

Absolute treatment effect by Pointwise meta-analysis - executive function score

Pearl Ante-Testard

2024-12-16

```
source(here::here("R", "00-config.R"))
```

Fit a cubic splines model per study

```
#SS.Comb <- df_analysis %>%
# split(.$study) %>%
# map_dfr(function(df_analysis) {
#   study_name <- unique(df_analysis$study)
#   if (study_name %in% c(study_name)) {
#     model <- gam(d12_execfun_el ~ arm_primary + iwi * arm_primary + s(iwi, bs = "cr",
#                               by = arm_primary),
#                 data = df_analysis,
#                 family = "gaussian", method = "REML")
#   }
#   ## Return a data frame with study_name and model
#   data.frame(study = study_name, model = I(list(model)))
#})

SS.Comb_exec <- df_analysis_exec %>%
split(.$study) %>%
map_dfr(function(df_analysis_exec) {
  study_name <- unique(df_analysis_exec$study)
  if (study_name %in% c("DOSE", "GHANA", "HAITI", "DYADG", "DYADM")) {
    model <- gam(d12_execfun_el ~ arms_maternal + iwi * arms_maternal + s(iwi, bs = "cr",
                                  by = arms_maternal),
                data = df_analysis_exec,
                family = "gaussian", method = "REML")
  } else {
    model <- gam(d12_execfun_el ~ arms_maternal + iwi * arms_maternal + s(iwi, bs = "cr",
                                  by = arms_maternal) +
                s(cluster, bs = "re", by = dummy),
                data = df_analysis_exec,
                family = "gaussian", method = "REML")
  }
  # Return a data frame with study_name and model
  data.frame(study = study_name, model = I(list(model)))
})
```

```

new.data_exec <- df_analysis_exec %>%
  select(study, cluster, arms_maternal) %>%
  mutate(dummy = 0) %>%
  distinct() %>%
  expand_grid(iwi = seq(0, 70, by = 1))

predictions.SS.Comb_exec <- new.data_exec %>%
  droplevels() %>%
  arrange(desc(study)) %>%
  group_by(study) %>%
  nest() %>%
  full_join(SS.Comb_exec, by = "study") %>%
  group_by(study) %>%
  do(augment(. $model[[1]], newdata = . $data[[1]], se_fit = T))

# View the resulting dataframe
print(predictions.SS.Comb_exec)

## # A tibble: 31,240 x 7
## # Groups:   study [6]
##   study cluster arms_maternal dummy   iwi .fitted .se.fit
##   <fct> <fct>   <fct>         <dbl> <dbl>   <dbl>   <dbl>
## 1 DOSE   ""      SQ-LNS             0     0 -0.117   0.118
## 2 DOSE   ""      SQ-LNS             0     1 -0.114   0.115
## 3 DOSE   ""      SQ-LNS             0     2 -0.112   0.112
## 4 DOSE   ""      SQ-LNS             0     3 -0.109   0.109
## 5 DOSE   ""      SQ-LNS             0     4 -0.107   0.106
## 6 DOSE   ""      SQ-LNS             0     5 -0.104   0.104
## 7 DOSE   ""      SQ-LNS             0     6 -0.102   0.101
## 8 DOSE   ""      SQ-LNS             0     7 -0.0993  0.0980
## 9 DOSE   ""      SQ-LNS             0     8 -0.0968  0.0952
## 10 DOSE  ""      SQ-LNS             0     9 -0.0943  0.0925
## # i 31,230 more rows

#write_csv(predictions.SS.Comb, here::here("output", "predictions_SSComb.csv"))
#write_csv(predictions.SS.Comb, here::here("output", "predictions_SSComb_nocluster.csv"))

plot_study_exec <- predictions.SS.Comb_exec %>%
  ggplot(., aes(iwi, .fitted, color = arms_maternal)) +
  geom_line(size = 1.25) +
  facet_wrap(~study, ncol = 2, nrow = 3) +
  scale_colour_manual(values = c('darkgrey', 'darkblue')) +
  scale_fill_manual(values = c('darkgrey', 'darkblue')) +
  geom_ribbon(mapping = aes(ymin = .fitted - 1.96*.se.fit,
                           ymax = .fitted + 1.96*.se.fit,
                           fill = arms_maternal), alpha = 0.1) +
  ylab("Executive function score") +
  xlab("IWI") +
  labs(color = 'Treatment', fill = 'Treatment') +
  theme_minimal() +

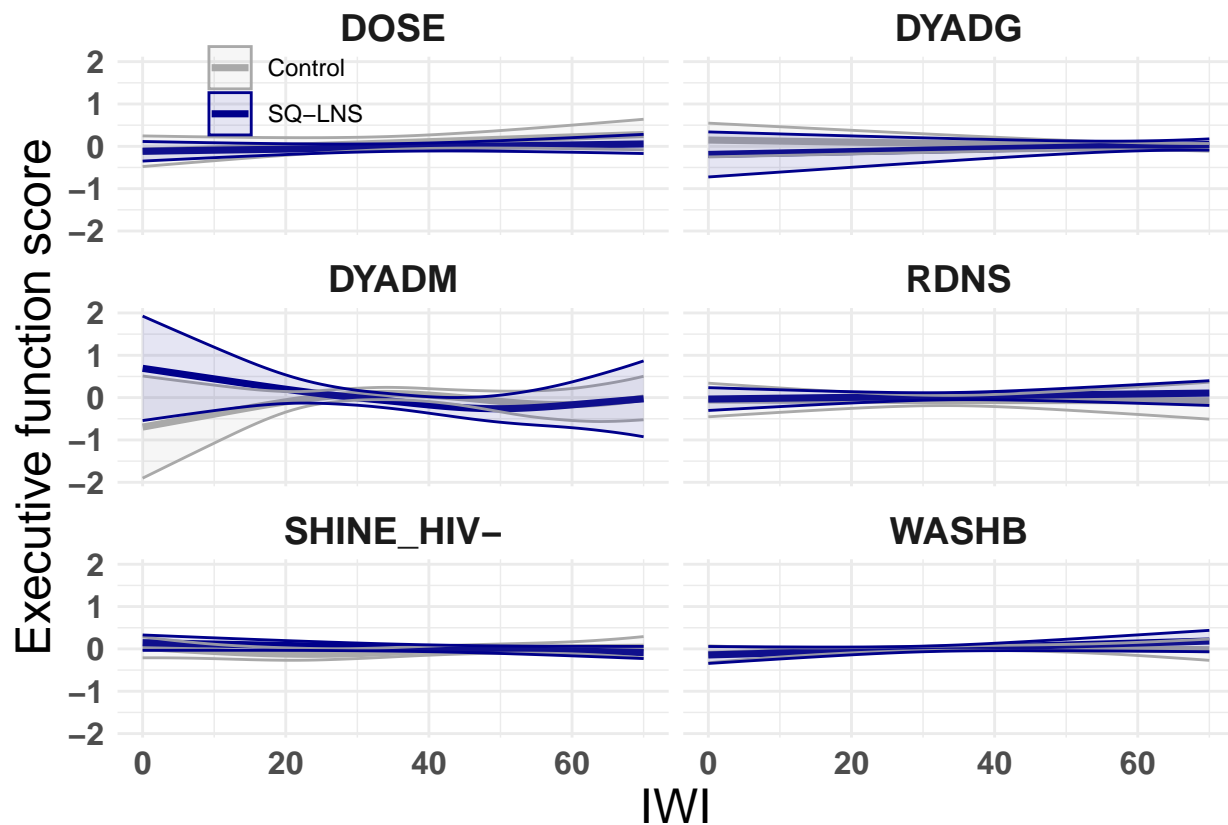
```

```

theme(legend.position = c(0.15,0.95),
      #"none",
      legend.title = element_blank(),
      axis.title.y = element_text(size = 18),
      axis.title.x = element_text(size = 18),
      strip.text = element_text(face="bold", size=14, hjust = 0.5),
      axis.text.y = element_text(face="bold", size=12),
      axis.text.x = element_text(face="bold", size=12))

print(plot_study_exec)

