

Inversion Contrast Negation Data - Analysis

2022-12-05

Horizontal face information is the main gateway to the configuration and surface cues to familiar face identity. Comparative analysis of inversion and negation effects

Abstract

Humans preferentially rely on horizontal cues when recognizing face identity. The reasons for this preference are still largely elusive. Past research has proposed the existence of two main sources of face identity information: feature configuration and surface properties, the access to which is disrupted by picture-plane inversion and contrast negation, respectively.

Our objective was to characterize the configural versus surface nature of the face information conveyed by the horizontal range. To do this, we tracked the effects of inversion and negation in the orientation domain.

Participants performed an identity recognition task using orientation-filtered (from 0° to 150° in steps of 30°) pictures of familiar male actors presented upright, inverted, or negated. We modelled the inversion and negation effects across orientation with a Gaussian function using a Bayesian nonlinear mixed-effect modelling approach.

Despite the absence of a significant correlation between the effects of inversion and negation for full-spectrum images, they showed a strikingly similar orientation tuning profile. Indeed, both effects peaked in the horizontal range and displayed a comparable tuning strength. This confirms past evidence that the horizontal tuning of face identification is due to this range facilitating the access to feature configuration; our findings further suggest that it is also the main carrier of the surface cues to identity.

Our findings that the horizontal tuning is similarly disrupted by inversion and negation indicates that the horizontally oriented contrast in the face stimulus provides a privileged access to the important cues for identity recognition, namely the configuration of features and the surface properties.

Minimum number of trial per condition

We set the minimum number of trials per condition to 60 (some participants did not finish all the sessions)

```
## [1] "Number of subjects before cut-off: 40"  
## [1] "Number of subjects with at least 60 trials per condition: 21"
```

Floor effect / Chance level check up

Checking that the results are not at floor level. Checking that the results for each condition are above chance level. Here we set the cutoff is at >0.5 for each condition (performances are always higher than chance level).

```
## [1] "Number of subjects after selection (above chance level): 21"
```

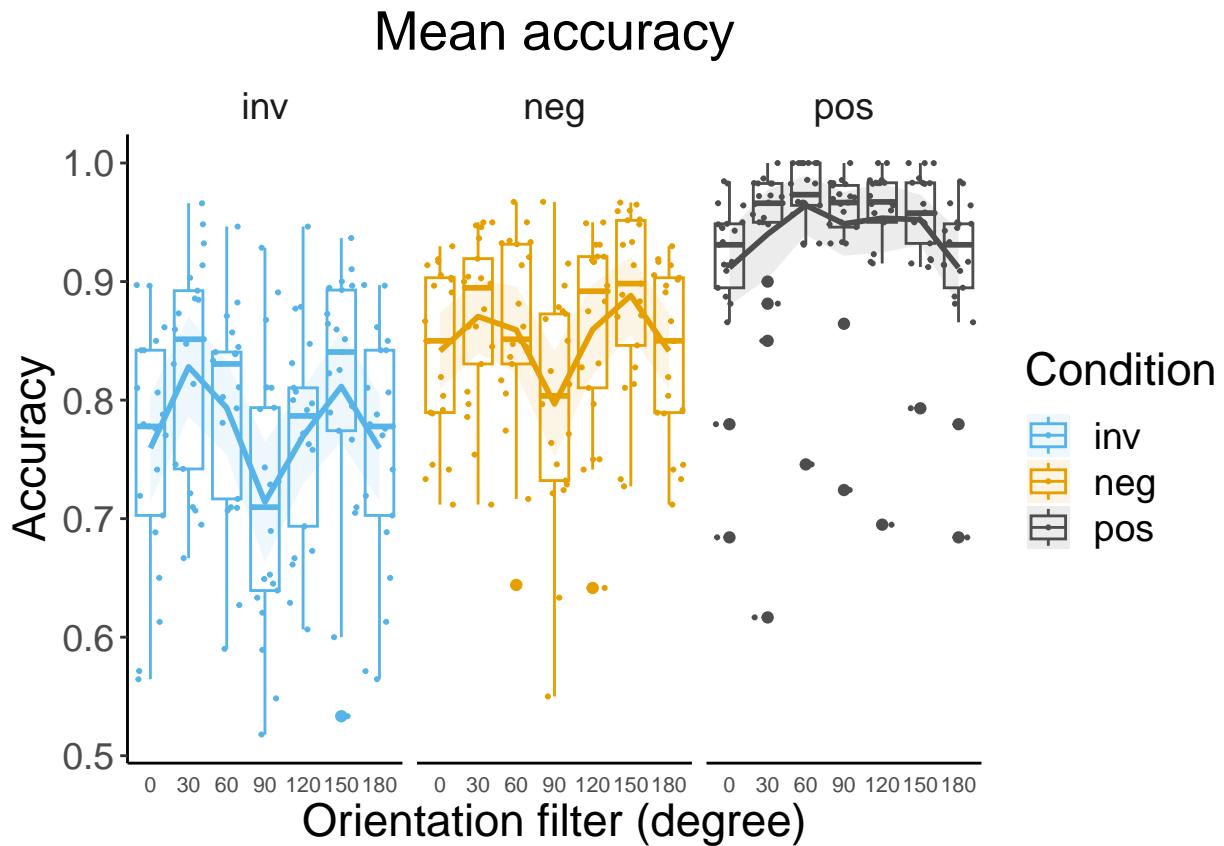
Response time

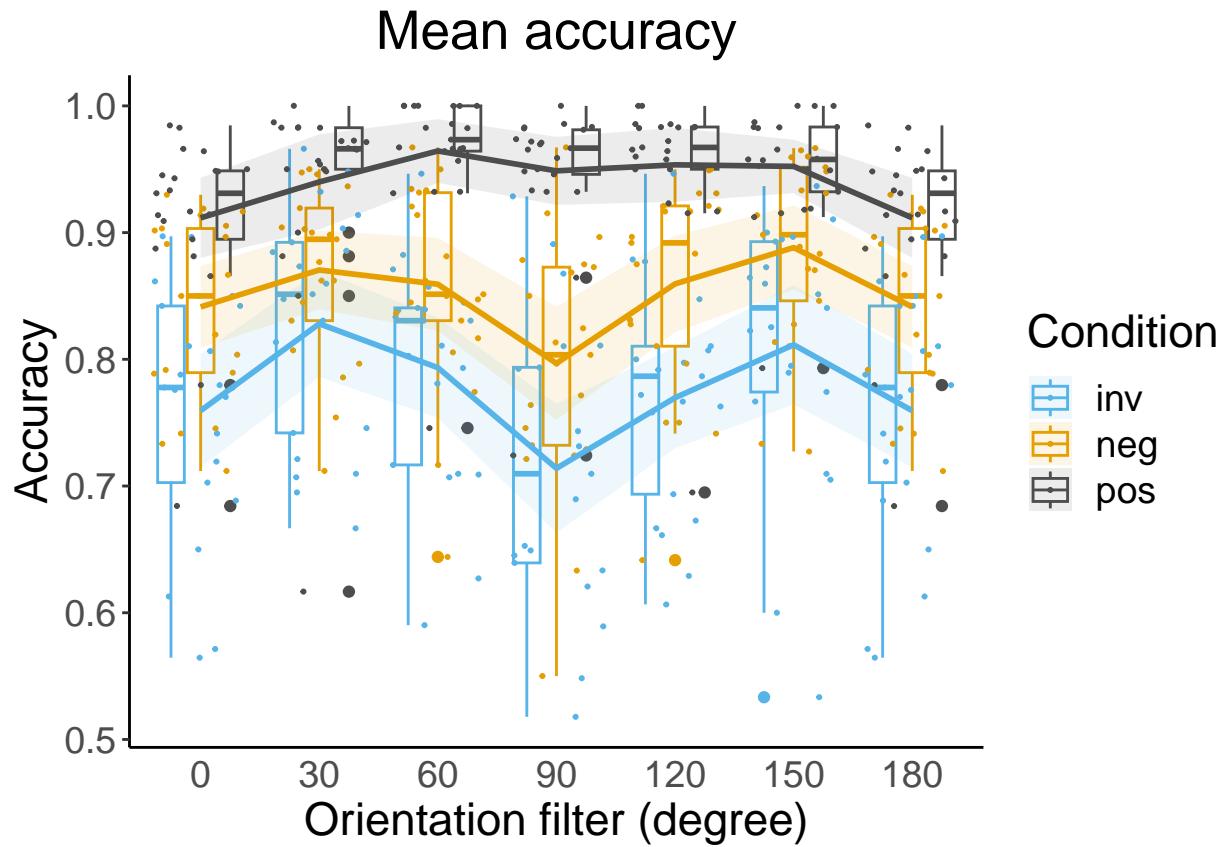
We use the response time to get rid of outlier responses. The response times are log10 transformed, the mean RT and sd is calculated for each participant. For each participants RT over or under the mean response time (mean \pm 2.5*sd) are considered outliers and taken out of the data.

```
## [1] "This results in a total deletion of : 3.26 % of the data"
```

Accuracy

Accuracy for each participant in each condition and filter





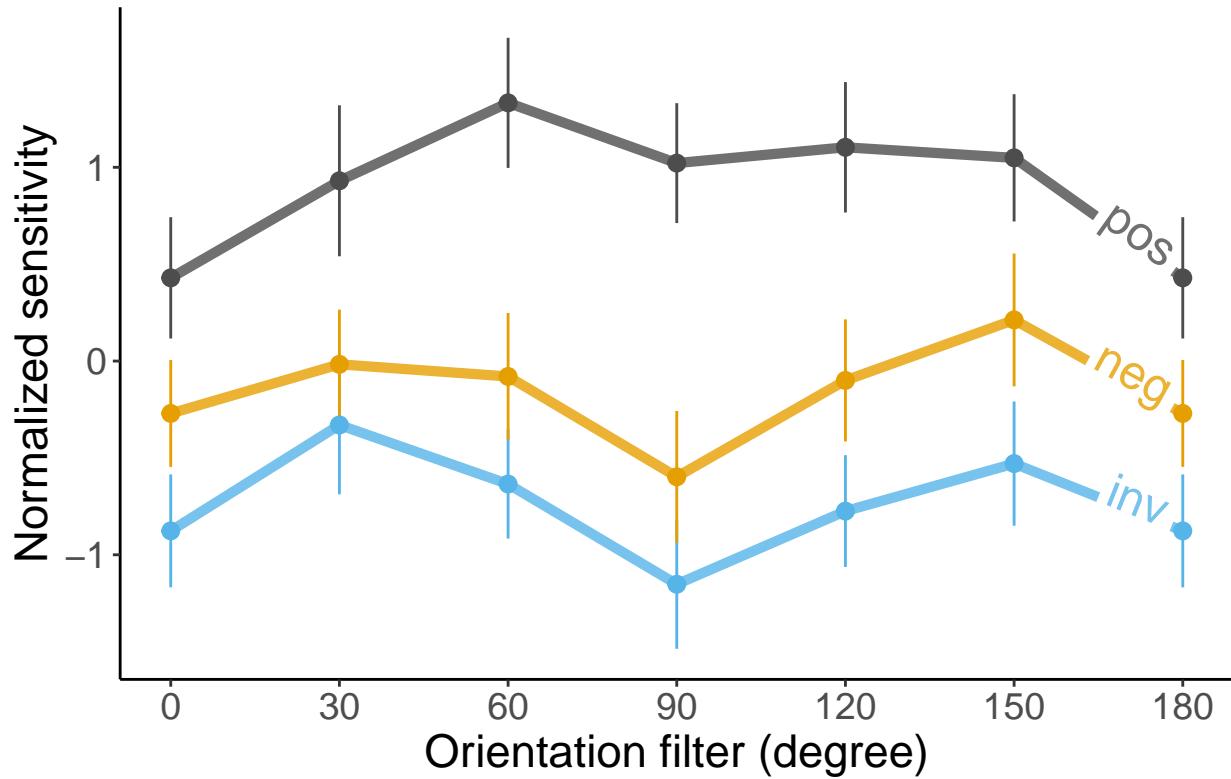
Calculation of the sensitivity (d')

The correction from Hautus 1995 of adding 0.5 to the count of Hits, False Alarm, Correct Rejection and Miss is applied here. The formula of the d' is the following : $d_{prime} = qnorm(Hit_Rate) - qnorm(FA_Rate)$

Normalized sensitivity

To normalize the sensitivity (d') I used the z transform following this formula $Z_{transformed} = (d' - \text{Mean}(d')) / \text{sd}(d')$. The mean and sd d_{prime} is calculated for each of the 3 conditions.

Normalized sensitivity at each filter

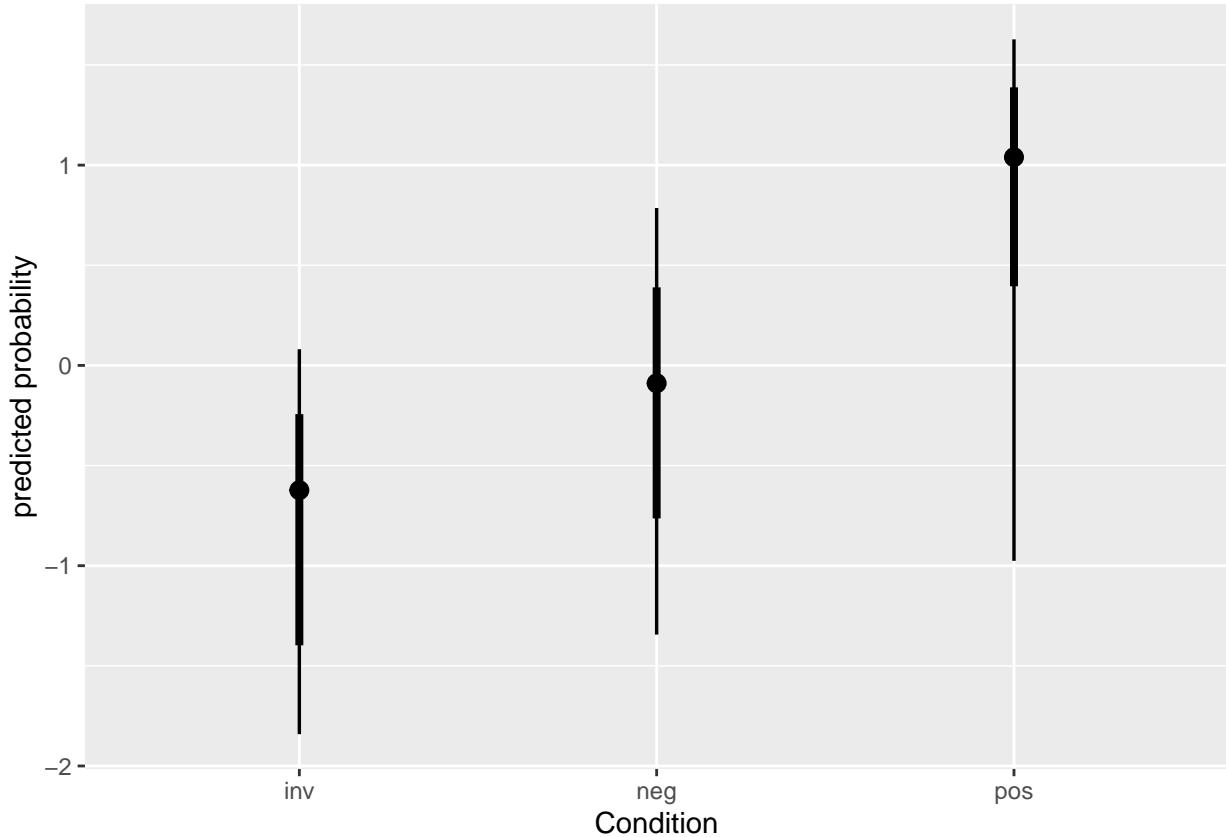


```
## pdf  
## 2
```

Comparison of the performance depending on the presentation condition - General Linear Mixed Model

To compare the recognition performances between the 3 presentation condition, we model our data (the *normd'*) using a GLMM.

```
## Family: student
## Links: mu = identity; sigma = identity; nu = identity
## Formula: Zscore ~ Condition + (Condition | Sujet)
## Data: Ztr (Number of observations: 441)
## Draws: 4 chains, each with iter = 6000; warmup = 2000; thin = 1;
##         total post-warmup draws = 16000
##
## Group-Level Effects:
## ~Sujet (Number of levels: 21)
##             Estimate Est.Error 1-95% CI u-95% CI Rhat
## sd(Intercept)      0.60     0.11    0.42    0.86 1.00
## sd(Conditionneg)   0.44     0.10    0.27    0.67 1.00
## sd(Conditionpos)   0.42     0.10    0.25    0.65 1.00
## cor(Intercept,Conditionneg) -0.22     0.24   -0.64    0.27 1.00
## cor(Intercept,Conditionpos) -0.13     0.25   -0.58    0.37 1.00
## cor(Conditionneg,Conditionpos)  0.36     0.25   -0.18    0.76 1.00
##                         Bulk_ESS Tail_ESS
## sd(Intercept)          5215     8039
## sd(Conditionneg)       5114     8996
## sd(Conditionpos)       5904     9739
## cor(Intercept,Conditionneg) 6852     9263
## cor(Intercept,Conditionpos) 8036    10371
## cor(Conditionneg,Conditionpos) 5827     8294
##
## Population-Level Effects:
##             Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept      -0.74     0.14   -1.02   -0.46 1.00      3338     5578
## Conditionneg    0.58     0.11    0.36    0.81 1.00      6471     8422
## Conditionpos    1.64     0.11    1.43    1.86 1.00      7706     9656
##
## Family Specific Parameters:
##             Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma        0.44     0.03    0.39    0.49 1.00     10720    11047
## nu          18.43    11.34    5.77   47.52 1.00     13496    12712
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```



```

## [1] "The highest Rhat values -- Should be under 1.01"

## r_Sujet[SAAY1407,Conditionpos]          b_Intercept
##                               1.000430      1.000436
## r_Sujet[BRMA1708,Intercept]           sd_Sujet__Conditionneg
##                               1.000525      1.000600
##                                         lp__        sd_Sujet__Intercept
##                               1.000827      1.001065

## [1] "The smallest Effective Sample Size -- should be over 100 * the number of chains"

##             lp__          b_Intercept
##           2752.664      3326.621
## sd_Sujet__Conditionneg    sd_Sujet__Intercept
##           5087.426      5116.219
## r_Sujet[ALRE2209,Intercept] r_Sujet[PHTA2206,Intercept]
##           5709.973      5752.447

## Family: student
## Links: mu = identity; sigma = identity; nu = identity
## Formula: Zscore ~ Condition + (Condition | Sujet)
## Data: Ztr (Number of observations: 441)
## Draws: 4 chains, each with iter = 6000; warmup = 2000; thin = 1;
##        total post-warmup draws = 16000
##

