

# An Empirical Assessment of Between-Study Heterogeneity in fMRI Studies

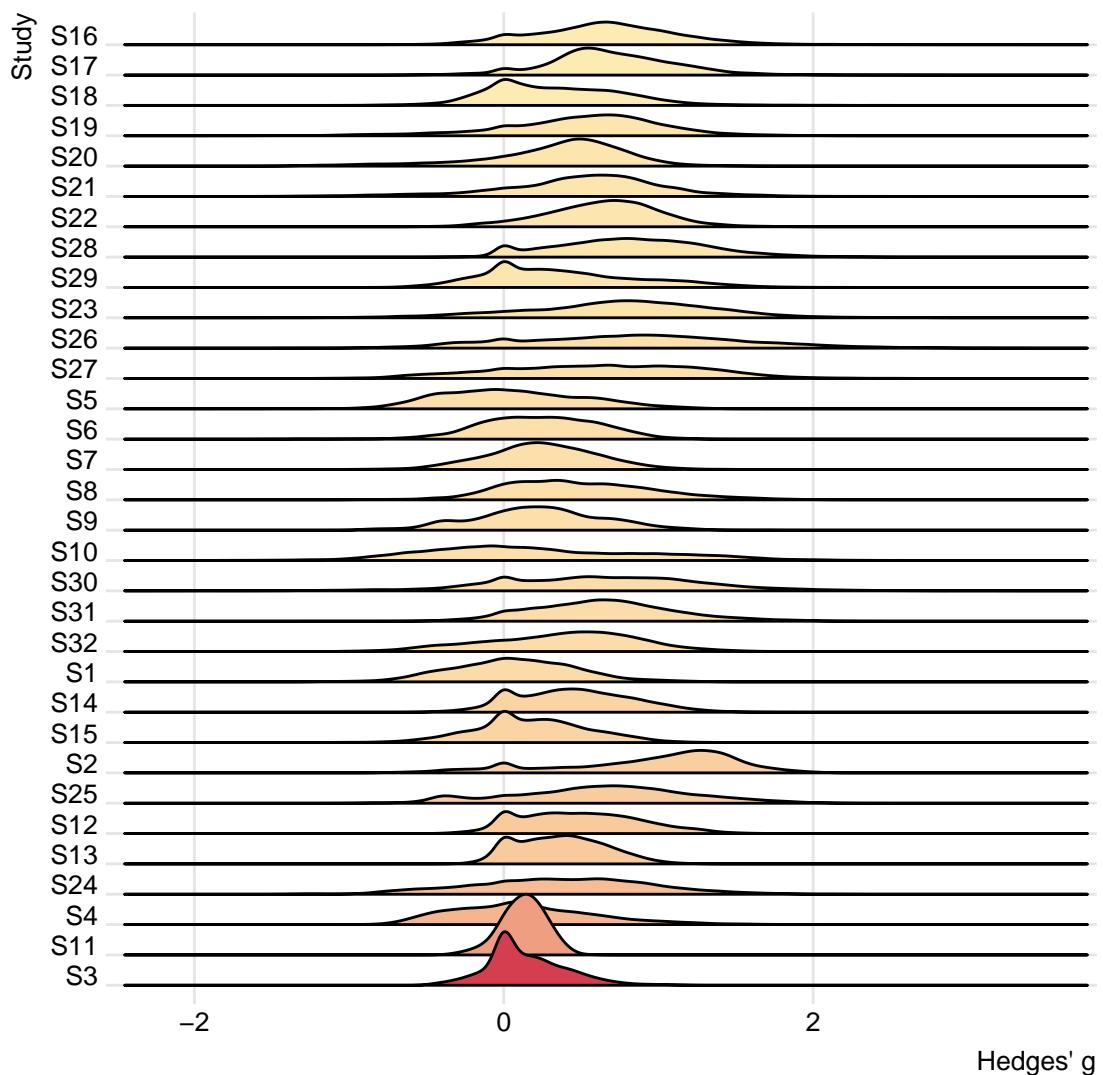
*Han Bossier*

15-1-2018

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Database</b>	<b>3</b>
2.1	Collection procedure . . . . .	3
2.2	Overview of database . . . . .	3
2.3	ROI . . . . .	4
2.4	Pre-processing . . . . .	5
<b>3</b>	<b>ROI Analysis</b>	<b>5</b>
3.1	Read in data . . . . .	5
3.2	Distributions of t-values . . . . .	6
3.3	Distributions of effect sizes . . . . .	9
3.4	Meta-analysis . . . . .	11
3.5	Brain images . . . . .	18
<b>4</b>	<b>Whole brain analysis</b>	<b>20</b>
4.1	Distributions of effect sizes . . . . .	20
4.2	Meta-analysis . . . . .	21
4.3	Brain images . . . . .	23
4.4	Comparing ROI with other brain areas . . . . .	25
<b>5</b>	<b>Moderator Analysis</b>	<b>29</b>
<b>6</b>	<b>Preliminary conclusion</b>	<b>29</b>

## Artwork



# 1 Introduction

In this report, we estimate the amount of observed between-study heterogeneity in a typical image-based fMRI meta-analysis. The goal is to obtain reasonable values for between-study heterogeneity so that we can use these in Monte-Carlo simulation studies.

At the moment, we have a database of 32 fMRI studies ( $N = 631$ ) involving a general experience of pain versus no pain. This database could be extended/curated.

Note: this report uses a regular analysis without control for some studies coming from the same laboratory. See section on moderator analysis for more information.

## 2 Database

The following link contains general information about the database. In general, we are interested in the effect of experiencing pain versus a baseline or versus experiencing no pain. We explicitly use a general definition of *pain* to obtain upper bounds on the observed between-study heterogeneity. Hence, we have studies that investigate the empathy of pain (i.e. seeing others having pain) or studies in which painful stimuli are given. There are multiple stimuli available ranging from auditory, thermal to mechanical stimuli.

### 2.1 Collection procedure

All studies are collected using the search term *pain* in NeuroVault at 11/01/2017. Results are manually checked and inputted in the database.

Note that we do not follow guidelines (such as the PRISMA guidelines) on how to collect/report studies for a meta-analysis/systematic review. Although we might add these in the future, it would involve putting more effort in collecting images that are not available through NeuroVault.

### 2.2 Overview of database

The database contains the following studies:

Study	Sample size	Type	Contrast
Braboszcz 2017	17	T-map	Painful images >Painless images (Normal state)
Hebestreit 2017	23	T-map	Ammonia stimulation (trigeminal pain) > Air in both sessions (medication and placebo)
Tamm 2017	86	T-map	Pain > no pain [no covariates]
Karjalainen 2017	35	T-map	Main effect of vicarious pain
Atlas 2010	15	beta-map <sup>1</sup>	Thermal high vs low stimulation pain
Wager 2013	15	beta-map <sup>1</sup>	Somatic pain vs baseline
Kano 2017	15	beta-map <sup>1</sup>	Visceral pain vs baseline
Rubio 2015	15	beta-map <sup>1</sup>	Visceral pain vs baseline
Unpublished	15	beta-map <sup>1</sup>	Mechanical high pressure pain vs baseline
Unpublished	15	beta-map <sup>1</sup>	Mechanical medium pressure pain vs baseline
Patil 2017	46	T-map	Outcome of pain induced to others versus baseline
Maumet 2016	Total = 334 <sup>2</sup>	T-maps	Pain versus baseline

Some notes:

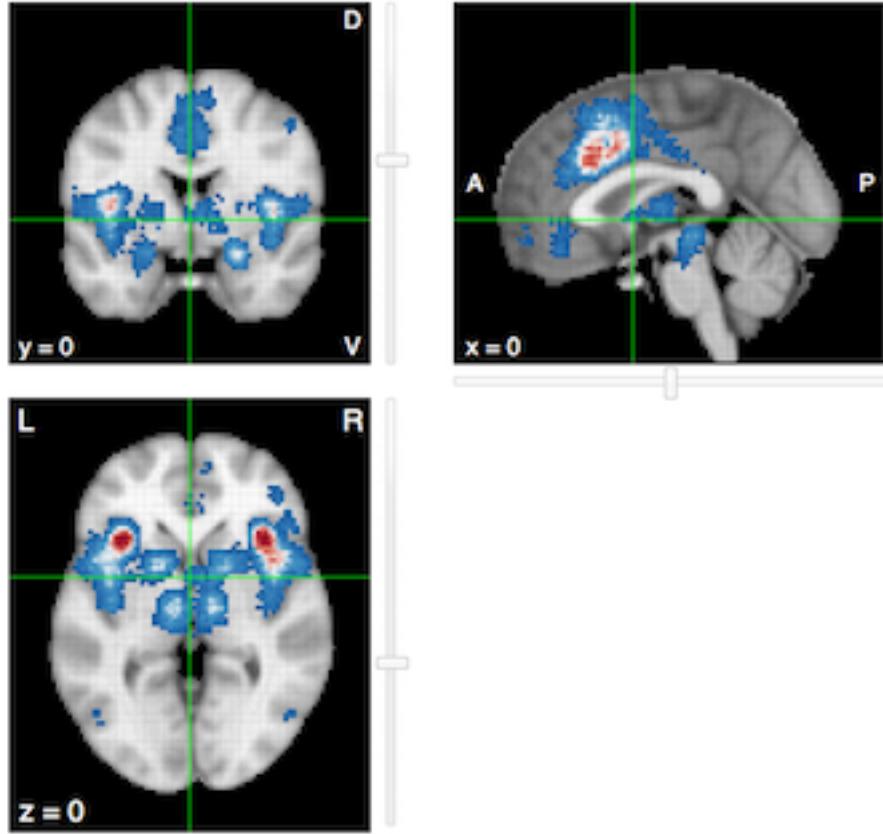


Figure 1: Region of Interest obtained using a forward inference meta-analysis with the keyword *pain*.

- We only include whole-brain analyses.
- 1) Data comes from Kragel et al, 2017. We need to pool the subjects using OLS. I assume the provided data from this study are first level beta parameter estimates each subject. Should verify with Tor Wager.
- 2) Study of Maumet et al., 2016 is about the *nidm* data structure. However, it contains **21** studies about pain. Note that these come from the same laboratory. In this report, we explicitly do not control for this. Need to verify where these studies came from.

### 2.3 ROI

In order to see whether between-study heterogeneity is different in brain areas known to be involved in pain processing, we also create a Region of Interest (ROI). To do so, we create a binary mask using NeuroSynth. After searching for the term *pain*, we obtain an automated meta-analysis of 420 studies. From here we use forward inference with a False Discovery Rate control at 0.01, the standard neurosynth procedure to create a mask. Forward inference is equal to:  $P(\text{Activation}|\text{Term})$ .

Figure 1 shows the obtained regions of interests using this procedure.

## 2.4 Pre-processing

The main pre-processing step involves getting the statistical parametric maps (mostly  $t$ -images) to the same MNI 2mm space. This is simply done through resampling within the FSL GUI. This technique involves changing the dimensions of the voxels so that the overall image matches in dimension with the MNI standard. We tried to do registration using *flirt* or *fnirt* to MNI space. However, we were unable to achieve reasonable registrations.