```



Point-wise Meta-analysis

```

# Create a new dataframe for the pointwise meta-analysis
# Since the predicted outcomes are similar with or without the random effect for the cluster variable,
new.data_exec <- df_analysis_exec %>%
  select(study, cluster, arms_maternal) %>%
  mutate(dummy = 0) %>%
  mutate(cluster = rep(unique(cluster), length.out = 1)) %>%
  distinct() %>%
  expand_grid(iwi = seq(0, 70, by = 1))

```

```

predictions.SS.Comb_exec <- new.data_exec %>%
  droplevels() %>%
  arrange(desc(study)) %>%
  group_by(study) %>%
  nest() %>%
  full_join(SS.Comb_exec, by = "study") %>%
  group_by(study)%>%
  do(augment(.$model[[1]], newdata = .$data[[1]], se_fit =T))

point.wise.DF.SS.Comb_exec = pointwise.ma(predictions.SS.Comb_exec,
  clustering.variable = "study",
  combining.variables = c("iwi","arms_maternal"),
  predicted.outcome = ".fitted",
  predicted.outcome.se = ".se.fit",
  predicted.outcome.CI = NULL,
  tau.method = "REML")

```

```

##
[1] "1%"
##
[1] "2%"
##
[1] "3%"
##
[1] "4%"
##
[1] "5%"
##
[1] "6%"
##
[1] "7%"
##
[1] "8%"
##
[1] "9%"
##
[1] "10%"
##
[1] "11%"
##
[1] "12%"
##
[1] "13%"
##
[1] "14%"
##
[1] "15%"
##
[1] "16%"
##
[1] "17%"
##
[1] "18%"

```

```
##  
[1] "19%"  
##  
[1] "20%"  
##  
[1] "21%"  
##  
[1] "22%"  
##  
[1] "23%"  
##  
[1] "24%"  
##  
[1] "25%"  
##  
[1] "26%"  
##  
[1] "27%"  
##  
[1] "28%"  
##  
[1] "29%"  
##  
[1] "30%"  
##  
[1] "31%"  
##  
[1] "32%"  
##  
[1] "33%"  
##  
[1] "34%"  
##  
[1] "35%"  
##  
[1] "36%"  
##  
[1] "37%"  
##  
[1] "38%"  
##  
[1] "39%"  
##  
[1] "40%"  
##  
[1] "41%"  
##  
[1] "42%"  
##  
[1] "43%"  
##  
[1] "44%"  
##  
[1] "45%"
```