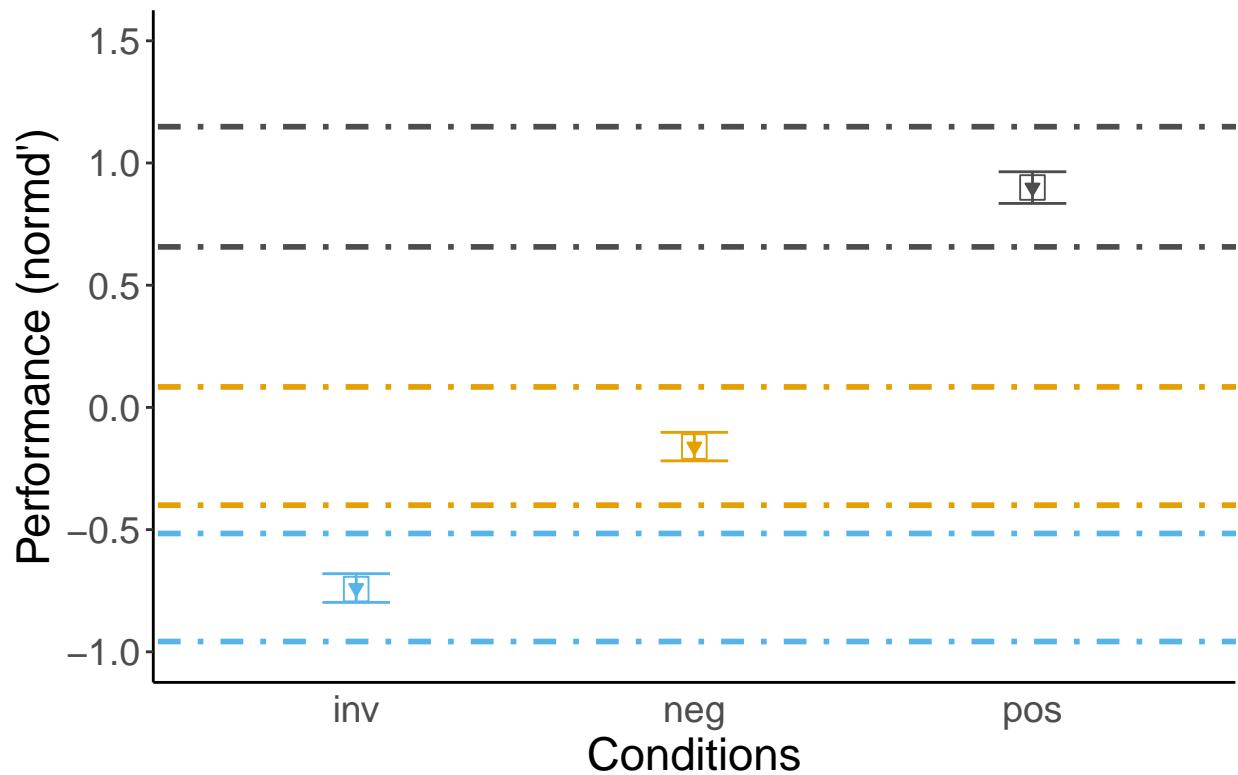
```

```

## Group-Level Effects:
## ~Sujet (Number of levels: 21)
##           Estimate Est.Error l-95% CI u-95% CI Rhat
## sd(Intercept)      0.60     0.11     0.42     0.86 1.00
## sd(Conditionneg)  0.44     0.10     0.27     0.67 1.00
## sd(Conditionpos)  0.42     0.10     0.25     0.65 1.00
## cor(Intercept,Conditionneg) -0.22    0.24    -0.64     0.27 1.00
## cor(Intercept,Conditionpos) -0.13    0.25    -0.58     0.37 1.00
## cor(Conditionneg,Conditionpos) 0.36    0.25    -0.18     0.76 1.00
##           Bulk_ESS Tail_ESS
## sd(Intercept)      5215     8039
## sd(Conditionneg)  5114     8996
## sd(Conditionpos)  5904     9739
## cor(Intercept,Conditionneg) 6852     9263
## cor(Intercept,Conditionpos) 8036    10371
## cor(Conditionneg,Conditionpos) 5827     8294
##
## Population-Level Effects:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept      -0.74     0.14    -1.02    -0.46 1.00      3338     5578
## Conditionneg   0.58     0.11     0.36     0.81 1.00      6471     8422
## Conditionpos   1.64     0.11     1.43     1.86 1.00      7706     9656
##
## Family Specific Parameters:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma        0.44     0.03     0.39     0.49 1.00     10720    11047
## nu          18.43    11.34    5.77    47.52 1.00     13496    12712
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).

```

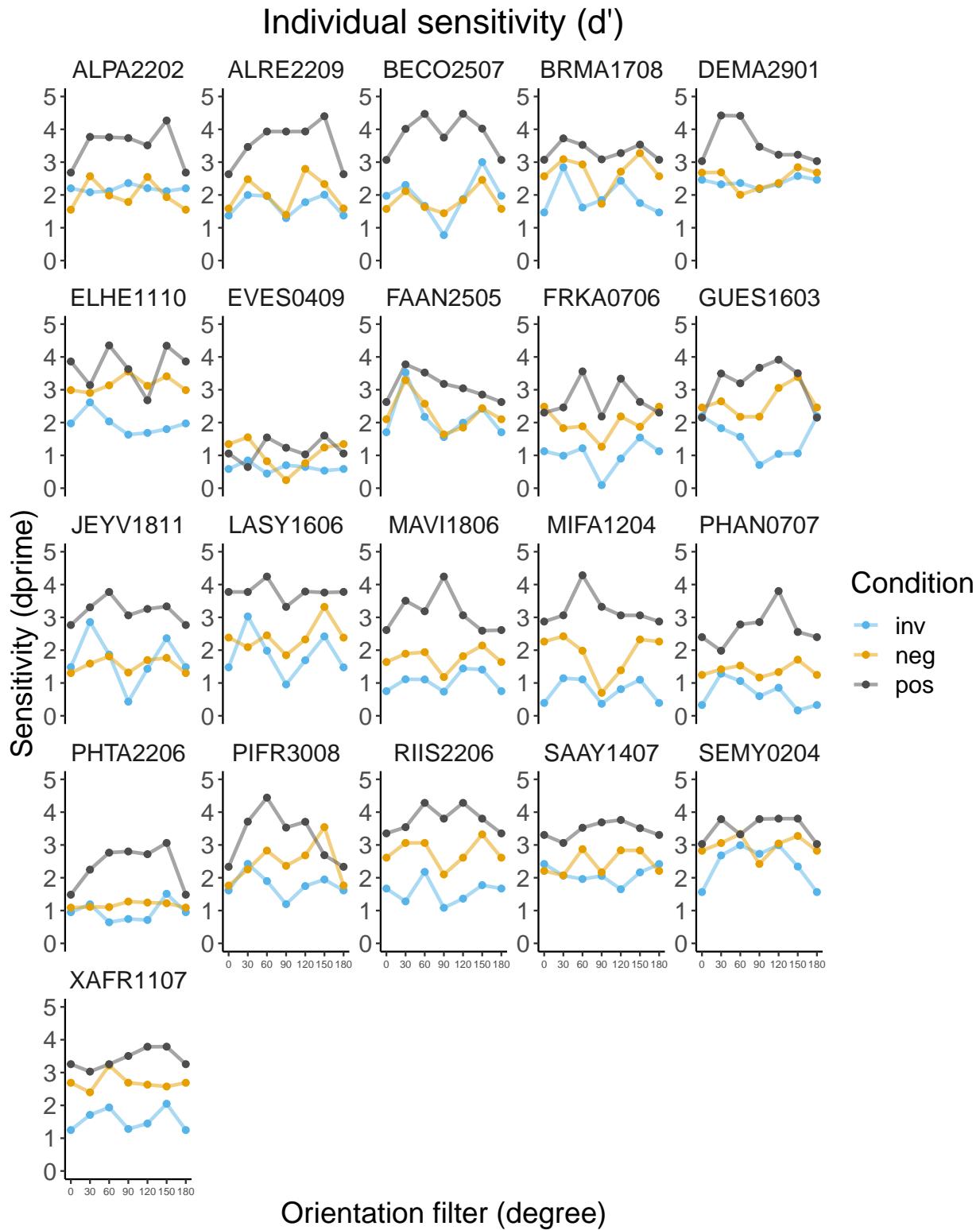
Mean and Crl predictions for each condition



```
## pdf  
## 2
```

Individual recognition performance of orientation-filtered images

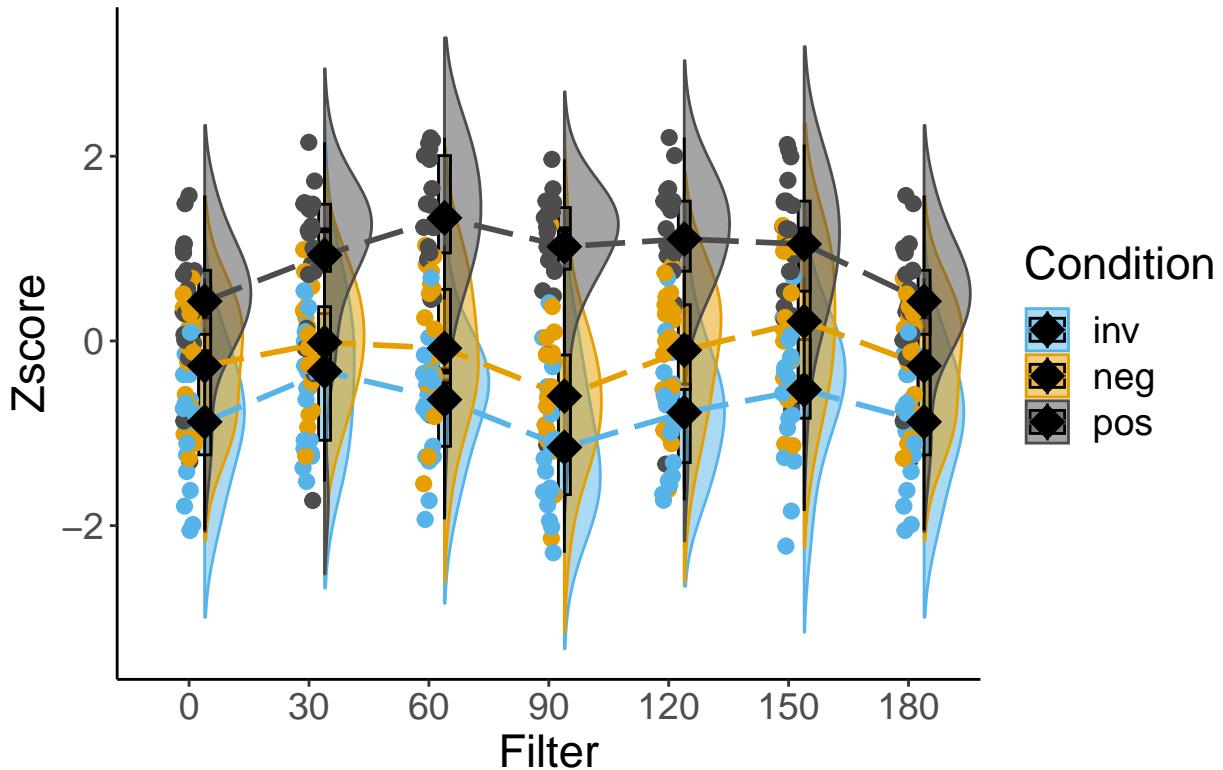
The $normd'$ for every subject



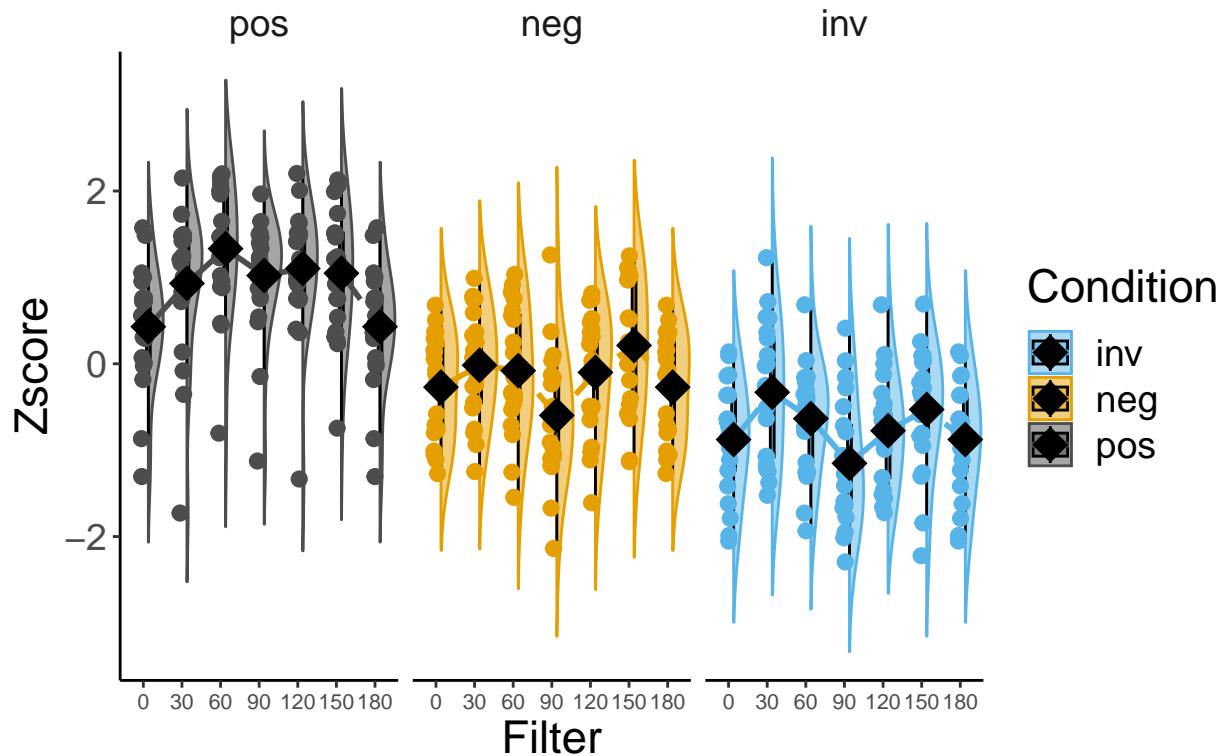
```
## pdf  
## 2
```

Population-level recognition performance of orientation-filtered images

Performance (nomrd') for each condition

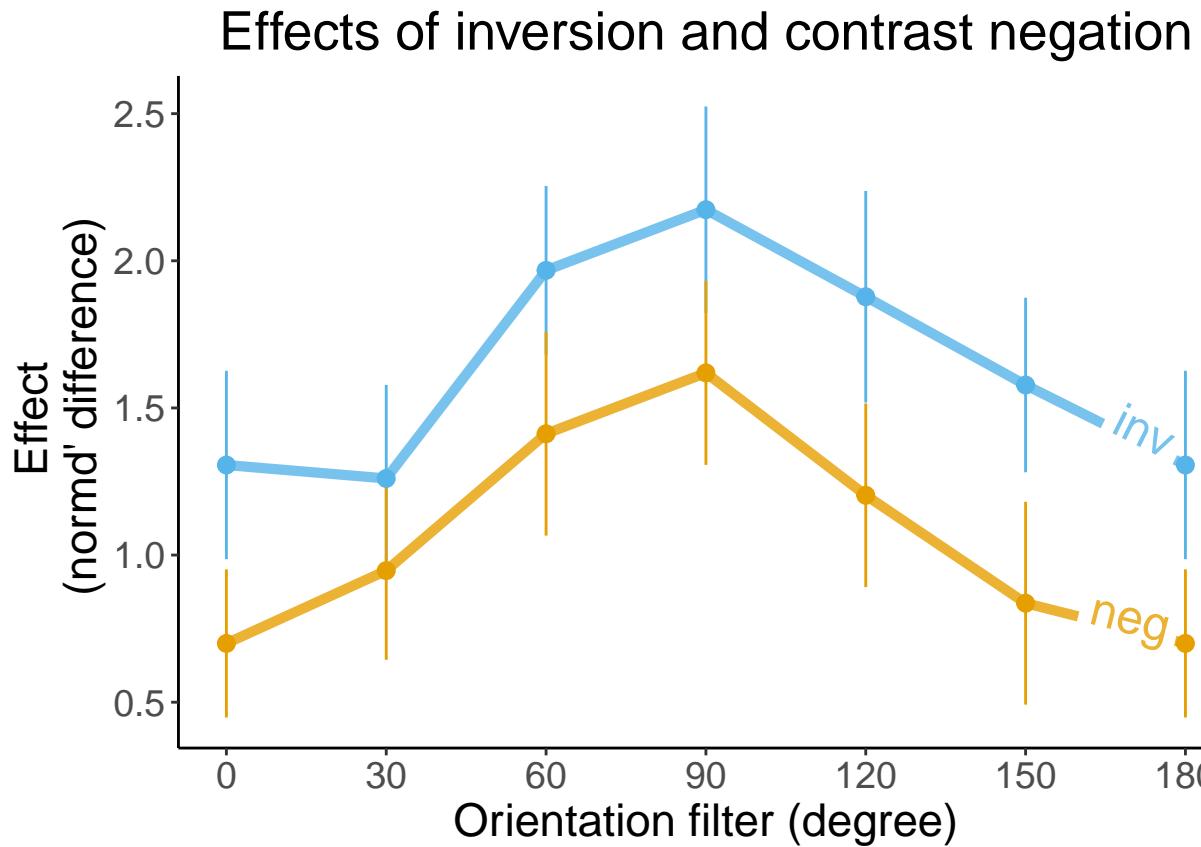


Performance (normd') for each condition



Effect of inversion and contrast negation

To calculate the effect and have it for each subject we decided to use differences : Effect of inversion = Up/Pos($normd'$) - inv($normd''$) Effect of contrast negation = Up/Pos($normd'$) - neg($normd'$)

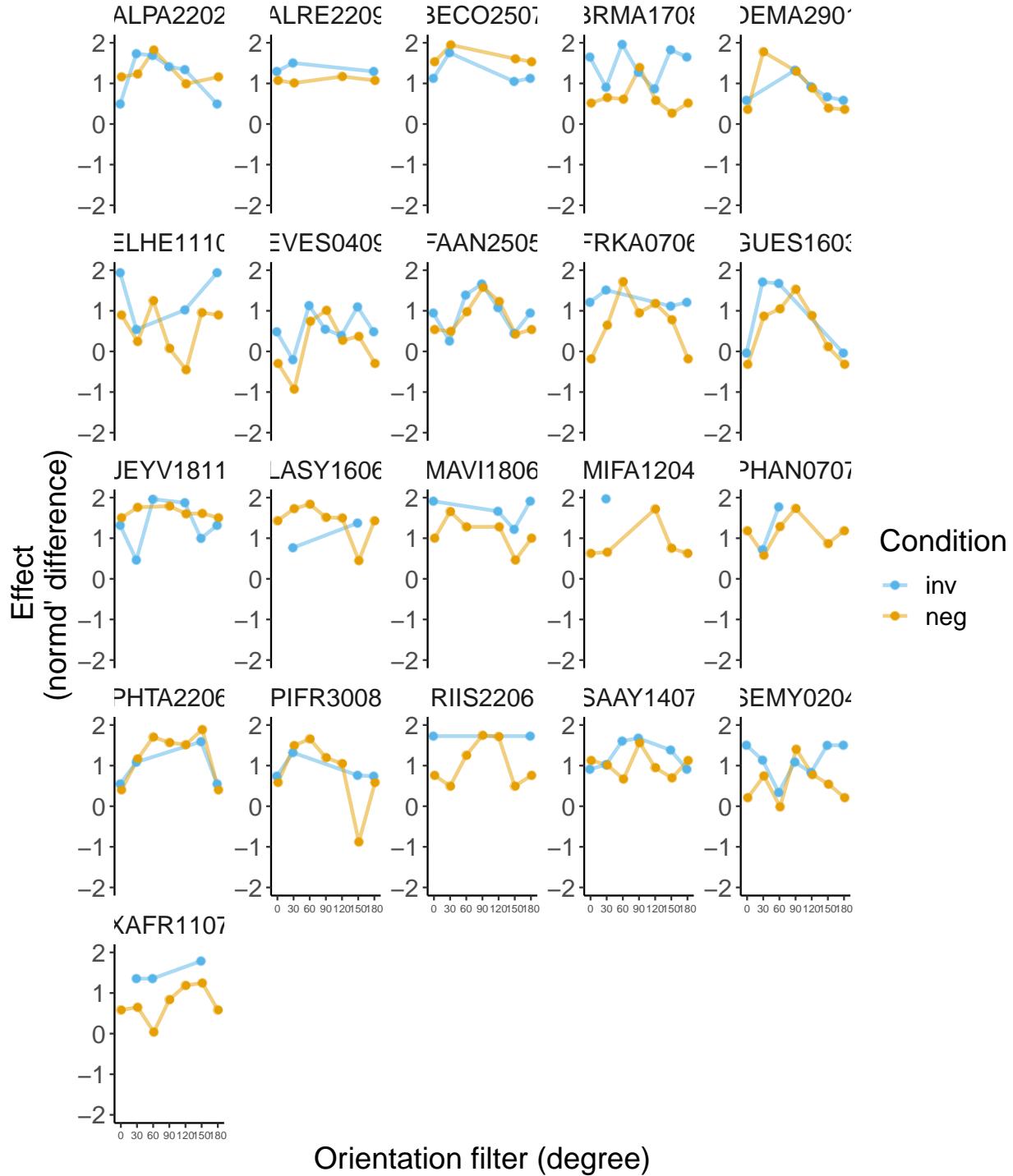


