Bayesian analysis of the NESTA study of interventions against verbal aggression online

Rafal Urbaniak

Contents

1 Exploration

Load the dataset and take a look first.

```
summaries <- read.csv(file = "datasets/Summaries.csv")
head(summaries) %>% kable( "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down") ,font_size = 9)
```

The basic variables we are dealing with are in the following table.

Further variables are defined in terms of those, in particular, we will be predicting AdiffS which is the standardized difference AA-AB, and AdiffS, which is the standardized difference CA-CB. Before we proceed, we will also standardize the predictors, and add a numerical index for the group:

```
summaries$ABS <- standardize(summaries$AB)
summaries$CBS <- standardize(summaries$CB)
summaries$AAS <- standardize(summaries$AA)
summaries$CAS <- standardize(summaries$CA)
summaries$CDS <- standardize(summaries$CD)
summaries$ADS <- standardize(summaries$AD)
summaries$group <- as.factor(summaries$group)
summaries$groupID <-  as.integer( as.factor(summaries$group) )
```

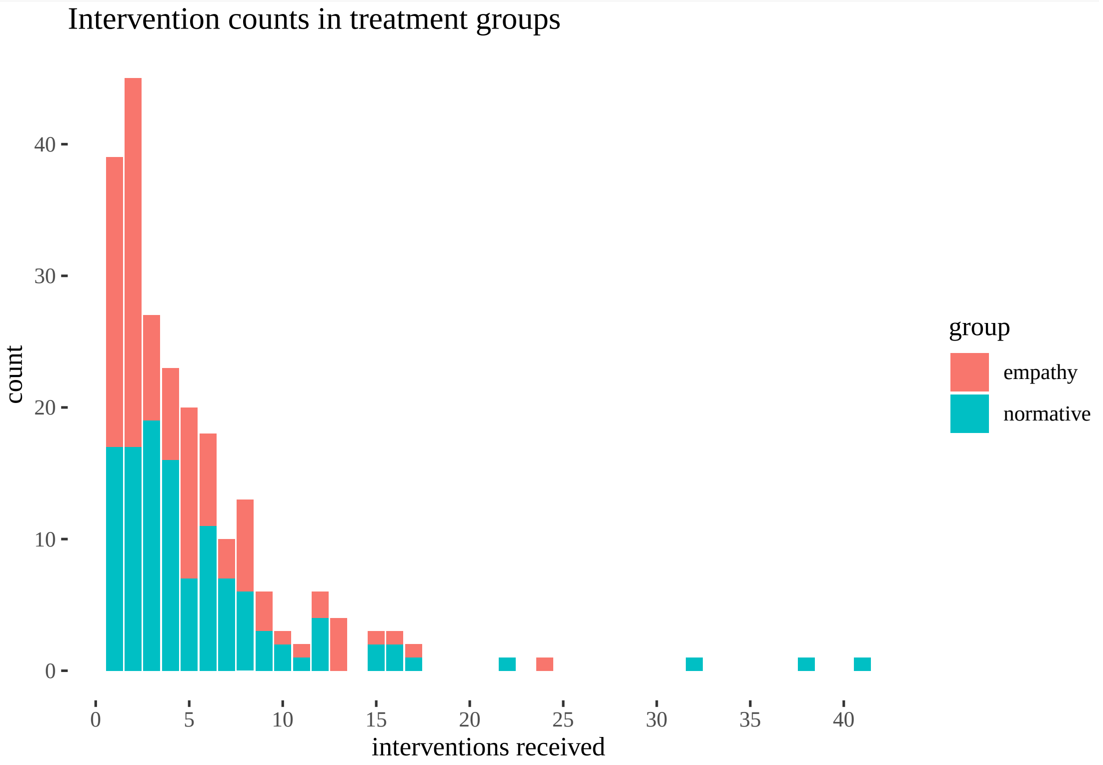
First, let's take a look at the distribution of IC in the treatment groups:

```
ggplot(summaries[summaries$group != "control",], aes(x = IC, fill = group))+
  geom_bar()+theme_tufte()+
  xlab("interventions received")+
```

X	author	AB	AD	AA	CB	CD	CA	Adiff	Cdiff	AdiffS	CdiffS	group	IC
1	_swf	19	1	0	720	25	28	-19	-692	-0.0245122	-0.3501491	normative	1
2	-Allergic	24	24	8	1614	1451	1237	-16	-377	0.0719197	0.1057675	normative	3
3	-funny-username-	23	6	12	847	497	721	-11	-126	0.2326395	0.4690535	control	0
4	-Johnny-	18	2	8	1465	408	684	-10	-781	0.2647835	-0.4789637	empathy	2
5	1secwhileiyeet3	15	3	4	1384	198	120	-11	-1264	0.2326395	-1.1780359	control	0
6	20CharsIsNotEnough	16	10	25	779	907	972	9	193	0.8755188	0.9307596	empathy	4

variable	explanation
AB	attacks before (pre-treatment)
AD	attacks during (the treatment period)
AA	attacks after (post-treatment)
CB	comments before
CD	comments during
CA	comments after
group	treatment group
IC	intervention count

```
  labs(title = "Intervention counts in treatment groups")+
  scale_x_continuous(breaks = seq(0,40,5))
```



Note there were much more empathetic interventions, this needs an explanation

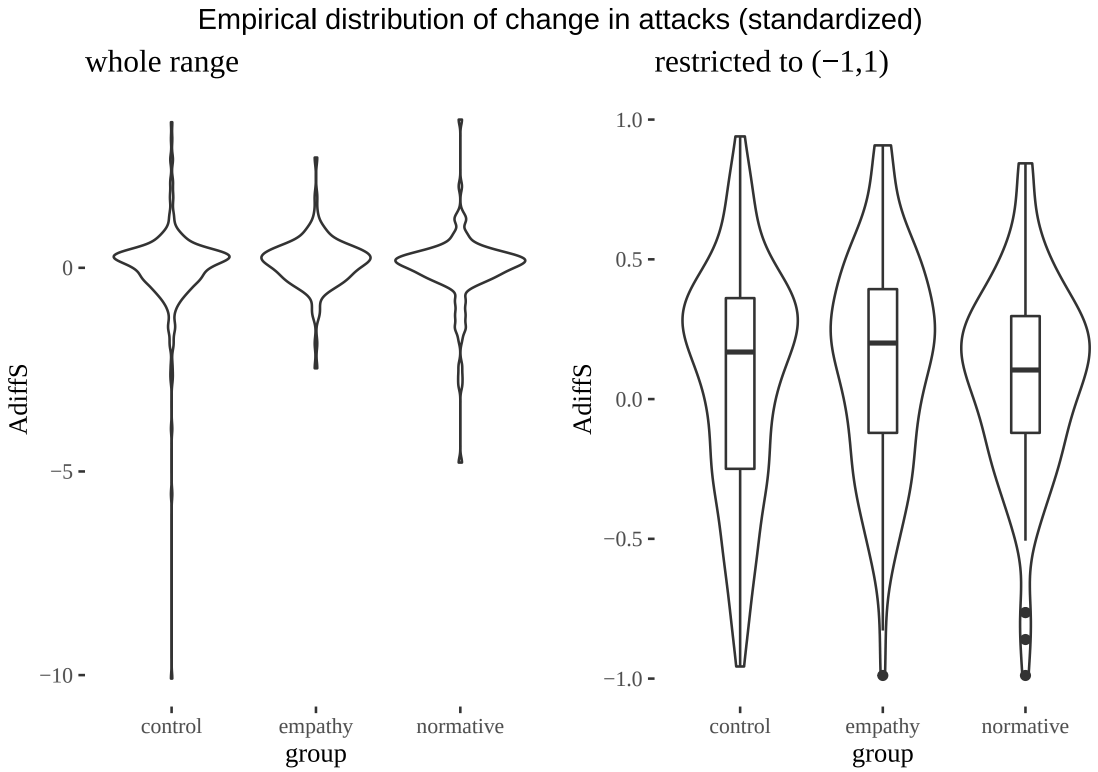
Question: intervention counts by group

Second, when we look at the distribution of standardized difference in attacks, when restricted to (-1,1), the peaks of distributions are shifted a bit, with lowest median for the normative group, but not too much:

```
violAdiffS <- ggplot(summaries, aes(x=group, y = AdiffS))+
  geom_violin() +theme_tufte()
violJoint <- ggarrange(violAdiffS+ggtitle("whole range"),
              violAdiffS + ylim(c(-1,1))+geom_boxplot(width = .2)+
              ggtitle("restricted to (-1,1)"))
```

```
## Warning: Removed 58 rows containing non-finite values (stat_ydensity).
## Warning: Removed 58 rows containing non-finite values (stat_boxplot).
```

```
violJointTitled <- annotate_figure(violJoint,
  top = text_grob("Empirical distribution of change in attacks (standardized)",
                size = 12))
violJointTitled
```