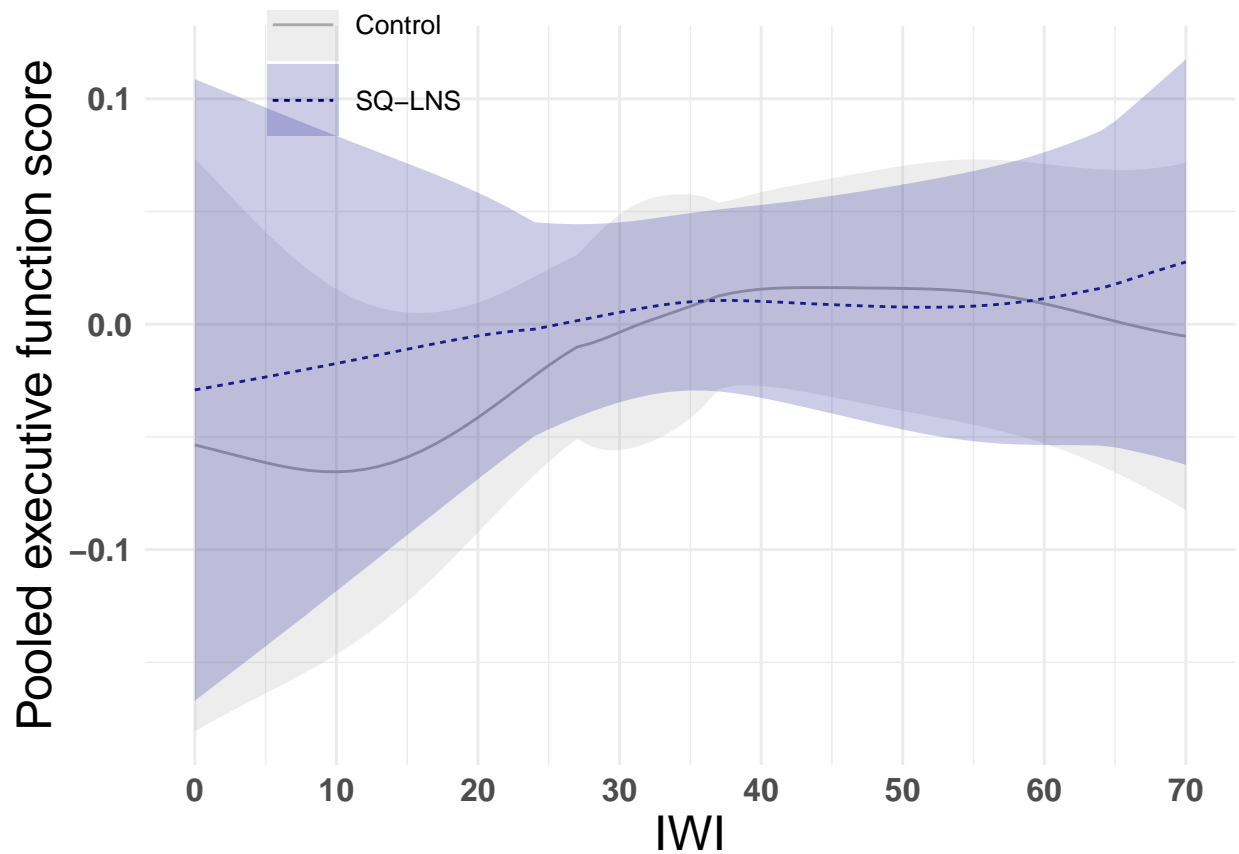
```
##
[1] "46%"
##
[1] "47%"
##
[1] "48%"
##
[1] "49%"
##
[1] "50%"
##
[1] "51%"
##
[1] "52%"
##
[1] "53%"
##
[1] "54%"
##
[1] "55%"
##
[1] "56%"
##
[1] "57%"
##
[1] "58%"
##
[1] "59%"
##
[1] "60%"
##
[1] "61%"
##
[1] "62%"
##
[1] "63%"
##
[1] "64%"
##
[1] "65%"
##
[1] "66%"
##
[1] "67%"
##
[1] "68%"
##
[1] "69%"
##
[1] "70%"
##
[1] "71%"
##
[1] "72%"
```

```
##  
[1] "73%"  
##  
[1] "74%"  
##  
[1] "75%"  
##  
[1] "76%"  
##  
[1] "77%"  
##  
[1] "78%"  
##  
[1] "79%"  
##  
[1] "80%"  
##  
[1] "81%"  
##  
[1] "82%"  
##  
[1] "83%"  
##  
[1] "84%"  
##  
[1] "85%"  
##  
[1] "86%"  
##  
[1] "87%"  
##  
[1] "88%"  
##  
[1] "89%"  
##  
[1] "90%"  
##  
[1] "91%"  
##  
[1] "92%"  
##  
[1] "93%"  
##  
[1] "94%"  
##  
[1] "95%"  
##  
[1] "96%"  
##  
[1] "97%"  
##  
[1] "98%"  
##  
[1] "99%"
```

```
##
[1] "100%"
```

```
## Backtransform predicted outcomes and their corresponding confidence intervals
point.wise.DF.SS.Comb_exec$RE.meta = point.wise.DF.SS.Comb_exec$RE.meta
point.wise.DF.SS.Comb_exec$RE.meta.upper = point.wise.DF.SS.Comb_exec$RE.meta.upper
point.wise.DF.SS.Comb_exec$RE.meta.lower = point.wise.DF.SS.Comb_exec$RE.meta.lower
```

```
point.wise.DF.SS.Comb.plot_exec = point.wise.DF.SS.Comb_exec%>%
  mutate(Treatment = as.factor(arms_maternal)) %>%
  ggplot(aes(x = iwi, y = RE.meta, fill = Treatment, linetype = Treatment)) +
  geom_line(size=0.5, aes(colour = Treatment)) +
  scale_color_manual(values = c('darkgrey', 'darkblue')) +
  scale_fill_manual(values = c('darkgrey', 'darkblue')) +
  geom_ribbon(aes(ymin = RE.meta.lower, ymax=RE.meta.upper), alpha=0.2) +
  #scale_color_jama(name= "Treatment")+
  #scale_linetype_discrete(name="Treatment")+ylab("") +
  #scale_linetype_manual(values=c("solid", "dashed")) +
  labs(color='Treatment', fill='Treatment', linetype='Treatment') +
  xlab(bquote('IWI')) +
  ylab(bquote('Pooled executive function score')) +
  theme_minimal() +
  theme(legend.position = c(0.20, 0.95),
        legend.title = element_blank(),
        axis.title.y = element_text(size = 18),
        axis.title.x = element_text(size = 18),
        strip.text = element_text(face="bold", size=14, hjust = 0.5),
        axis.text.y = element_text(face="bold", size=12),
        axis.text.x = element_text(face="bold", size=12),
        legend.key.size = unit(1, "cm"),
        legend.key.width = unit(1, "cm"),
        legend.text=element_text(size=10, hjust = 0)) +
  scale_x_continuous(breaks = seq(min(point.wise.DF.SS.Comb_exec$iwi), max(point.wise.DF.SS.Comb_exec$
print(point.wise.DF.SS.Comb.plot_exec)
```

```
# Save the plot
#ggsave(here::here("output", "gam_pooled_pointwise_laz.png"),
#       plot = point.wise.DF.SS.Comb.plot, width = 6, height = 6, dpi = 300)
```