```
## pdf  
## 2
```

Effect of inversion and negation for each subject

The effects of inversion and contrast negation (calculated with the difference between the upright/positive condition and each of the two other conditions) is shown for each subject

Effect of inversion and contrast negation for each subject



Comparison of the effects of inversion and negation - Bayesian modeling

Seeing the effects of inversion and negation for each orientation filter, the data is shaped like a Gaussian. To compare precisely the effects of inversion and negation in terms on the recognition performance in term of the orientation-filter at which the effect is strongest, the selectivity of the effect over the orientation and the strength of the effect, we decided to model our data with a Gaussian Bayesian (non-linear) mixed model.

```

Diff <- Diff %>%
  rename(Subjects = Sujet,
        Orifilter = Filter,
        Effect = diff) %>%
  mutate(Orifilter = as.numeric(Orifilter))

formula <- brmsformula(
  #gaussian model y ~ (1/(StDev * sqrt(2*pi))) * exp(-(x - mu)^2 / (2 * StDev ^2))
  # with y : the difference and x: the orientation filter, mu:the mean centered on the peaklocation
  Effect ~ BaseAmpl + PeakAmpl * exp(-(Orifilter - PeakLoc)^2 / (2 * StDev^2)),

  #All 4 parameters : BaseAmpl , PeakAmpl, PeakLoc, StDev
  #VD ~ 1 + effet(s) fixe(s) + (XX / XX)
  # Peak location parameter (PeakLoc)
  PeakLoc ~ Condition + ( Condition | Subjects ),

  # Standard Deviation parameter (StDev)
  StDev ~ Condition + ( Condition | Subjects),

  # Base amplitude parameter (baseAmpl)
  BaseAmpl ~ Condition + ( Condition | Subjects),

  # Peak amplitude parameter (peakAmpl)
  PeakAmpl ~ Condition + ( Condition | Subjects),

  # non-linear model = TRUE
  nl = TRUE
)

# Priors
priors <- c(
  # peak location:
  prior(normal(90, 20), class = "b", nelpar = "PeakLoc", coef = "Intercept"), #90,40
  #for the intercept we can set an informative prior / or set it on the center
  # 0-180 the middle is 90
  prior(normal(0, 20), nelpar = "PeakLoc", class = "b"),
  # even if we expect a difference we set the difference to 0 +/- 5
  #to not influence the model
  prior(exponential(0.1), nelpar = "PeakLoc", class = "sd"),
  #we chose exp because it is always positive
)

```

```

# StDev:
prior(normal(35, 20), class = "b", nlpar = "StDev", coef = "Intercept"), #40,20
prior(normal(0, 30), nlpars = "StDev", class = "b"), #0,20/ 0,30
prior(exponential(0.1), nlpars = "StDev", class = "sd"),

# base amplitude:
prior(normal(1.5 , 1), class = "b", nlpars = "BaseAmpl", coef = "Intercept"), # -0.5, 1
prior(normal(0, 1.5), nlpars = "BaseAmpl", class = "b"), #0, 1
prior(exponential(0.05), nlpars = "BaseAmpl", class = "sd"),

# peak amplitude:
prior(normal(1.5, 1), class = "b", nlpars = "PeakAmpl", coef = "Intercept"), #1,2
prior(normal(0, 1.5), nlpars = "PeakAmpl", class = "b"), #0,1/#0,2
prior(exponential(0.1), nlpars = "PeakAmpl", class = "sd"),

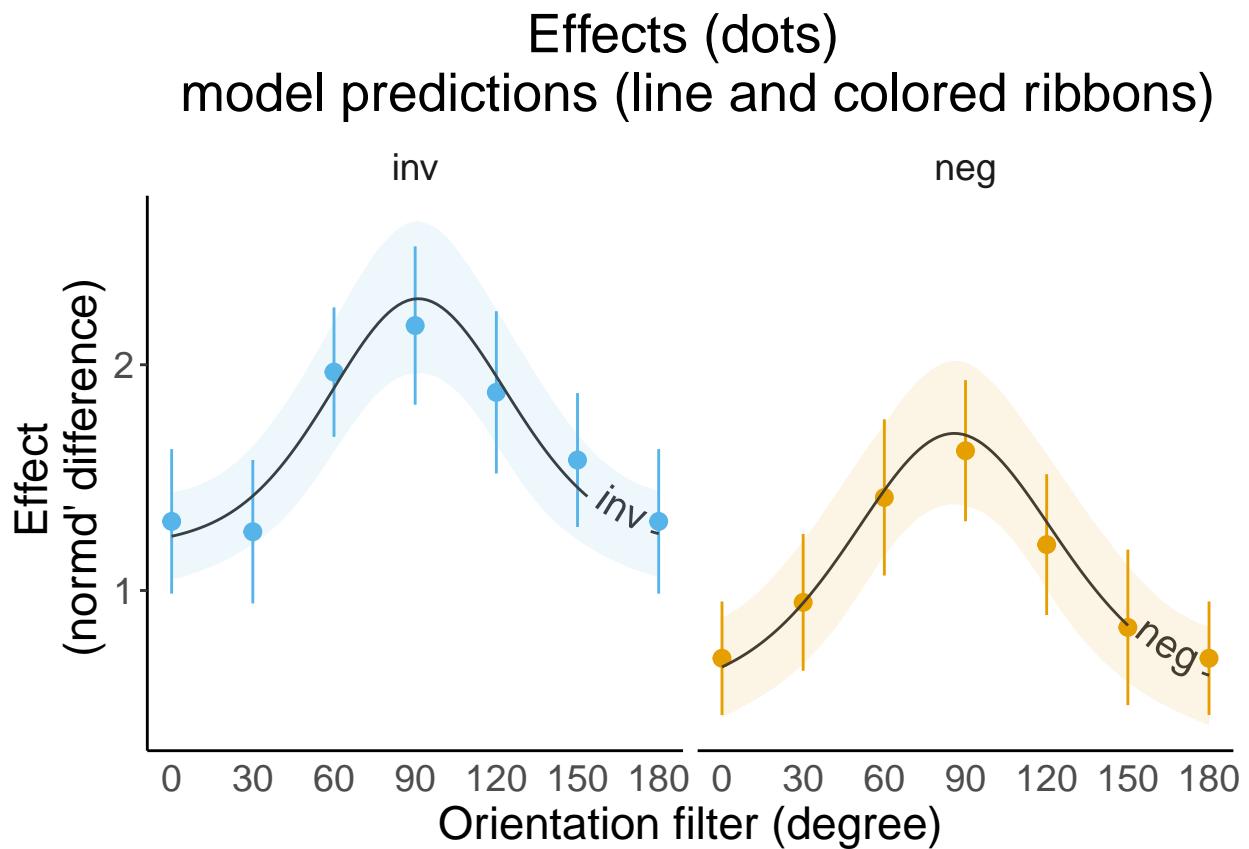
# sigma parameter of the Gaussian model
# (accounts for the variability of the residuals)
prior(exponential(0.1), class = "sigma")
)

# Fitting

fit <- brm(
  formula = formula,
  data = Diff,
  family = gaussian("identity"),
  prior = priors,
  warmup = 3000, #1000
  iter = 6000, #4000
  cores = parallel::detectCores(),
  chains = 4, #2
  control = list(adapt_delta = 0.9),
  file = "FinMod70.rds",
  seed = 986,
  sample_prior = "yes"  #permet de REGARDER
)

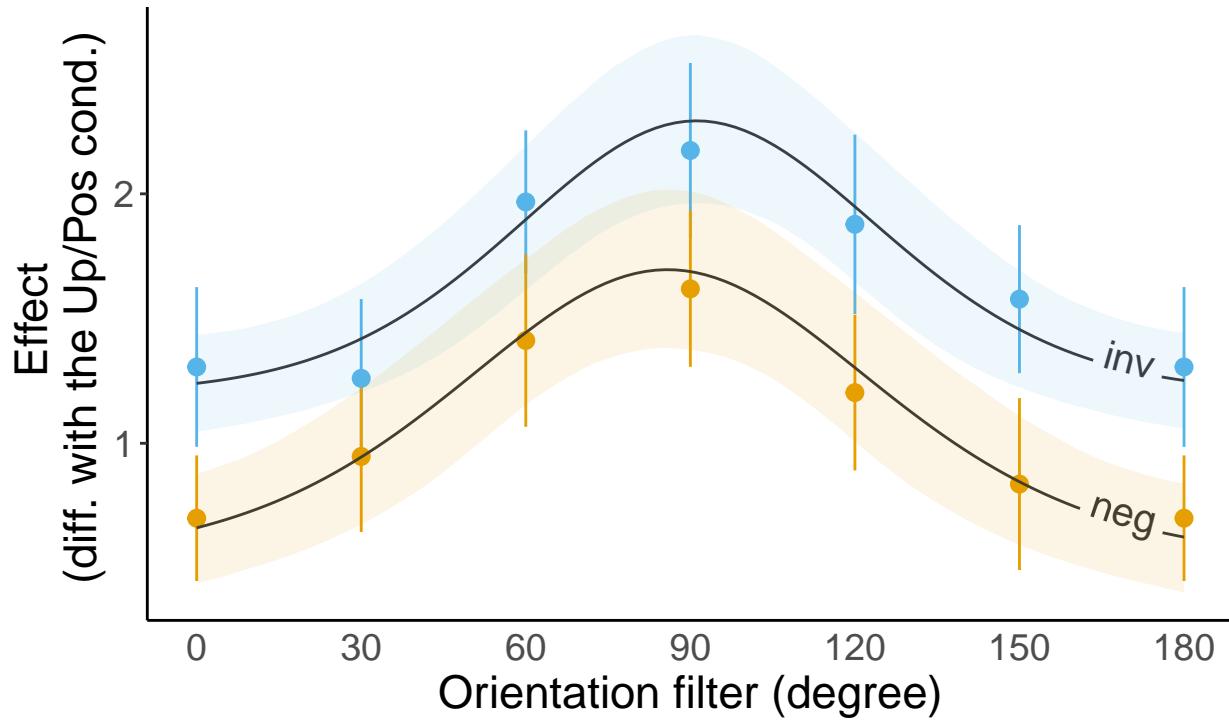
```

First Overview of the model

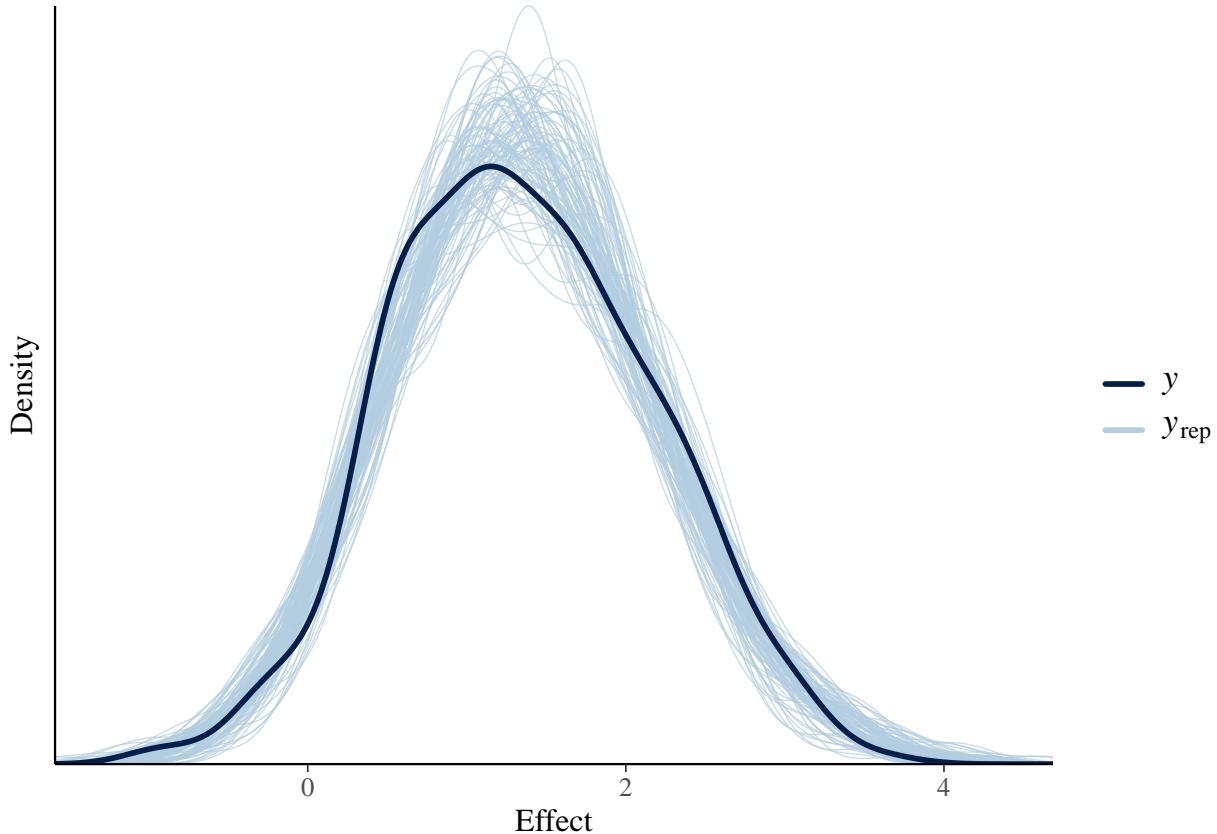


```
## pdf  
## 2
```

Effects (dots)
model predictions (line and colored ribbons)



```
## pdf  
## 2
```



Model summary and diagnostics

Checking to see if our model is good : A good model means that the chains (MCMC) converged. A small Rhat (smaller than 1.1) indicates small variance between each chain hence, a good convergence. A big effective sample size (ESS) The ESS corresponds to the number of independent samples with the same estimation power as the N autocorrelated samples. It is is a measure of “how much independent information there is in autocorrelated chains” (Kruschke 2015, p182-3).

```

## Family: gaussian
## Links: mu = identity; sigma = identity
## Formula: Effect ~ BaseAmpl + PeakAmpl * exp(-(Orifilter - PeakLoc)^2/(2 * StDev^2))
##           PeakLoc ~ Condition + (Condition | Subjects)
##           StDev ~ Condition + (Condition | Subjects)
##           BaseAmpl ~ Condition + (Condition | Subjects)
##           PeakAmpl ~ Condition + (Condition | Subjects)
## Data: Diff (Number of observations: 294)
## Draws: 4 chains, each with iter = 6000; warmup = 3000; thin = 1;
##        total post-warmup draws = 12000
##
## Group-Level Effects:
## ~Subjects (Number of levels: 21)
## sd(PeakLoc_Intercept)                         Estimate   Est.Error 1-95% CI
##                                         13.35      4.24    5.27
## sd(PeakLoc_Conditionneg)                      4.23      3.47    0.16
## sd(StDev_Intercept)                           7.65      6.27    0.25

```

```

## sd(StDev_Conditionneg)           10.65    8.91    0.27
## sd(BaseAmpl_Intercept)          0.45     0.10    0.27
## sd(BaseAmpl_Conditionneg)       0.36     0.14    0.06
## sd(PeakAmpl_Intercept)          0.58     0.18    0.24
## sd(PeakAmpl_Conditionneg)       0.23     0.17    0.01
## cor(PeakLoc_Intercept,PeakLoc_Conditionneg) -0.09    0.56   -0.95
## cor(StDev_Intercept,StDev_Conditionneg)  0.03     0.57   -0.94
## cor(BaseAmpl_Intercept,BaseAmpl_Conditionneg) -0.44    0.30   -0.91
## cor(PeakAmpl_Intercept,PeakAmpl_Conditionneg) -0.28    0.52   -0.97
##
##                                         u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(PeakLoc_Intercept)            22.43  1.00    2989   2781
## sd(PeakLoc_Conditionneg)        12.96  1.00    5544   5911
## sd(StDev_Intercept)             22.63  1.02    563    1028
## sd(StDev_Conditionneg)          31.92  1.00    813    1090
## sd(BaseAmpl_Intercept)          0.68   1.00    2951   4866
## sd(BaseAmpl_Conditionneg)       0.63   1.00    1187   1211
## sd(PeakAmpl_Intercept)          0.96   1.00    2089   1597
## sd(PeakAmpl_Conditionneg)       0.63   1.00    3065   4507
## cor(PeakLoc_Intercept,PeakLoc_Conditionneg) 0.93   1.00    8691   7116
## cor(StDev_Intercept,StDev_Conditionneg)  0.95   1.00    2421   4219
## cor(BaseAmpl_Intercept,BaseAmpl_Conditionneg) 0.26   1.00    3651   3054
## cor(PeakAmpl_Intercept,PeakAmpl_Conditionneg) 0.87   1.00    5607   6426
##
## Population-Level Effects:
##                                         Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS
## PeakLoc_Intercept      91.52      4.74   82.29  101.08 1.00    3927
## PeakLoc_Conditionneg  -5.53      4.78  -15.22   3.62 1.00    5793
## StDev_Intercept        34.06      6.19   21.80  46.92 1.00    1486
## StDev_Conditionneg     5.34      9.12  -11.32  25.15 1.00    3091
## BaseAmpl_Intercept     1.20      0.14    0.91   1.45 1.00    3855
## BaseAmpl_Conditionneg -0.67      0.18  -1.04  -0.33 1.00    3011
## PeakAmpl_Intercept     1.11      0.20    0.73   1.51 1.01    1679
## PeakAmpl_Conditionneg  0.07      0.21  -0.33   0.49 1.00    2633
##
##                                         Tail_ESS
## PeakLoc_Intercept      5430
## PeakLoc_Conditionneg  6626
## StDev_Intercept        1574
## StDev_Conditionneg     4144
## BaseAmpl_Intercept     5182
## BaseAmpl_Conditionneg  5138
## PeakAmpl_Intercept     2707
## PeakAmpl_Conditionneg  4990
##
## Family Specific Parameters:
##                                         Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma      0.48      0.03     0.43    0.53 1.00    2917    5683
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).