## 3 ROI Analysis

### 3.1 Read in data

First we read in the meta-data about the database (name of files and sample sizes). We also read in the ROI mask.

```
# Number of studies
nstud <- 32
database <- read.csv2('..../database.csv', header = TRUE, stringsAsFactors = FALSE)
# Read in ROI mask
ROI <- readNIfTI2('..../pain_pAgF_z_FDR_0.01_forward_mask')[,,]
# Dimension in 3D
DIM3D <- dim(ROI)
# Switch to array
ROI <- array(ROI, dim = prod(DIM3D))
# MNI standard (2mm)
MNI <- readNIfTI2('..../MNI152_T1_2mm_brain')[,,]
# MNI mask
MNImask <- readNIfTI2('..../MNI152_T1_2mm_brain_mask')[,,]
```

Now we load in all t-maps.

```
# Empty data matrix
allStud <- matrix(NA, nrow = prod(DIM3D), ncol = nstud)

# For loop over the studies
for(i in 1:nstud){
  # Name of dataset is first column of database
  studDat <- readNIfTI2(paste('..../Data/', database[i,'img'], '.nii', sep = ''))[,,]
  # Check if dimensions match
  if(all(dim(studDat) != DIM3D)) stop(paste0('Dimension of image ', i,
                                                'does not match MNI space'))
  # Switch to vector
  studDat <- array(studDat, dim = prod(DIM3D))
  # Values outside MNI mask to NA
  studDat[MNImask == 0] <- NA
  # Bind to data matrix
  allStud[,i] <- studDat
  # Reset
  rm(studDat)
}

# Distribution of all t-values
distrAllT <- data.frame(allStud) %>% as.tibble()
```

```

names(distrAllT) <- paste('S', 1:nstud, sep = '')
distrAllT <- gather(distrAllT, key = 'study', value = 'Tvalue')
# Now only for masked values
distrAllT <- data.frame(distrAllT, ROImask = rep(ROI, nstud)) %>% as.tibble()
maskedVox <- filter(distrAllT, ROImask == 1)
# Create factor of study
maskedVox$study <- factor(maskedVox$study, levels = paste('S', 1:nstud, sep = ''))

```

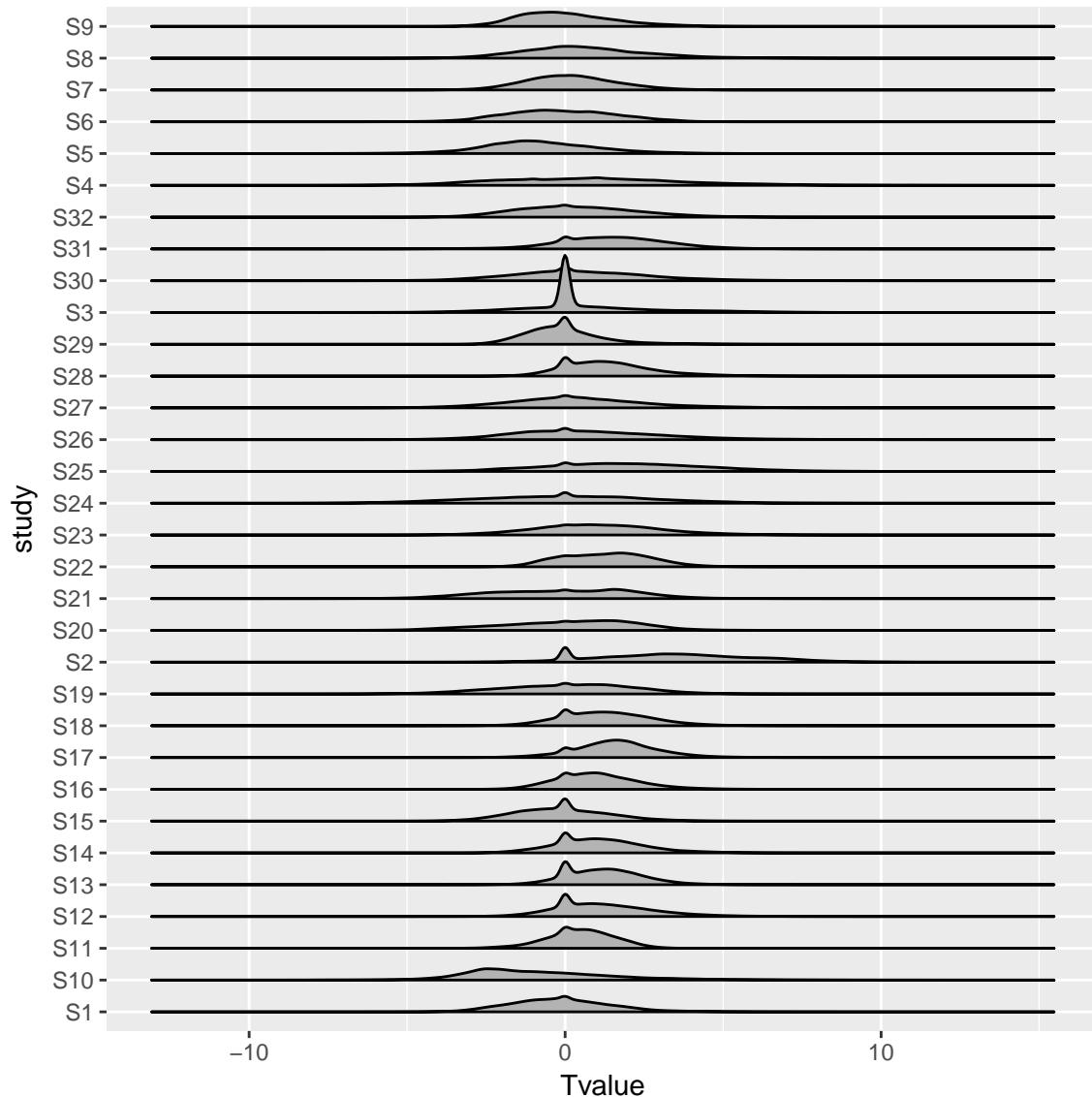
## 3.2 Distributions of t-values

Before focussing on distributions within ROI, let us look at the distribution of the t-values over all voxels in each study.

```

# Raw ridge plot
ggplot(distrAllT, aes(y = study, x = Tvalue)) +
  geom_density_ridges2()

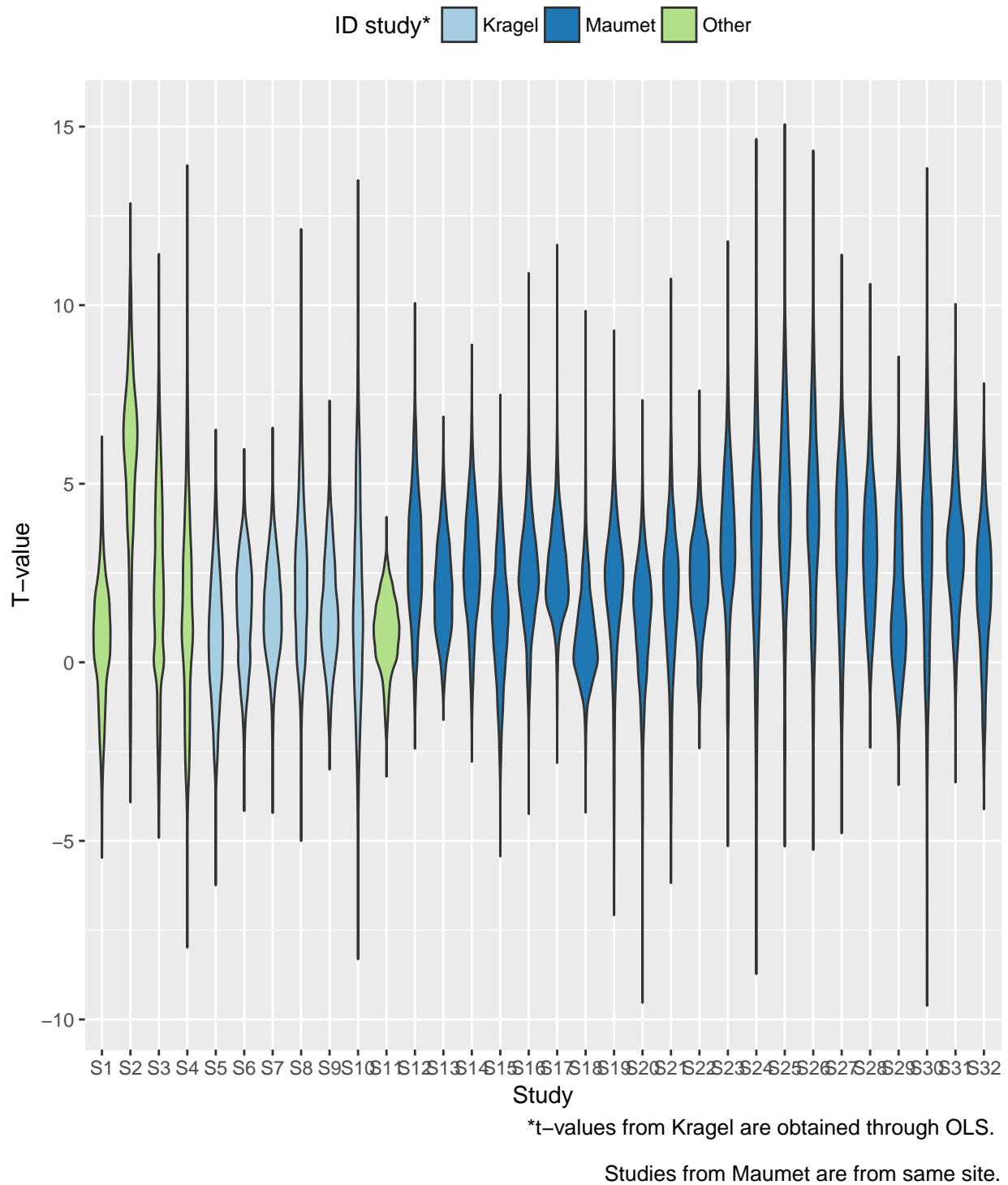
```



Now we plot the distributions of the t-values within the ROI. See caption of note on subdivision.