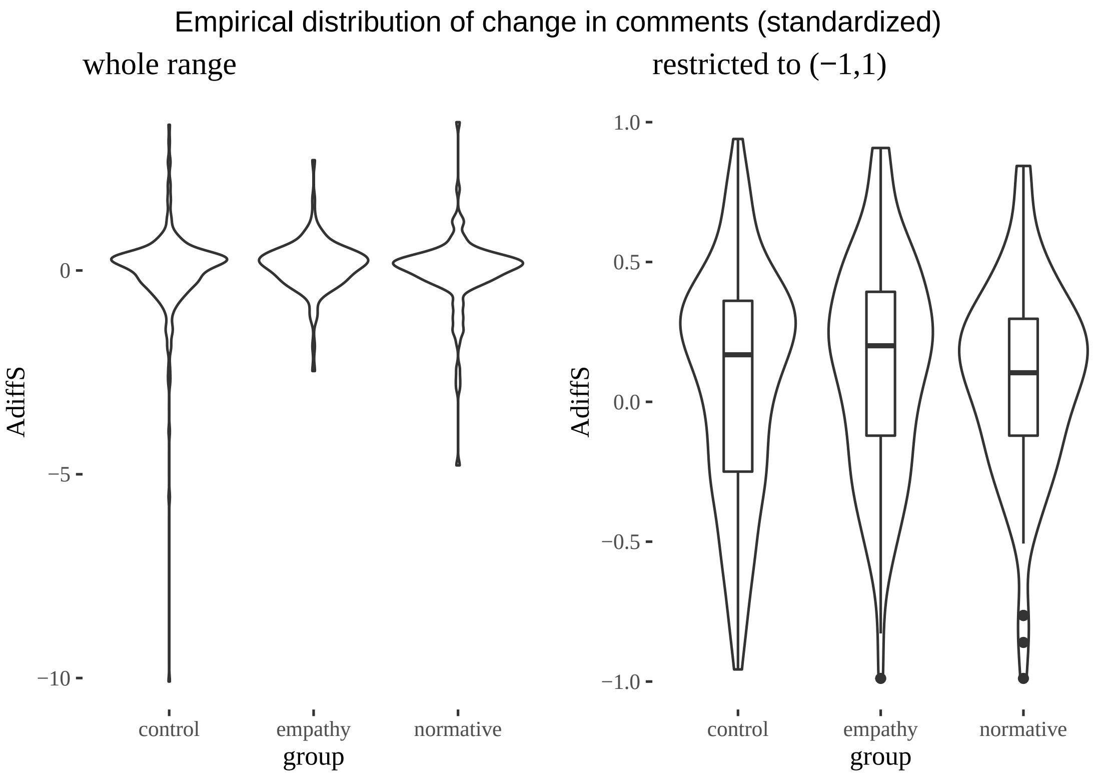


Analogous plot for comments does not reveal this slight downward shift for normative, but otherwise the visualisation migth suggest no strong impact of interventions on attacks, and no impact on comments.

```
violCdiffS <- ggplot(summaries, aes(x=group, y = CdiffS))+
  geom_violin() +theme_tufte()
violJointC <- ggarrange(violCdiffS+ggtitle("whole range"),
              violCdiffS + ylim(c(-1,1))+geom_boxplot(width = .2)+
              ggtitle("restricted to (-1,1)"))
```

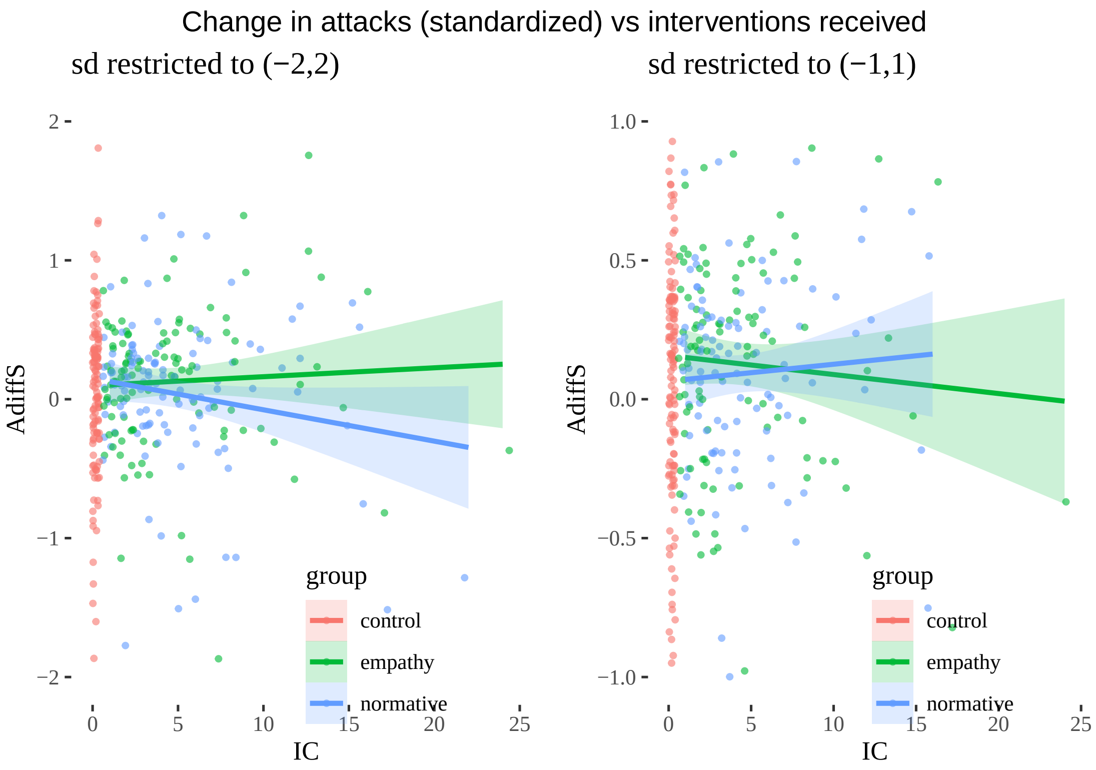
```
## Warning: Removed 90 rows containing non-finite values (stat_ydensity).
## Warning: Removed 90 rows containing non-finite values (stat_boxplot).
```

```
violJointCTitled <- annotate_figure(violJoint,
  top = text_grob("Empirical distribution of change in comments (standardized)",
                  size = 12))
violJointCTitled
```



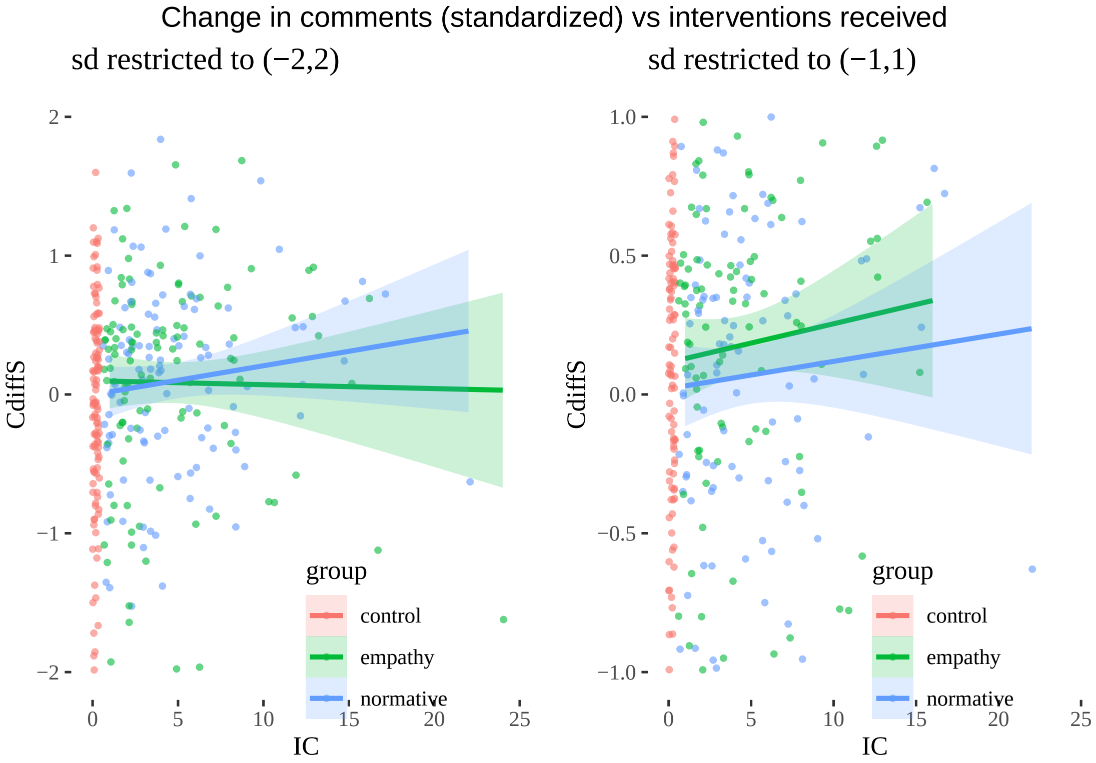
However, plotting changes against intervention counts reveals that restricting attention to various activity levels drastically changes the regression lines.

```
icplot1 <- ggplot(summaries, aes(x = IC, y = AdiffS, color = group, fill = group))+
  geom_jitter(alpha = 0.6, size =.8)+theme_tufte()+
  geom_smooth(alpha = 0.2, method = "lm")+
  xlim(c(0,25))+ylim(c(-2,2))+
  ggtitle("sd restricted to (-2,2)")+
  theme(legend.position = c(0.65, 0.1))

icplot2 <-  ggplot(summaries, aes(x = IC, y = AdiffS, color = group, fill = group))+
  geom_jitter(alpha = 0.6, size =.8)+theme_tufte()+
  geom_smooth(alpha = 0.2, method = "lm")+
  xlim(c(0,25))+ylim(c(-1,1))+ggtitle("sd restricted to (-1,1)")+
  theme(legend.position = c(0.65, 0.1))

icplotJoint <- ggarrange(icplot1, icplot2)
icplotTitled <- annotate_figure(icplotJoint,
  top = text_grob("Change in attacks (standardized) vs interventions received",  size = 12))
icplotTitled
```