## [1] "The highest Rhat values -- Should be under 1.01"

## r_Subjects__StDev[BEC02507,Intercept] r_Subjects__StDev[ELHE1110,Intercept]

```

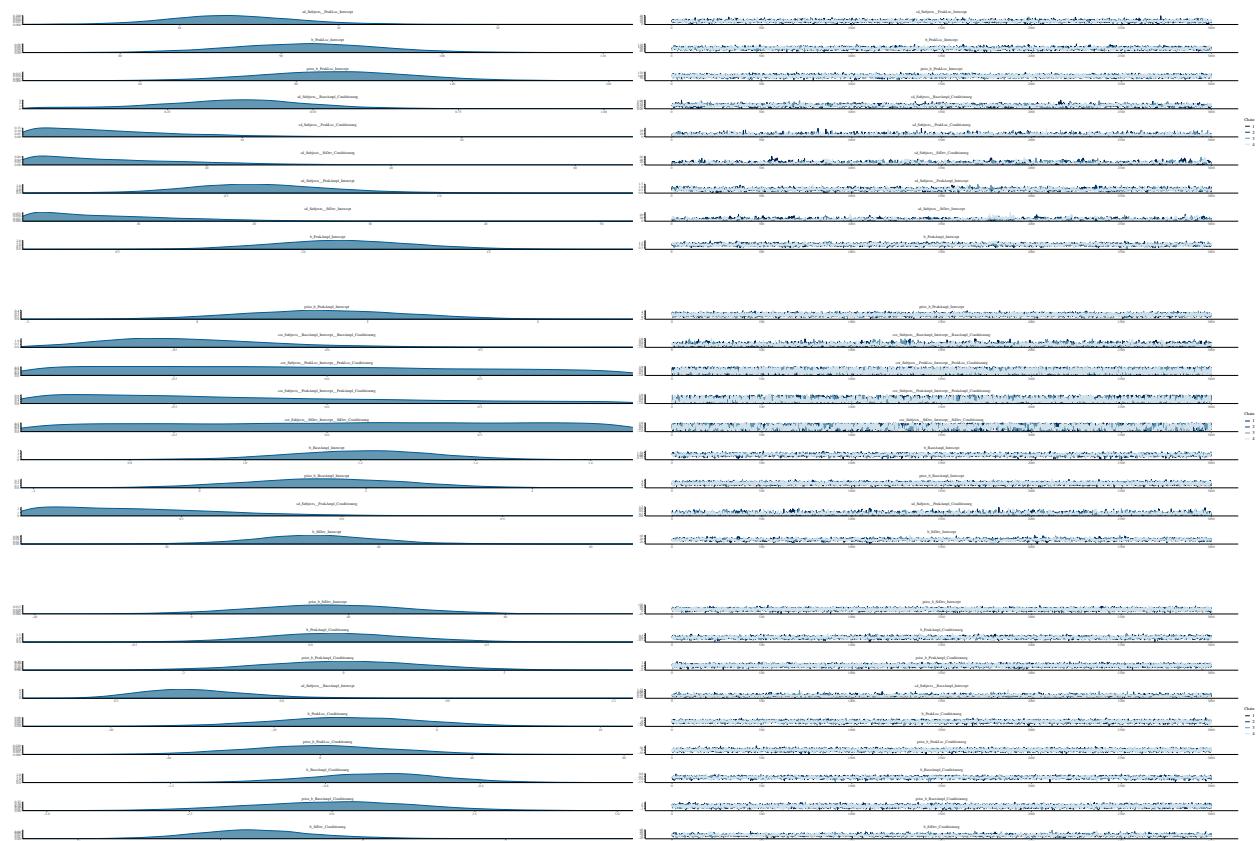
```

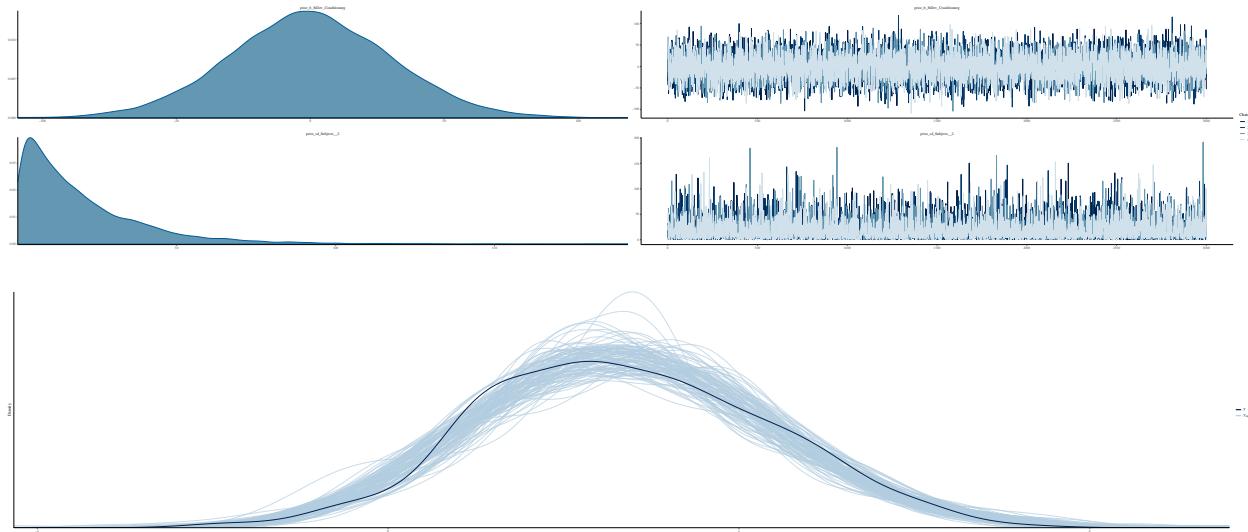
##                               1.007869                               1.008574
## r_Subjects__StDev[PHTA2206,Intercept] r_Subjects__StDev[SEMY0204,Intercept]
##                               1.008947                               1.009774
## r_Subjects__StDev[MAVI1806,Intercept]      sd_Subjects__StDev_Intercept
##                               1.016524                               1.018659

## [1] "The smallest Effective Sample Size -- should be over 100 * the number of chains"

##          sd_Subjects__StDev_Intercept
##                               451.4870
##          r_Subjects__StDev[MAVI1806,Intercept]
##                               555.3371
##          sd_Subjects__StDev_Conditionneg
##                               673.2604
##          r_Subjects__StDev[MAVI1806,Conditionneg]
##                               677.3807
##          lprior
##                               779.7634
##          r_Subjects__StDev[FRKA0706,Intercept]
##                               813.9308

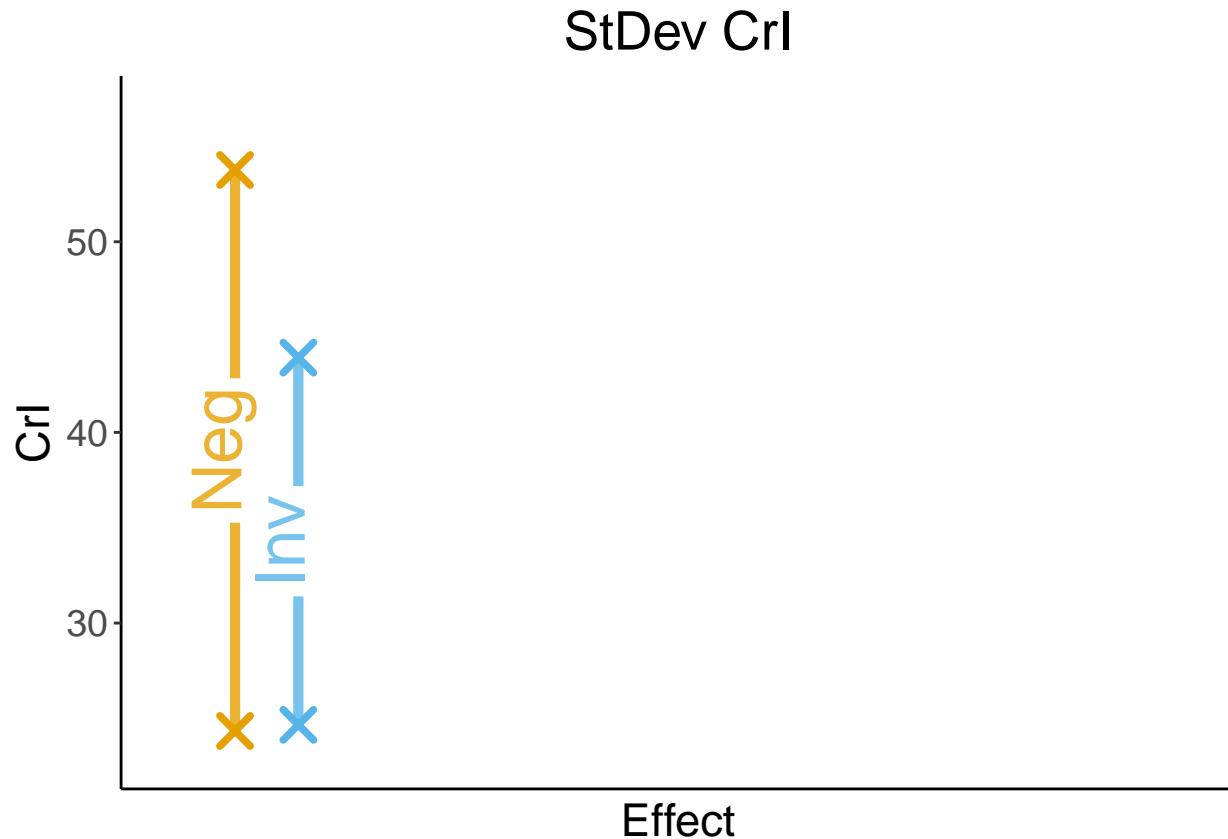
```





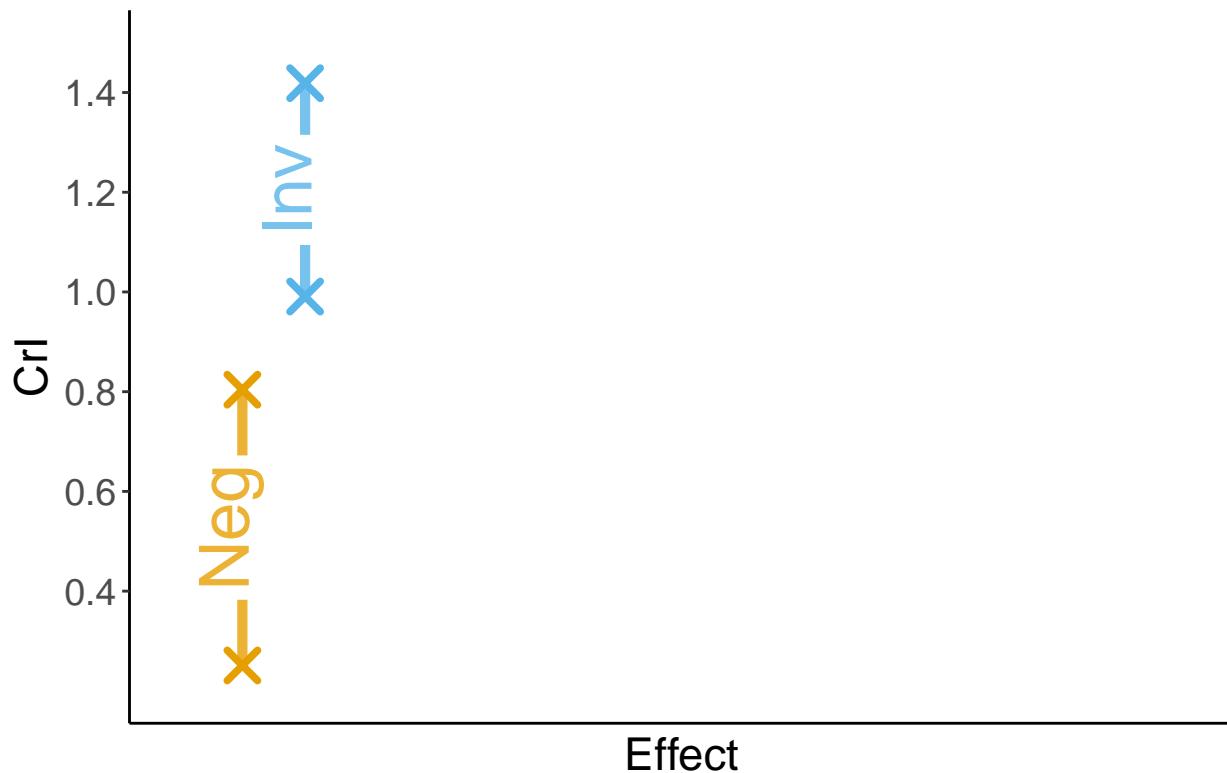
Credible Intervals

If the credible intervals of two conditions do not overlap, then the difference between the two conditions is considered significant.



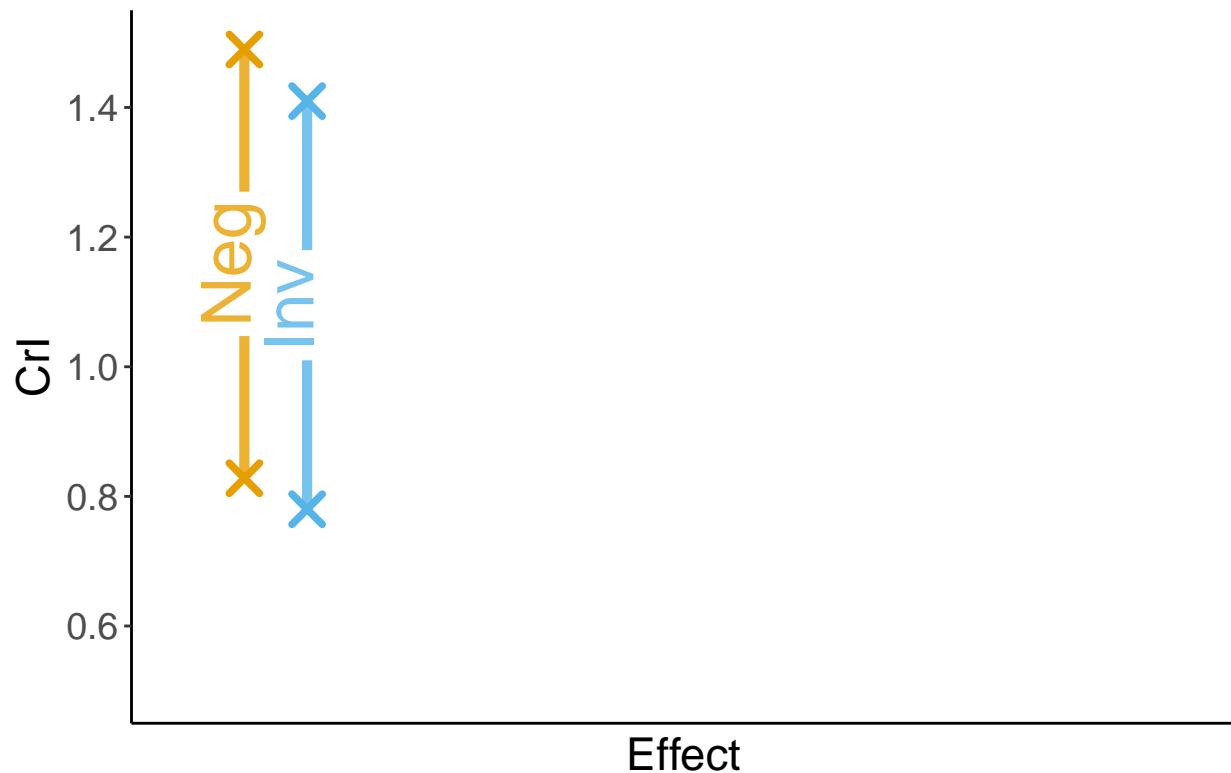
```
## pdf
## 2
```

BaseAmp Crl



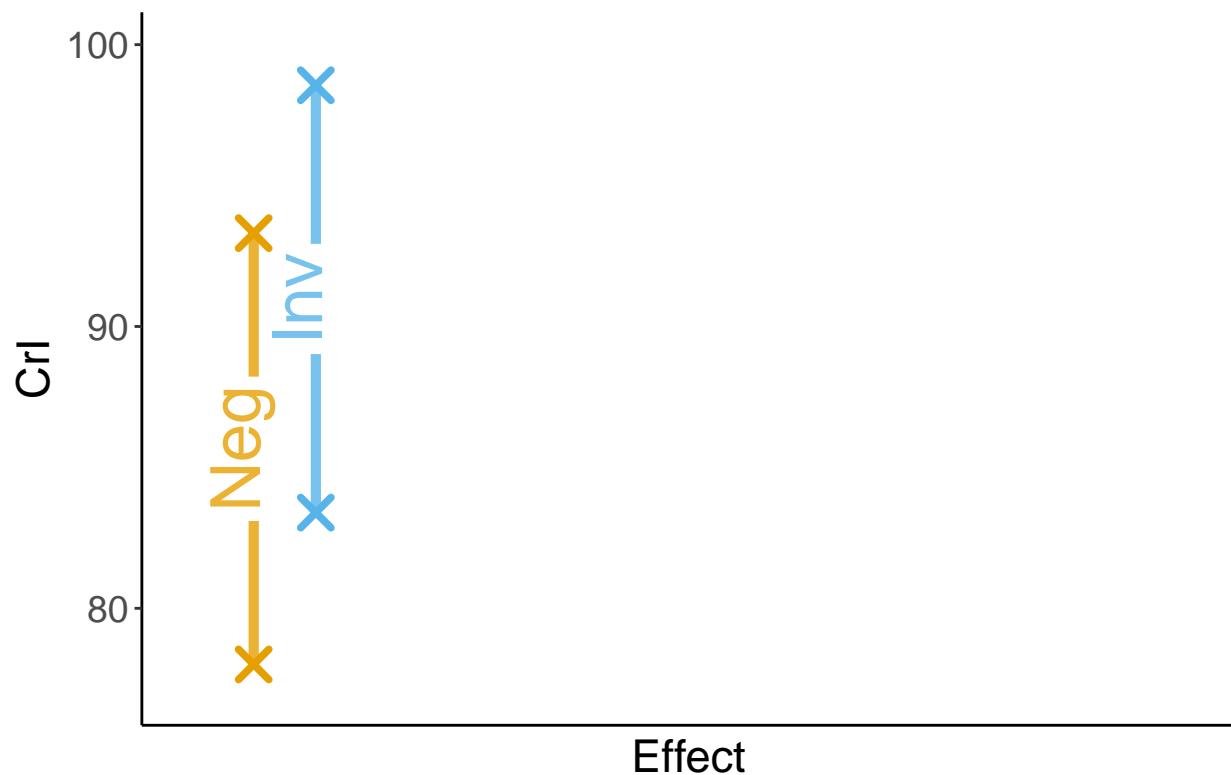
```
## pdf  
## 2
```

PeakAmp CrI