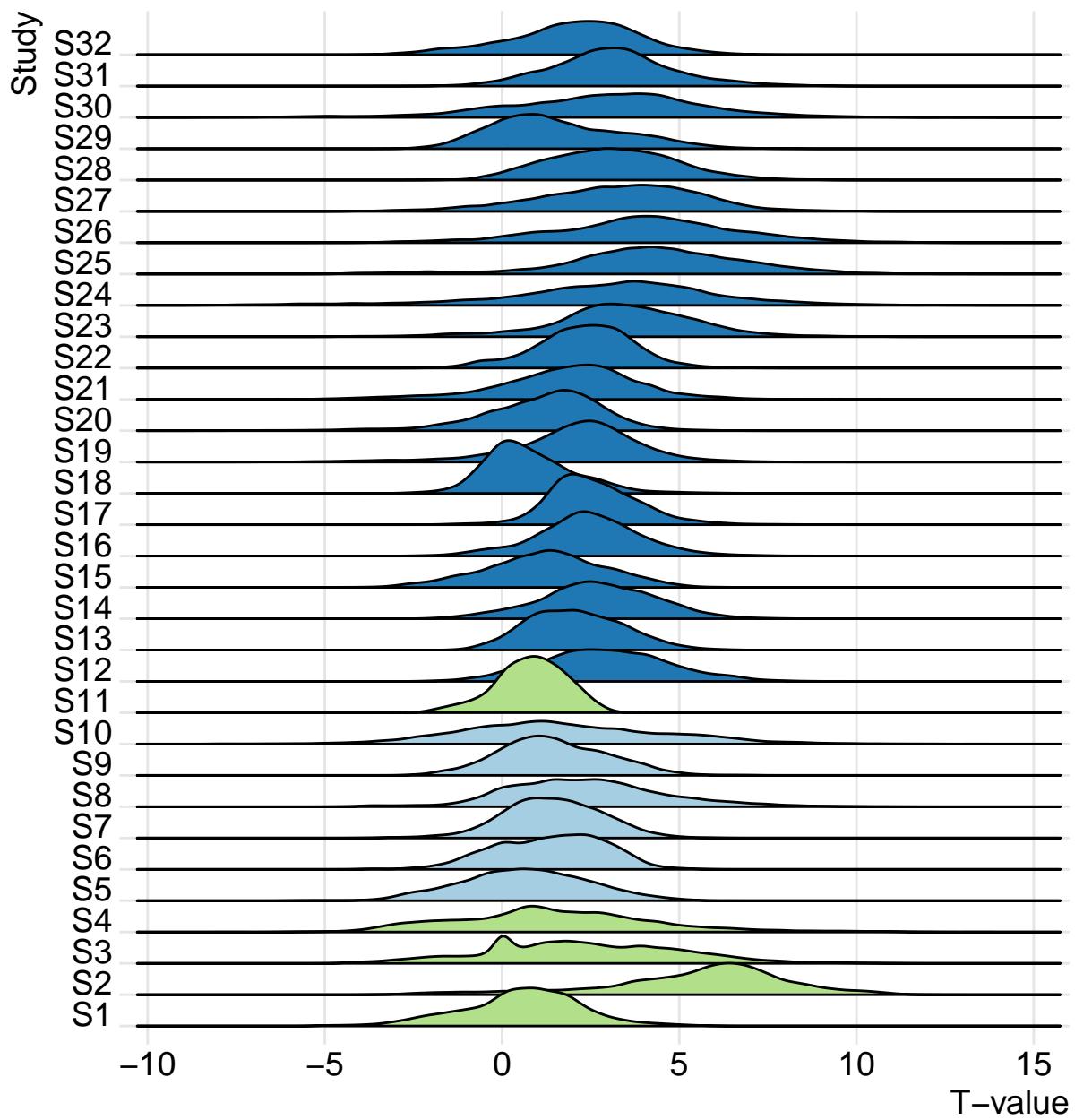
```
# Add variable for Kragel, Maumet and other study
IDvar <- c(rep('Other', 4), rep('Kragel', 6), 'Other', rep('Maumet', 21))
maskedVox$IDvar <- factor(rep(IDvar, each = sum(ROI)))
# Violin plot
ggplot(maskedVox, aes(x = study, y = Tvalue)) +
  geom_violin(aes(fill = IDvar)) +
  scale_y_continuous('T-value') +
  scale_x_discrete('Study') +
  scale_fill_brewer('ID study*', type = 'qual', palette = 3) +
  ggtitle("T-values from voxels in ROI") +
  theme(legend.position = 'top') +
  labs(caption = "*t-values from Kragel are obtained through OLS. \n
    Studies from Maumet are from same site.")
# Ridge plot
ggplot(maskedVox, aes(y = study, x = Tvalue)) +
  geom_density_ridges2(aes(fill = IDvar)) +
  scale_y_discrete('Study', expand = c(0.01, 0)) +
  scale_x_continuous('T-value', expand = c(0.01, 0)) +
  scale_fill_brewer('ID study*', type = 'qual', palette = 3) +
  ggtitle("T-values from voxels in ROI") +
  labs(caption = "*t-values from Kragel are obtained through OLS. \n
    Studies from Maumet are from same site.") +
  theme_ridges() +
  theme(legend.position = 'top')
```

## T-values from voxels in ROI



### T-values from voxels in ROI

ID study\* Kragel Maumet Other



\*t-values from Kragel are obtained through OLS.

Studies from Maumet are from same site.

### 3.3 Distributions of effect sizes

First we transform the reported t-value for each voxel to Hedges'  $g$ .

This is done through:

$$g = \frac{t}{\sqrt{N}} \times J, \quad (1)$$

with  $N$  the study sample size and  $J$  a correction factor, defined as:

$$J = 1 - \left( \frac{3}{(4 \times (N - 1)) - 1} \right). \quad (2)$$

We also calculate the within-study variance through:

$$\text{Var}(g) = \frac{1}{N} + \left[ 1 - \left( \frac{\Gamma((N - 2)/2)}{\Gamma((N - 1)/2)} \right)^2 \times \frac{(N - 3)}{2} \right] \times g^2 \quad (3)$$

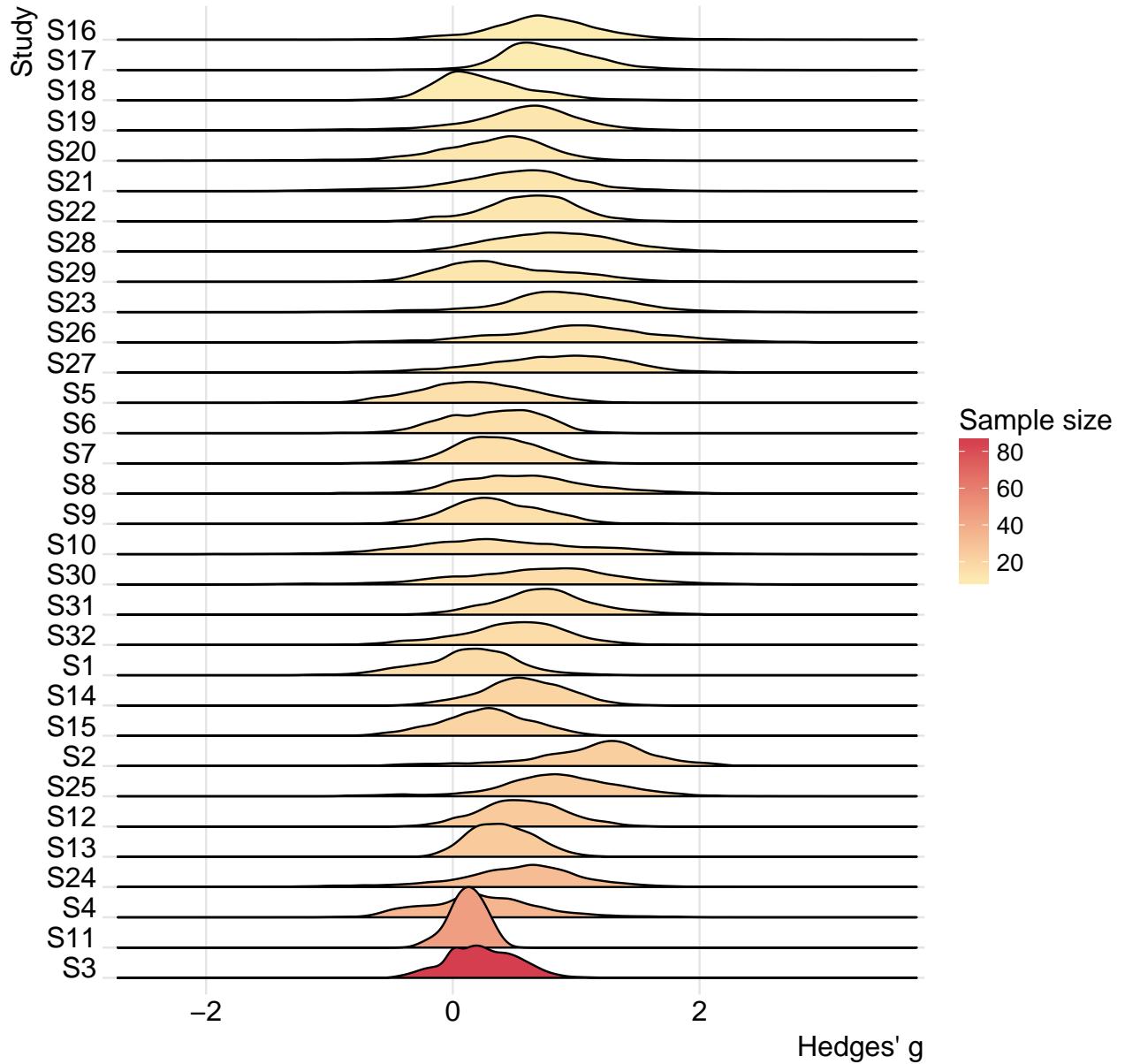
```
dataG <- database %>% as.tibble() %>%
  mutate(study = factor(paste('S', 1:nstud, sep = '')),
         levels = paste('S', 1:nstud, sep = ''))) %>%
  select(-img) %>% right_join(., maskedVox, by = 'study') %>%
  arrange(samplesize) %>%
  # Now transform to hedges' g and calculate variance
  mutate(hedgeG = hedgeG(t = Tvalue, N = samplesize),
         varG = varHedge(g = hedgeG, N = samplesize))
```

Now we plot the distributions of  $g$  within the ROI, according to the sample size.

```
OrderFactorSS <- database %>% as.tibble() %>%
  mutate(study = factor(paste('S', 1:nstud, sep = '')),
         levels = paste('S', 1:nstud, sep = ''))) %>%
  select(-img) %>% arrange(samplesize)
dataG$study <- factor(dataG$study, levels = rev(OrderFactorSS$study))
ggplot(dataG, aes(y = study, x = hedgeG)) +
  geom_density_ridges(aes(fill = samplesize), scale = 2) +
  scale_y_discrete('Study', expand = c(0.01, 0)) +
  scale_x_continuous("Hedges' g", expand = c(0.01, 0)) +
  scale_fill_gradient2('Sample size',
                       low = '#3288bd',
                       mid = '#ffffbf',
                       high = '#d53e4f') +
  ggtitle("Hedges' g from voxels within ROI",
          subtitle = 'Sorted according to study sample size (bottom to top: high N to low)') +
  theme_ridges()
```

### Hedges' $g$ from voxels within ROI

Sorted according to study sample size (bottom to top: high N to low)



Before we continue, note that smaller studies tend to have more positive distributions. In fact, this could indicate that we have publication bias in our dataset. However, the smallest 6 studies are actually obtained by fitting an OLS model to 15 randomly chosen subjects from their corresponding study dataset. Hence these studies are not an actual representation of the published work. Aside from these 6, the smaller studies do tend to show positive effect sizes. Though so do some larger studies.

### 3.4 Meta-analysis

Now let us estimate the weighted averages using a random-effects model with the method of moments estimator for between-study heterogeneity. We will extract the weighted average (denoted as  $\beta$ ) and the method of moments estimate for  $\tau^2$ , the between-study variability. Note that the latter is a standardized measure and hence operates on the same scale as Hedges'  $g$ . We also extract a measure for heterogeneity  $I^2$

defined as:

$$I^2 = \frac{Q - df}{Q} \times 100\%.$$

Here,  $Q$  is the amount of observed dispersion between studies. Since this quantity is measured on a standardized scale, its expected value equals  $df$ , the degrees of freedom. Therefore,  $I^2$  is the ratio between excess dispersion over total (observed) dispersion. It is a descriptive statistic of the amount of study heterogeneity in relation to the total variability. Note though that we assume homogeneous within-study variability for the interpretation of  $I^2$ . This is probably not the case! We also extract  $H^2$ . This is defined as:

$$H^2 = \frac{Q}{k - 1}.$$

Moreover, we add a moderator variable to control for acquisition site. Studies from the Maumet et al. database are acquired in the same laboratory. Hence we add a variable to code the 32 studies (*Maumet* versus *Other*). This corresponds to a categorical dummy variable with *Maumet* being the reference level. Note that this might reduce the amount of unexplained variability, which is not what we are aiming for.

