Auditory Mountain

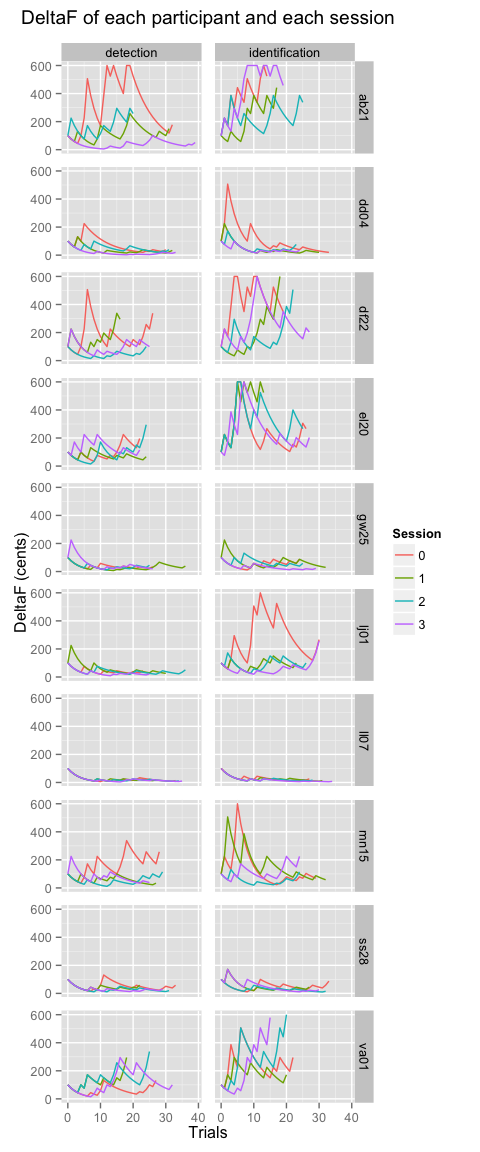
# Individual analyses

## Frequency threshold task

The frequency thresholds are calculated from an adaptive procedure where deltaF is decreased when a wrong answer is provided and increased for a right answer.

### Detection and identification deltaF

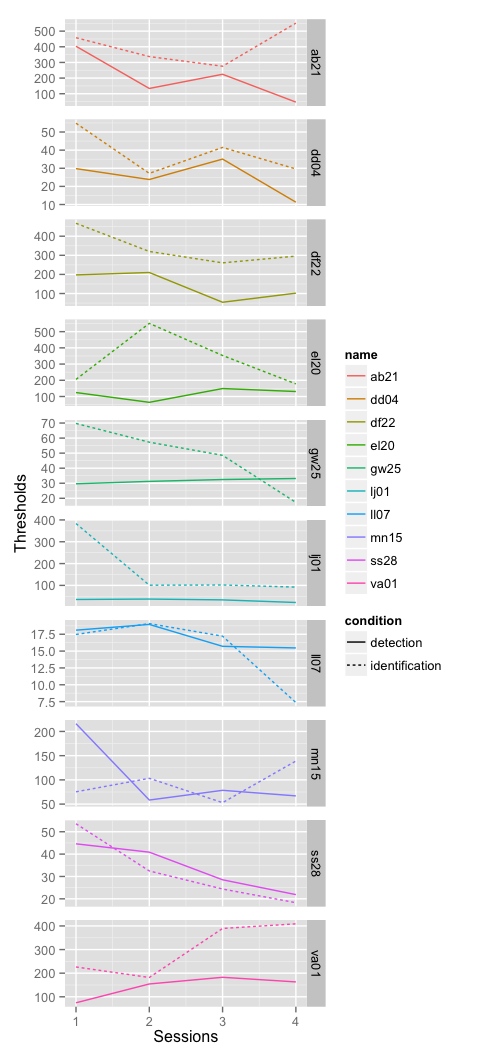
The evolution of the delta F values across trials for the detection and identification tasks is ploted bellow:

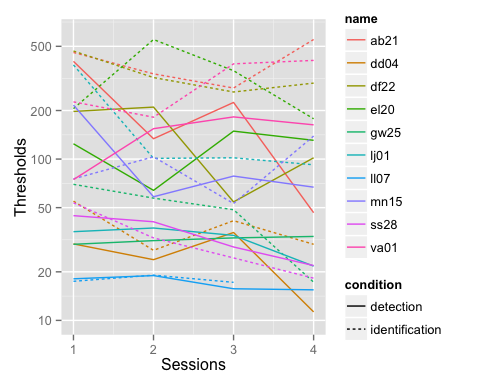


### Thresholds

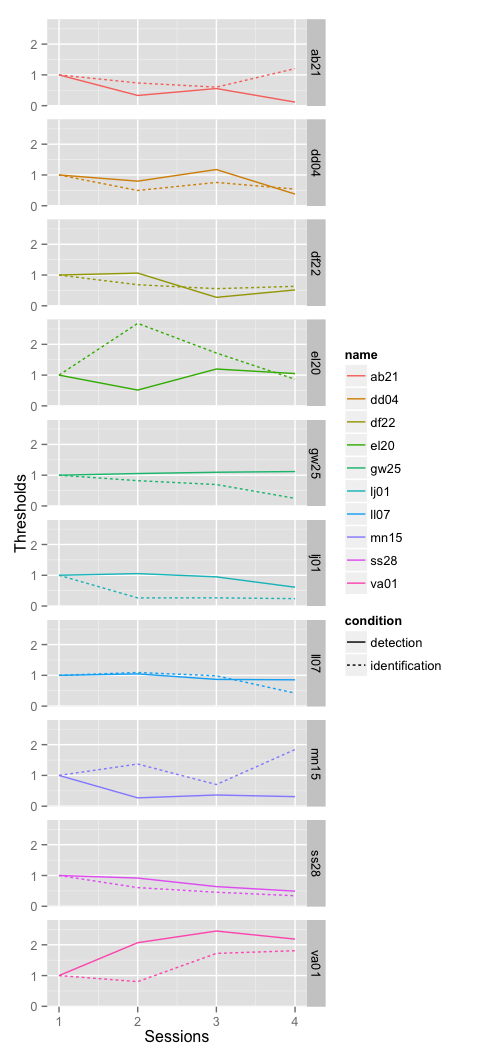
#### Each participant among sessions

The thresholds are calculated from the DeltaF values corresponding to the last 10 reversals. The procedure was repeated two times before training (sessions 1 and 2) and two times after (sessions 3 and 4). The figure bellow shows these thresholds across sessions for the detection and identification conditions:



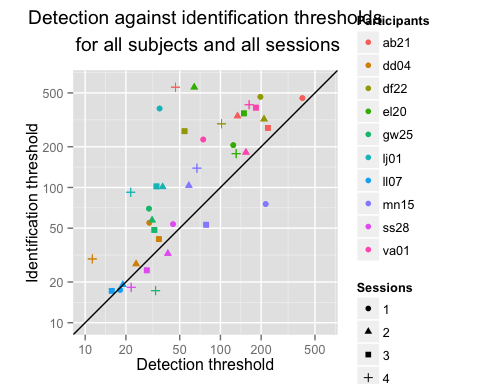


#### Each participant among sessions in percent of the first session



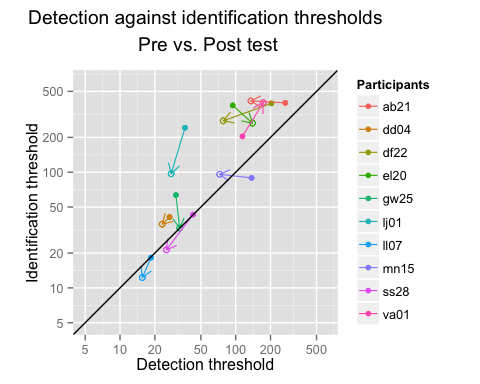
### Detection threshold in function of identification threshold

#### For each session

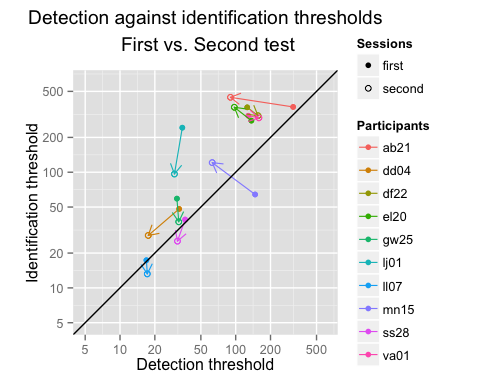


#### Pre vs. post tests

The threshold scores are the means of the four sessions (two pre-test and two post-test)



#### First vs. second tests



## Auditory mountain task

Here are the plots of scores, accuracy and duration in function of trials. The accuracy is the difference between the choosen tone and the highest tone. The duration is the length in seconds of each trial from the begining of the sound to the answer. The score is a mixed value calculated from accuracy and duration according to this equation:

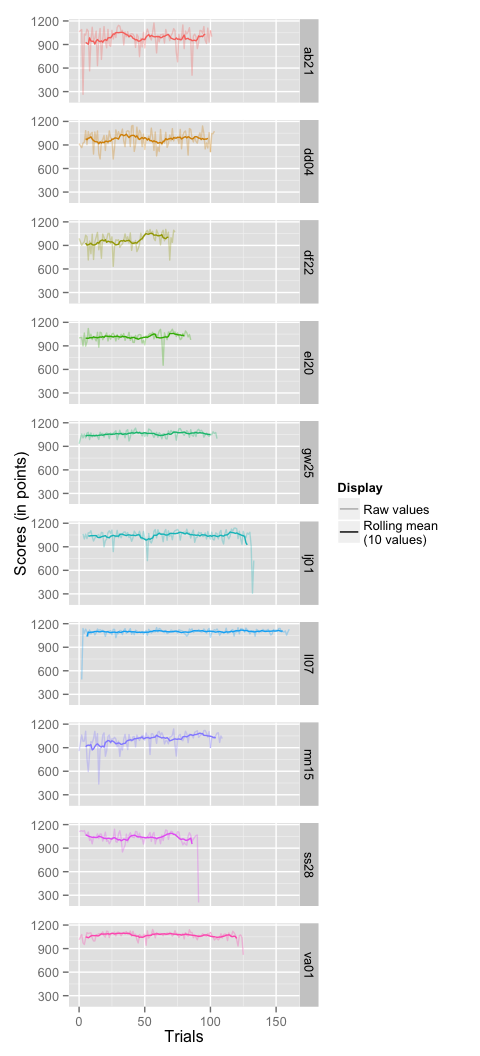
Thus, the time score is a multiplier between 1 for a very long trial and 2 for a very short one (few seconds). Thus, the accuracy score can be doubled with speed.

The accuracy score is 600 maximum (the idea was to make a score in order to have the pleasure to cross 1000 points when the performance is good). For accuracy and time, the power function is used to provide the maximum of points near the perfect score and have a lot of difference when it is harder.

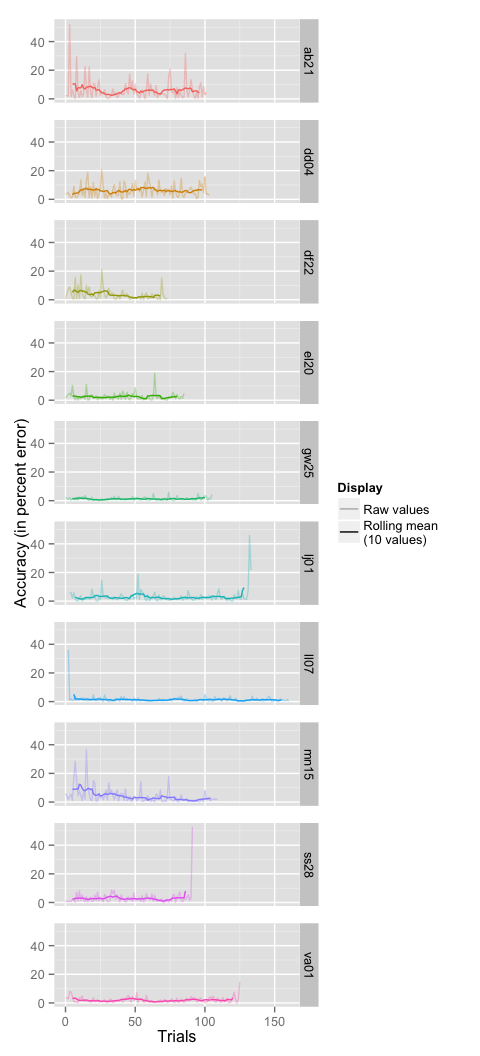
UPDATE:

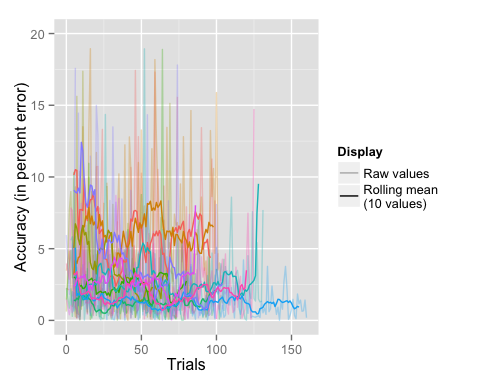
The equation used from va01 (20/07/15) was modified to have a more steeper curve and invite people to be more accurate and avoid the strategy to make a lot of trial quickly and win a lot of points:

### Participant's scores

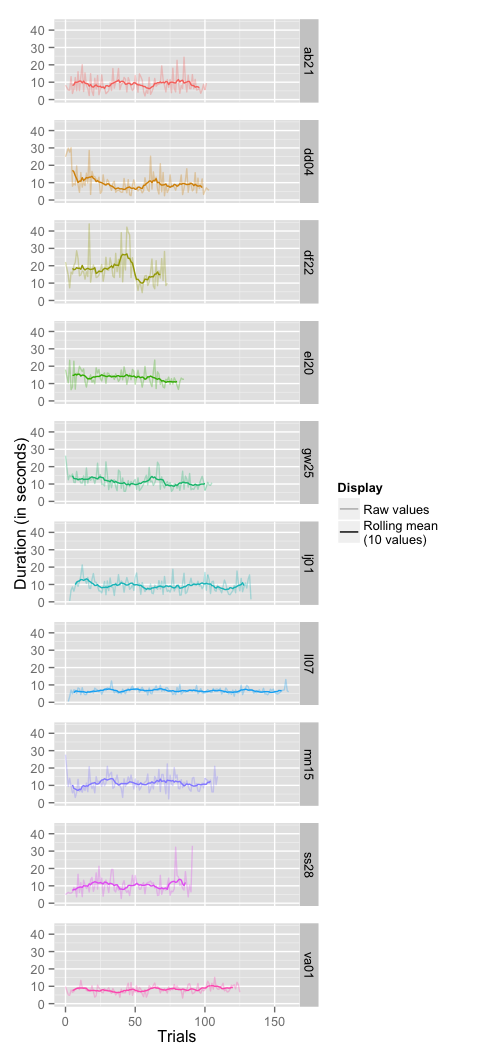


### Accuracy of trials

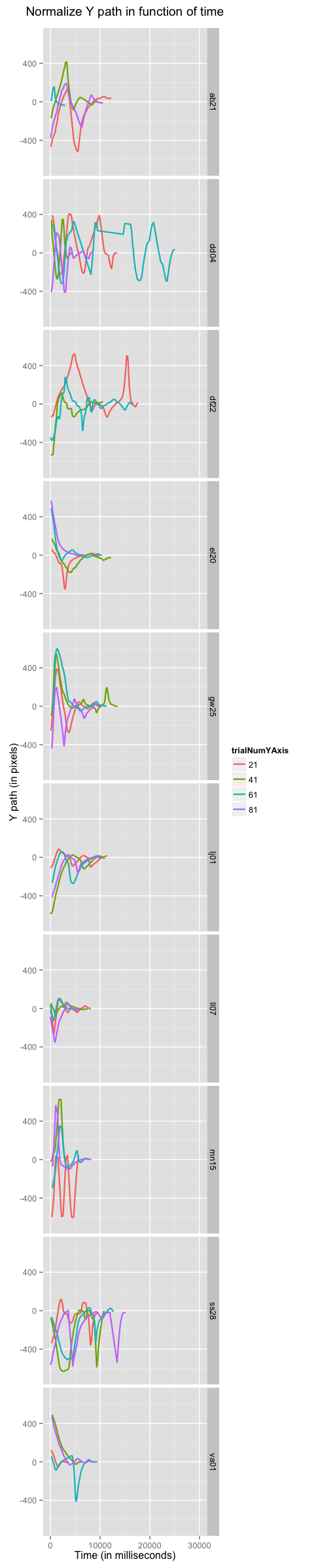




### Durations of trials

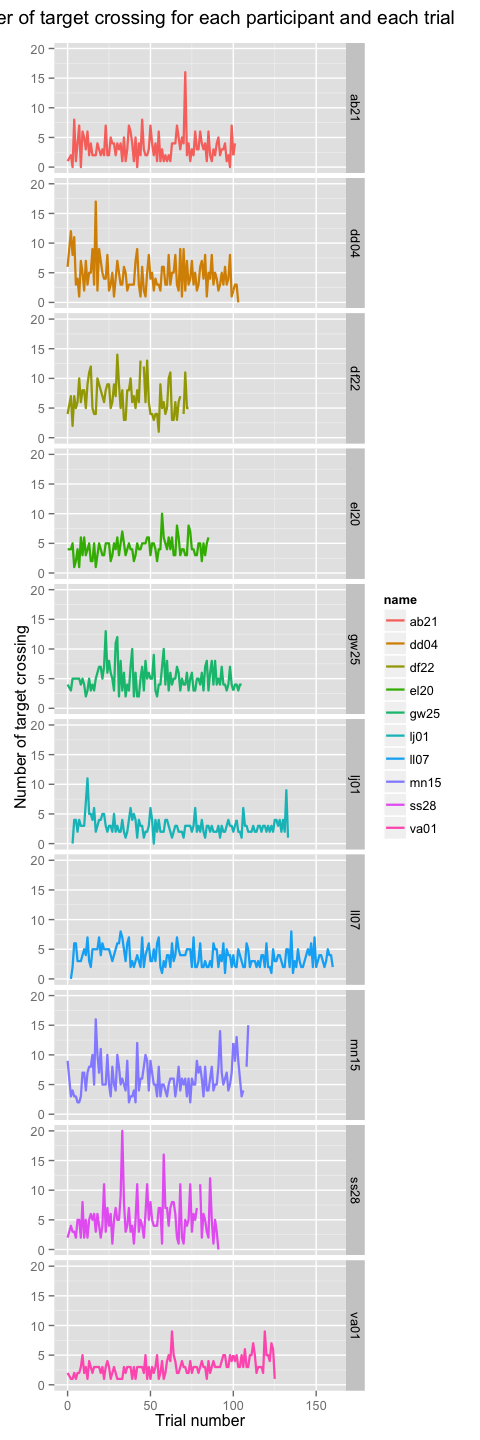


### Path coordinates



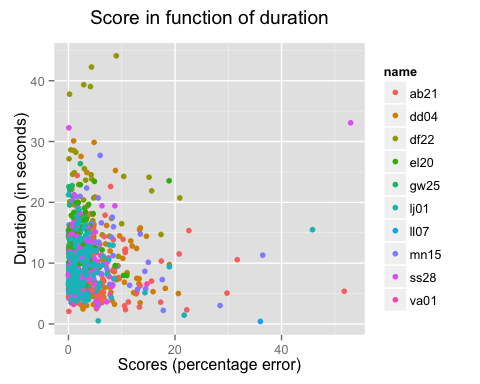
### Number of target crossings

The number of target crossing in the auditory mountain is the number of time the participant cross the target.