Absolute treatment effect

```
predictions.SS.Comb_exec=predictions.SS.Comb_exec%>%
  mutate(Lower = .fitted - 1.96*.se.fit,
         Upper = .fitted + 1.96*.se.fit)%>%
  mutate(fit = .fitted, Lower = Lower, Upper = Upper) %>%
  mutate(iwi = as.character(iwi))

absolute_diff_SS.Comb_exec = risk.diff.creator(dataframe = predictions.SS.Comb_exec,
                                              treatment = "arms_maternal", outcome = NULL,
                                              matching.variables = c("iwi","study"),
                                              predicted.outcome = "fit",
                                              predicted.CI = c("Lower","Upper"))

absolute_diff_SS.Comb_exec= absolute_diff_SS.Comb_exec%>%
  select(study, iwi, fit.diff, diff.lower, diff.upper)
```

```

point.wise.absolute_diff_SS.Comb_exec = pointwise.ma(data = absolute_diff_SS.Comb_exec,
                                                    clustering.variable = "study",
                                                    combining.variables = c("iwi"),
                                                    predicted.outcome = "fit.diff",
                                                    predicted.outcome.se = NULL,
                                                    predicted.outcome.CI = c("diff.lower", "diff.upper"),
                                                    tau.method = "REML"
)

```

```

##
[1] "1%"
##
[1] "3%"
##
[1] "4%"
##
[1] "6%"
##
[1] "7%"
##
[1] "8%"
##
[1] "10%"
##
[1] "11%"
##
[1] "13%"
##
[1] "14%"
##
[1] "15%"
##
[1] "17%"
##
[1] "18%"
##
[1] "20%"
##
[1] "21%"
##
[1] "23%"
##
[1] "24%"
##
[1] "25%"
##
[1] "27%"
##
[1] "28%"
##
[1] "30%"
##
[1] "31%"

```

```
##  
[1] "32%"  
##  
[1] "34%"  
##  
[1] "35%"  
##  
[1] "37%"  
##  
[1] "38%"  
##  
[1] "39%"  
##  
[1] "41%"  
##  
[1] "42%"  
##  
[1] "44%"  
##  
[1] "45%"  
##  
[1] "46%"  
##  
[1] "48%"  
##  
[1] "49%"  
##  
[1] "51%"  
##  
[1] "52%"  
##  
[1] "54%"  
##  
[1] "55%"  
##  
[1] "56%"  
##  
[1] "58%"  
##  
[1] "59%"  
##  
[1] "61%"  
##  
[1] "62%"  
##  
[1] "63%"  
##  
[1] "65%"  
##  
[1] "66%"  
##  
[1] "68%"  
##  
[1] "69%"
```

```
##
[1] "70%"
##
[1] "72%"
##
[1] "73%"
##
[1] "75%"
##
[1] "76%"
##
[1] "77%"
##
[1] "79%"
##
[1] "80%"
##
[1] "82%"
##
[1] "83%"
##
[1] "85%"
##
[1] "86%"
##
[1] "87%"
##
[1] "89%"
##
[1] "90%"
##
[1] "92%"
##
[1] "93%"
##
[1] "94%"
##
[1] "96%"
##
[1] "97%"
##
[1] "99%"
##
[1] "100%"
```

```
point.wise.absolute_diff_SS.Comb_exec = point.wise.absolute_diff_SS.Comb_exec%>%
  mutate(iwi = as.numeric(iwi))
```