```
## pdf  
## 2
```

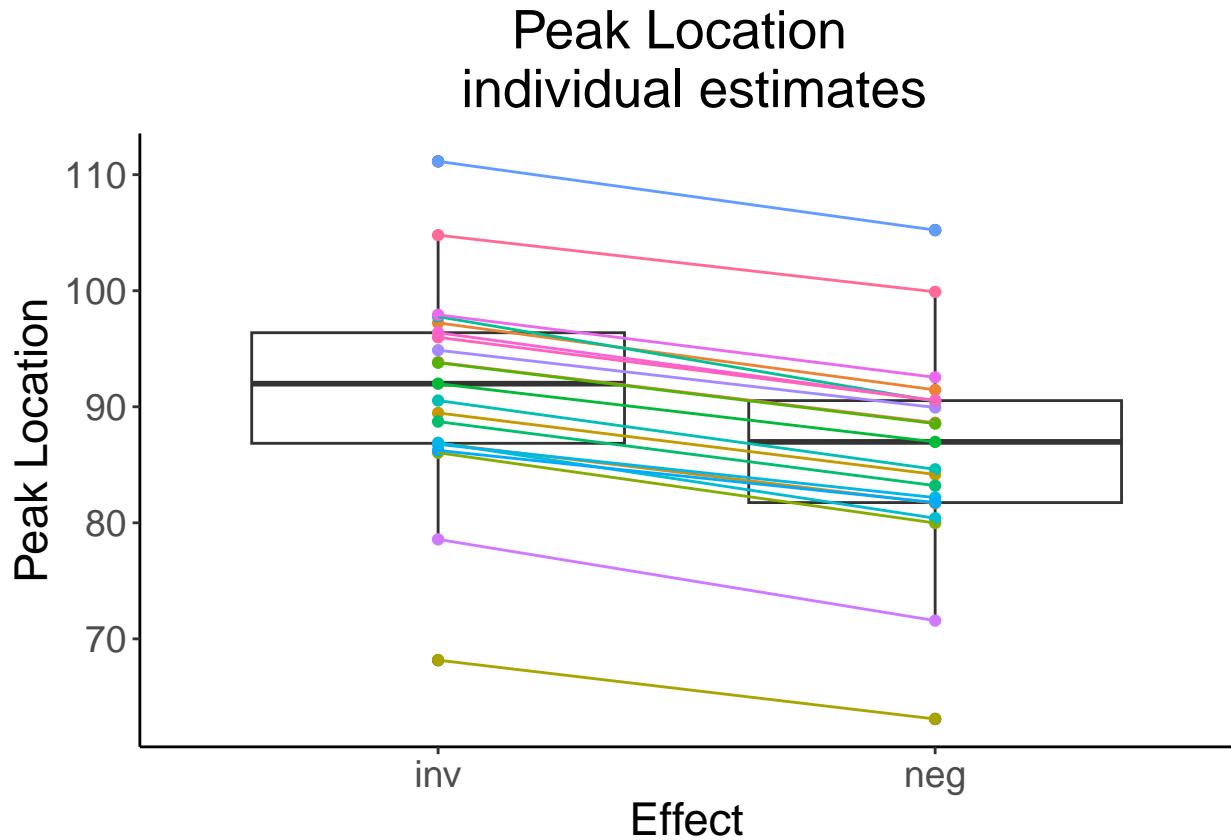
PeakLoc Crl



```
## pdf  
## 2
```

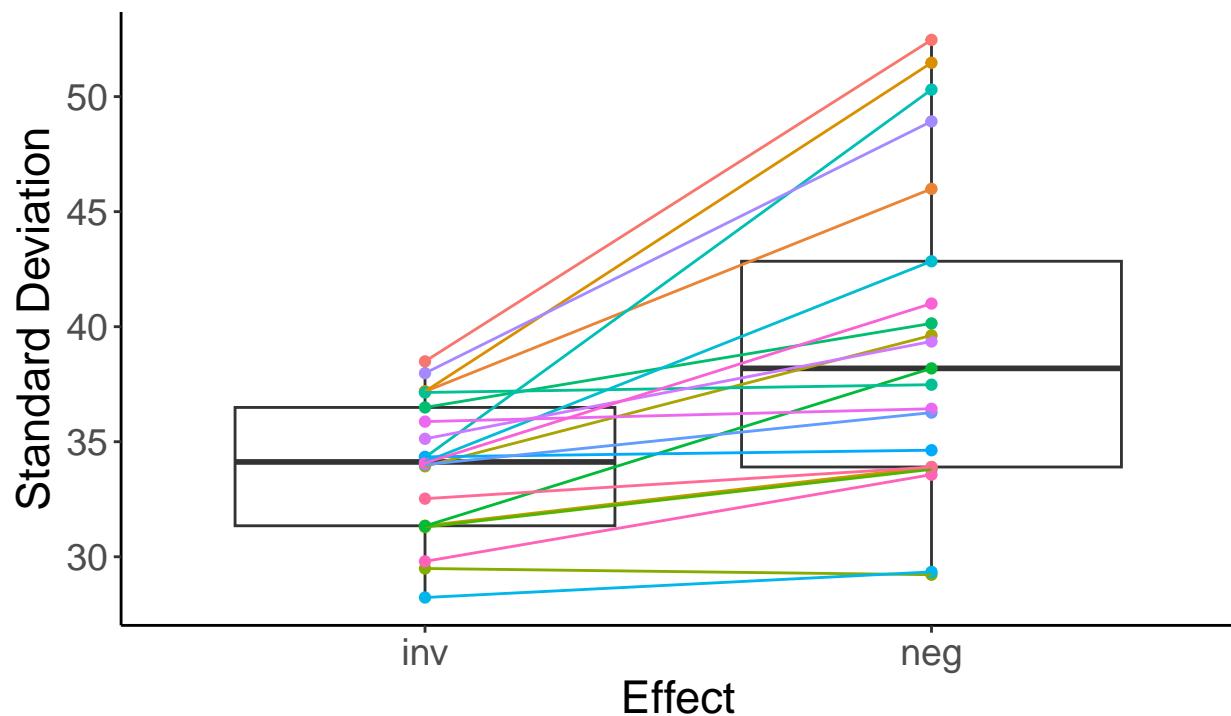
Individual Parameters estimation

For each of the 4 Gaussian parameters we take the individual level model estimates. We use it to the the difference of the parameters estimates for each subjects between the inversion and the contrast negation effect.



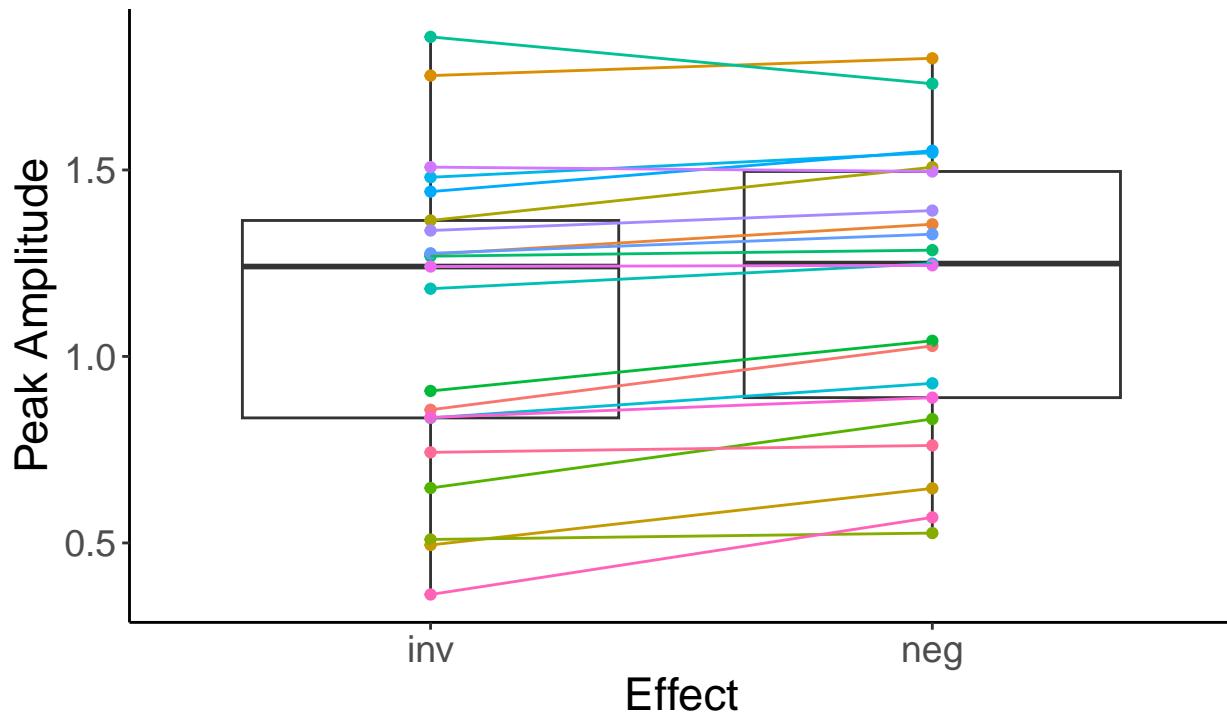
```
## pdf  
## 2
```

Standart Deviation individual estimates



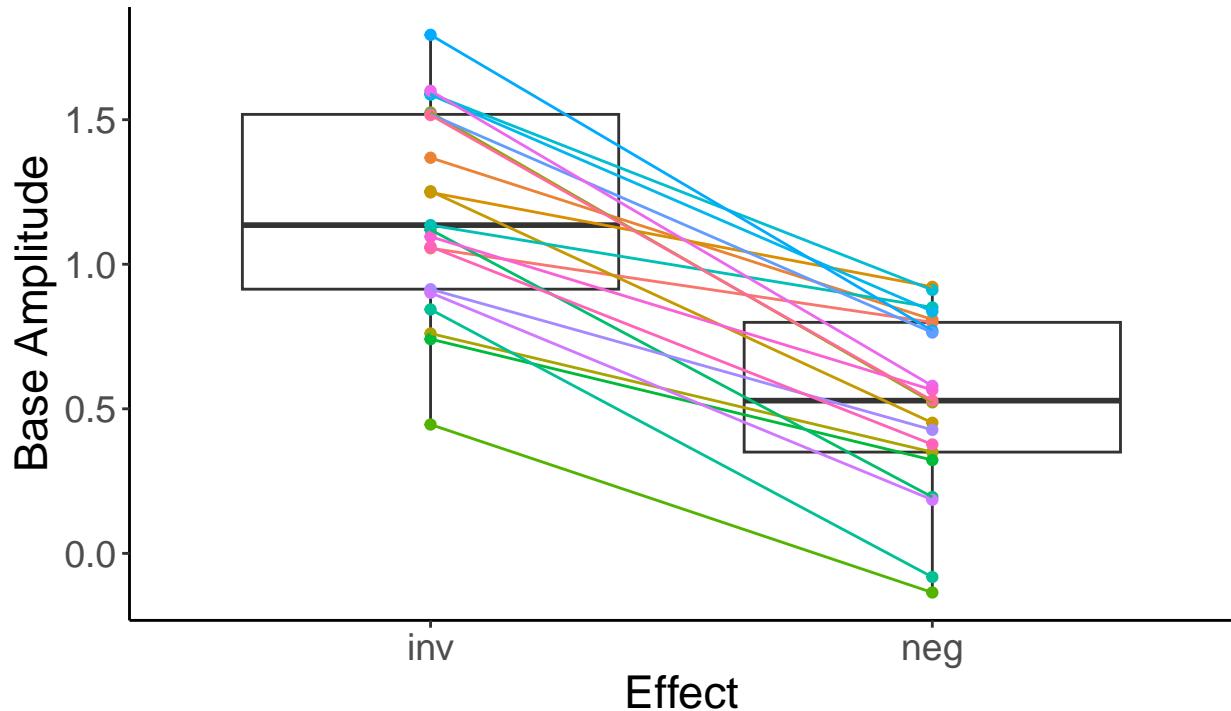
```
## pdf  
## 2
```

Peak Amplitude individual estimates



```
## pdf  
## 2
```

Base Amplitude individual estimates



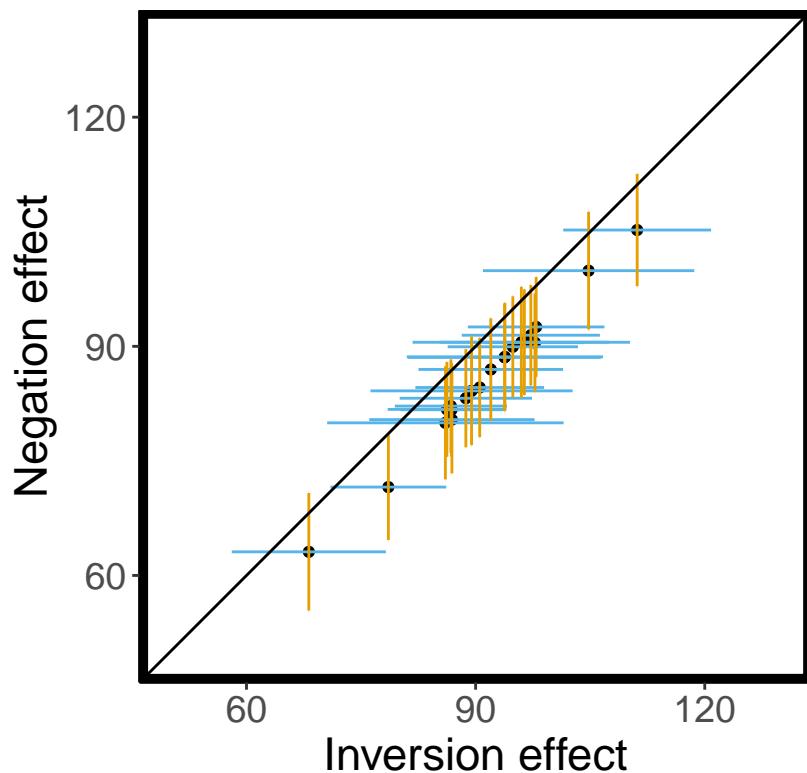
```
## pdf
## 2
```

Comparison of inversion and contrast negation effect based on individual parameters estimates

Here we represent each individual parameter estimate as a dot on a graph with the inversion effect in abscisse and the contrast negation effect in y.

The comparison is also done through a correlation between the parameter estimates of the inversion effect and the contrast negation effect.

Comparison of estimates for Peak Location

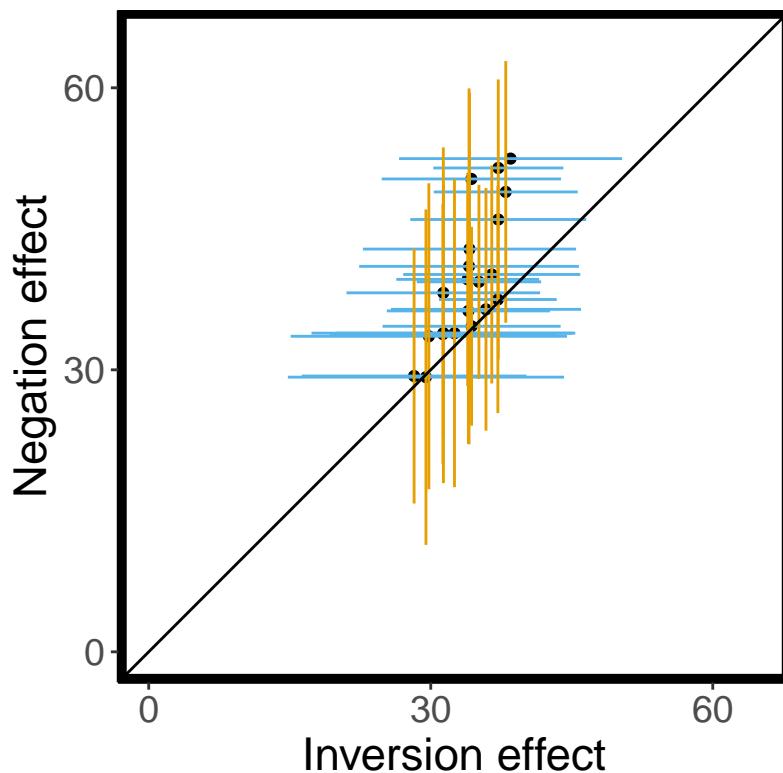


```
## pdf
## 2

## [1] "The correlation coefficient for peak location is : "

##
## Pearson's product-moment correlation
##
## data: peakLoc1$inv and peakLoc1$neg
## t = 53.419, df = 19, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9916760 0.9986838
## sample estimates:
##        cor
## 0.9966874
```

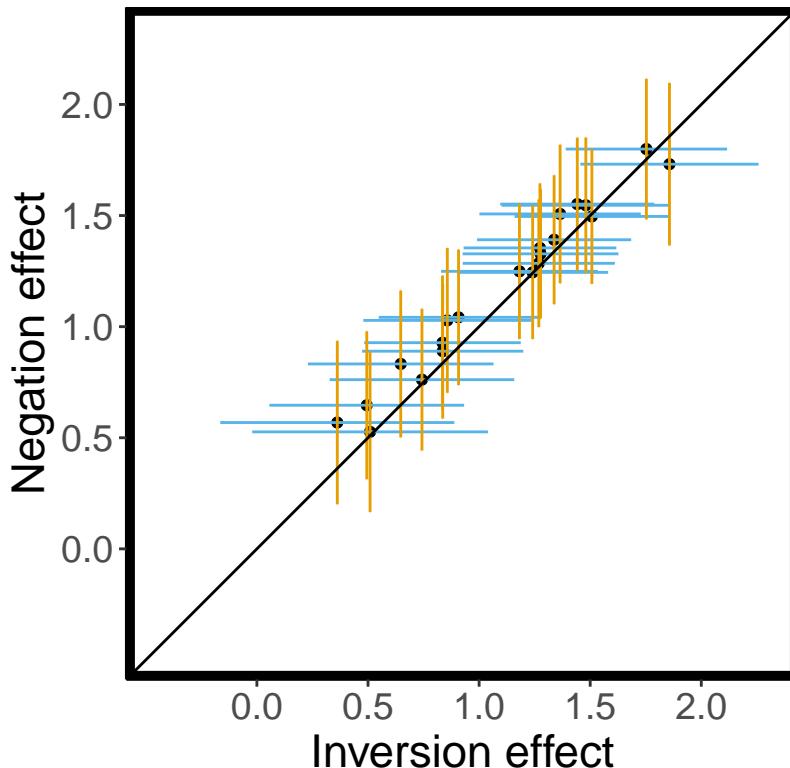
Comparison estimates for Standard Deviation