Some interactions are also suggested by the differences in linear smoothing when attention is restricted when it comes to change in comments.

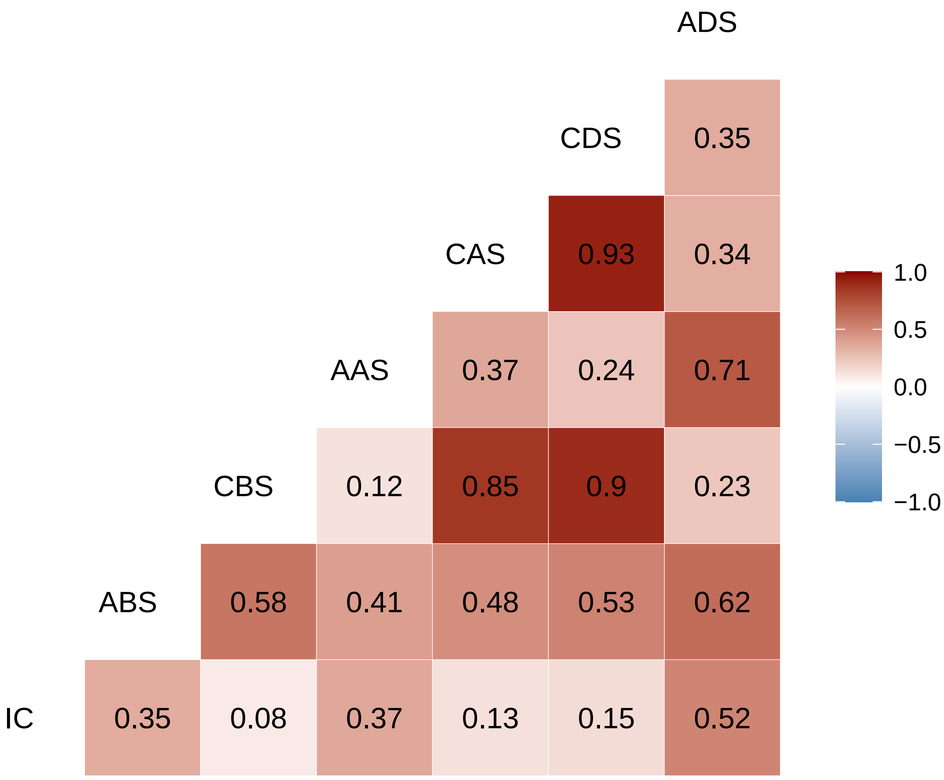
```
icCplot1 <- ggplot(summaries, aes(x = IC, y = CdiffS, color = group, fill = group))+
  geom_jitter(alpha = 0.6, size =.8)+theme_tufte()+
  geom_smooth(alpha = 0.2, method = "lm")+
  xlim(c(0,25))+ylim(c(-2,2))+
  ggtitle("sd restricted to (-2,2)")+
  theme(legend.position = c(0.65, 0.1))

icCplot2 <-  ggplot(summaries, aes(x = IC, y = CdiffS, color = group, fill = group))+
  geom_jitter(alpha = 0.6, size =.8)+theme_tufte()+
  geom_smooth(alpha = 0.2, method = "lm")+
  xlim(c(0,25))+ylim(c(-1,1))+ggtitle("sd restricted to (-1,1)")+
  theme(legend.position = c(0.65, 0.1))

icCplotJoint <- ggarrange(icCplot1, icCplot2)
icCplotTitled <- annotate_figure(icCplotJoint,
  top = text_grob("Change in comments (standardized) vs interventions received",
   size = 12))
icCplotTitled
```



This suggests we should keep an eye out for interactions in the analysis, and that the intial comparison of means or medians between groups might be misleading if the effects in different volume groups are different and cancel each other.

Now, let's inspect correlations between the variables involved in the model:

```
summariesCorr <- select(summaries, IC, ABS, CBS, AAS, CAS, CDS, ADS)
ggcorr(summariesCorr, method = c("pairwise"),
       digits = 4, low = "steelblue", mid = "white",
       high = "darkred", midpoint =0,
       geom = "tile", label = TRUE, label_size=4, label_round =2, layout.exp =1,
       label_alpha = FALSE,hjust = 0.75)
```



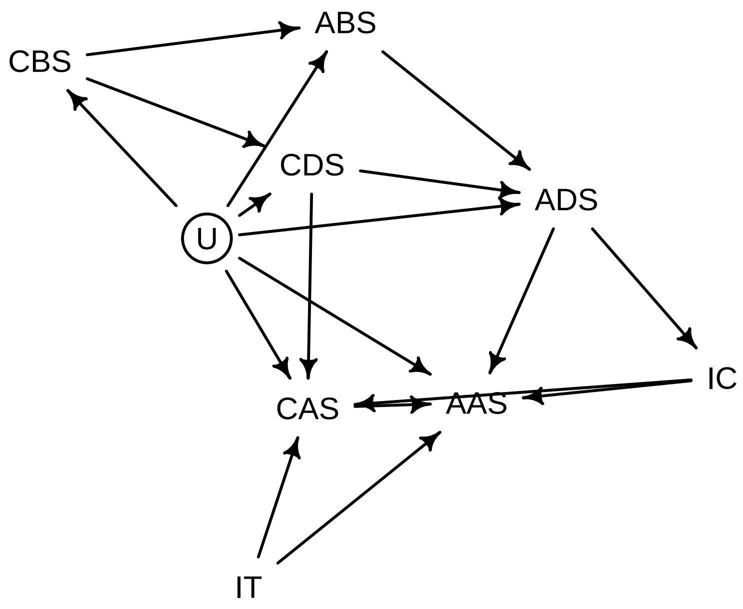
	ABS	CBS	AAS	CAS	CDS	ADS
CDS						0.35
CAS					0.93	0.34
AAS				0.37	0.24	0.71
CBS			0.12	0.85	0.9	0.23
ABS		0.58	0.41	0.48	0.53	0.62
IC	0.35	0.08	0.37	0.13	0.15	0.52

This tells us that almost no predictors are strongly correlated, except for pairs CBS-CDS, so we drop CDS from the analysis and avoid using them in the same model to avoid multicolinearity issues. These are just comments during the intervention period, which, unsurprisingly are also a good proxy for comments before and comments after.

2 Causal inference

To identify the right variables to condition (or not condition) on to identify the causal effect of the interventions, we first need to think about the causal structure of the problem. Here's a plausible causal structure that we will be working with:

```
dag <- dagitty("
  dag{
  CDS -> ADS -> IC  ons
               U [unobserved]
               U -> CBS -> ABS
               U -> ABS
               U -> CDS -> ADS
               U -> ADS
               U -> CAS -> AAS
               U -> AAS
               IC -> AAS
               IC -> CAS
               IT -> CAS
               IT -> AAS
               CBS -> CDS -> CAS
               ABS -> ADS -> AAS
               }")
set.seed(123)
drawdag(dag)
```



ons

Comments during impact attacks during, which trigger interventions. Unmeasured user features cause comments before, which impact attacks before, and also attacks before directly. Comments during (their impact on ADS is areadly included) impact attacks during during directly and comments after, which impact attacks after and attacks after directly. Intervention count impacts attacks after and comments after. The same directions of impact are included for intervention type. Finally, comments through time are connected causally, and so are attacks.