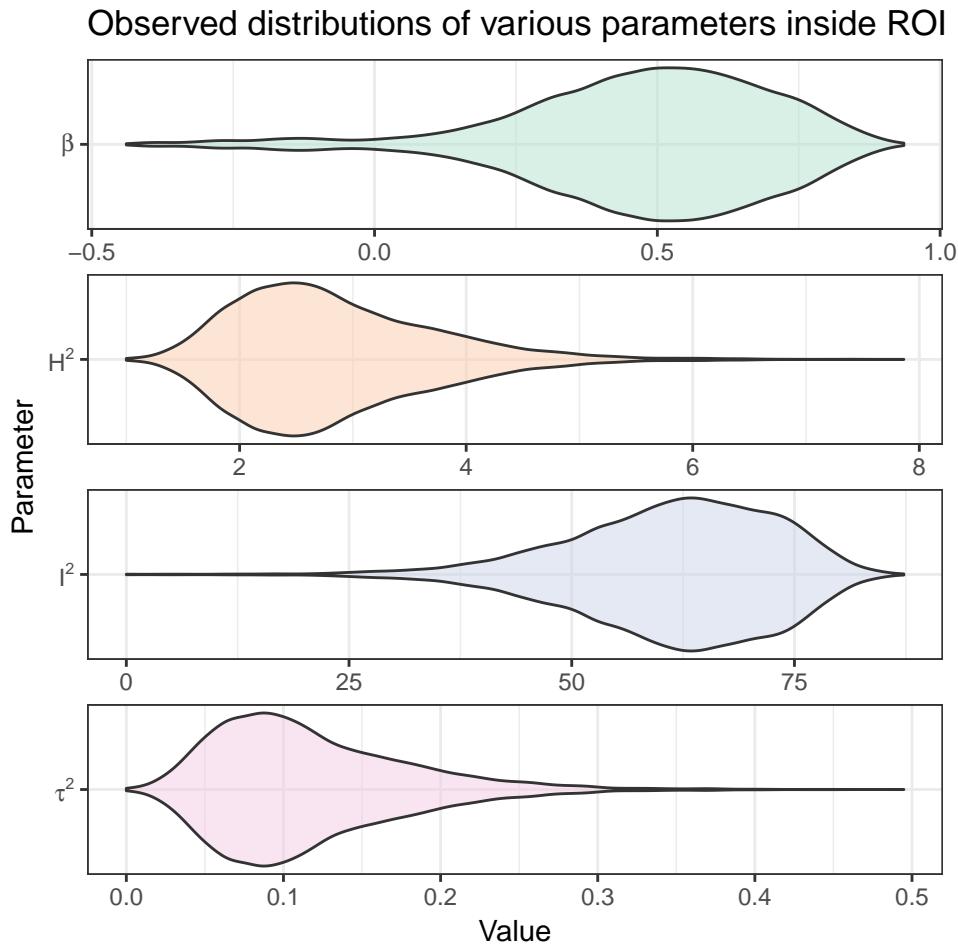
```
# First have vector with study ID if coming from Maumet database
# Note, we do not use this variable here!
IDmaumet <- database %>% mutate(study = paste('S', 1:nstud, sep = '')) %>%
  mutate(CheckMaumet = ifelse(str_detect(img, 'Maumet'), 'Maumet', 'Other')) %>%
  filter(CheckMaumet == 'Maumet') %>%
  select(study) %>% unlist()
names(IDmaumet) <- 1:length(IDmaumet)

# Now add voxel ID to dataframe and create a factor for the moderator variable.
# This is to control for the fact that studies from Maumet et al. are from same site.
voxID <- dataG %>% mutate(voxel = rep(1:sum(ROI), nstud)) %>%
  mutate(Group = factor(ifelse(study %in% IDmaumet,
                               'Maumet', 'Other')))

# Although purrr is a bit slower, it is possible to conveniently
# extract multiple coefficients.
# To do so, we use our custom function fitDL. This takes data from each voxel,
# fits the model and extracts I2, H2 and a beta estimate (which is the weighted average).
estTau <- voxID %>%
  split(.\$voxel) %>%
  map_df(~ fitDL(data=.x))
```

Let us plot the distributions over all voxels in the ROI of these 4 parameters.

```
estTau %>% gather(key = 'Parameter', value = 'Value') %>%
  ggplot(., aes(x = Parameter, y = Value)) +
  geom_violin(aes(fill = Parameter), alpha = 0.5) +
  scale_fill_brewer('', type = 'qual', palette = 5) +
  facet_wrap(~ Parameter, scales = 'free', ncol = 1) +
  coord_flip() +
  scale_x_discrete('Parameter', labels = c('beta' = expression(beta),
                                             'H2' = expression(H^2),
                                             'I2' = expression(I^2),
                                             'tau2' = expression(tau^2))) +
  guides(fill = FALSE) +
  ggtitle('Observed distributions of various parameters inside ROI') +
  theme_bw() +
  theme(strip.background = element_blank(),
        strip.text.x = element_blank())
```



Overall, we observe mostly positive effect sizes within the ROI. For most voxels within the ROI, we estimate a modest amount of between-study heterogeneity ( $\tau^2$ ). However, the maximum observed estimate for  $\tau^2$  equals 0.4948224. This seems not trivial considering we only observe a maximum effect size of 0.9358934. Recall that we could roughly state that 95% of the true effects will fall in the range of:

$$\hat{\beta} \pm 1.96\hat{\sigma}$$

For instance, consider the voxel with the highest value for  $\hat{\tau}$ :

```
filter(estTau, tau2 == max(tau2))
```

```
# A tibble: 1 x 4
  tau2      I2      H2      beta
  <dbl>    <dbl>    <dbl>    <dbl>
1 0.4948224 87.23366 7.833097 0.8432343
```

We then get 95% of the true effects for this voxel within the interval: [-0.5355005; 2.2219691].

To see which values for the estimated effect sizes are associated with  $\tau^2 > 0.10$ , we look at the following two figures:

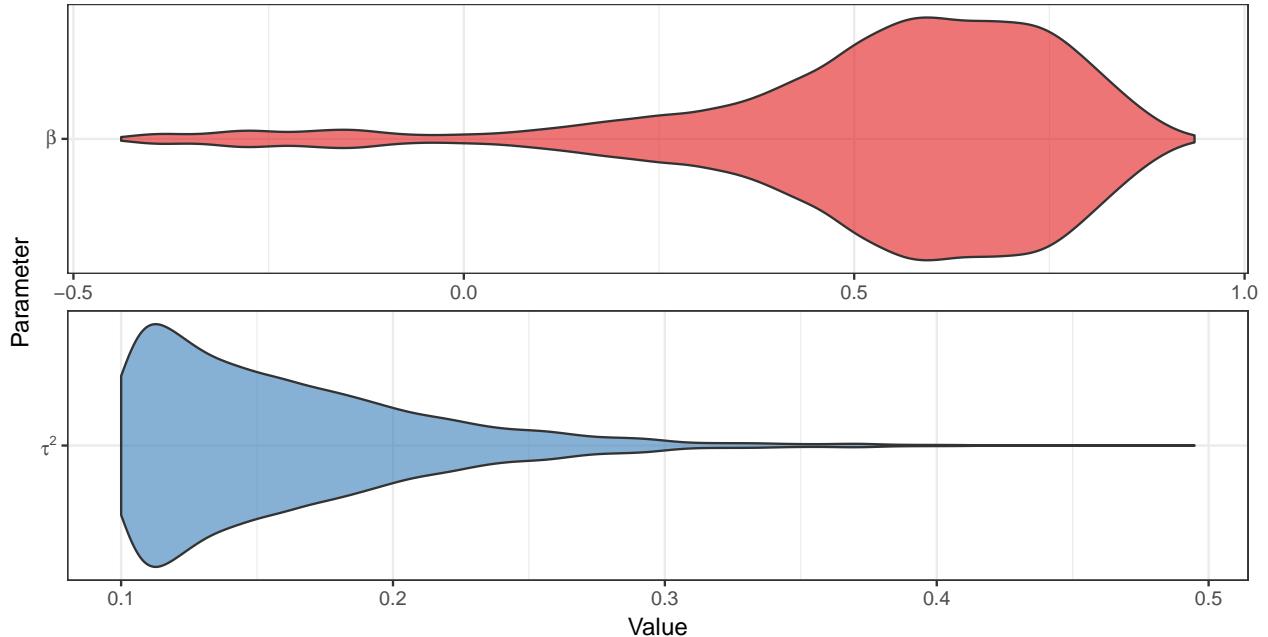
```
estTau %>% filter(tau2 >= 0.10) %>% select(tau2, beta) %>%
  gather(key = 'Parameter', value = 'Value') %>%
  ggplot(., aes(x = Parameter, y = Value)) +
  geom_violin(aes(fill = Parameter), alpha = 0.6) +
  scale_fill_brewer('!', type = 'qual', palette = 6) +
  facet_wrap(~ Parameter, scales = 'free', ncol = 1) +
```

```

coord_flip() +
scale_x_discrete('Parameter', labels = c('beta' = expression(beta),
  'tau2' = expression(tau^2))) +
guides(fill = FALSE) +
ggtitle(expression(Observed ~ distributions ~ of ~ weighted ~ average ~ when ~ tau^2 >= 0.10)) +
theme_bw() +
theme(strip.background = element_blank(),
  strip.text.x = element_blank())