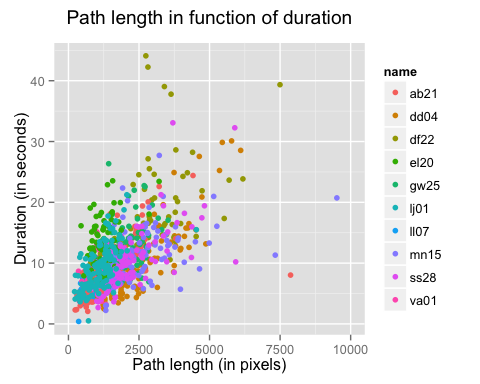
### Accuracy in function of duration for each trial

The variability of the results is not associated with duration. Each point is a trial.



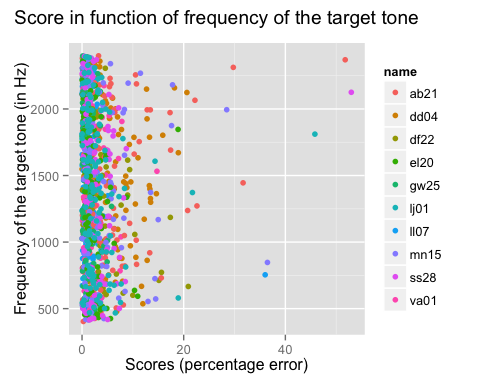
### Path length in function of duration for each trial

A correlation seems to exists between the length of the displacement on the screen and the duration of the trial

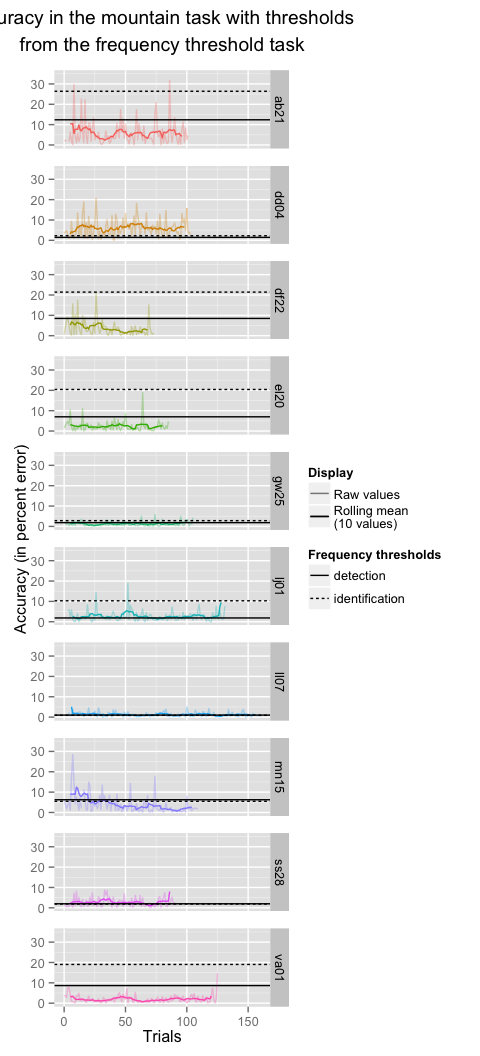


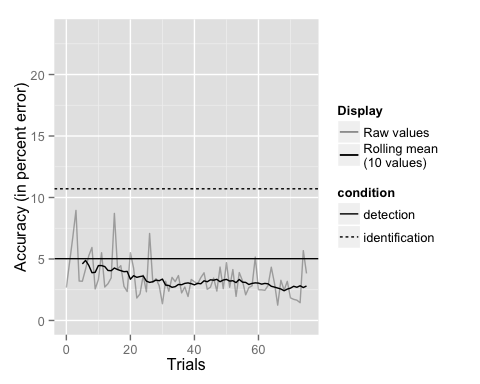
### Accuracy in function of frequency of the target tone

The frequencies used in the mountain task are randomly taken from 400 Hz to 2400 Hz.