We already know not to condition on CDS if we condition on CAS or CBS. What else? IT has no bacwkard paths, but IC does. Let's identify all paths from IC to AAS:

```r
paths(dag, from = c("IC"), to = "AAS")
```

```
## $paths
##  [1] "IC -> AAS"
##  [2] "IC -> CAS -> AAS"
##  [3] "IC -> CAS <- CDS -> ADS -> AAS"
##  [4] "IC -> CAS <- CDS -> ADS <- ABS <- CBS <- U -> AAS"
##  [5] "IC -> CAS <- CDS -> ADS <- ABS <- U -> AAS"
##  [6] "IC -> CAS <- CDS -> ADS <- U -> AAS"
##  [7] "IC -> CAS <- CDS <- CBS -> ABS -> ADS -> AAS"
##  [8] "IC -> CAS <- CDS <- CBS -> ABS -> ADS <- U -> AAS"
##  [9] "IC -> CAS <- CDS <- CBS -> ABS <- U -> AAS"
## [10] "IC -> CAS <- CDS <- CBS -> ABS <- U -> ADS -> AAS"
## [11] "IC -> CAS <- CDS <- CBS <- U -> AAS"
## [12] "IC -> CAS <- CDS <- CBS <- U -> ABS -> ADS -> AAS"
## [13] "IC -> CAS <- CDS <- CBS <- U -> ADS -> AAS"
## [14] "IC -> CAS <- CDS <- U -> AAS"
## [15] "IC -> CAS <- CDS <- U -> ABS -> ADS -> AAS"
## [16] "IC -> CAS <- CDS <- U -> ADS -> AAS"
## [17] "IC -> CAS <- CDS <- U -> CBS -> ABS -> ADS -> AAS"
## [18] "IC -> CAS <- IT -> AAS"
## [19] "IC -> CAS <- U -> AAS"
## [20] "IC -> CAS <- U -> ABS -> ADS -> AAS"
## [21] "IC -> CAS <- U -> ABS <- CBS -> CDS -> ADS -> AAS"
## [22] "IC -> CAS <- U -> ADS -> AAS"
## [23] "IC -> CAS <- U -> CBS -> ABS -> ADS -> AAS"
## [24] "IC -> CAS <- U -> CBS -> CDS -> ADS -> AAS"
## [25] "IC -> CAS <- U -> CDS -> ADS -> AAS"
## [26] "IC -> CAS <- U -> CDS <- CBS -> ABS -> ADS -> AAS"
## [27] "IC <- ADS -> AAS"
## [28] "IC <- ADS <- ABS <- CBS -> CDS -> CAS -> AAS"
## [29] "IC <- ADS <- ABS <- CBS -> CDS -> CAS <- IT -> AAS"
## [30] "IC <- ADS <- ABS <- CBS -> CDS -> CAS <- U -> AAS"
## [31] "IC <- ADS <- ABS <- CBS -> CDS <- U -> AAS"
## [32] "IC <- ADS <- ABS <- CBS -> CDS <- U -> CAS -> AAS"
## [33] "IC <- ADS <- ABS <- CBS -> CDS <- U -> CAS <- IT -> AAS"
## [34] "IC <- ADS <- ABS <- CBS <- U -> AAS"
## [35] "IC <- ADS <- ABS <- CBS <- U -> CAS -> AAS"
## [36] "IC <- ADS <- ABS <- CBS <- U -> CAS <- IT -> AAS"
## [37] "IC <- ADS <- ABS <- CBS <- U -> CDS -> CAS -> AAS"
## [38] "IC <- ADS <- ABS <- CBS <- U -> CDS -> CAS <- IT -> AAS"
## [39] "IC <- ADS <- ABS <- U -> AAS"
## [40] "IC <- ADS <- ABS <- U -> CAS -> AAS"
## [41] "IC <- ADS <- ABS <- U -> CAS <- IT -> AAS"
## [42] "IC <- ADS <- ABS <- U -> CBS -> CDS -> CAS -> AAS"
## [43] "IC <- ADS <- ABS <- U -> CBS -> CDS -> CAS <- IT -> AAS"
## [44] "IC <- ADS <- ABS <- U -> CDS -> CAS -> AAS"
## [45] "IC <- ADS <- ABS <- U -> CDS -> CAS <- IT -> AAS"
## [46] "IC <- ADS <- CDS -> CAS -> AAS"
## [47] "IC <- ADS <- CDS -> CAS <- IT -> AAS"
## [48] "IC <- ADS <- CDS -> CAS <- U -> AAS"
## [49] "IC <- ADS <- CDS <- CBS -> ABS <- U -> AAS"
## [50] "IC <- ADS <- CDS <- CBS -> ABS <- U -> CAS -> AAS"
## [51] "IC <- ADS <- CDS <- CBS -> ABS <- U -> CAS <- IT -> AAS"
## [52] "IC <- ADS <- CDS <- CBS <- U -> AAS"
## [53] "IC <- ADS <- CDS <- CBS <- U -> CAS -> AAS"
## [54] "IC <- ADS <- CDS <- CBS <- U -> CAS <- IT -> AAS"
## [55] "IC <- ADS <- CDS <- U -> AAS"
## [56] "IC <- ADS <- CDS <- U -> CAS -> AAS"
## [57] "IC <- ADS <- CDS <- U -> CAS <- IT -> AAS"
## [58] "IC <- ADS <- U -> AAS"
## [59] "IC <- ADS <- U -> ABS <- CBS -> CDS -> CAS -> AAS"
## [60] "IC <- ADS <- U -> ABS <- CBS -> CDS -> CAS <- IT -> AAS"
## [61] "IC <- ADS <- U -> CAS -> AAS"
## [62] "IC <- ADS <- U -> CAS <- IT -> AAS"
## [63] "IC <- ADS <- U -> CBS -> CDS -> CAS -> AAS"
## [64] "IC <- ADS <- U -> CBS -> CDS -> CAS <- IT -> AAS"
## [65] "IC <- ADS <- U -> CDS -> CAS -> AAS"
## [66] "IC <- ADS <- U -> CDS -> CAS <- IT -> AAS"
##
## $open
##  [1]  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [25] FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE
## [37]  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE
## [49] FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE
## [61]  TRUE FALSE  TRUE FALSE  TRUE FALSE
```

Crucially, all backdoor paths go through ADS, which then becomes either a fork or a pipe, so all backdoor paths can be closed by conditioning on ADS. Moreover there is only one directed indirect path, it goes through CAS, so we should not condition on it if we are to identify causal effect on attacks mediated by impact on comments (unless we care about the direct effect of IC and IT on AAS, but that's a separate question). This is in line with the adjustment set identified algorithmically, and the same move makes sense when we want to predict CAS.

```
adjustmentSets(dag, exposure = c("IC", "IT"), outcome = "AAS")
```

```
## { ADS }
```

```
adjustmentSets(dag, exposure = c("IC", "IT"), outcome = "CAS")
```

```
## { ADS }
```

It's open season for other variables, and our decision to include them in the model will be guided by information-theoretic criteria of predictive power.

In fact, we will be predicting the difference between attacks before and after, and the difference between comments, before and after. Let's add them to the dag to double-check our selection of variables.