```

Observed distributions of weighted average when  $\tau^2 \geq 0.1$



```

lineData <- estTau %>% mutate(tau = sqrt(tau2)) %>%
  select(tau, beta) %>%
  mutate(x = 'beta', xend = 'tau') %>%
  mutate(censor_tau =
    ifelse(tau > quantile(tau, probs = 0.95), 'high','low'),
    censor_beta =
    ifelse(beta > quantile(beta, probs = 0.95), 'high','low'))

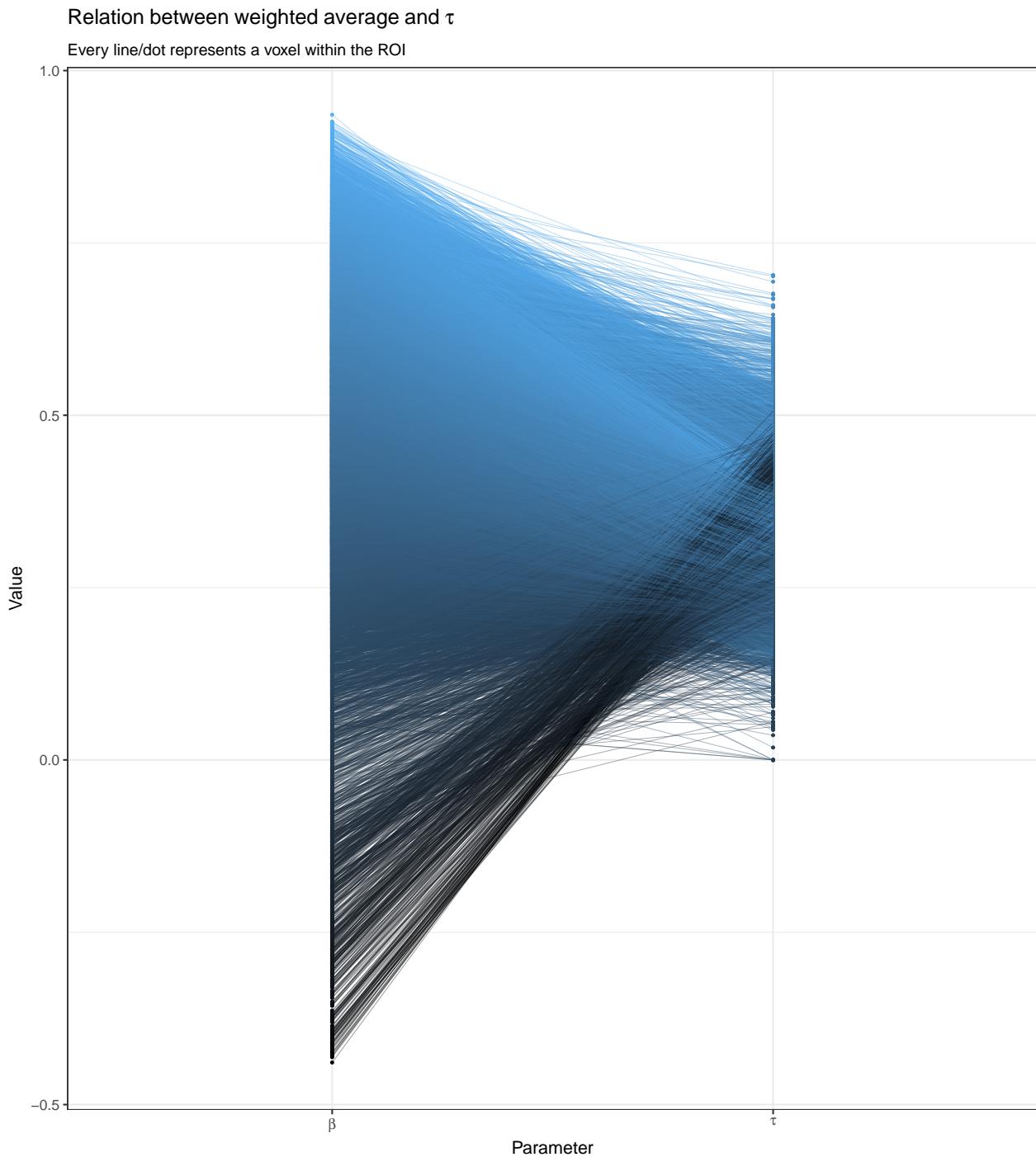
estTau %>% mutate(tau = sqrt(tau2)) %>%
  select(tau, beta) %>%
  gather(key = 'Parameter', value = 'Value') %>%
  ggplot(aes(x = Parameter, y = Value)) +
  geom_point(aes(colour = Value), size = 0.5) +
  geom_segment(data = lineData,
    aes(x = x, xend = xend,
        y = beta, yend = tau,
        colour = beta),
    size = 0.05,
    alpha = 0.4) +
  scale_colour_gradient(low = "#000000", high = "#56B1F7") +
  guides(colour = FALSE)

```

```

scale_x_discrete(name = 'Parameter',
                  labels = c(beta = expression(beta),
                             tau = expression(tau))) +
ggtitle(expression(Relation ~ between ~ weighted ~ average ~ and ~ tau),
       subtitle = 'Every line/dot represents a voxel within the ROI') +
theme_bw()

```



The latter plot connects every estimated weighted average with its corresponding estimate for  $\tau$ . Since the range of  $\tau$  is larger than  $\tau^2$ , we use the latter for visualization purposes. As can be seen, both large *and* low

estimated effect sizes are associated with higher values for  $\hat{\tau}$ . Though high effect sizes are mainly associated with higher values for  $\tau$ .

In the following figure, we plot on the left hand side the relation between the weighted averages and the voxels with the estimate for  $\tau$  exceeding its 95% quantile. On the right hand side, we plot the 95% highest observed weighted average and their corresponding estimated  $\hat{\tau}$ .

```
Pcen_tau <- estTau %>% mutate(tau = sqrt(tau2)) %>%
  select(tau, beta) %>%
  gather(key = 'Parameter', value = 'Value') %>%
  ggplot(aes(x = Parameter, y = Value)) +
  geom_point(aes(colour = Value), size = 0.5) +
  geom_segment(data = lineData,
    aes(x = x, xend = xend,
        y = beta, yend = tau,
        colour = beta,
        alpha = censor_tau),
    size = 0.05) +
  scale_alpha_manual(values = c('high' = 1, 'low' = 0)) +
  scale_colour_gradient(low = "#000000", high = "#56B1F7") +
  guides(colour = FALSE) +
  guides(alpha = FALSE) +
  scale_x_discrete(name = 'Parameter',
    labels = c(beta = expression(beta),
               tau = expression(tau > 95^th ~ percentile))) +
  ggtitle(expression(Relation ~ between ~ w ~ av ~ and ~ tau),
         subtitle = expression(Criterium: ~ tau > its ~ 95^th ~ percentile)) +
  theme_bw()

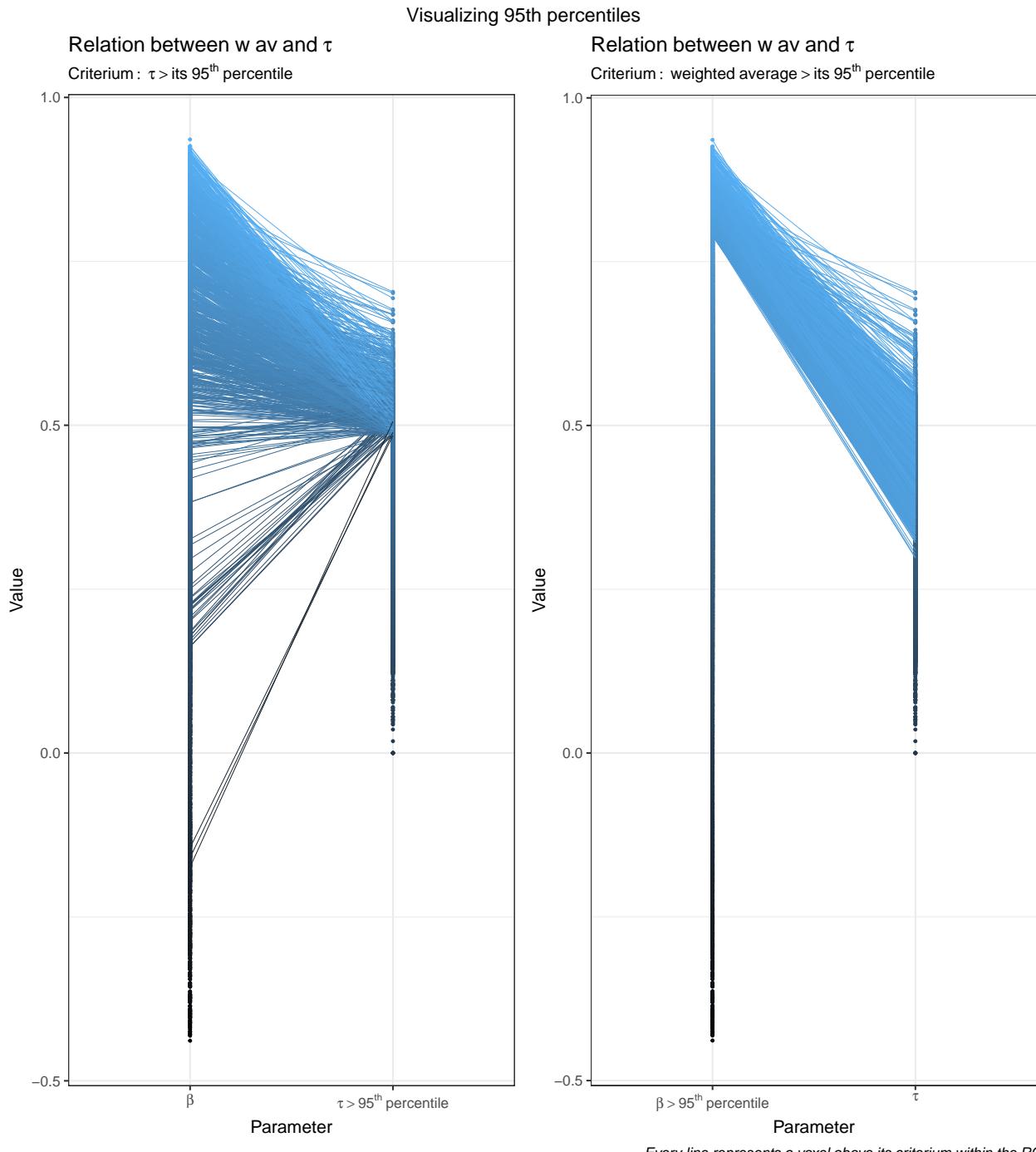
Pcen_beta <- estTau %>% mutate(tau = sqrt(tau2)) %>%
  select(tau, beta) %>%
  gather(key = 'Parameter', value = 'Value') %>%
  ggplot(aes(x = Parameter, y = Value)) +
  geom_point(aes(colour = Value), size = 0.5) +
  geom_segment(data = lineData,
    aes(x = x, xend = xend,
        y = beta, yend = tau,
        colour = beta,
        alpha = censor_beta),
    size = 0.05) +
  scale_alpha_manual(values = c('high' = 1, 'low' = 0)) +
  scale_colour_gradient(low = "#000000", high = "#56B1F7") +
  guides(colour = FALSE) +
  guides(alpha = FALSE) +
  scale_x_discrete(name = 'Parameter',
    labels = c(beta = expression(beta > 95^th ~ percentile),
               tau = expression(tau))) +
  ggtitle(expression(Relation ~ between ~ w ~ av ~ and ~ tau),
         subtitle = expression(Criterium: ~ weighted ~ average > its ~ 95^th ~ percentile)) +
  theme_bw()

gridExtra::grid.arrange(
  Pcen_tau,
  Pcen_beta,
  nrow = 1,
```

```

top = "Visualizing 95th percentiles",
bottom = grid::textGrob(
  "Every line represents a voxel above its criterium within the ROI",
  gp = grid::gpar(fontface = 3, fontsize = 9),
  hjust = 1,
  x = 1
)
)

```



Next, we also observe fairly large values for  $I^2$ . Higgins et al. (2003) suggested the following benchmarks for

Table 2: Quantiles for estimated between-study variability in ROI

$\tau^2$	$I^2$	Quantile
0.000	0.000	0%
0.040	40.322	5%
0.050	45.896	10%
0.058	49.417	15%
0.065	52.241	20%
0.072	54.376	25%
0.078	56.392	30%
0.085	58.172	35%
0.091	59.732	40%
0.097	61.229	45%
0.104	62.605	50%
0.110	63.959	55%
0.118	65.353	60%
0.127	66.804	65%
0.137	68.313	70%
0.149	69.853	75%
0.163	71.525	80%
0.179	73.248	85%
0.200	75.102	90%
0.234	77.540	95%
0.495	87.283	100%

$I^2$ : 25% = low, 50% = moderate and 75% = high.