### Comparison of frequency thresholds and accuracy



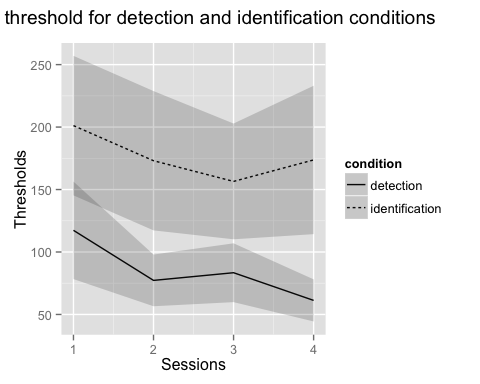


# Global analyses

## Frequency threshold task

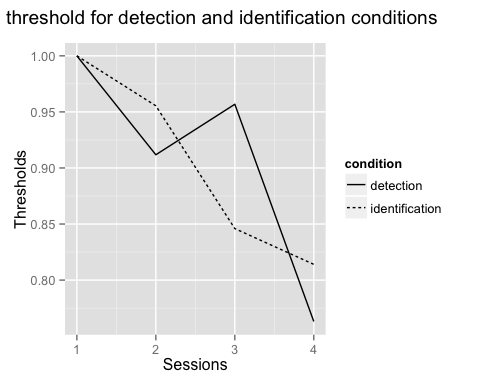
### Mean of participant's thresholds

The mean thresholds are calculated from all participant's deltaF for each session (first and second in pre-test and third and fourth in post-test) and condition (detection and identification).



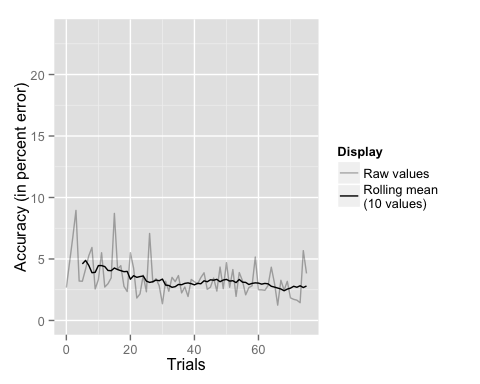
### Mean of participant's thresholds (percent of the first session)

Thresholds are also calculated in percent of the first session deltaF.



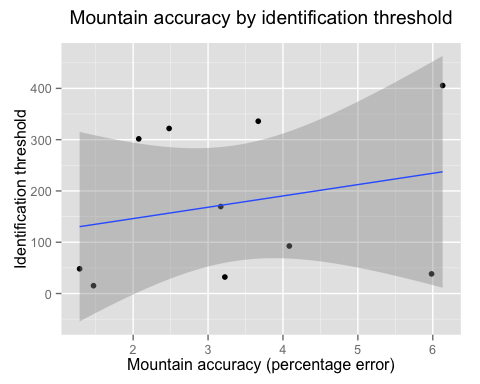
## Auditory mountain task

### Mean accuracy of trials

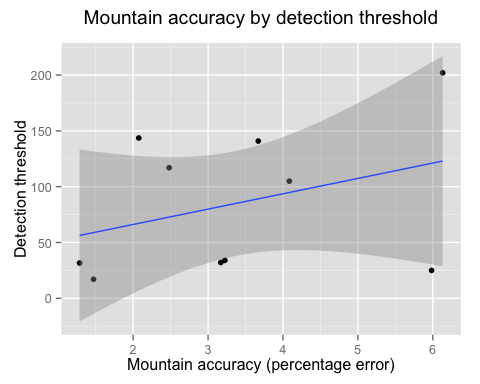


### Correlation between mountain task accuracy and frequency thresholds

#### Identification frequency threhsold



#### Detection frequency threhsold



# Longitudinal analysis

Two participants (va01 and df22) who did the experiment were invited to do four other sessions (one a day) identical to the the first session (pre-test, training, post-test).

The protocol was also identical but the participants were told that the better one could win 10 euros more and that the total score would be calculated from all the training sessions and all the threshold measures.

Thus, the equation used to calculate the total score take into account the global scores to the mountain task of each session, the mean accuracy of all mountain task sessions and the results to each threshold tasks in this way:

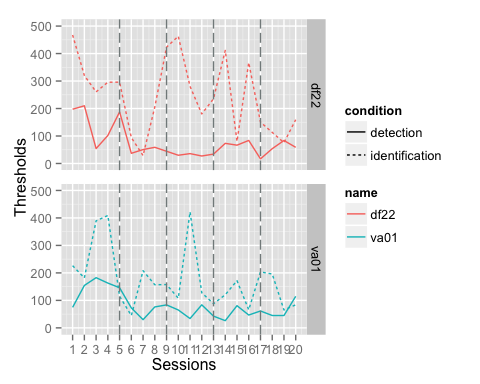
#### va01:

#### df22:

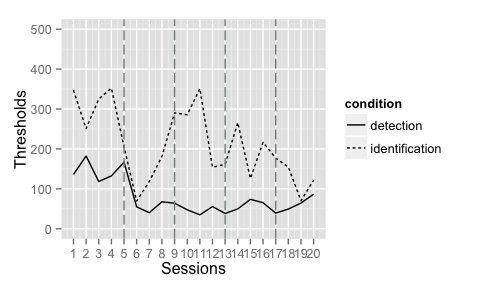
## Frequency threshold task

The frequency threshold tests are done before and after each training session. The dotted vertical lines correspond to each first threshold task.