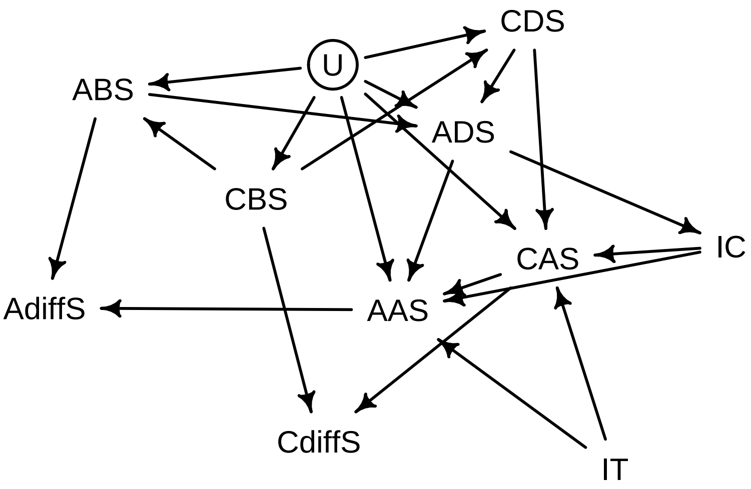
```
dag2 <- dagitty("
  dag{
              CDS -> ADS -> IC  ons
              U [unobserved]
              U -> CBS -> ABS
```

```
                U -> ABS
                U -> CDS -> ADS
                U -> ADS
                U -> CAS -> AAS
                U -> AAS
                IC -> AAS
                IC -> CAS
                IT -> CAS
                IT -> AAS
                CBS -> CDS -> CAS
                ABS -> ADS -> AAS
                ABS -> AdiffS
                AAS -> AdiffS
                CBS -> CdiffS
                CAS -> CdiffS
                }")
set.seed(123)
drawdag(dag2)
adjustmentSets(dag2, exposure = c("IC", "IT"), outcome = "AdiffS")
```

```
## { ADS }
```

```
adjustmentSets(dag2, exposure = c("IC", "IT"), outcome = "CdiffS")
```

```
## { ADS }
```



ons

3 Bayesian models, priors and diagnostics

We will focus on a class of additive models where the outcome variable is normally distributed around the predicted mean, which is a linear function of predictors (possibly with some interactions). To spoil the story, we will end up using a model, whose specification is as follows:

$$
\begin{aligned}
\mathsf{AdiffS} &\sim \mathsf{Norm}(\mu,\sigma)\\
\mu_i &= \alpha + \beta_{\mathsf{ADS}}[\mathsf{group}_i]\times \mathsf{ADS} + \beta_{\mathsf{group}_i} + \beta_{\mathsf{IC}}[\mathsf{group}_i]\times \mathsf{IC}+\\
&\quad + \beta_{\mathsf{ADSIC}}\times \mathsf{ADS}\times \mathsf{IC} + \beta_{\mathsf{CBS}}[\mathsf{group}_i]\times \mathsf{CBS}\\
\alpha &\sim \mathsf{Norm}(0,.3)\\
\beta_{\mathsf{ADS}}[\mathsf{group}_i] &\sim \mathsf{Norm}(0,.3)\\
\beta_{\mathsf{group}_i} &\sim \mathsf{Norm}(0,.3)\\
\beta_{\mathsf{IC}}[\mathsf{group}_i] &\sim \mathsf{Norm}(0,.3)\\
\beta_{\mathsf{ADSIC}} &\sim \mathsf{Norm}(0,.3)\\
\beta_{\mathsf{CBS}}[\mathsf{group}_i] &\sim \mathsf{Norm}(0,.3)
\end{aligned}
$$

That is, we take the resulting mean to be the result of the general average (α) and the impact of the following coefficients: group-specific coefficient for ADS, group coefficient, group-specific coefficient for IC, interaction coefficient for ADS and IC, and group-specific coeffient for CBS. This is plausible prima facie which group a user belongs to might have impact on how attacks during the treatment is related to attacks after, the role of the intervention count, and the role of comments before. Moreover, the levels of agressive behavior displayed by the user during treament might have impact on the role played by the intervention count. Later on we will see that there are information-theoretic reasons to include these interactions.

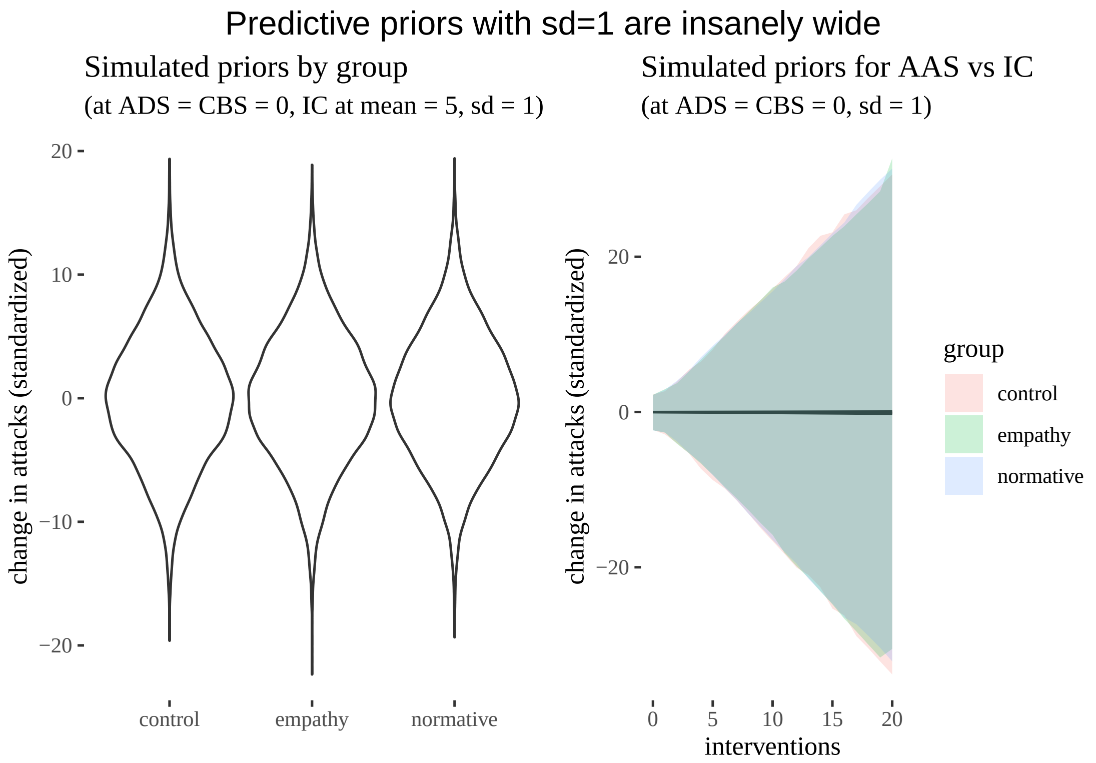
Now for the priors. One might be suspicious of $\sigma = .3$ we employed and suggest using standard normal distributions with $\sigma = 1$ instead. However, a quick prior predictive check shows that this results in insanely wide priors that are competely unrealistic. (For computational reasons, instead of running the simulations, we load pre-compiled models, but we include the code used to build them).

```
# building model with sd=1
# InteractionsModelDiffSD1 <- ulam(
#   alist(
#     AdiffS ~ dnorm( mu, sigma ),
#     mu <- a + bADS[groupID] * ADS +  bIT[groupID] + bIC[groupID] * IC+
#     bADSIC * ADS * IC+ bCBS[groupID] *CBS,
#     a ~ dnorm (0,1),
#     bADS[groupID] ~ dnorm(0,1),
#     bADSIC ~ dnorm(0,1),
#     bCBS[groupID] ~ dnorm(0,1),
#     bIT[groupID] ~ dnorm(0,1),
#     bIC[groupID] ~ dnorm(0,1),
#     sigma  ~ dexp(1)
#   ),
#   data = summaries
# )
#
# saveRDS(InteractionsModelDiffSD1, file = "models/InteractionsModelDiffSD1.rds")
InteractionsModelDiffSD1 <- readRDS(file = "models/InteractionsModelDiffSD1.rds")


#now model with prior sd = .3
# InteractionsModelDiff <- ulam(
#   alist(
#     AdiffS ~ dnorm( mu, sigma ),
#     mu <- a + bADS[groupID] * ADS +  bIT[groupID] + bIC[groupID] * IC +
#     bADSIC * ADS * IC+ bCBS[groupID] *CBS,
#     a ~ dnorm (0,0.3),
#     bADS[groupID] ~ dnorm(0,.3),
#     bADSIC ~ dnorm(0,.3),
#     bCBS[groupID] ~ dnorm(0,.3),
#     bIT[groupID] ~ dnorm(0,.3),
#     bIC[groupID] ~ dnorm(0,.3),
#     sigma  ~ dexp(1)
#   ),
#   data = summaries
# )
```

```
#saveRDS(InteractionsModelDiff, file = "models/InteractionsModelDiff.rds")

InteractionsModelDiff <- readRDS(file = "models/InteractionsModelDiff.rds")

##prior predictive checks sd =1
ADS <- 0
CBS <- 0
groupID <- 1:3
IC <- 5  #mean for interventions in treatment
data <- expand.grid(ADS = ADS,groupID = groupID, CBS = CBS, IC =  IC)
prior <- extract.prior(InteractionsModelDiffSD1, n = 1e4)
mu <- link( InteractionsModelDiffSD1 , post=prior , data=data )
colnames(mu) <- levels(summaries$group)
muLong <- melt(mu)
colnames(muLong) <- c("id", "group", "AdiffS")

priorGroupsSD1 <- ggplot(muLong)+
  geom_violin(aes(x = group, y = AdiffS))+
  theme_tufte()+xlab("")+
  labs(title = "Simulated priors by group",
  subtitle = "(at ADS = CBS = 0, IC at mean = 5, sd = 1)")+
  ylab("change in attacks (standardized)")

ADS <- 0
CBS <- 0
groupID <- 1:3
IC <- 0:20
data <- expand.grid(ADS = ADS,groupID = groupID, CBS = CBS, IC =  IC)

prior <- extract.prior(InteractionsModelDiffSD1, n = 1e4)
```

```
## recompiling to avoid crashing R session
```