```
## pdf
## 2

## [1] "The correlation coefficient for standard deviation is : "
## 
## Pearson's product-moment correlation
## 
## data: stDev1$inv and stDev1$neg
## t = 5.4601, df = 19, p-value = 2.875e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5279325 0.9071536
## sample estimates:
##        cor
## 0.7815116
```

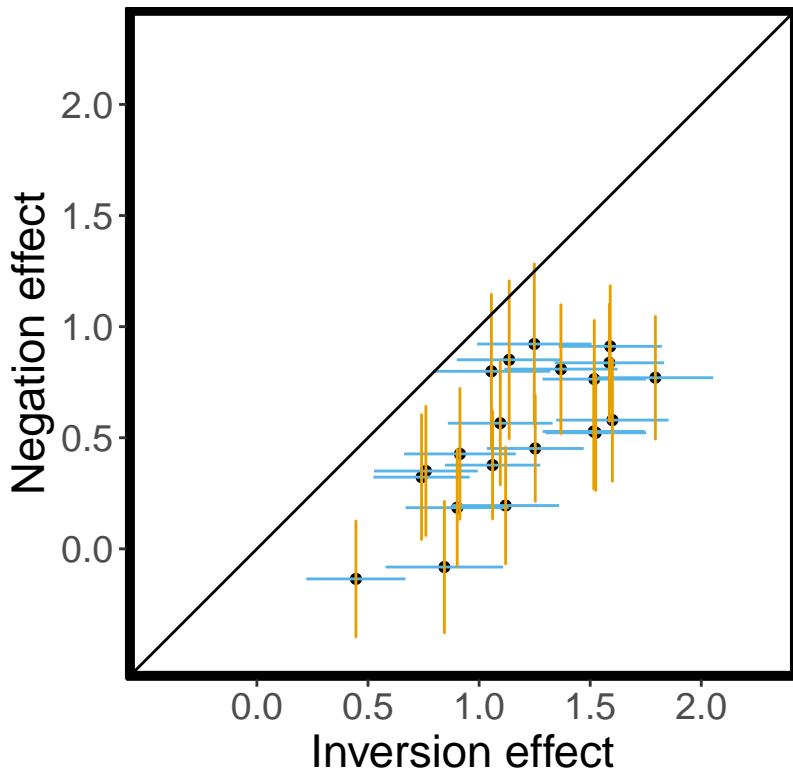
Comparison estimates for Peak Amplitude



```
## pdf
## 2

## [1] "The correlation coefficient for peak amplitude is : "
## 
## Pearson's product-moment correlation
## 
## data: peakAmpl1$inv and peakAmpl1$neg
## t = 26.023, df = 19, p-value = 2.539e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9657429 0.9945230
## sample estimates:
##        cor
## 0.9862596
```

Comparison estimates for Base Amplitude

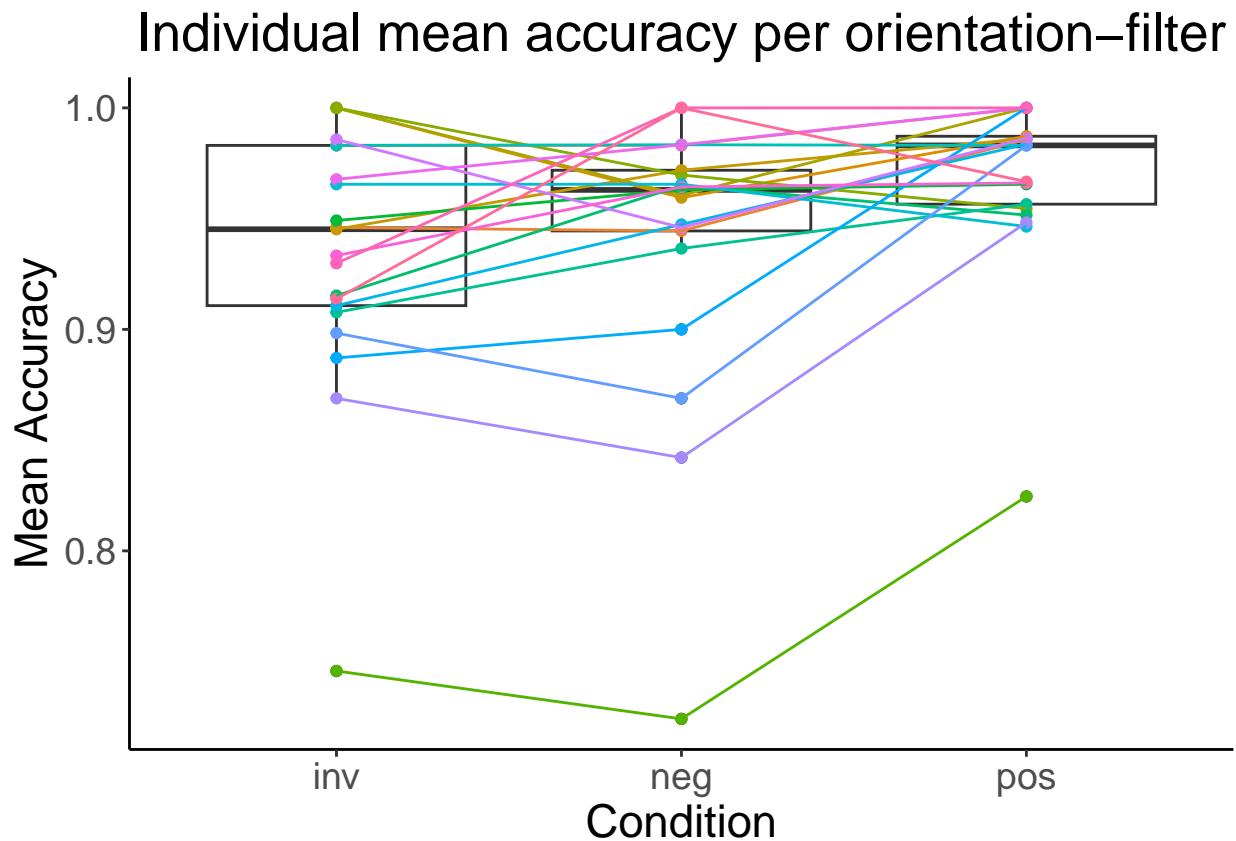


```
## pdf
## 2

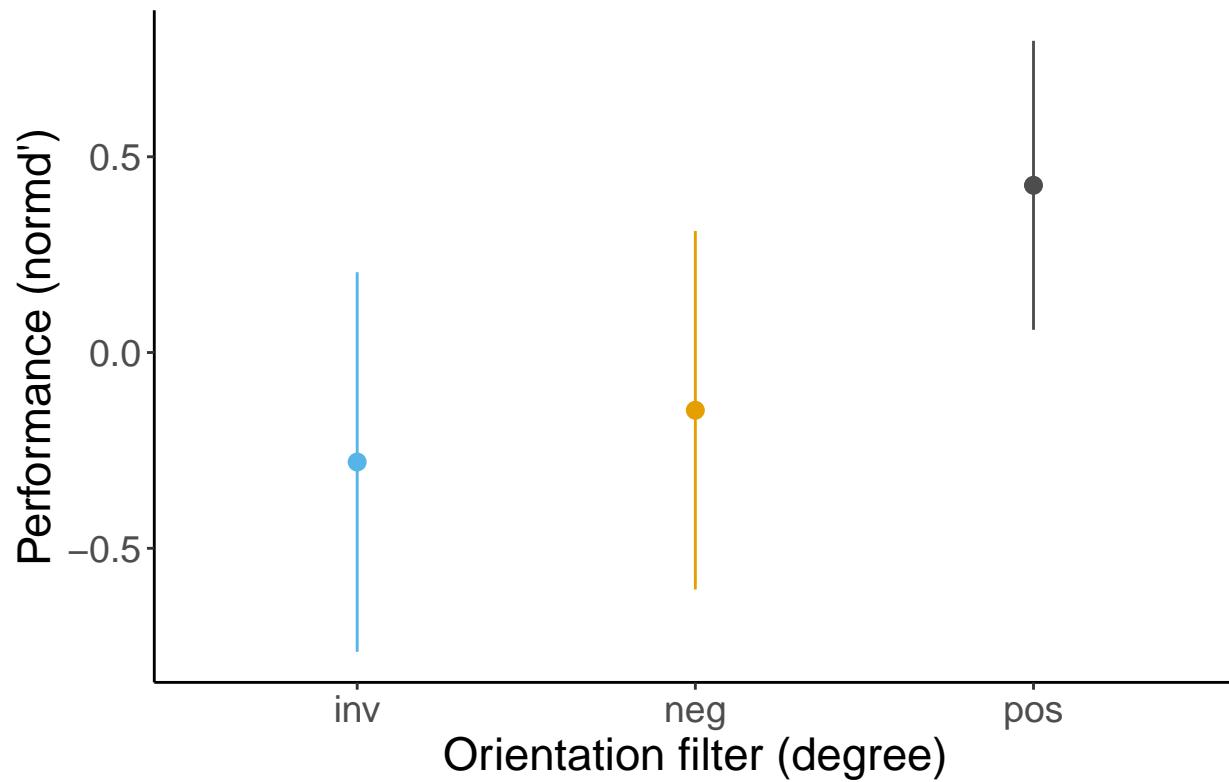
## [1] "The correlation coefficient for base amplitude is : "
## 
## Pearson's product-moment correlation
## 
## data: baseAmpl1$inv and baseAmpl1$neg
## t = 4.5123, df = 19, p-value = 0.0002384
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4170155 0.8782378
## sample estimates:
##        cor
## 0.719227
```

Full Spectrum analysis

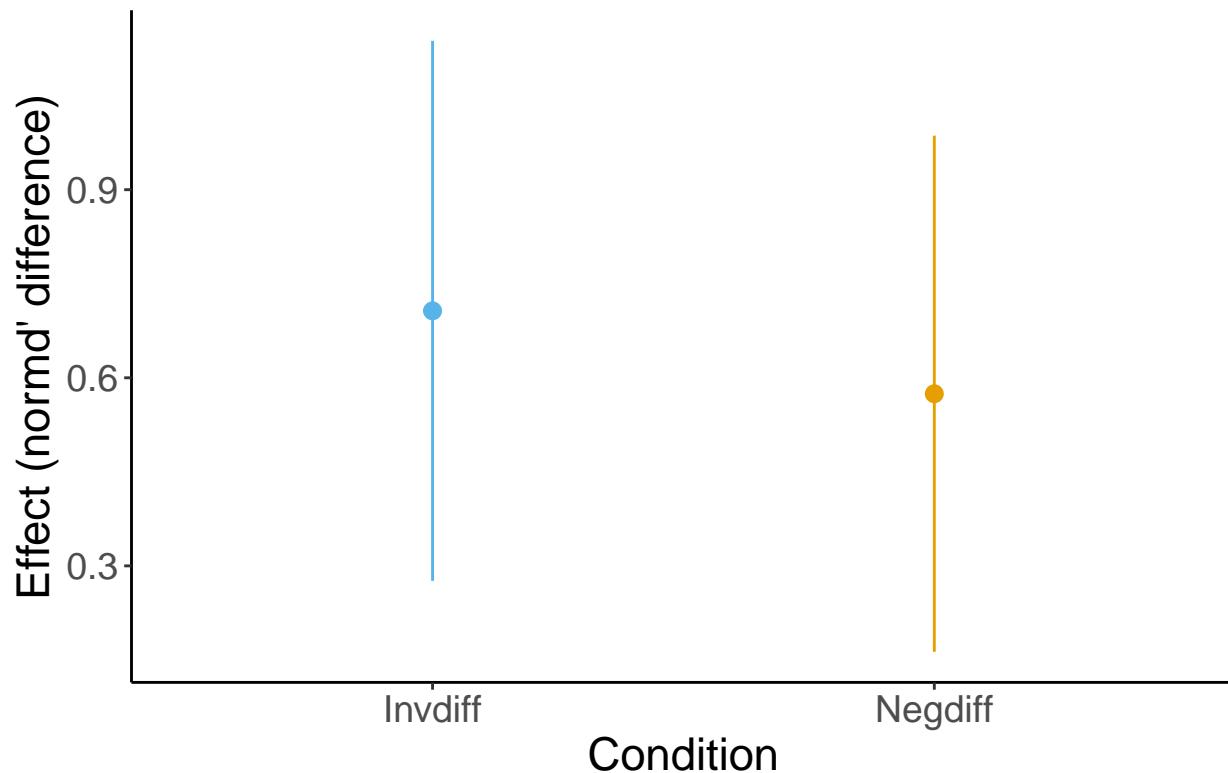
For the main analysis we do not take into account the results for non filtered images. The accuracy and dprime for full spectrum analysis is reported in the Supplementary, as well as the effects of inversion and negation for full spectrum images.



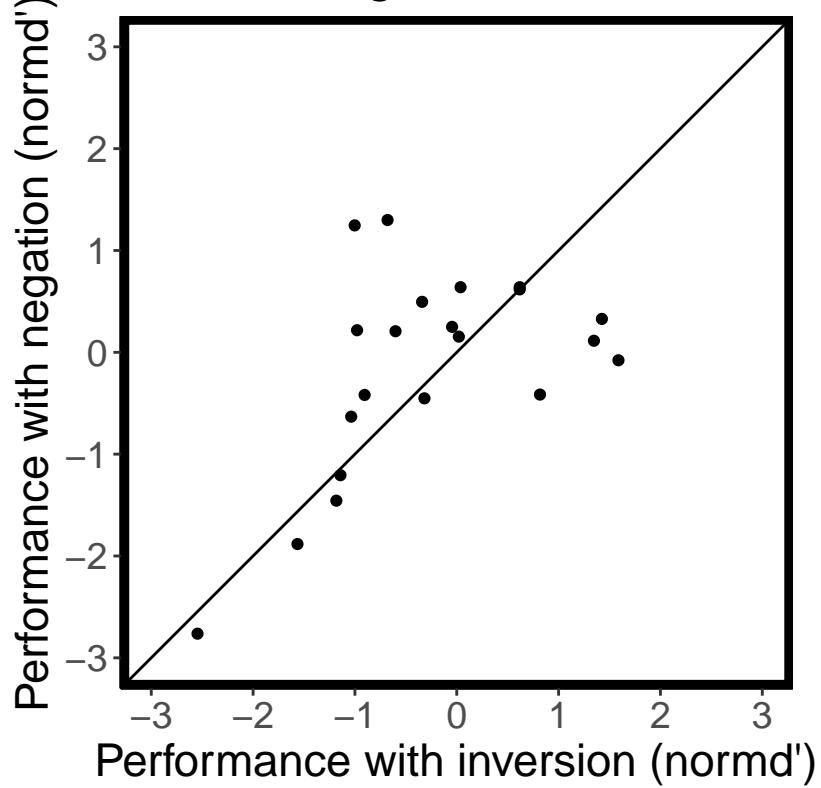
Normalized sensitivity at each filter



Full spectrum inversion and negation effects

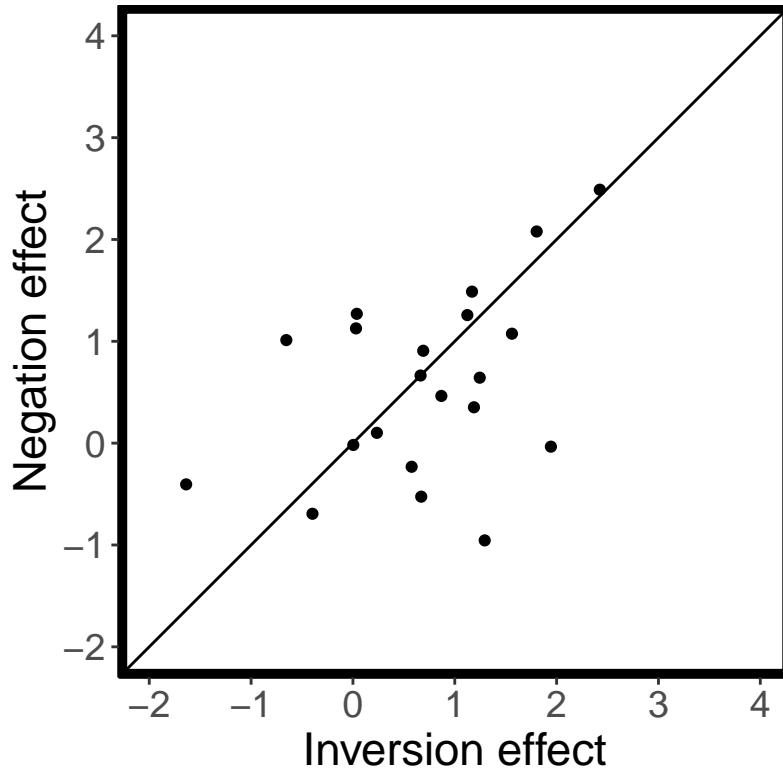


Performance for FS images with inversion and negation



```
## pdf  
## 2
```

Effects comparison for FS images



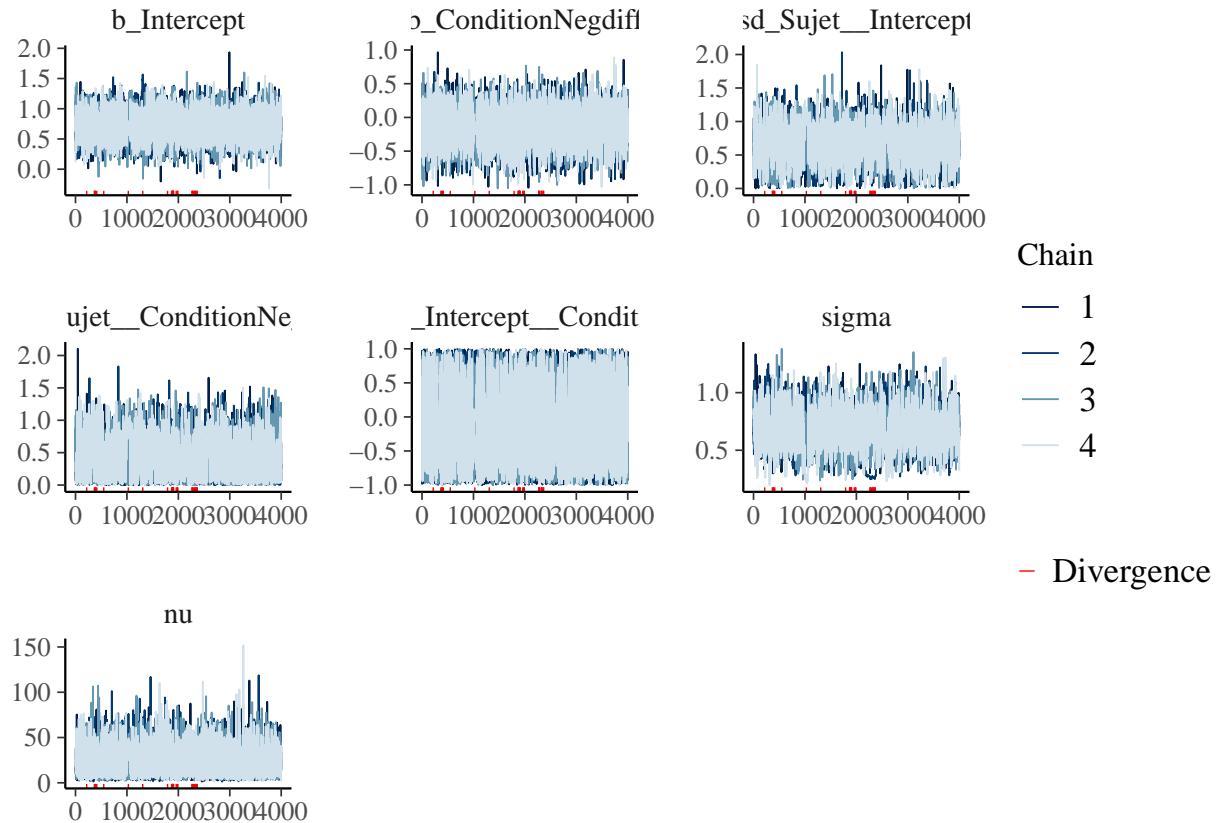
```
## pdf
## 2