To summarise, we provide the quantiles for  $\tau^2$  and  $I^2$  in table 2.

```
knitr::kable(data.frame(tau2 = quantile(estTau$tau2, probs = seq(0,1,by = 0.05)),
                        I2 = quantile(estTau$I2, probs = seq(0,1,by = 0.05)),
                        Quantile = paste(seq(0,1,by = 0.05) * 100, '%', sep = '')),
               caption = 'Quantiles for estimated between-study variability in ROI',
               row.names = FALSE, format = 'latex', booktabs = TRUE,
               col.names = c("$\\tau^2$", "$I^2$\", "Quantile"),
               digits = 3)
```

### 3.5 Brain images

Let us plot the observed weighted averages onto the anatomical MNI image.

```
# Weighted averages back to 3D
wAvg_3D <- array(NA, dim = DIM3D)
wAvg_3D[ROI == 1] <- estTau$beta

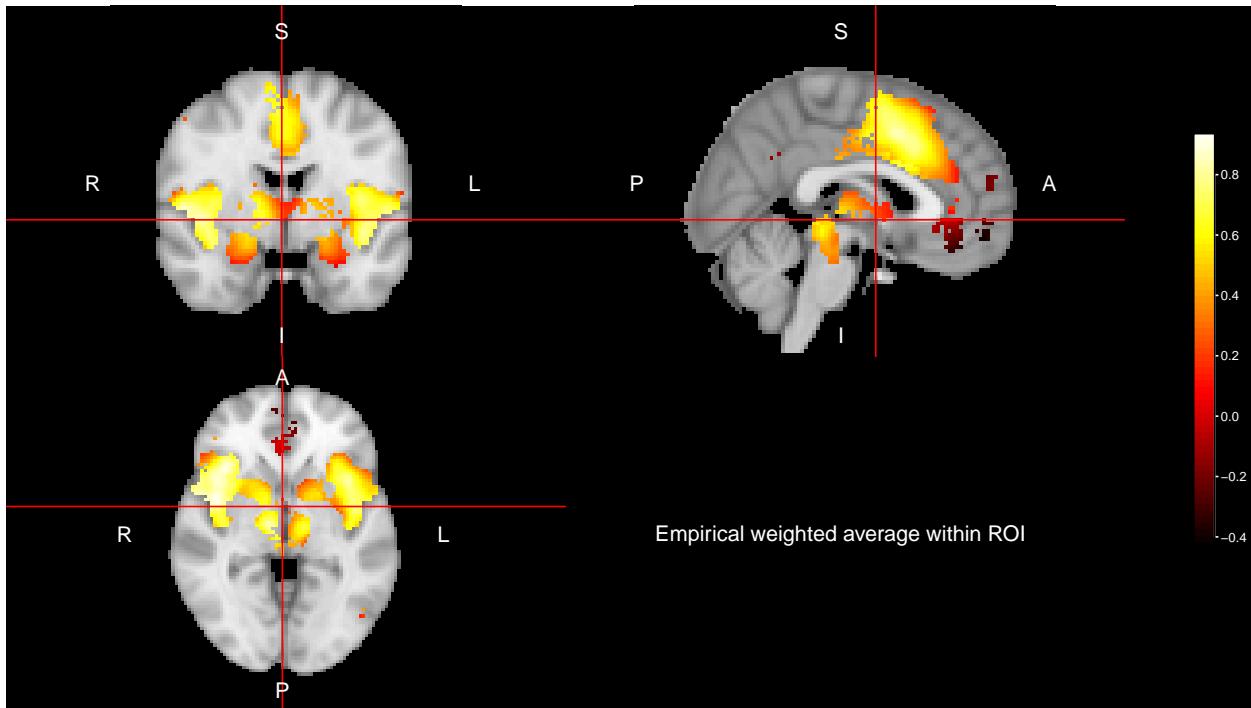
# Coordinate: center of MNI
xyz <- c(45, 63, 36)

# Plot using ortho2
ortho2(x = MNI, y = wAvg_3D, xyz = xyz, ycolorbar = TRUE,
       ybreaks = c(seq(min(wAvg_3D, na.rm = TRUE),
```

```

max(wAvg_3D, na.rm = TRUE), length.out = 65)),
text = "Empirical weighted average within ROI",
text.cex = 1)

```



And now the same for between-study variability.

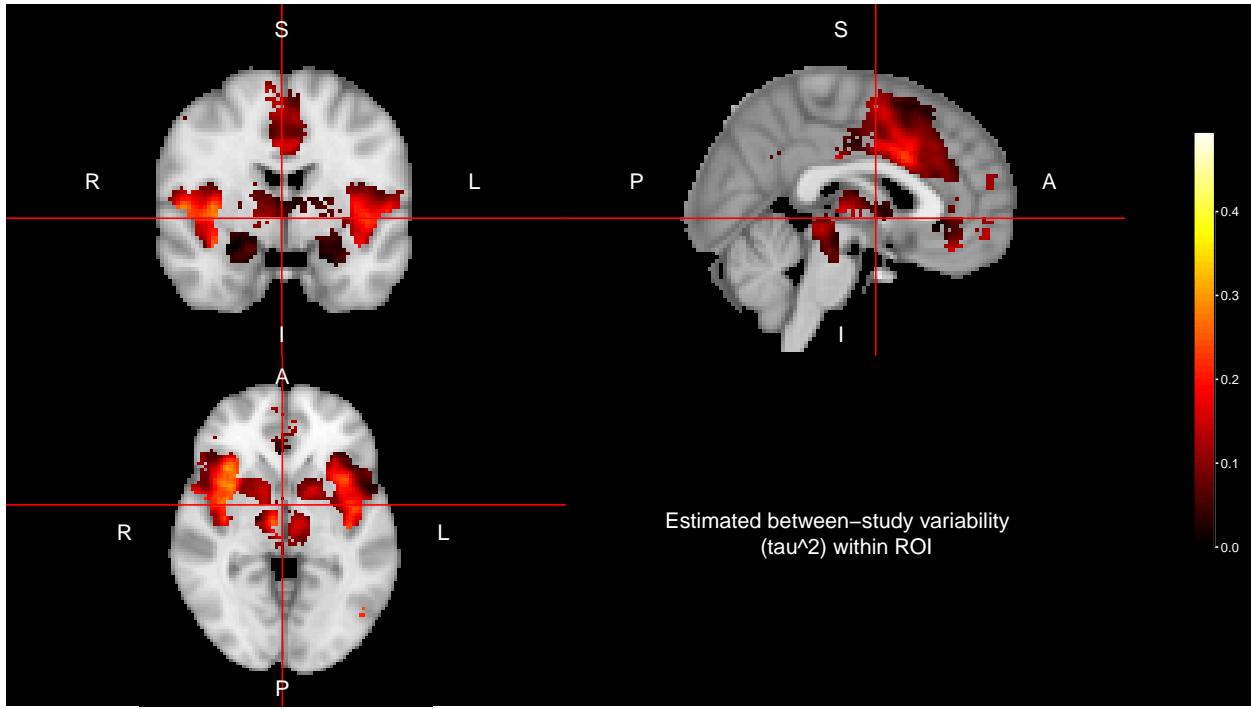
```

# Weighted averages back to 3D
tau_3D <- array(NA, dim = DIM3D)
tau_3D[ROI == 1] <- estTau$tau2

# Coordinate: center of MNI
xyz <- c(45, 63, 36)

# Plot using ortho2
ortho2(x = MNI, y = tau_3D, xyz = xyz, ycolorbar = TRUE,
       ybreaks = c(seq(min(tau_3D, na.rm = TRUE),
                      max(tau_3D, na.rm = TRUE), length.out = 65)),
       text = "Estimated between-study variability \n (tau^2) within ROI",
       text.cex = 1)

```



## 4 Whole brain analysis

In this section, we compare the ROI analysis with a whole brain analysis. I am especially interested in checking whether there would be a different distribution of between-study heterogeneity in regions that are not systematically involved in *pain* processing.

### 4.1 Distributions of effect sizes

Now let us look at the distributions of the effect sizes in the entire brain for each study.

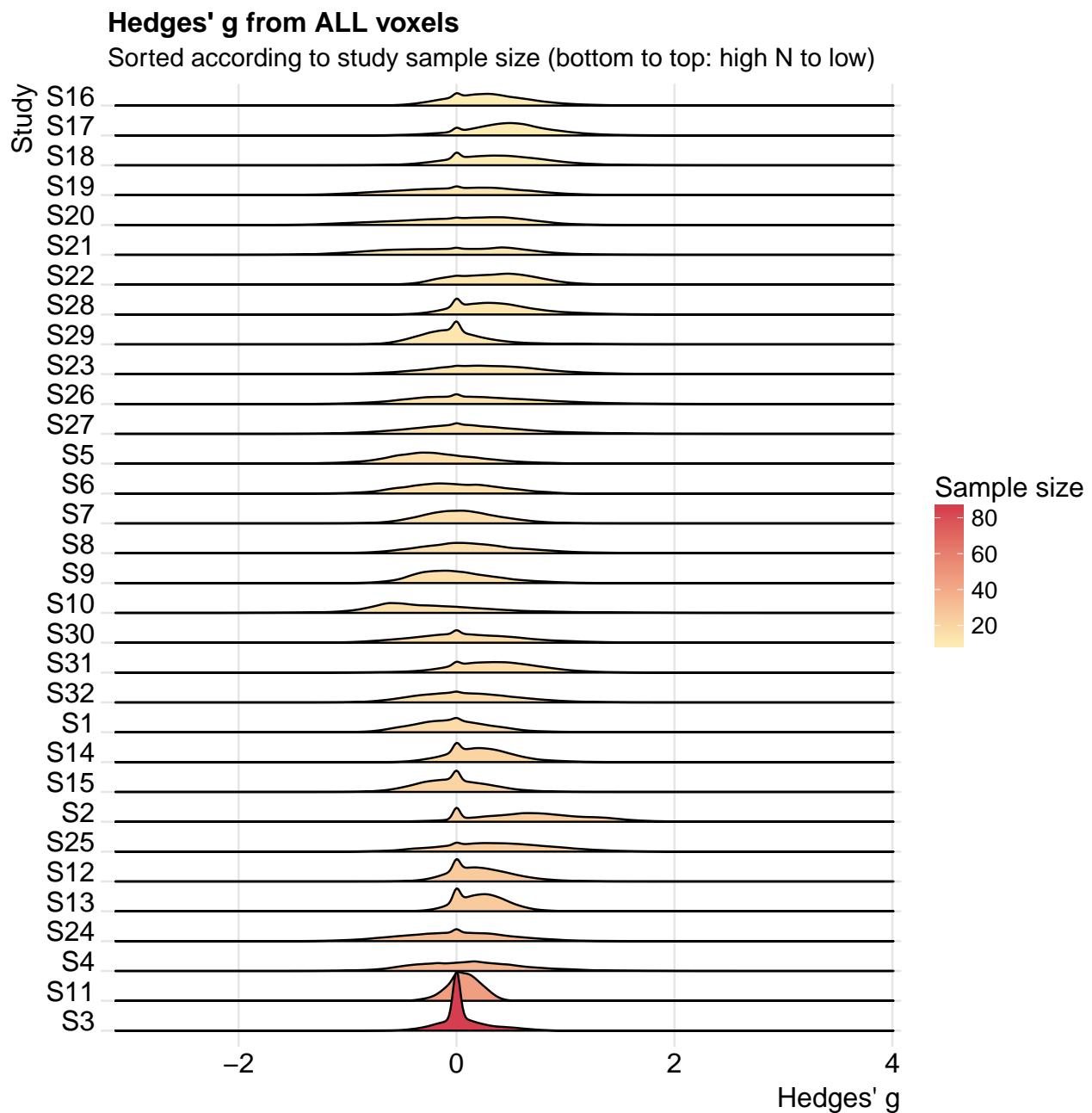
```
data_all_G <- database %>% as.tibble() %>%
  mutate(study = factor(paste('S', 1:nstud, sep = '')),
         levels = paste('S', 1:nstud, sep = ''))) %>%
  select(-img) %>% right_join(., distrAllT, by = 'study') %>%
  arrange(samplesize) %>%
  # Now transform to hedges' g and calculate variance
  mutate(hedgeG = hedgeG(t = Tvalue, N = samplesize),
         varG = varHedge(g = hedgeG, N = samplesize))

data_all_G$study <- factor(data_all_G$study, levels = rev(OrderFactorSS$study))
ggplot(data_all_G, aes(y = study, x = hedgeG)) +
  geom_density_ridges2(aes(fill = samplesize), scale = 2) +
  scale_y_discrete('Study', expand = c(0.01, 0)) +
  scale_x_continuous("Hedges' g", expand = c(0.01, 0)) +
  scale_fill_gradient2('Sample size',
                       low = '#3288bd',
                       mid = '#ffffbf',
                       high = '#d53e4f') +
  ggtitle("Hedges' g from ALL voxels",
```

```

subtitle = 'Sorted according to study sample size (bottom to top: high N to low)' +
theme_ridges()

```



There is a clear dominance of zero valued voxels. However, there are also voxels with negative values for Hedges'  $g$ .