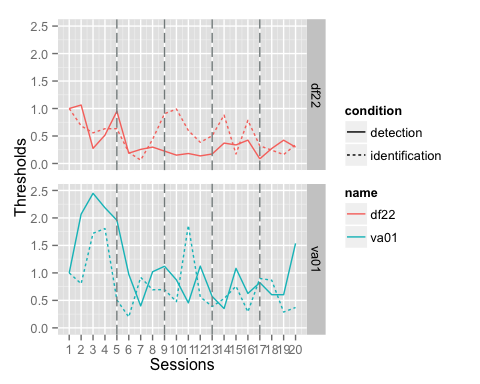
### All Subjects



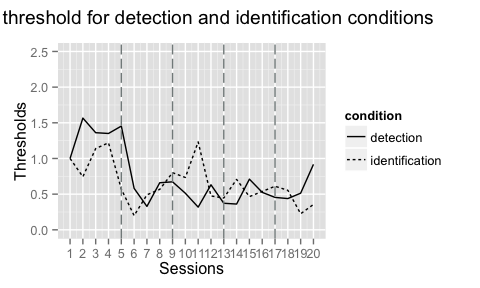
#### Mean



### All Subjects percent

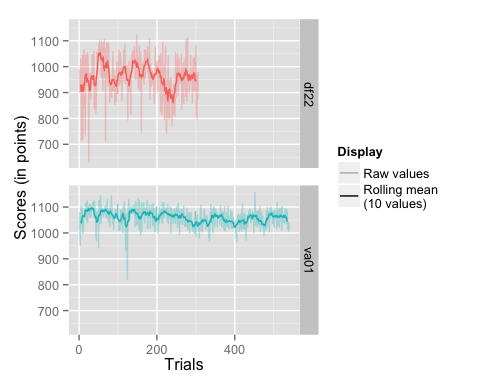


### Mean percent

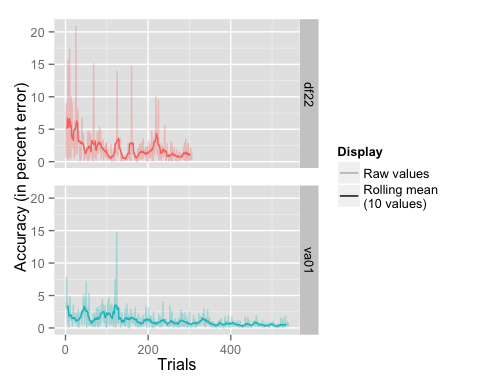


## Auditory mountain task

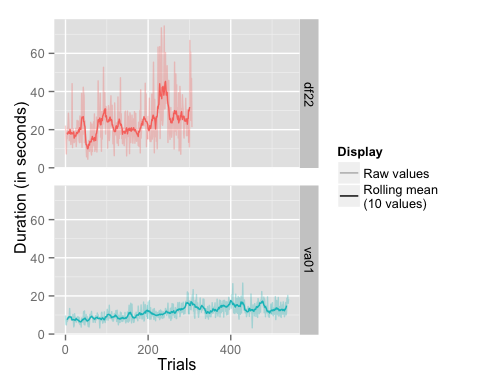
### Participant's scores



### Accuracy of trials

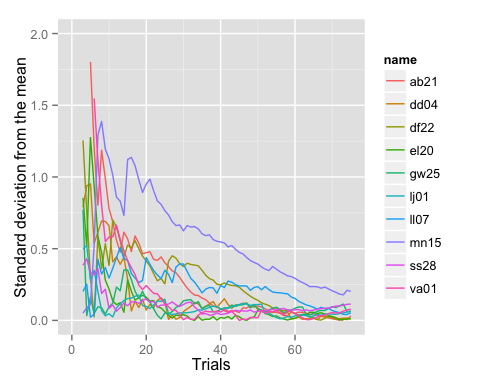


### Durations of trials



# Auditory mountain as a frequency threshold measure

We calculated the efficiency of the auditory mountain task to give an estimate of the frequency threshold. It represents for each trial and each participant the difference between the whole mean of accuracy and the mean from trial 1 to the one presented. This difference is expressed in standard deviation (calculated with the whole trials of each participant). For instance, the value of the trial 21 for mn15 is around one standard deviation. This means that for this participant, the mean of the 21 first trials is away of the global mean by one standard deviation.



The same thing can be done with the mean among participants for each trial.

