multitask\_prac\_paramsBehav

K. Garner

29/08/2018

### Summary

Having identified the candidate network underpinning modulations in connectivity due to multitasking practice, we sought to identify whether parameter estimates for each subject correlate with the observed behaviour, and whether correlation differences are different between groups. We would expect the parameter estimates to correspond to practice-related reductions in multitasking costs, rather than to reductions of response time under single-task conditions (i.e. when there is no multitasking).

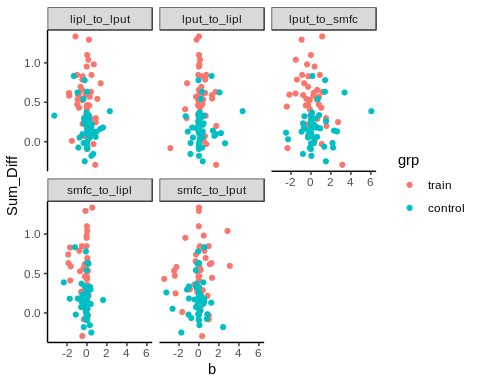
First I present the correlations between parameter estimates and multitask reduction costs (for the summed multitasking cost reductions - see s1 individual differences analysis for definitions). The correlation between LIPL -> LPut is statistically significant after adjustment for multiple comparisons. The strength of the correlation is not statistically different between the training and the control groups.

### What this code does:

1. load the subject individual parameter estimates (Session 1 DCM - influence of multitasking)
2. load the behaviour and extract variables of interest - join to the parameter estimate data
3. Plot correlations between parameter estimates and behaviour and perform correlation tests between estimated parameters and observed data
4. Load subject parameter estimates
5. Load behavioural data and extract variables of interest, and join to parameter data
6. Plot the data and compute pearson and spearman correlations

# params vs. Sum DC Diff

draw.scatters.grp(data, "Sum\_Diff")



lapply(unique(data$con), get.cors, data = data, dv = "Sum\_Diff", method="spearman")

## Warning in cor.test.default(x, y, method = method): Cannot compute exact p-  
## value with ties  
  
## Warning in cor.test.default(x, y, method = method): Cannot compute exact p-  
## value with ties  
  
## Warning in cor.test.default(x, y, method = method): Cannot compute exact p-  
## value with ties  
  
## Warning in cor.test.default(x, y, method = method): Cannot compute exact p-  
## value with ties

## [[1]]  
##   
## Spearman's rank correlation rho  
##   
## data: x and y  
## S = 176150, p-value = 0.00785  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## -0.2726304   
##   
##   
## [[2]]  
##   
## Spearman's rank correlation rho  
##   
## data: x and y  
## S = 134580, p-value = 0.7909  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.02770694   
##   
##   
## [[3]]  
##   
## Spearman's rank correlation rho  
##   
## data: x and y  
## S = 151390, p-value = 0.3688  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## -0.09374052   
##   
##   
## [[4]]  
##   
## Spearman's rank correlation rho  
##   
## data: x and y  
## S = 158000, p-value = 0.1733  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## -0.1415237   
##   
##   
## [[5]]  
##   
## Spearman's rank correlation rho  
##   
## data: x and y  
## S = 138930, p-value = 0.9716  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## -0.003720776

Grp\_Sum\_DC\_Diff\_ps = c(0.00785, 0.7909, 0.3688, 0.1733, 0.9716)  
p.adjust(Grp\_Sum\_DC\_Diff\_ps, method="fdr")

## [1] 0.0392500 0.9716000 0.6146667 0.4332500 0.9716000

The correlation between LIPL -> LPut is statistically significant

### Testing difference in correlation size between the groups

train.cor = get.cors(data[data$grp=="train", ], iv = "lipl\_to\_lput", dv = "Sum\_Diff", method="spearman")  
ctrl.cor = get.cors(data[data$grp=="control", ], iv = "lipl\_to\_lput", dv = "Sum\_Diff", method="spearman")  
cor.test.between = paired.r(-0.178045 , -0.1698871, n=45, n2=47, twotailed=TRUE)  
cor.test.between

## Call: paired.r(xy = -0.178045, xz = -0.1698871, n = 45, n2 = 47, twotailed = TRUE)  
## [1] "test of difference between two independent correlations"  
## z = 0.04 With probability = 0.97

Test between correlation values does not show a difference between the two groups - there is a negative relationship between the strength of the LIPL -> LPut parameter and improvements to multitasking, invariant to group.

### Testing group differences on parameter values per se

grps.mod = lme(b ~ con + grp + con\*grp, random = ~1|sub,   
 data = data,  
 method = "REML")  
Anova(grps.mod, type="II")

## Analysis of Deviance Table (Type II tests)  
##   
## Response: b  
## Chisq Df Pr(>Chisq)   
## con 19.4498 4 0.0006411 \*\*\*  
## grp 3.9381 1 0.0472048 \*   
## con:grp 3.6319 4 0.4581139   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

There is a main effect of parameter and a main effect of group (using type II SS as no statistical interaction).

Plotting b parameters by parameter and by group

plot.grp.violins(data)