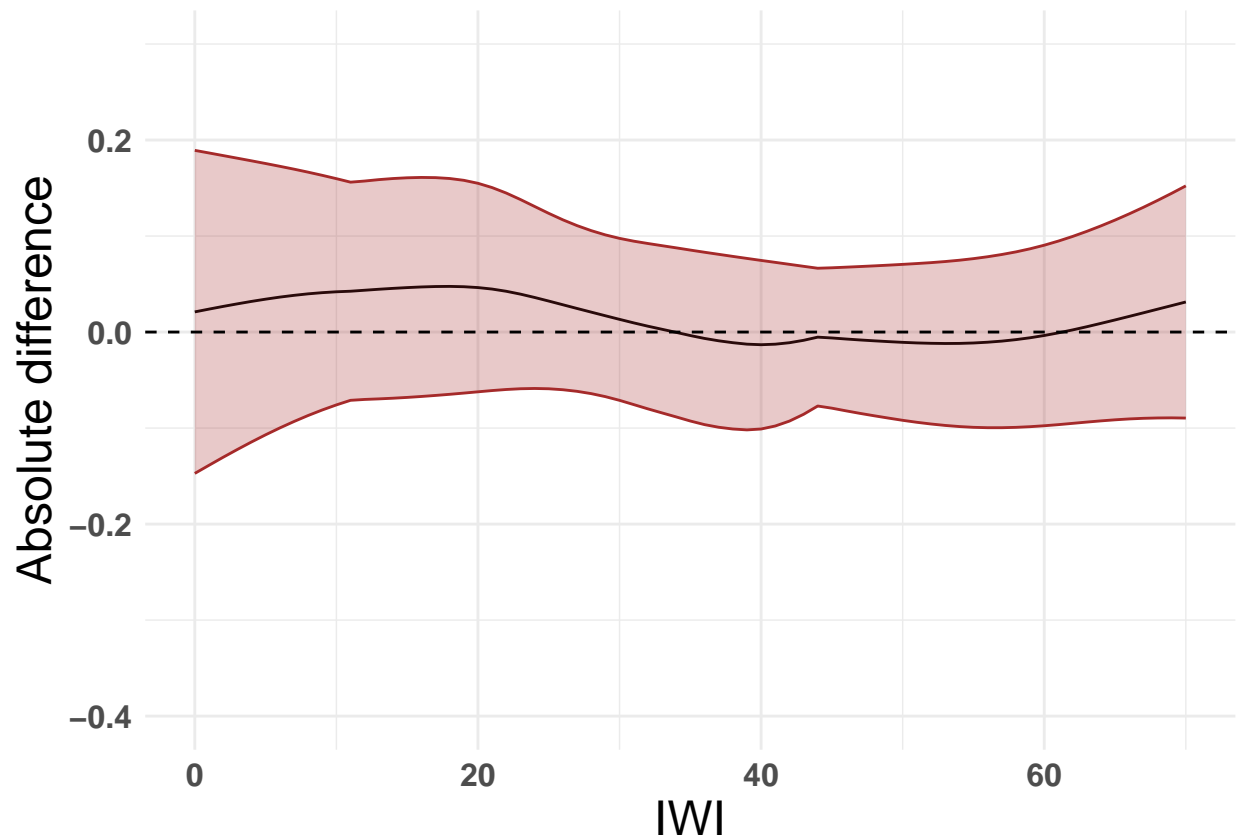
```
point.wise.DF.SS.Comb.diff.plot_exec=point.wise.absolute_diff_SS.Comb_exec%>%
  ggplot(aes(x = iwi,y=RE.meta)) + geom_line(size=0.5)+
  geom_ribbon(mapping = aes(ymin=RE.meta.lower, ymax=RE.meta.upper),alpha=0.25,
             color="brown", fill="brown")+
  geom_hline(yintercept = 0, linetype=2)+ylab("") +
  xlab(bquote('IWI')) +
```

```

ylab(bquote('Absolute difference')) +
theme_minimal() +
theme(axis.title.y = element_text(size = 18),
      axis.title.x = element_text(size = 18),
      strip.text = element_text(face="bold", size=14, hjust = 0.5),
      axis.text.y = element_text(face="bold", size=12),
      axis.text.x = element_text(face="bold", size=12)) + ylim(c(-0.4,0.3))

print(point.wise.DF.SS.Comb.diff.plot_exec)