## 4.2 Meta-analysis

Now let us again estimate a random effects model, controlling for studies from the same site.

```

# Add voxel ID to dataframe and create a factor for the moderator variable.
# Note, we do not use this variable.
voxID_all <- data_all_G %>% mutate(voxel = rep(1:prod(DIM3D), nstud)) %>%
    mutate(Group = factor(ifelse(study %in% IDmaumet,
                                'Maumet', 'Other')))

# Although purrr is a bit slower, it is possible to conveniently
# extract multiple coefficients.
# To do so, we use our custom function fitDL. This takes data from each voxel,
# fits the model and extracts I2, H2 and a beta estimate (which is the weighted average).
# Note that we only process the masked voxels.
estTau_all <- voxID_all %>%
    mutate(MNImask = rep(array(MNImask, dim = prod(DIM3D)), nstud)) %>%
    filter(MNImask == 1) %>%
    split(.~voxel) %>%
    map_df(~ fitDL(data=.x))

```

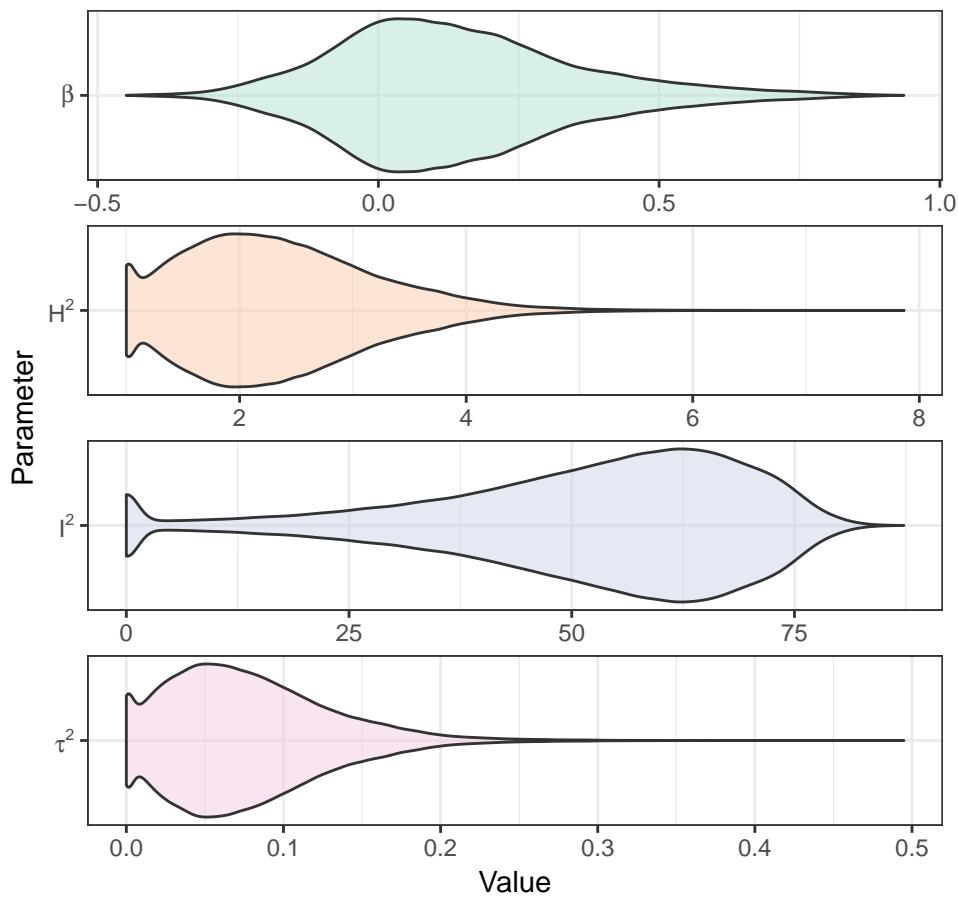
Let us plot the distributions of these 4 parameters over all voxels in the MNI mask.

```

estTau_all %>% gather(key = 'Parameter', value = 'Value') %>%
    ggplot(., aes(x = Parameter, y = Value)) +
    geom_violin(aes(fill = Parameter), alpha = 0.5) +
    scale_fill_brewer(' ', type = 'qual', palette = 5) +
    facet_wrap(~ Parameter, scales = 'free', ncol = 1) +
    coord_flip() +
    scale_x_discrete('Parameter', labels = c('beta' = expression(beta),
                                              'H2' = expression(H^2),
                                              'I2' = expression(I^2),
                                              'tau2' = expression(tau^2))) +
    guides(fill = FALSE) +
    ggtitle('Whole brain distributions of various parameters') +
    theme_bw() +
    theme(strip.background = element_blank(),
          strip.text.x = element_blank())

```

### Whole brain distributions of various parameters



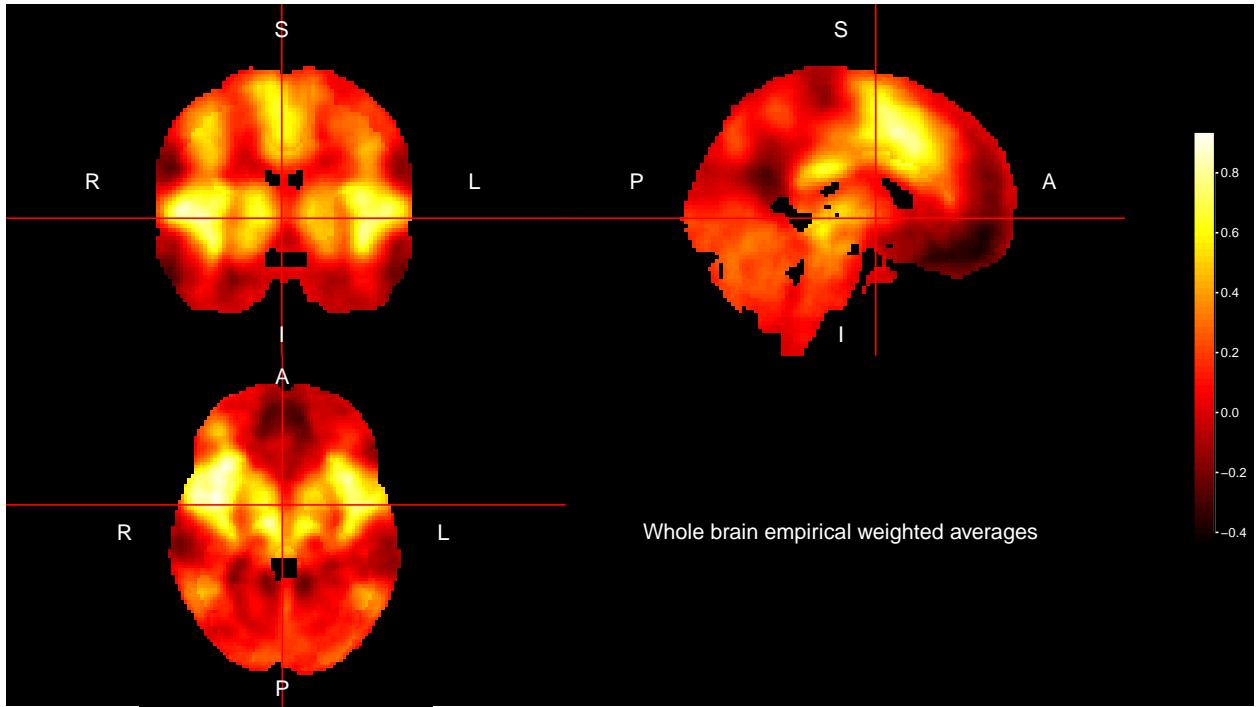
### 4.3 Brain images

Let us plot the observed weighted averages onto the anatomical MNI image.

```
# Weighted averages back to 3D
wAvg_3D <- array(NA, dim = DIM3D)
wAvg_3D[MNImask == 1] <- estTau_all$beta

# Coordinate: center of MNI
xyz <- c(45, 63, 36)

# Plot using ortho2
ortho2(x = MNI, y = wAvg_3D, xyz = xyz, ycolorbar = TRUE,
       NA.y = FALSE,
       ybreaks = c(seq(min(wAvg_3D, na.rm = TRUE),
                      max(wAvg_3D, na.rm = TRUE), length.out = 65)),
       text = "Whole brain empirical weighted averages",
       text.cex = 1)
```