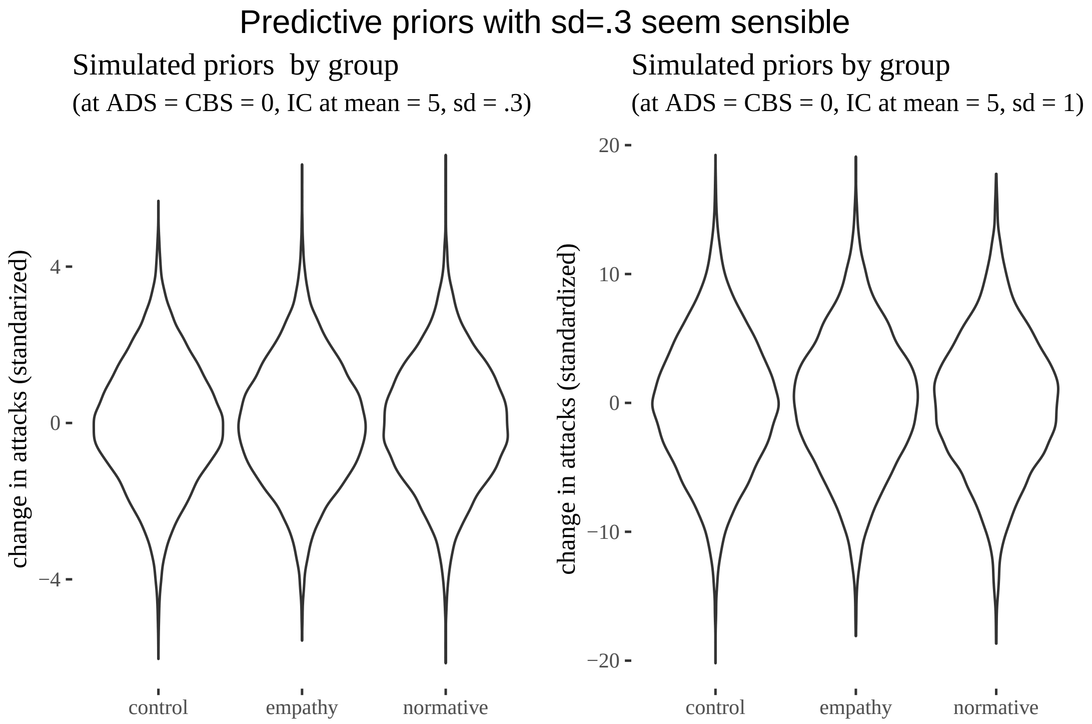
```
mu <- link(InteractionsModelDiffSD1 , post=prior , data=data )
mu.mean <- apply( mu , 2, mean )
mu.HPDI <- data.frame(t(apply( mu , 2 , HPDI )))
priorDF <- cbind(data, mu.mean, mu.HPDI)
priorDF$groupID <- as.factor(groupID)
levels(priorDF$groupID) <- c("control", "empathy", "normative")
colnames(priorDF)[2]<- "group"


priorICSD1  <- ggplot(priorDF, aes(x = IC, y  = mu.mean,  fill = group))+
  geom_line()+geom_ribbon(aes(ymin = X.0.89, ymax = X0.89.), alpha = 0.2)+
  theme_tufte()+ylab("change in attacks (standardized)")+
  labs(title = "Simulated priors for AAS vs IC",
      subtitle = "(at ADS = CBS = 0, sd = 1)")+xlab("interventions")


priorJoint1 <- ggarrange(priorGroupsSD1,priorICSD1, ncol = 2)
priorJoint1Titled <- annotate_figure(priorJoint1,
  top = text_grob("Predictive priors with sd=1 are insanely wide",
                  size = 14))
priorJoint1Titled
```



Some experimentation leads to the value of $\sigma = .3$, which leads to the following priors:

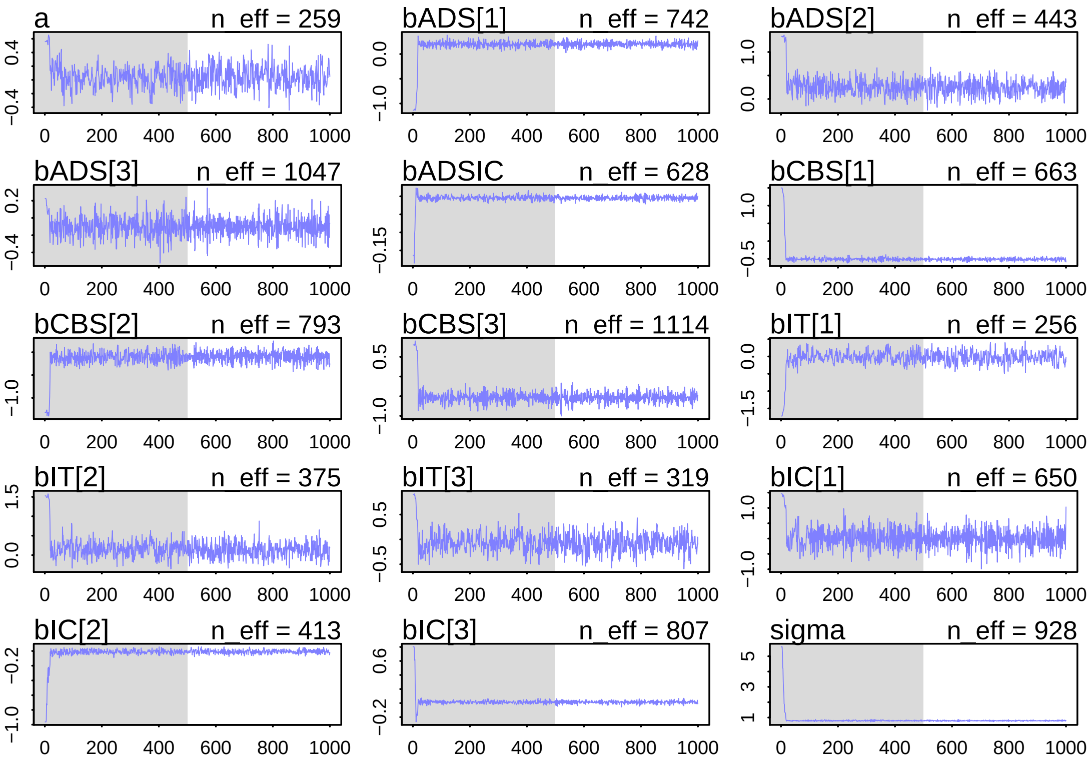
```
#prior predictive check sd =.3
ADS <- 0
CBS <- 0
groupID <- 1:3
IC <- 5  #mean for interventions in treatment
data <- expand.grid(ADS = ADS,groupID = groupID, CBS = CBS, IC =  IC)
prior <- extract.prior(InteractionsModelDiff, n = 1e4)
mu <- link(InteractionsModelDiff , post=prior , data=data )
colnames(mu) <- levels(summaries$group)
muLong <- melt(mu)
colnames(muLong) <- c("id", "group", "AdiffS")
head(muLong)

priorGroupSD03 <- ggplot(muLong)+
  geom_violin(aes(x = group, y = AdiffS))+theme_tufte()+
  xlab("")+
  labs(title = "Simulated priors  by group",
  subtitle = "(at ADS = CBS = 0, IC at mean = 5, sd = .3)")+
  ylab("change in attacks (standarized)")

ADS <- 0
CBS <- 0
groupID <- 1:3
IC <- 5  #mean for interventions in treatment
data <- expand.grid(ADS = ADS,groupID = groupID, CBS = CBS, IC =  IC)
prior <- extract.prior(InteractionsModelDiffSD1, n = 1e4)
mu <- link( InteractionsModelDiffSD1 , post=prior , data=data )
colnames(mu) <- levels(summaries$group)
muLong <- melt(mu)
colnames(muLong) <- c("id", "group", "AdiffS")
head(muLong)

priorICSD03 <- ggplot(muLong)+
  geom_violin(aes(x = group, y = AdiffS))+
  theme_tufte()+xlab("")+
  labs(title = "Simulated priors by group",
  subtitle = "(at ADS = CBS = 0, IC at mean = 5, sd = 1)")+
  ylab("change in attacks (standardized)")

priorJoint03 <- ggarrange(priorGroupSD03,priorICSD03, ncol = 2)
```

```
priorJoint03Titled <- annotate_figure(priorJoint03,
  top = text_grob("Predictive priors with sd=.3 seem sensible",
                  size = 14))
priorJoint03Titled
```