```

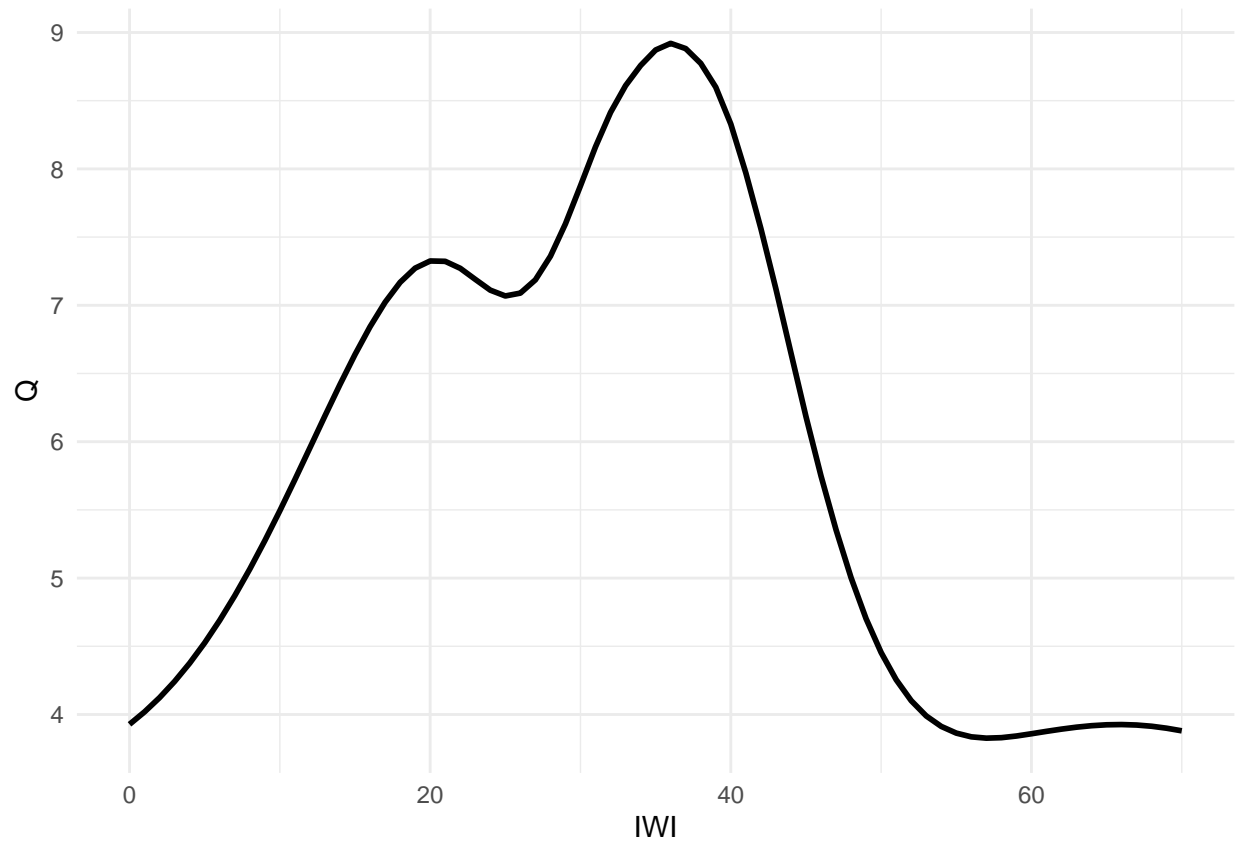


```

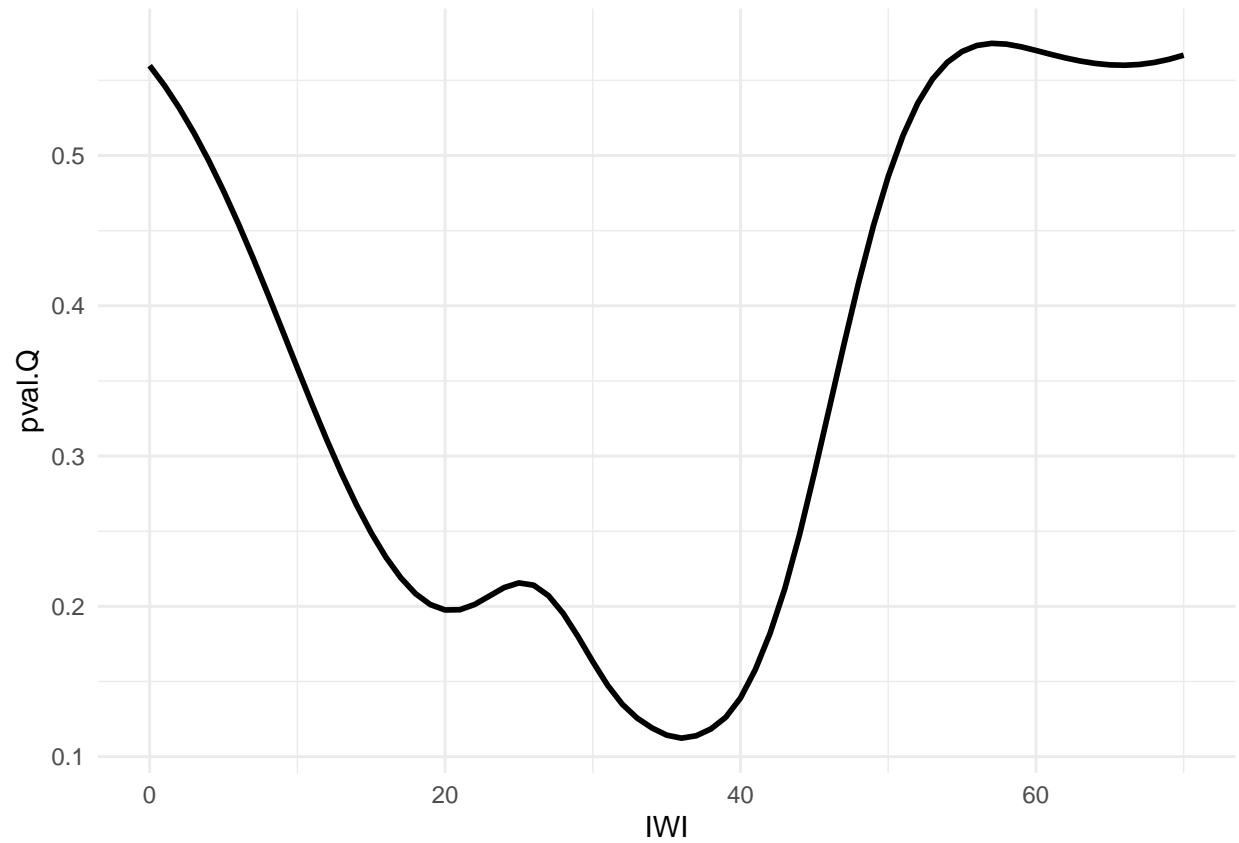
point.wise.Q.plot_exec=point.wise.absolute_diff_SS.Comb_exec%>%
  ggplot(aes(x = iwi,y=Q)) + geom_line(size=1)+
  xlab(bquote('IWI')) +
  theme_minimal()

print(point.wise.Q.plot_exec)

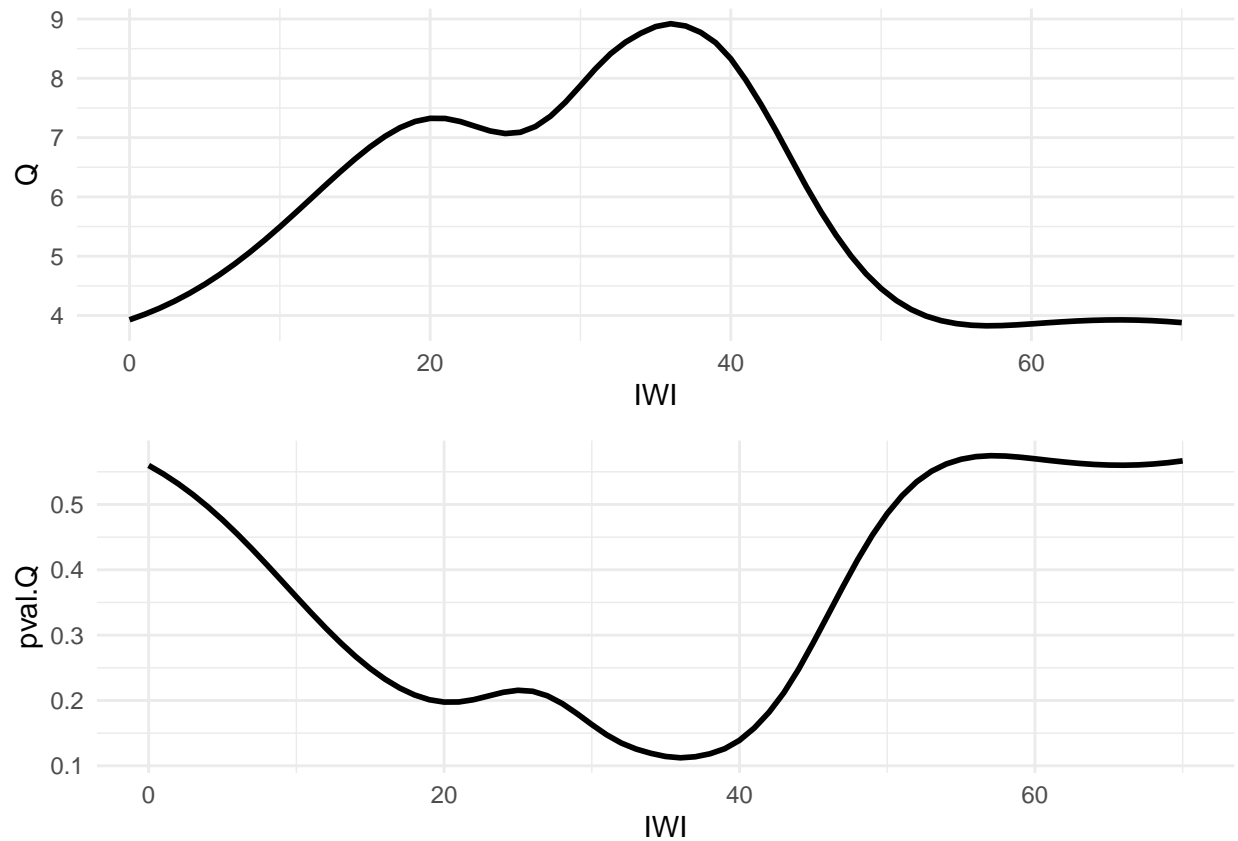
```



```
point.wise.pvalQ.plot_exec=point.wise.absolute_diff_SS.Comb_exec%>%  
  ggplot(aes(x = iwi,y=pval.Q)) + geom_line(size=1)+  
  xlab(bquote('IWI')) +  
  theme_minimal()  
  
print(point.wise.pvalQ.plot_exec)
```



```
plot_composite_hetero_exec <- plot_grid(point.wise.Q.plot_exec, point.wise.pvalQ.plot_exec,  
                                         nrow=2, align = "h")  
plot_composite_hetero_exec
```