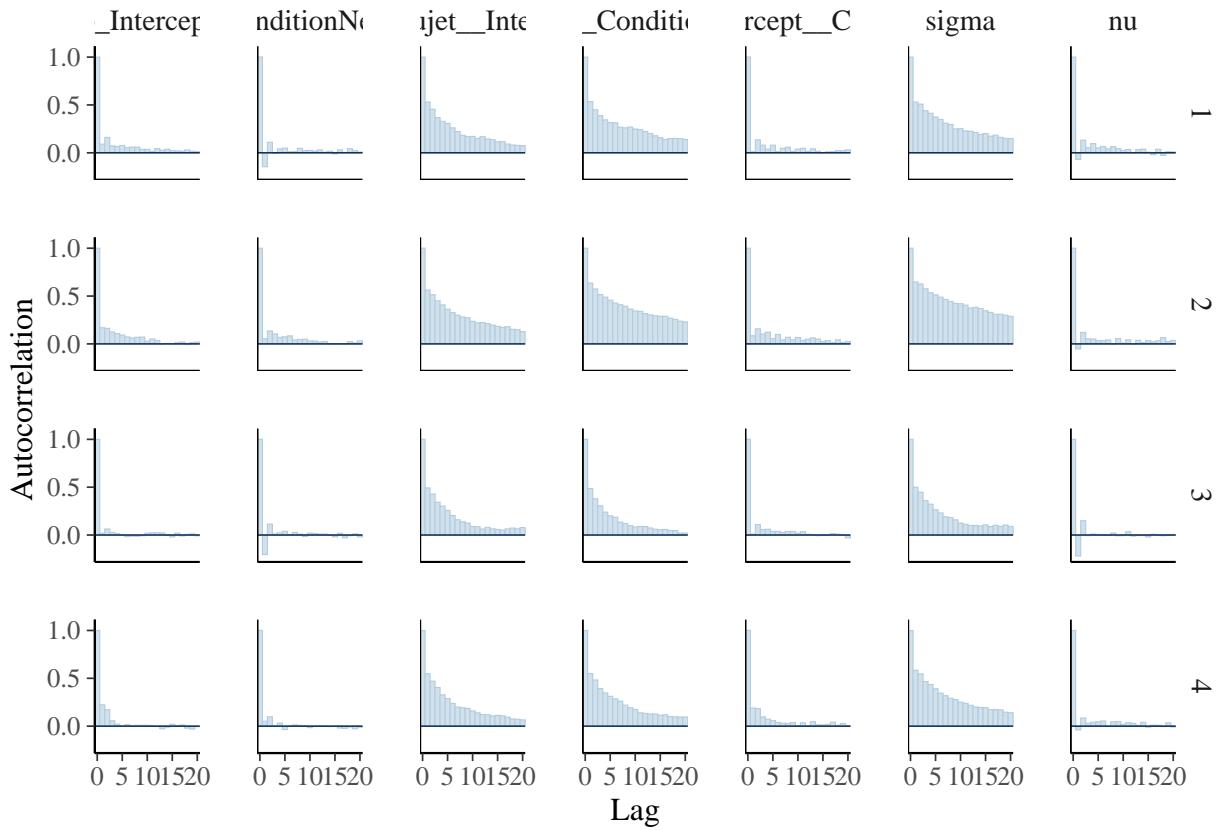
## [1] "Correlation for FS images between the performance in inversion and negation c(t = 2.841587904308"
## [2] "Correlation for FS images between the performance in inversion and negation c(df = 19)"
## [3] "Correlation for FS images between the performance in inversion and negation 0.0104324647722316"
## [4] "Correlation for FS images between the performance in inversion and negation c(cor = 0.546109706"
## [5] "Correlation for FS images between the performance in inversion and negation c(correlation = 0)"
## [6] "Correlation for FS images between the performance in inversion and negation two.sided"
## [7] "Correlation for FS images between the performance in inversion and negation Pearson's product-moment"
## [8] "Correlation for FS images between the performance in inversion and negation diff.fs$inv and diff."
## [9] "Correlation for FS images between the performance in inversion and negation c(0.149718912442251

## [1] "Correlation for FS images between the effects of inversion and negation c(t = 2.0799695507215)"
## [2] "Correlation for FS images between the effects of inversion and negation c(df = 19)"
## [3] "Correlation for FS images between the effects of inversion and negation 0.0513062276123572"
## [4] "Correlation for FS images between the effects of inversion and negation c(cor = 0.43065976304205"
## [5] "Correlation for FS images between the effects of inversion and negation c(correlation = 0)"
## [6] "Correlation for FS images between the effects of inversion and negation two.sided"
## [7] "Correlation for FS images between the effects of inversion and negation Pearson's product-moment"
## [8] "Correlation for FS images between the effects of inversion and negation diff.fs$Negdiff and diff."
## [9] "Correlation for FS images between the effects of inversion and negation c(-0.00126155168082106,
```

General Linear Mixed Model

The same General Linear Mixed model used for orientation-filtered images was run to compare the recognition performance of full spectrum images between the effects of inversion and negation.





```

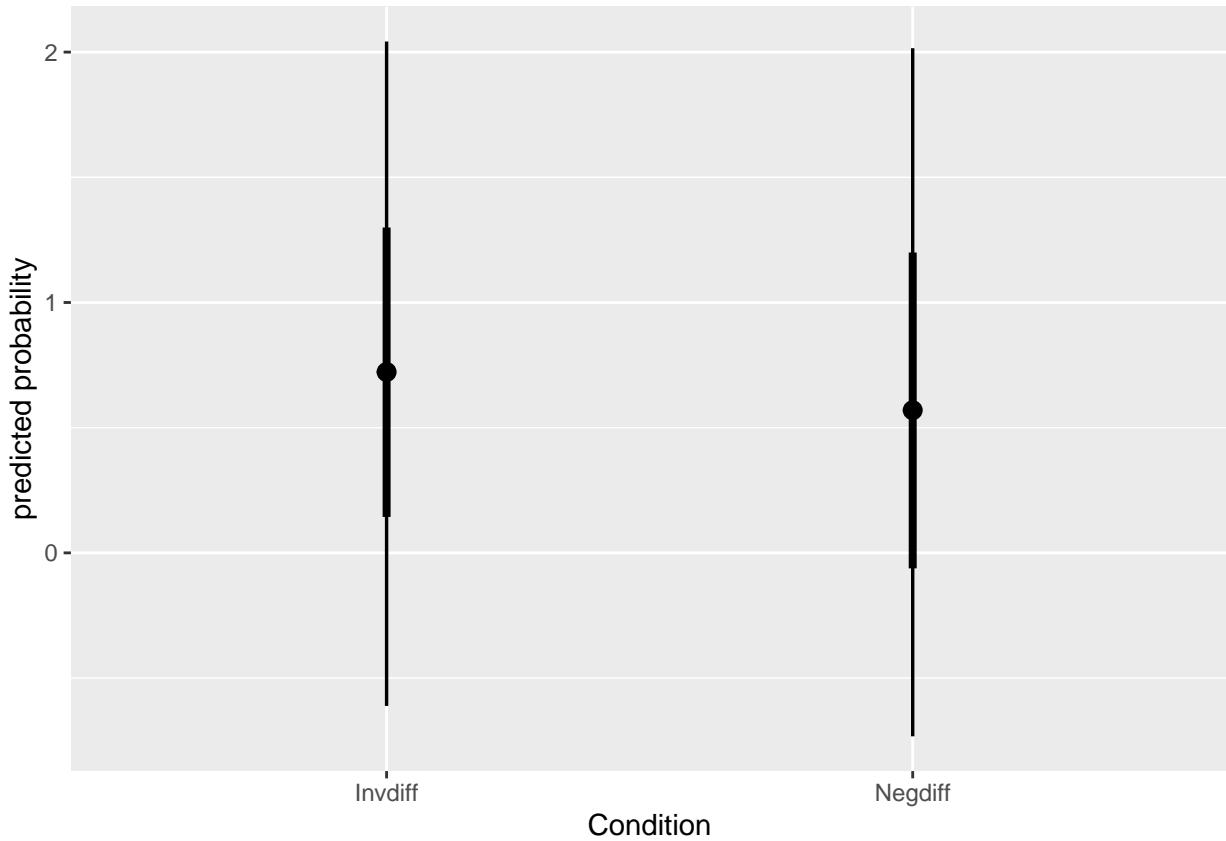
## Family: student
## Links: mu = identity; sigma = identity; nu = identity
## Formula: dprime ~ Condition + (Condition | Sujet)
## Data: diffFS (Number of observations: 42)
## Draws: 4 chains, each with iter = 6000; warmup = 2000; thin = 1;
##        total post-warmup draws = 16000
##
## Group-Level Effects:
## ~Sujet (Number of levels: 21)
##                                         Estimate Est.Error 1-95% CI u-95% CI Rhat
## sd(Intercept)                      0.64      0.27    0.09    1.16 1.00
## sd(ConditionNegdiff)               0.42      0.30    0.02    1.12 1.00
## cor(Intercept,ConditionNegdiff)   -0.15      0.52   -0.94    0.91 1.00
##                                         Bulk_ESS Tail_ESS
## sd(Intercept)                      1166      2497
## sd(ConditionNegdiff)               837       369
## cor(Intercept,ConditionNegdiff)   7106      8463
##
## Population-Level Effects:
##                                         Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept                  0.72      0.22    0.30    1.14 1.00    7019     3690
## ConditionNegdiff          -0.14      0.25   -0.63    0.35 1.00   10048     8789
##
## Family Specific Parameters:
##                                         Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma                     0.66      0.17    0.31    0.99 1.01      539      240

```

```

## nu      21.45     13.87     4.24    57.19  1.00      7582     7995
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).

```



```

## [1] "The highest Rhat values -- Should be under 1.01"

## r_Sujet[BEC02507,ConditionNegdiff] r_Sujet[SEMY0204,ConditionNegdiff]
##                               1.004822          1.005142
## sd_Sujet__ConditionNegdiff r_Sujet[XAFR1107,ConditionNegdiff]
##                               1.005531          1.006047
## sigma                      lp__
##                               1.006520          1.013759

## [1] "The smallest Effective Sample Size -- should be over 100 * the number of chains"

##                               lp__ r_Sujet[XAFR1107,ConditionNegdiff]
## 241.0937                  494.2899
## sigma          sd_Sujet__ConditionNegdiff
## 504.1986                  576.0855
## r_Sujet[SEMY0204,ConditionNegdiff] r_Sujet[BEC02507,ConditionNegdiff]
## 584.7329                  605.7227

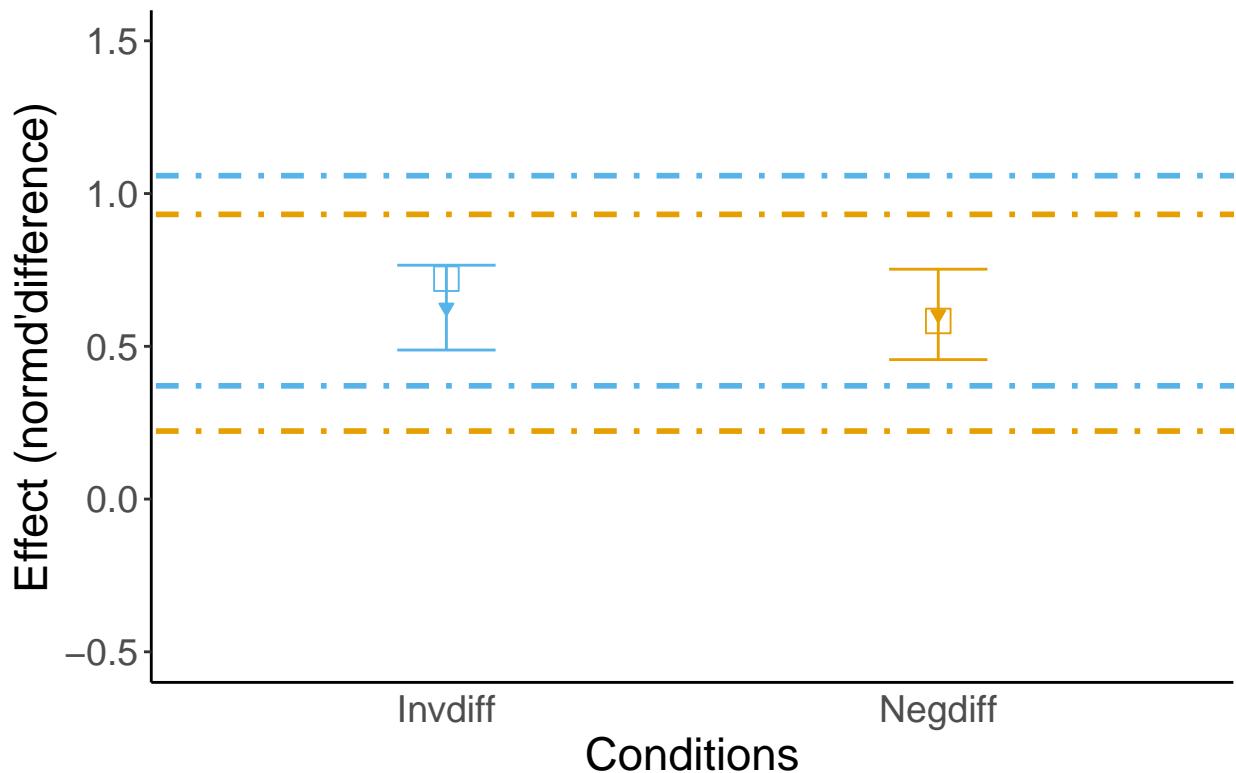
```

```

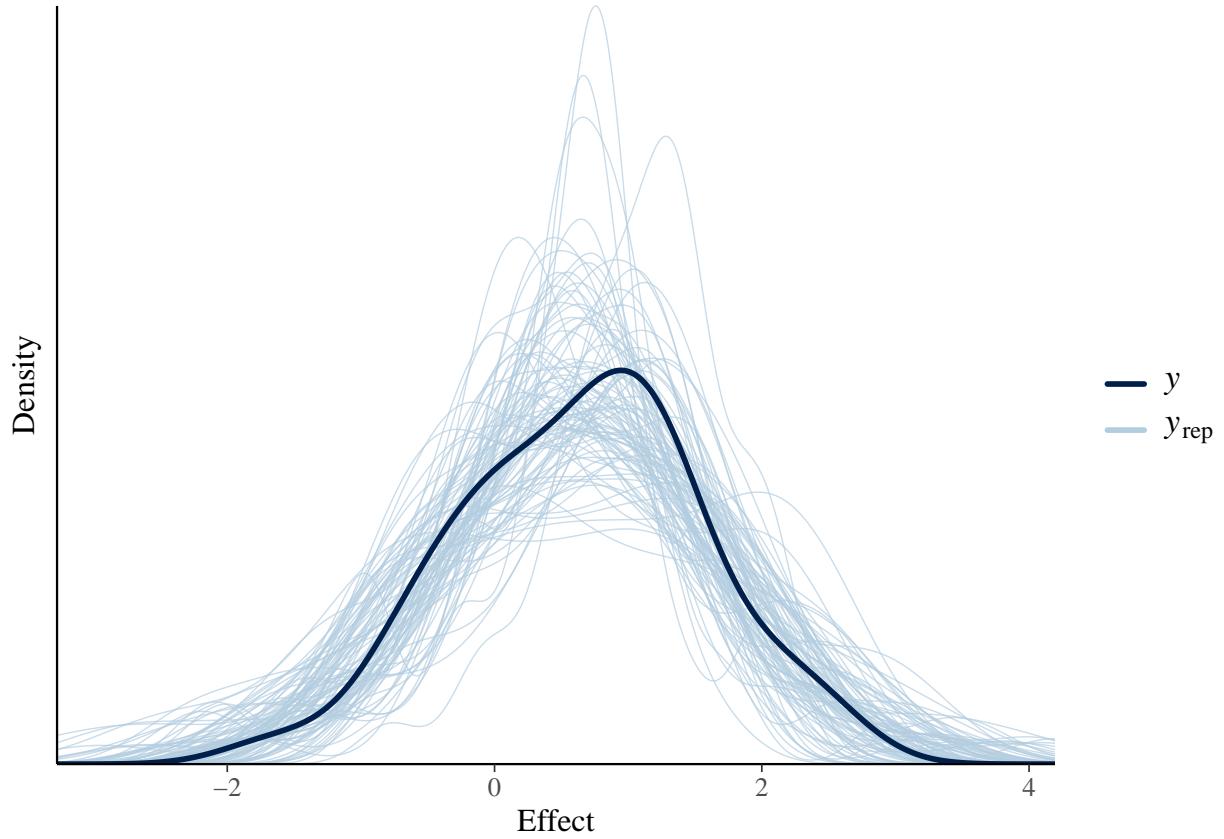
## Family: student
## Links: mu = identity; sigma = identity; nu = identity
## Formula: dprime ~ Condition + (Condition | Sujet)
## Data: diffFS (Number of observations: 42)
## Draws: 4 chains, each with iter = 6000; warmup = 2000; thin = 1;
##         total post-warmup draws = 16000
##
## Group-Level Effects:
## ~Sujet (Number of levels: 21)
##                                         Estimate Est.Error 1-95% CI u-95% CI Rhat
## sd(Intercept)                      0.64      0.27    0.09     1.16 1.00
## sd(ConditionNegdiff)               0.42      0.30    0.02     1.12 1.00
## cor(Intercept,ConditionNegdiff)   -0.15      0.52   -0.94     0.91 1.00
##                                         Bulk_ESS Tail_ESS
## sd(Intercept)                      1166      2497
## sd(ConditionNegdiff)                837       369
## cor(Intercept,ConditionNegdiff)    7106      8463
##
## Population-Level Effects:
##                                         Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept                         0.72      0.22    0.30     1.14 1.00    7019    3690
## ConditionNegdiff                  -0.14      0.25   -0.63     0.35 1.00   10048    8789
##
## Family Specific Parameters:
##                                         Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma      0.66      0.17    0.31     0.99 1.01      539     240
## nu        21.45    13.87    4.24    57.19 1.00     7582    7995
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).

```

Mean and CrI estimates for each effect



```
## pdf  
## 2
```



```
##Upright-positive condition
```

Because we use the upright-positive condition and a control condition, and we use the performance in this condition to calculate the effects of inversion and negation, we wanted to test whether the recognition performance in this upright-positive condition influenced the striking similarities between the effects of inversion and negation.

Gaussian Bayesian non linear mixed modeling

Because the similarities were found at the level of the model parameter estimates of the effects of inversion and negation, we decided to model the normd'of the upright-positive condition using the same model.