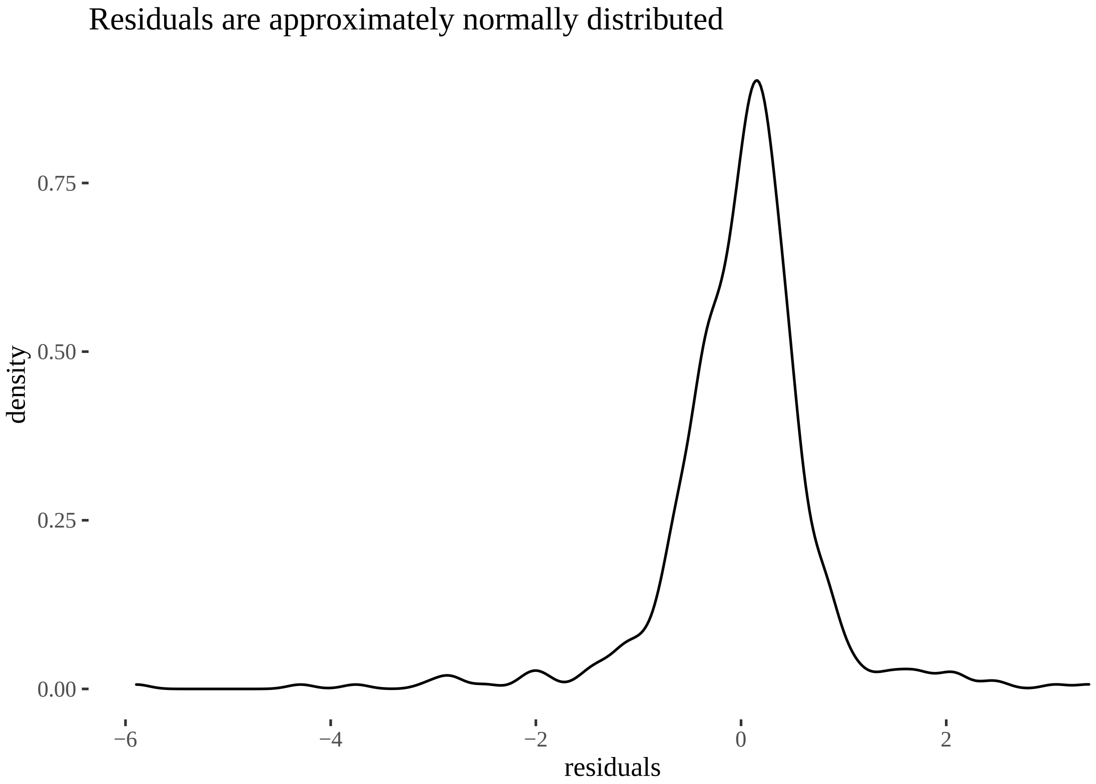
Now, some model diagnostics before we move on. What we are witnessing is (1) stationarity (the chains stay mostly in the most probable regions), (2) good mixing (they explore a range of options in the beginning), and (3) convergence (they stabilize as they progress).

```
traceplot( InteractionsModelDiff )
```



Finally, let's inspect the distribution of residuals. That is, we calculate all predictions, their distance from the actual values, and inspect the distribution of the distances:

```
mu <- link(InteractionsModelDiff)
mu_mean <- apply( mu , 2 , mean )
mu_resid <- summaries$AdiffS - mu_mean
ggplot()+geom_density(aes(x = mu_resid))+theme_tufte()+
  ggtitle("Residuals are approximately normally distributed")+xlab("residuals")
```



4 Model selection

How did we get to this fairly complicated model though? Once preliminary causal considerations guided our restrictions on variable selection, we proceed by building models of increasing complexity, and comparing them in terms of Widely Acceptable Information Criterion. The models differ mostly in the underlying linear formulae. For computational ease we will here use quadratic approximations, while in the final analysis we will deploy Hamiltionian Monte Carlo. The names are meant to decode the model structure: the predictors are listed before dashes, whereas interactions are listed after dashes.

$$\mu_i = \alpha \tag{Null}$$

$$\mu_i = \alpha + \beta_{\mathsf{ADS}} \times \mathsf{ADS} \tag{ADS}$$

$$\mu_i = \alpha + \beta_{\mathsf{ADS}} \times \mathsf{ADS} + \beta_{\mathsf{IC}} \times \mathsf{IC} \tag{ADSIC}$$

$$\mu_i = \beta_{\mathsf{group}[i]} \tag{IT}$$

$$\mu_i = \alpha + \beta_{\mathsf{ADS}} \times \mathsf{ADS} + \beta_{\mathsf{group}[i]} \tag{ADSIT}$$

$$\mu_i = \alpha + \beta_{\mathsf{ADS}} \times \mathsf{ADS} + \beta_{\mathsf{group}[i]} + \beta_{\mathsf{IC}} \times \mathsf{IC} \tag{ADSITIC}$$

$$\mu_i = \alpha + \beta_{\mathsf{ADS}} \times \mathsf{ADS} + \beta_{\mathsf{group}[i]} + \beta_{\mathsf{IC}} \times \mathsf{IC} + \beta_{\mathsf{ADSIC}} \times \mathsf{ADS} \times \mathsf{IC} \tag{ADSITIC-ADSIC}$$

$$\begin{aligned}\mu_i = \alpha + \beta_{\mathsf{ADS}}[\mathsf{group}_i] \times \mathsf{ADS} + \beta_{\mathsf{group}[i]} + \\ + \beta_{\mathsf{IC}} \times \mathsf{IC} + \beta_{\mathsf{ADSIC}} \times \mathsf{ADS} \times \mathsf{IC}\end{aligned} \tag{ADSITIC-ADSIC-ADSIT}$$

$$\mu_i = \alpha + \beta_{\mathsf{ADS}}[\mathsf{group}_i] \times \mathsf{ADS} + \beta_{\mathsf{group}[i]} \tag{ADSIT-ADSIT}$$

$$\begin{aligned}\mu_i = \alpha + \beta_{\mathsf{ADS}}[\mathsf{group}_i] \times \mathsf{ADS} + \beta_{\mathsf{group}[i]} + \beta_{\mathsf{IC}}[\mathsf{group}_i] \times \mathsf{IC} + \\ + \beta_{\mathsf{ADSIC}} \times \mathsf{ADS} \times \mathsf{IC}\end{aligned} \tag{ADSITIC-ADSIT-ITIC-ADSIC}$$

$$\begin{aligned}\mu_i = \alpha + \beta_{\mathsf{ADS}}[\mathsf{group}_i] \times \mathsf{ADS} + \beta_{\mathsf{group}[i]} + \beta_{\mathsf{IC}}[\mathsf{group}_i] \times \mathsf{IC} + \\ + \beta_{\mathsf{CBS}} \times \mathsf{CBS} + \beta_{\mathsf{ADSIC}} \times \mathsf{ADS} \times \mathsf{IC}\end{aligned} \tag{ADSITICCBS-ITIC-ADSIC}$$

$$\begin{aligned}\mu_i = \alpha + \beta_{\mathsf{ADS}}[\mathsf{group}_i] \times \mathsf{ADS} + \beta_{\mathsf{group}_i} + \beta_{\mathsf{IC}}[\mathsf{group}_i] \times \mathsf{IC} + \\ + \beta_{\mathsf{ADSIC}} \times \mathsf{ADS} \times \mathsf{IC} + \beta_{\mathsf{CBS}}[\mathsf{group}_i] \times \mathsf{CBS}\end{aligned} \tag{Final}$$

$$\begin{aligned}\mu_i = \alpha + \beta_{\mathsf{ADS}}[\mathsf{group}_i] \times \mathsf{ADS} + \beta_{\mathsf{group}_i} + \beta_{\mathsf{IC}}[\mathsf{group}_i] \times \mathsf{IC} + && \text{(tooFAr)} \\ + \beta_{\mathsf{ADSIC}} \times \mathsf{ADS} \times \mathsf{IC} + \beta_{\mathsf{CBS}}[\mathsf{group}_i] \times \mathsf{CBS} + \beta_{\mathsf{CBSIC}} \times \mathsf{CBS} \times \mathsf{IC} && \text{(1)}\end{aligned}$$

