0` == nt[i]))*1,length(which(types$`0.8` == nt[i]))*0.8,
                      length(which(types$`0.6` == nt[i]))*0.6,length(which(types$`0.4` == nt[i]))*0.4,
                      length(which(types$`0.2` == nt[i]))*.2)
}
neuro
```

```
##   player type1a type1b type2a type2b type3
## 1      0   32.2   30.6   39.6   44.4  41.6
```

Create Function

```
player <- results[1,]
neuro[1,1] <- player[2]

neuro_types <- function(a) {
  types <- data.frame("1.0"=rep(0,64),"0.8"=rep(0,64),"0.6"=rep(0,64),"0.4"=rep(0,64),"0.2"=rep(0,64))
  colnames(types) <- c("1.0","0.8","0.6","0.4","0.2")
  total <- nrow(a)
  neuro <- data.frame("player"=rep(0,total),"type1a"=rep(0,total),"type1b"=rep(0,total),
                      "type2a"=rep(0,total),"type2b"=rep(0,total),"type3"=rep(0,total))
  for (i in 1:total) {
    athlete <- a[i,]
    neuro[i,1] <- athlete[2]
    player <- as.numeric(athlete[,-c(1:3)])
    for (j in 1:64) {
      temp <- get(paste("q",j,sep=""))
      types[j,] <- temp[,colnames(temp) == player[j]]
    }
    nt <- c("1a","1b","2a","2b","3")
    for (k in 1:5) {
      neuro[i,k+1] <- sum(length(which(types$`1.0` == nt[k]))*1,length(which(types$`0.8` == nt[k]))*0.8
                          length(which(types$`0.6` == nt[k]))*0.6,length(which(types$`0.4` == nt[k]))*0
                          length(which(types$`0.2` == nt[k]))*0.2)
    }
  }
  return(neuro)
}
```

Function Test and Run with Team Results

```
suppressWarnings(neuro_types(results)[,2:6])
```

```
##   type1a type1b type2a type2b type3
## 1   37.8   43.0   40.2   34.4  34.4
## 2   32.2   30.6   39.6   44.4  41.6
## 3   35.8   35.0   42.0   39.4  36.6
## 4   36.6   39.2   40.4   38.8  35.8
```

Here we see that the largest result of the row is the primary neurotype of our athlete, followed by the second largest, ans so on.