# Statistical tests

Test of normality for each session and condition

##   
## Shapiro-Wilk normality test  
##   
## data: subset(data, condition == cond & session == sess & threshold < thresholdMax)$threshold  
## W = 0.88, p-value = 0.3137

##   
## Shapiro-Wilk normality test  
##   
## data: subset(data, condition == cond & session == sess & threshold < thresholdMax)$threshold  
## W = 0.91, p-value = 0.4567

##   
## Shapiro-Wilk normality test  
##   
## data: subset(data, condition == cond & session == sess & threshold < thresholdMax)$threshold  
## W = 0.91, p-value = 0.4919

##   
## Shapiro-Wilk normality test  
##   
## data: subset(data, condition == cond & session == sess & threshold < thresholdMax)$threshold  
## W = 0.76, p-value = 0.03682

##   
## Shapiro-Wilk normality test  
##   
## data: subset(data, condition == cond & session == sess & threshold < thresholdMax)$threshold  
## W = 0.88, p-value = 0.255

##   
## Shapiro-Wilk normality test  
##   
## data: subset(data, condition == cond & session == sess & threshold < thresholdMax)$threshold  
## W = 0.94, p-value = 0.6105

##   
## Shapiro-Wilk normality test  
##   
## data: subset(data, condition == cond & session == sess & threshold < thresholdMax)$threshold  
## W = 0.88, p-value = 0.2484

##   
## Shapiro-Wilk normality test  
##   
## data: subset(data, condition == cond & session == sess & threshold < thresholdMax)$threshold  
## W = 0.89, p-value = 0.2952

# Session info

devtools::session\_info()

## Session info --------------------------------------------------------------

## setting value   
## version R version 3.1.3 (2015-03-09)  
## system x86\_64, darwin13.4.0   
## ui X11   
## language (EN)   
## collate en\_US.UTF-8   
## tz Europe/Paris

## Packages ------------------------------------------------------------------

## package \* version date source   
## assertthat 0.1 2013-12-06 CRAN (R 3.1.2)  
## colorspace 1.2-6 2015-03-11 CRAN (R 3.1.3)  
## curl 0.9.1 2015-07-04 CRAN (R 3.1.3)  
## DBI 0.3.1 2014-09-24 CRAN (R 3.1.2)  
## devtools \* 1.8.0 2015-05-09 CRAN (R 3.1.3)  
## digest 0.6.8 2014-12-31 CRAN (R 3.1.2)  
## dplyr \* 0.4.3 2015-09-01 CRAN (R 3.1.3)  
## evaluate 0.7 2015-04-21 CRAN (R 3.1.3)  
## formatR 1.2 2015-04-21 CRAN (R 3.1.3)  
## ggplot2 \* 1.0.1 2015-03-17 CRAN (R 3.1.3)  
## git2r 0.10.1 2015-05-07 CRAN (R 3.1.3)  
## gridExtra \* 0.9.1 2012-08-09 CRAN (R 3.1.2)  
## gtable 0.1.2 2012-12-05 CRAN (R 3.1.2)  
## htmltools 0.2.6 2014-09-08 CRAN (R 3.1.2)  
## jsonlite 0.9.16 2015-04-11 CRAN (R 3.1.3)  
## knitr 1.10.5 2015-05-06 CRAN (R 3.1.3)  
## labeling 0.3 2014-08-23 CRAN (R 3.1.2)  
## lattice 0.20-30 2015-02-22 CRAN (R 3.1.3)  
## lazyeval 0.1.10 2015-01-02 CRAN (R 3.1.2)  
## magrittr 1.5 2014-11-22 CRAN (R 3.1.2)  
## MASS 7.3-39 2015-02-24 CRAN (R 3.1.3)  
## memoise 0.2.1 2014-04-22 CRAN (R 3.1.2)  
## munsell 0.4.2 2013-07-11 CRAN (R 3.1.2)  
## plyr 1.8.3 2015-06-12 CRAN (R 3.1.3)  
## proto 0.3-10 2012-12-22 CRAN (R 3.1.2)  
## psyphy 0.1-9 2014-01-22 CRAN (R 3.1.2)  
## R6 2.1.0 2015-07-04 CRAN (R 3.1.3)  
## RColorBrewer \* 1.1-2 2014-12-07 CRAN (R 3.1.2)  
## Rcpp 0.11.6 2015-05-01 CRAN (R 3.1.3)  
## reshape2 \* 1.4.1 2014-12-06 CRAN (R 3.1.2)  
## rjson \* 0.2.15 2014-11-03 CRAN (R 3.1.2)  
## rmarkdown 0.5.1 2015-01-26 CRAN (R 3.1.2)  
## rstudioapi 0.3.1 2015-04-07 CRAN (R 3.1.3)  
## rversions 1.0.1 2015-06-06 CRAN (R 3.1.3)  
## scales \* 0.2.5 2015-06-12 CRAN (R 3.1.3)  
## stringi 0.4-1 2014-12-14 CRAN (R 3.1.2)  
## stringr 1.0.0 2015-04-30 CRAN (R 3.1.3)  
## xml2 0.1.1 2015-06-02 CRAN (R 3.1.3)  
## yaml 2.1.13 2014-06-12 CRAN (R 3.1.2)  
## zoo \* 1.7-12 2015-03-16 CRAN (R 3.1.3)