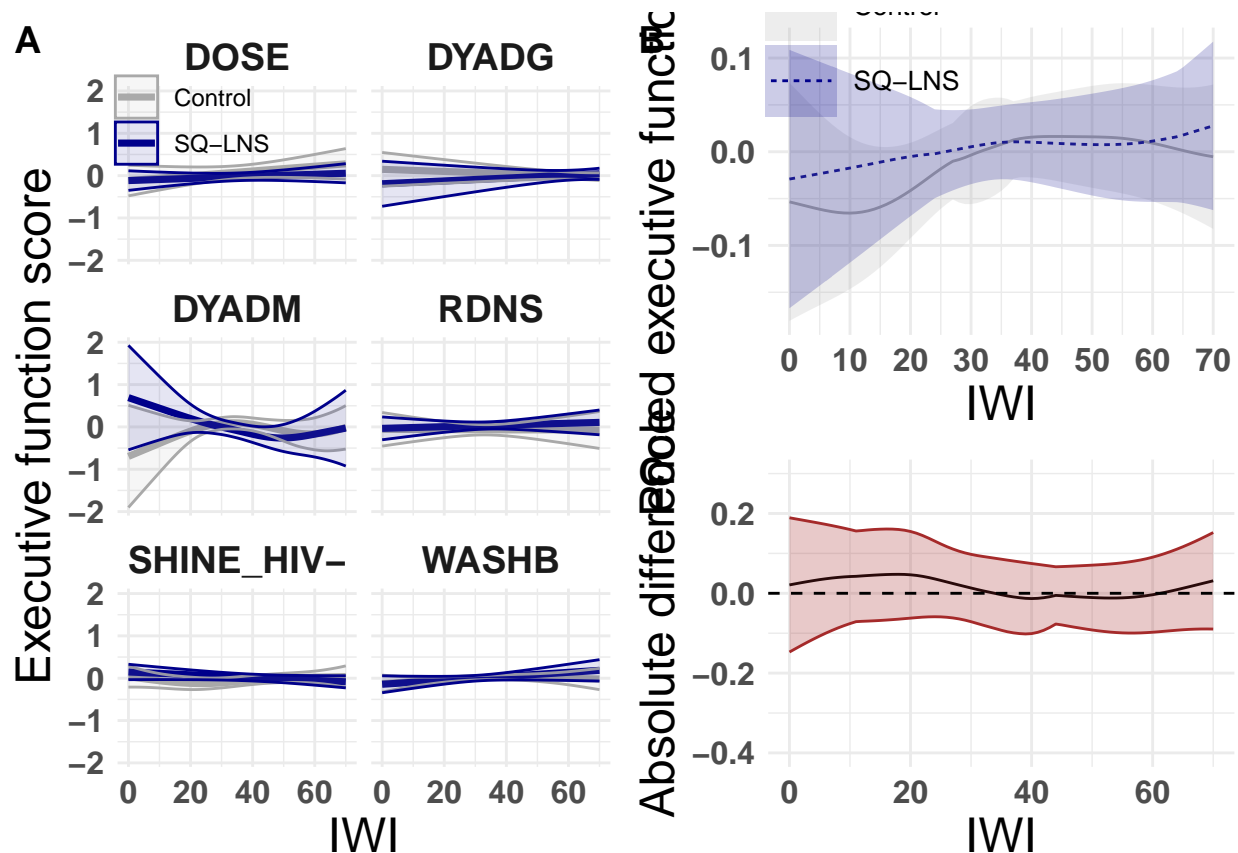


```
# Save the plot
#ggsave(here::here("output", "qplot_pooled_pointwise_laz.png"),
#       plot = point.wise.Q.plot, width = 6, height = 6, dpi = 300)
```

```
plot_comp_effect_exec <- plot_grid(point.wise.DF.SS.Comb.plot_exec, point.wise.DF.SS.Comb.diff.plot_exec,
```

```
plot_composite_exec <- plot_grid(plot_study_exec, plot_comp_effect_exec,
                                ncol=2, labels = c("A", "", ""))
```

```
plot_composite_exec
```

```
# Save the plot
ggsave(here::here("output", "gam_pooled_pointwise_d12_execfun_el_arm_maternal.png"),
       plot = plot_composite_exec, width = 14, height = 16, dpi = 300)
```