```
PosDp <- Ztr %>%
  dplyr::filter(Condition == 'pos') %>%
  dplyr::select(Sujet, Filter, Zscore) %>%
  dplyr::rename(Subjects = Sujet,
    Orifilter = Filter,
    Dp = Zscore) %>%
  dplyr::mutate(Orifilter = as.numeric(Orifilter))

formula <- brmsformula(
  #gaussian model y ~ (1/(StDev * sqrt(2*pi))) * exp(- (x - mu)^2 / (2 * StDev ^2))
  # with y : the difference and x: the orientation filter, mu:the mean centered on the peaklocation
  Dp ~ BaseAmpl + PeakAmpl * exp(-(Orifilter - PeakLoc)^2 / (2 * StDev^2)),
  #All 4 parameters : BaseAmpl , PeakAmpl, PeakLoc, StDev
  #VD ~ 1 + effet(s) fixe(s) + (XX / XX)
  # Peak location parameter (PeakLoc)
  PeakLoc ~ 1 + (1 | Subjects),
  # Standard Deviation parameter (StDev)
  StDev ~ 1 + (1 | Subjects),
  # Base amplitude parameter (baseAmpl)
  BaseAmpl ~ 1 + (1 | Subjects),
  # Peak amplitude parameter (peakAmpl)
  PeakAmpl ~ 1 + (1 | Subjects),
  # non-linear model = TRUE
  nl = TRUE
)

# Priors
priors <- c(
  # peak location:
  prior(normal(80, 30), class = "b", npar = "PeakLoc", coef = "Intercept"), #90,40
  #for the intercept we can set an informative prior / or set it on the center
```

```

# 0-180 the middle is 90
#prior(normal(0, 20), nelpar = "PeakLoc", class = "b"),
# even if we expect a difference we set the difference to 0 +/- 5
#to not influence the model
prior(exponential(0.1), nelpar = "PeakLoc", class = "sd"),
#we chose exp because it is always positive

# StDev:
prior(normal(50, 20), class = "b", nelpar = "StDev", coef = "Intercept"), #40,20
#prior(normal(0, 30), nelpar = "StDev", class = "b"), #0,20/ 0,30
prior(exponential(0.1), nelpar = "StDev", class = "sd"),

# base amplitude:
prior(normal(2 , 1), class = "b", nelpar = "BaseAmpl", coef = "Intercept"), # -0.5, 1
# prior(normal(0, 1.5), nelpar = "BaseAmpl", class = "b"), #0, 1
prior(exponential(0.05), nelpar = "BaseAmpl", class = "sd"),

# peak amplitude:
prior(normal(1, 1), class = "b", nelpar = "PeakAmpl", coef = "Intercept"), #1,2
# prior(normal(0, 1.5), nelpar = "PeakAmpl", class = "b"), #0,1/#0,2
prior(exponential(0.1), nelpar = "PeakAmpl", class = "sd"),

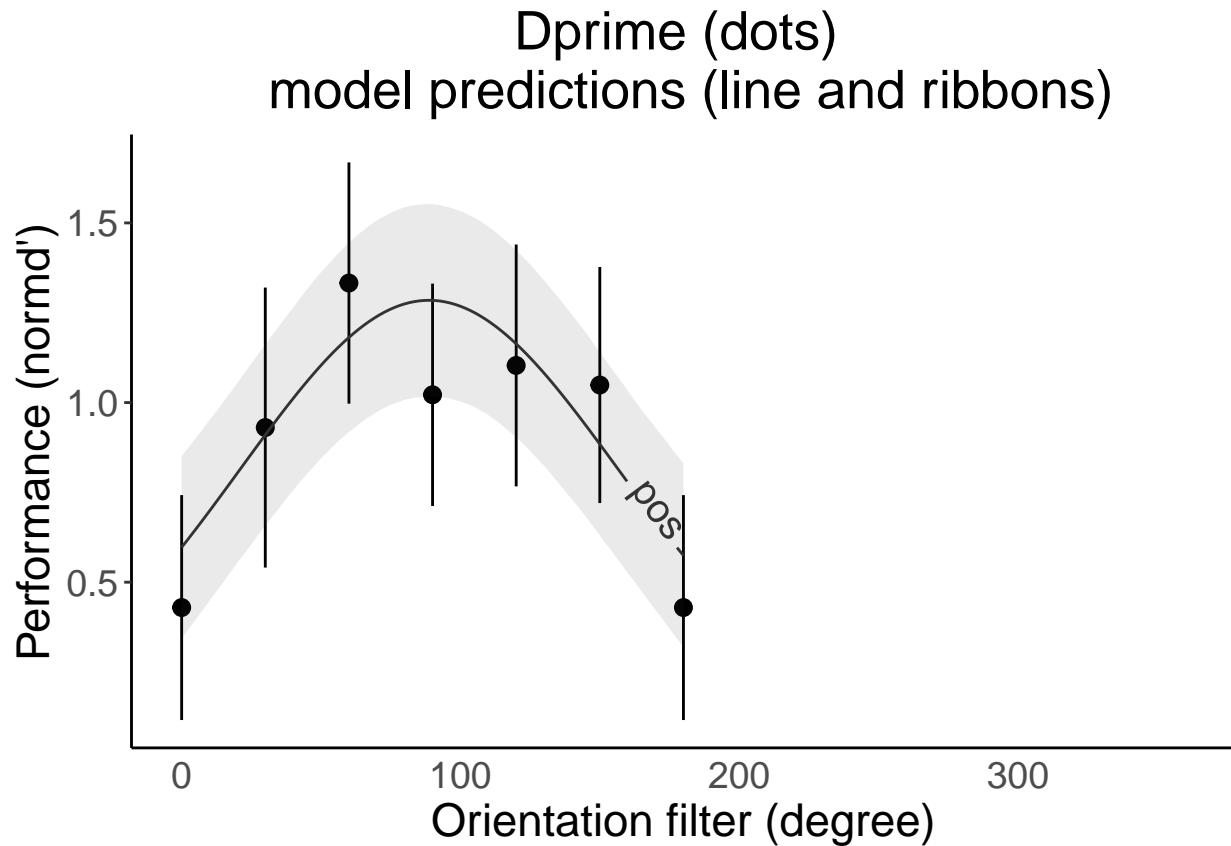
# sigma parameter of the Gaussian model
# (accounts for the variability of the residuals)
prior(exponential(0.1), class = "sigma")
)

# Fitting

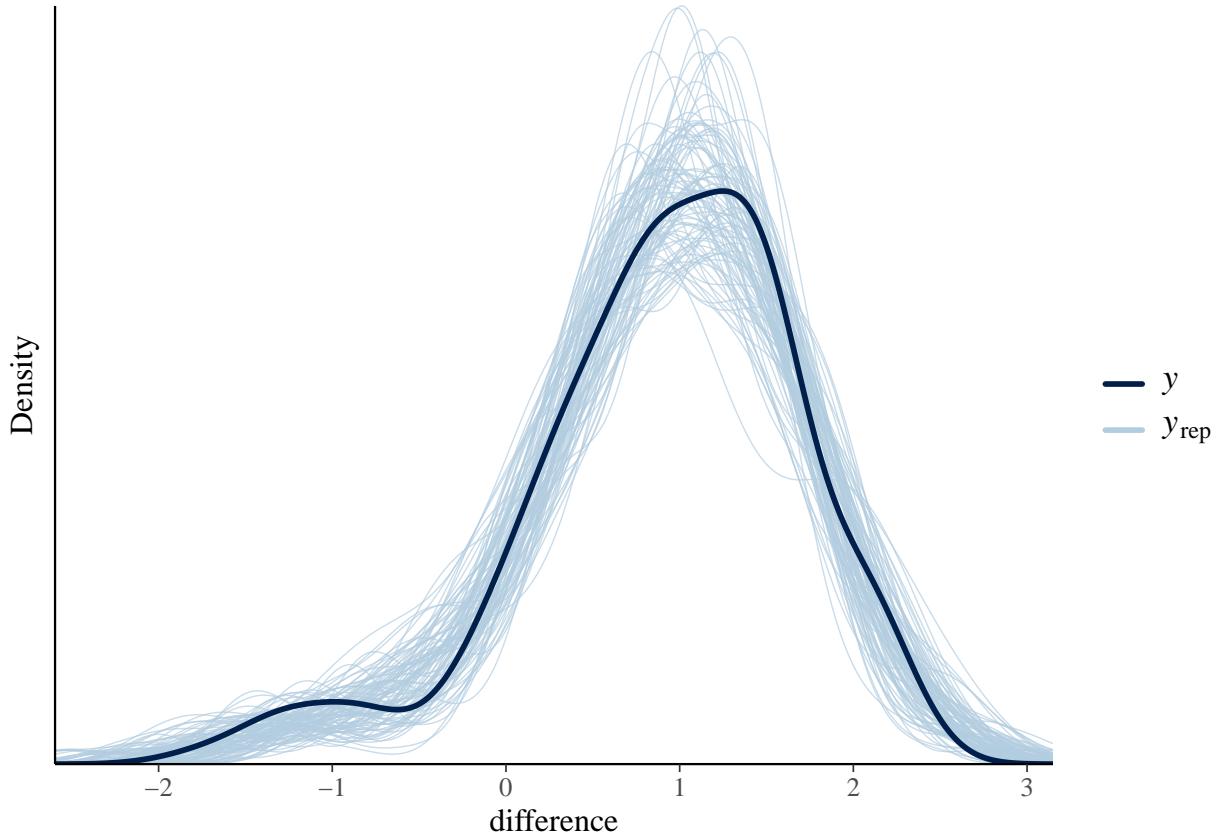
fit <- brm(
  formula = formula,
  data = PosDp,
  family = gaussian("identity"),
  prior = priors,
  warmup = 3000, #1000
  iter = 6000, #4000
  cores = parallel::detectCores(),
  chains = 4, #2
  control = list(adapt_delta = 0.9),
  file = "posDP14.rds",
  seed = 986,
  sample_prior = "yes"  #permet de REGARDER
)

```

Uptight-positive check of bayesian modeling



```
## pdf  
## 2
```



Upright-positive model diagnostic

```

##  Family: gaussian
##  Links: mu = identity; sigma = identity
## Formula: Dp ~ BaseAmpl + PeakAmpl * exp(-(Orifilter - PeakLoc)^2/(2 * StDev^2))
##            PeakLoc ~ 1 + (1 | Subjects)
##            StDev ~ 1 + (1 | Subjects)
##            BaseAmpl ~ 1 + (1 | Subjects)
##            PeakAmpl ~ 1 + (1 | Subjects)
##      Data: PosDp (Number of observations: 147)
##    Draws: 4 chains, each with iter = 6000; warmup = 3000; thin = 1;
##           total post-warmup draws = 12000
##
## Group-Level Effects:
## ~Subjects (Number of levels: 21)
##                               Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS
## sd(PeakLoc_Intercept)      5.51     4.44    0.18   16.26 1.00    5579
## sd(StDev_Intercept)       7.06     6.29    0.22   23.58 1.00    4725
## sd(BaseAmpl_Intercept)    0.61     0.14    0.37    0.90 1.00    3846
## sd(PeakAmpl_Intercept)    0.31     0.19    0.02    0.72 1.00    2107
##                               Tail_ESS
## sd(PeakLoc_Intercept)     6396
## sd(StDev_Intercept)      3004
## sd(BaseAmpl_Intercept)   3233
## sd(PeakAmpl_Intercept)   4142

```

```

## 
## Population-Level Effects:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## PeakLoc_Intercept     88.76      4.68    79.26   97.83 1.00    14983    8837
## StDev_Intercept       70.09     11.14    49.43   92.94 1.00     5898    6319
## BaseAmpl_Intercept     0.02      0.31    -0.65    0.55 1.00     5373    5721
## PeakAmpl_Intercept     1.27      0.29     0.77    1.93 1.00     6801    5962
## 
## Family Specific Parameters:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma        0.44      0.03     0.39     0.51 1.00    11097    8978
## 
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).

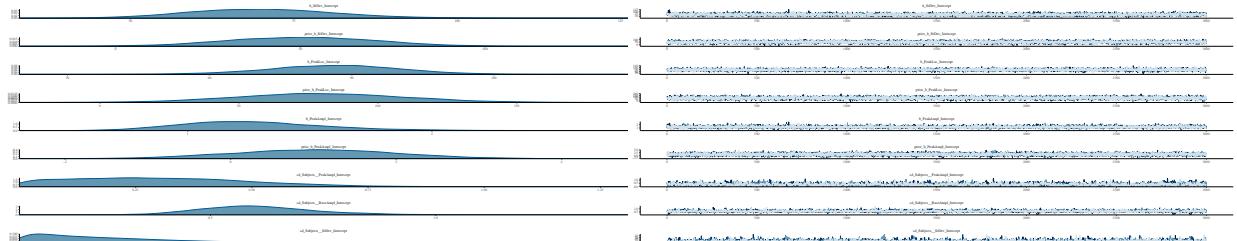
## [1] "The highest Rhat values -- Should be under 1.01"

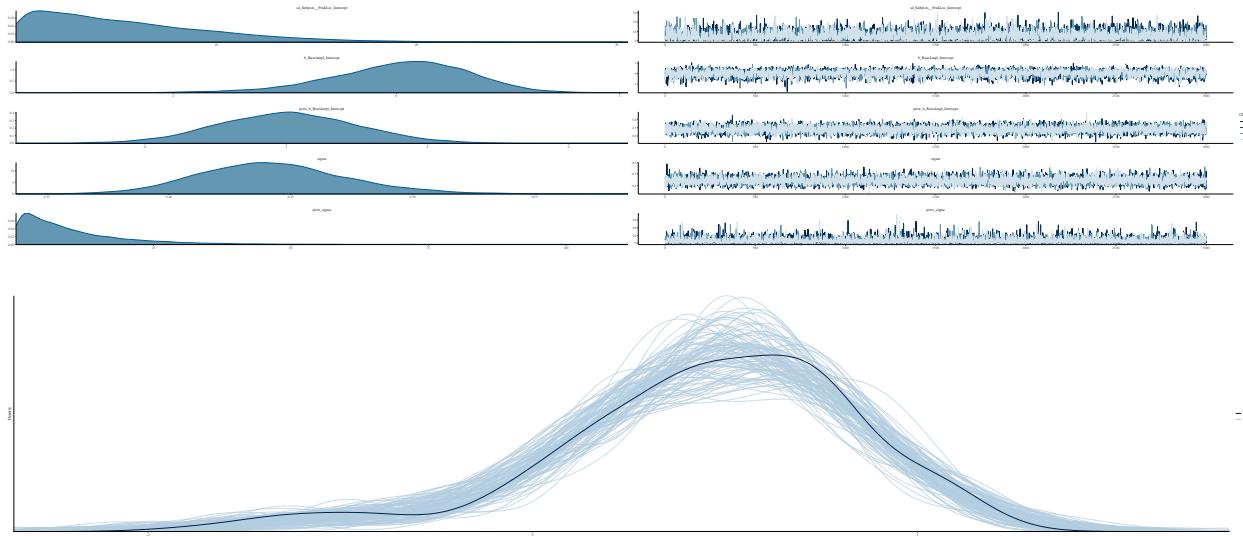
##                      lp_-
##                      1.000690
## r_Subjects__StDev[EVES0409,Intercept]
##                      1.000801
## r_Subjects__BaseAmpl[RIIS2206,Intercept]
##                      1.000805
## r_Subjects__PeakAmpl[RIIS2206,Intercept]
##                      1.000844
## sd_Subjects__PeakAmpl_Intercept
##                      1.000936
## sd_Subjects__StDev_Intercept
##                      1.000941

## [1] "The smallest Effective Sample Size -- should be over 100 * the number of chains"

## r_Subjects__StDev[EVES0409,Intercept]
##                      2000.108
## sd_Subjects__PeakAmpl_Intercept
##                      2030.931
## r_Subjects__PeakAmpl[EVES0409,Intercept]
##                      2424.116
##                      lp_-
##                      2786.127
## r_Subjects__BaseAmpl[EVES0409,Intercept]
##                      2791.235
## sd_Subjects__StDev_Intercept
##                      3485.047

```





Partial correlations between the parameter estimates of the effects of inversion and negation controlling for the influence of the upright-positive condition.

```

## $estimate
##          P A pos      inv      neg
## P A pos 1.00000000 0.1563259 -0.02924284
## inv      0.15632594 1.0000000  0.97828887
## neg     -0.02924284 0.9782889  1.00000000
##
## $p.value
##          P A pos      inv      neg
## P A pos 0.0000000 5.104348e-01 9.025959e-01
## inv      0.5104348 0.000000e+00 9.409968e-14
## neg     0.9025959 9.409968e-14 0.000000e+00
##
## $statistic
##          P A pos      inv      neg
## P A pos 0.0000000 0.6714904 -0.12412
## inv      0.6714904 0.0000000 20.02706
## neg     -0.1241200 20.0270627 0.00000
##
## $n
## [1] 21
##
## $gp
## [1] 1
##
## $method
## [1] "pearson"

## $estimate
##          B A pos      inv      neg
## B A pos 1.0000000 0.2447709 0.2829485
## inv      0.2447709 1.0000000 0.5995942
## neg     0.2829485 0.5995942 1.0000000

```

```

##
## $p.value
##          BPos      inv      neg
## BPos 0.0000000 0.298296355 0.226739106
## inv   0.2982964 0.000000000 0.005201179
## neg   0.2267391 0.005201179 0.000000000
##
## $statistic
##          BPos      inv      neg
## BPos 0.0000000 1.071055 1.251595
## inv   1.071055 0.0000000 3.178619
## neg   1.251595 3.178619 0.0000000
##
## $n
## [1] 21
##
## $gp
## [1] 1
##
## $method
## [1] "pearson"

##
## Pearson's product-moment correlation
##
## data: BaseAmpAll$BPos and BaseAmpAll$inv
## t = 2.7959, df = 19, p-value = 0.01153
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.1410951 0.7879413
## sample estimates:
## cor
## 0.5398973

##
## Pearson's product-moment correlation
##
## data: BaseAmpAll$BPos and BaseAmpAll$neg
## t = 2.899, df = 19, p-value = 0.009199
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.1604516 0.7953350
## sample estimates:
## cor
## 0.5537809

```