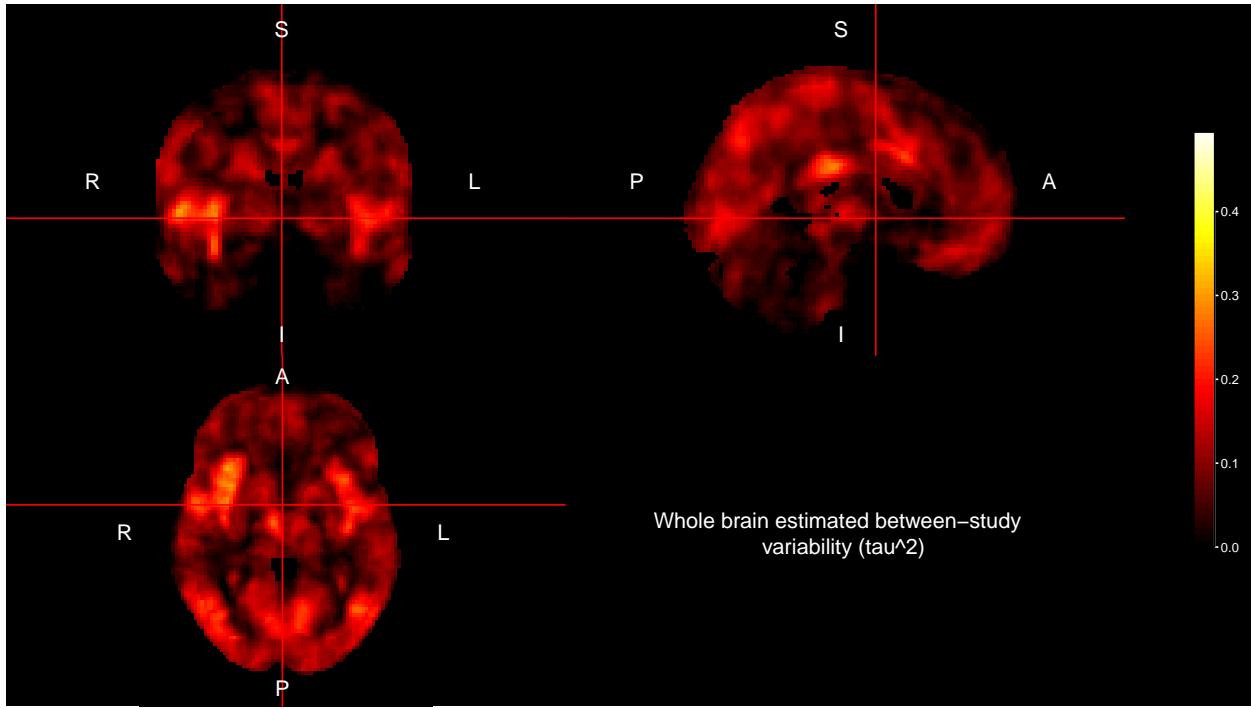


And now the same for between-study variability.

```
# Weighted averages back to 3D
tau_3D <- array(NA, dim = DIM3D)
tau_3D[MNImask == 1] <- estTau_all$tau2

# Coordinate: center of MNI
xyz <- c(45, 63, 36)

# Plot using ortho2
ortho2(x = MNI, y = tau_3D, xyz = xyz, ycolorbar = TRUE,
       NA.y = FALSE,
       ybreaks = c(seq(min(tau_3D, na.rm = TRUE),
                      max(tau_3D, na.rm = TRUE), length.out = 65)),
       text = "Whole brain estimated between-study \n variability (tau^2)",
       text.cex = 1)
```



#### 4.4 Comparing ROI with other brain areas

In the following, we plot the distributions of the 4 estimated parameters for the voxels inside the ROI versus those outside the ROI but still inside MNI mask.

```
# First we identify the voxels inside MNI space that are within mask
voxID_MNI <- data.frame(MNI = array(MNImask,
                                         dim = prod(dim(MNImask)))) %>% as.tibble() %>%
  mutate(voxID = 1:prod(dim(MNImask))) %>%
  filter(MNI == 1)

# Same, but for voxels inside ROI
voxID_ROI <- data.frame(ROI = ROI) %>% as.tibble() %>%
  mutate(voxID = 1:length(ROI)) %>%
  filter(ROI == 1)

# We add these voxel IDs to the estimated parameters from whole brain
# We start through a full join
ROIvsOther_param <- full_join(
  # We have data from the entire brain, gathered and info about voxID
  estTau_all %>% gather(key = 'Parameter', value = 'Value') %>%
    mutate(WB = 1,
      voxID = rep(voxID_MNI$voxID, 4))
  ,
  # Now we have data from the ROI, gathered and info about voxID
  estTau %>% gather(key = 'Parameter', value = 'Value') %>%
    mutate(ROI = 1,
      voxID = rep(voxID_ROI$voxID, 4))
  # Now we join by the parameter, its value and voxID
  # This is possible as the whole brain also contains all the ROI values.
```

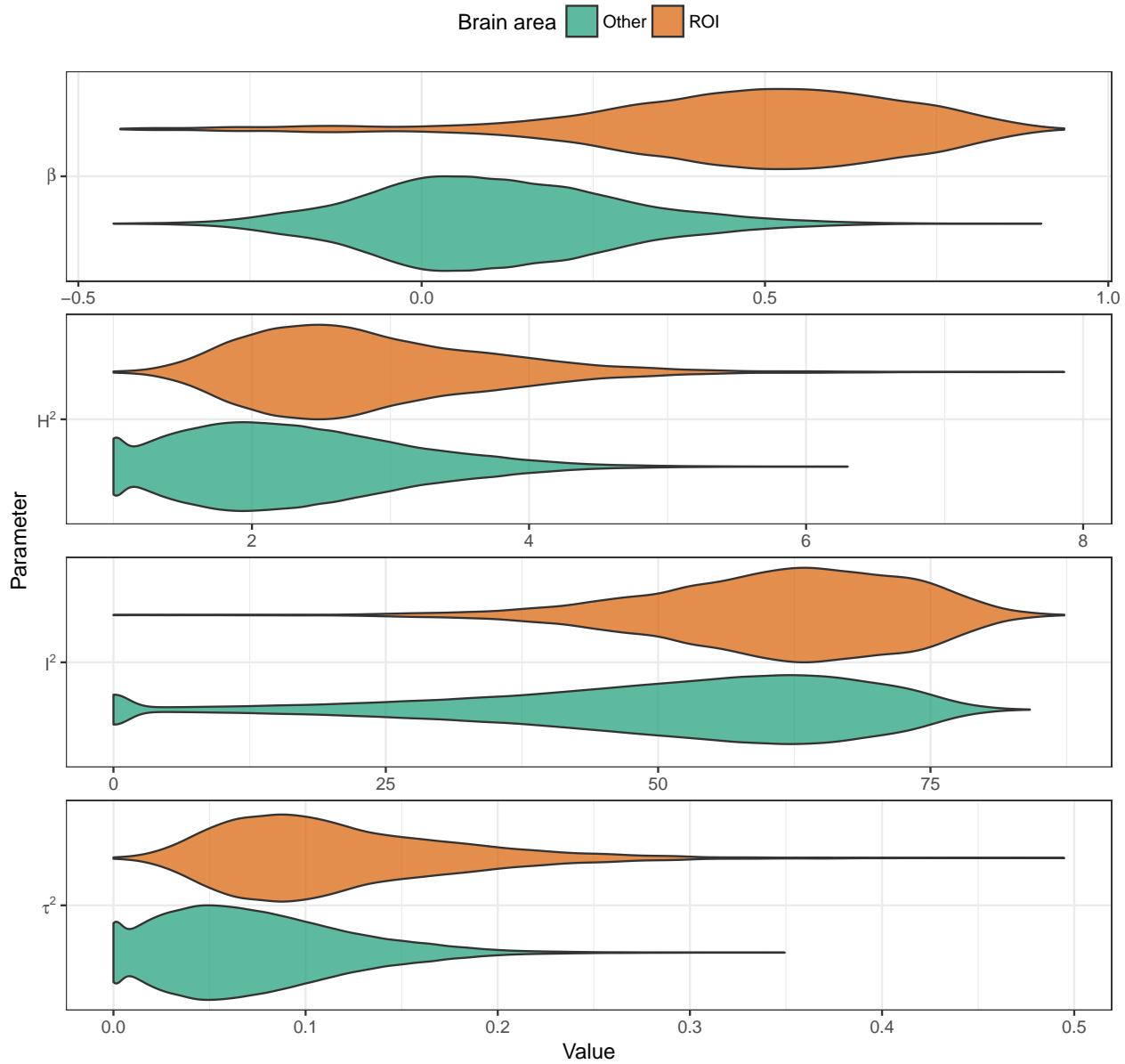
```

# So these are duplicated at the moment.
, by = c("Parameter", "Value", "voxID")) %>%
# We continue by creating a label. If ROI is missing, then this voxel is
#     outside ROI. Otherwise inside ROI.
mutate(Area = ifelse(is.na(ROI), 'Other', 'ROI'))

# Let us now plot these two distributions
ROIvsOther_param %>%
  ggplot(., aes(x = Parameter, y = Value)) +
  geom_violin(aes(fill = Area), alpha = 0.7) +
  scale_fill_brewer('Brain area', type = 'qual', palette = 2) +
  facet_wrap(~ Parameter, scales = 'free', ncol = 1) +
  coord_flip() +
  scale_x_discrete('Parameter', labels = c('beta' = expression(beta),
    'H2' = expression(H^2),
    'I2' = expression(I^2),
    'tau2' = expression(tau^2))) +
  ggtitle('Distributions of parameters within ROI versus other masked brain areas.') +
  theme_bw() +
  theme(strip.background = element_blank(),
    strip.text.x = element_blank(),
    legend.position = 'top')

```

Distributions of parameters within ROI versus other masked brain areas.



Note the following two main observations. First there are more voxels with higher effect sizes within the ROI compared to other brain areas (top row in previous figure). Second, we observe more or less the amount of between-study heterogeneity comparing both brain areas.

There is no real point in running a t-test to compare the means of both groups with estimated values for  $\tau^2$  as we have so many datapoints (there will be a significant difference). For completeness:

```
other_tau2 <- filter(ROIvsOther_param, Area == 'Other' & Parameter == 'tau2') %>%
  select(Value) %>% unlist(.) %>% as.numeric()
ROI_tau2 <- filter(ROIvsOther_param, Area == 'ROI' & Parameter == 'tau2') %>%
  select(Value) %>% unlist(.) %>% as.numeric()
t.test(x = ROI_tau2, y = other_tau2,
       alternative = 'two.sided', paired = FALSE)
```

Welch Two Sample t-test

```
data: ROI_tau2 and other_tau2
t = 95.149, df = 22051, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.04201429 0.04378169
sample estimates:
 mean of x  mean of y
0.11624022 0.07334223
```

## 5 Moderator Analysis

As stated in the introduction, we have used a variable to control for the fact that 21 studies come from the same laboratory. This would be the approach in a typical meta-analysis as one is interested in explaining between-study heterogeneity by adding covariates. This might reduce the standard errors associated with the estimated parameters of interest.

However in our analysis, we are not interested in reducing between-study heterogeneity. In fact, we would like to obtain upper bounds for between-study heterogeneity. For this reason, we might prefer to use the meta-analysis without any moderator.

Still under discussion.

## 6 Preliminary conclusion

In this report, we analyzed a database of 32 fMRI (whole-brain) studies involving the experience of *pain*. We were interested in estimating the empirical distribution of between-study heterogeneity, controlling for acquisition site. In order to obtain estimates within brain areas that are known to be involved in *pain* processing, we first generated a mask containing a Region of Interest (ROI). This was done using forward inference ( $P[\text{Activation}|\text{Term}]$ ) with NeuroSynth. Next we did a meta-analysis with a random-effects model and a moderator variable for site. We used the method of moments estimator for between-study variability and obtain the following conclusions:

- While the majority of voxels within the ROI show a fairly low amount of excess heterogeneity, between-study variability is not negligible.
- To see, consider that the top 5% of observed values for Hedges'  $g$  within the ROI are between [0.789; 0.936]. Then, the top 5% of observed values for  $\hat{\tau}^2$  are between [0.234; 0.495]. And the top 5% of estimated values for  $I^2$  are between [77.54; 87.283]. These values do reveal between-study heterogeneity which is not accounted for in the analysis.
- Furthermore, these higher values for between-study heterogeneity tend to co-occur *mainly* with higher estimates for the effect sizes.
- However, we do also observe high between-study variability for voxels with low effect sizes.
- A comparison of the brain regions outside the ROI with those defined in our ROI indeed reveals two elements. First we observe lower effect sizes for brain regions not included in the ROI. Second, we observe roughly the same distributions for the estimated between-study variability comparing these two brain areas (within versus outside ROI).