Session info

```
sessionInfo()
```

```
## R version 4.3.3 (2024-02-29)
## Platform: aarch64-apple-darwin20 (64-bit)
## Running under: macOS Sonoma 14.5
##
## Matrix products: default
## BLAS: /System/Library/Frameworks/Accelerate.framework/Versions/A/Frameworks/vecLib.framework/Versions/A/
## LAPACK: /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRlapack.dylib; LAPACK v
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## time zone: America/Los_Angeles
## tzcode source: internal
##
## attached base packages:
## [1] stats graphics grDevices utils datasets methods base
##
```

```

## other attached packages:
## [1] geepack_1.3.10      estimatr_1.0.4      sandwich_3.1-0      msm_1.7.1
## [5] lmtest_0.9-40       zoo_1.8-12          tidymv_3.4.2        meta_7.0-0
## [9] metadat_1.2-0       mgcv_1.9-1          nlme_3.1-164        patchwork_1.2.0
## [13] RColorBrewer_1.1-3 cowplot_1.1.3       gridExtra_2.3        ggrepel_0.9.5
## [17] ggsci_3.0.1         broom_1.0.5         rmarkdown_2.26       skimr_2.1.5
## [21] readxl_1.4.3        lubridate_1.9.3     forcats_1.0.0        stringr_1.5.1
## [25] purrr_1.0.2         readr_2.1.5         tidyr_1.3.1          tibble_3.2.1
## [29] ggplot2_3.5.1       tidyverse_2.0.0     haven_2.5.4          table1_1.4.3
## [33] dplyr_1.1.4         here_1.0.1
##
## loaded via a namespace (and not attached):
## [1] tidyselect_1.2.1    farver_2.1.2        fastmap_1.1.1        CompQuadForm_1.4.3
## [5] mathjaxr_1.6-0      digest_0.6.35       timechange_0.3.0     lifecycle_1.0.4
## [9] survival_3.5-8      magrittr_2.0.3      compiler_4.3.3       sass_0.4.9
## [13] rlang_1.1.4         tools_4.3.3         utf8_1.2.4           yaml_2.3.8
## [17] knitr_1.45          labeling_0.4.3       xml2_1.3.6           repr_1.1.7
## [21] expm_0.999-9        withr_3.0.0         numDeriv_2016.8-1.1 grid_4.3.3
## [25] fansi_1.0.6         colorspace_2.1-0    scales_1.3.0         MASS_7.3-60.0.1
## [29] tinytex_0.50        cli_3.6.2           mvtnorm_1.2-4        metafor_4.6-0
## [33] ragg_1.3.0          generics_0.1.3      rstudioapi_0.15.0    tzdb_0.4.0
## [37] cachem_1.0.8        minqa_1.2.6         splines_4.3.3        cellranger_1.1.0
## [41] base64enc_0.1-3     vctrs_0.6.5         boot_1.3-29          Matrix_1.6-5
## [45] jsonlite_1.8.8      hms_1.1.3           Formula_1.2-5        systemfonts_1.0.6
## [49] jquerylib_0.1.4     glue_1.7.0          nloptr_2.0.3         stringi_1.8.4
## [53] gtable_0.3.5        lme4_1.1-35.1       munsell_0.5.1        pillar_1.9.0
## [57] htmltools_0.5.7     R6_2.5.1            textshaping_0.3.7    rprojroot_2.0.4
## [61] evaluate_0.23       lattice_0.22-5      highr_0.10           backports_1.4.1
## [65] bslib_0.6.1         Rcpp_1.0.12         xfun_0.42            pkgconfig_2.0.3

```