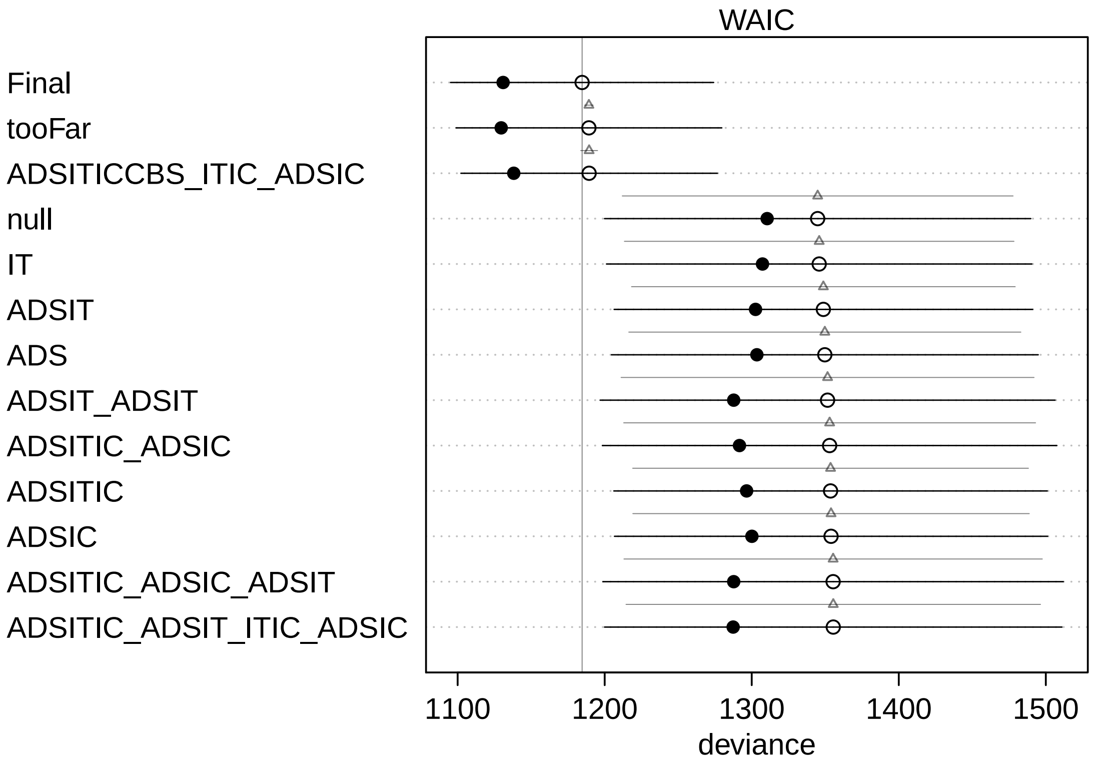
```
null <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu ~ dnorm (0,0.3),
    sigma  ~ dexp(1)
  ),
  data = summaries
)

ADS <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu <-  a + bADS * ADS,
    a ~ dnorm (0,0.3),
    bADS ~ dnorm(0,0.3),
    sigma  ~ dexp(1)
  ),
  data = summaries
)

ADSIC <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu <-  a + bADS * ADS+ bIC * IC,
    a ~ dnorm (0,0.3),
    bADS ~ dnorm(0,0.3),
    bIC ~ dnorm(0,0.3),
    sigma  ~ dexp(1)
  ),
  data = summaries
)


IT <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu <-  bIT[groupID] ,
    bIT[groupID] ~ dnorm(0,.3),
    sigma  ~ dexp(1)
  ),
  data = summaries
)


ADSIT <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu <- a + bADS * ADS +  bIT[groupID],
    a ~ dnorm (0,0.3),
    bADS ~ dnorm(0,.3),
    bIT[groupID] ~ dnorm(0,.3),
    sigma  ~ dexp(1)
  ),
  data = summaries
)


ADSITIC <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu <- a + bADS * ADS +  bIT[groupID] + bIC * IC,
    a ~ dnorm (0,0.3),
    bADS ~ dnorm(0,.3),
    bIT[groupID] ~ dnorm(0,.3),
    bIC ~ dnorm(0,.3),
    sigma  ~ dexp(1)
  ),
  data = summaries
)


ADSITIC_ADSIC <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu <- a + bADS * ADS +  bIT[groupID] + bIC * IC + bADSIC * ADS * IC,
    a ~ dnorm (0,0.3),
    bADS ~ dnorm(0,.3),
    bADSIC ~ dnorm(0,.3),
    bIT[groupID] ~ dnorm(0,.3),
    bIC ~ dnorm(0,.3),
    sigma  ~ dexp(1)
  ),
  data = summaries
)


ADSITIC_ADSIC_ADSIT <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu <- a + bADS[groupID] * ADS +  bIT[groupID] + bIC * IC + bADSIC * ADS * IC,
    a ~ dnorm (0,0.3),
    bADS[groupID] ~ dnorm(0,.3),
    bADSIC ~ dnorm(0,.3),
    bIT[groupID] ~ dnorm(0,.3),
    bIC ~ dnorm(0,.3),
    sigma  ~ dexp(1)
  ),
  data = summaries
)


ADSIT_ADSIT <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu <- a + bADS[groupID] * ADS +  bIT[groupID] ,
    a ~ dnorm (0,0.3),
    bADS[groupID] ~ dnorm(0,.3),
    #bADSIC ~ dnorm(0,.5),
    bIT[groupID] ~ dnorm(0,.3),
    #bIC ~ dnorm(0,.5),
    sigma  ~ dexp(1)
  ),
  data = summaries
)


ADSITIC_ADSIT_ITIC_ADSIC <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu <- a + bADS[groupID] * ADS +  bIT[groupID] + bIC[groupID] * IC +
      bADSIC * ADS * IC,
    a ~ dnorm (0,0.3),
    bADS[groupID] ~ dnorm(0,.3),
    bADSIC ~ dnorm(0,.3),
    bIT[groupID] ~ dnorm(0,.3),
    bIC[groupID] ~ dnorm(0,.3),
    sigma  ~ dexp(1)
  ),
  data = summaries
)


ADSITICCBS_ITIC_ADSIC <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu <- a + bADS[groupID] * ADS +  bIT[groupID] + bIC[groupID] * IC +
      bADSIC * ADS * IC+ bCBS *CBS,
    a ~ dnorm (0,0.3),
    bADS[groupID] ~ dnorm(0,.3),
    bADSIC ~ dnorm(0,.3),
    bCBS ~ dnorm(0,.3),
    bIT[groupID] ~ dnorm(0,.3),
    bIC[groupID] ~ dnorm(0,.3),
    sigma  ~ dexp(1)
  ),
  data = summaries
)


Final <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu <- a + bADS[groupID] * ADS +  bIT[groupID] + bIC[groupID] * IC +
      bADSIC * ADS * IC+ bCBS[groupID] *CBS,
    a ~ dnorm (0,0.3),
    bADS[groupID] ~ dnorm(0,.3),
    bADSIC ~ dnorm(0,.3),
    bCBS[groupID] ~ dnorm(0,.3),
    bIT[groupID] ~ dnorm(0,.3),
    bIC[groupID] ~ dnorm(0,.3),
    sigma  ~ dexp(1)
  ),
  data = summaries
)


tooFar <- quap(
  alist(
    AdiffS ~ dnorm( mu, sigma ),
    mu <- a + bADS[groupID] * ADS +  bIT[groupID] + bIC[groupID] * IC +
      bADSIC * ADS * IC+ bCBS[groupID] *CBS + bCBSIC * CBS * IC,
    a ~ dnorm (0,0.3),
    bADS[groupID] ~ dnorm(0,.3),
    bADSIC ~ dnorm(0,.3),
    bCBS[groupID] ~ dnorm(0,.3),
    bIT[groupID] ~ dnorm(0,.3),
    bIC[groupID] ~ dnorm(0,.3),
     bCBSIC ~ dnorm(0, .3),
    sigma  ~ dexp(1)
  ),
  data = summaries
```

```
)
```

```
comparison<- compare(null,ADS,ADSIC,IT,ADSIT,ADSITIC,ADSITIC_ADSIC,
                     ADSITIC_ADSIC_ADSIT,ADSIT_ADSIT,ADSITIC_ADSIT_ITIC_ADSIC,
                     ADSITICCBS_ITIC_ADSIC,Final, tooFar)
mykable(data.frame(comparison ))
```

	WAIC	SE	dWAIC	dSE	pWAIC	weight
Final	1184.641	89.47387	0.000000	NA	26.85809	0.8393664
tooFar	1189.257	90.39801	4.616272	2.884162	29.81499	0.0834720
ADSITICCBS_ITIC_ADSIC	1189.414	87.19709	4.773490	5.762571	25.62296	0.0771616
null	1344.835	144.95033	160.193855	132.894118	17.16957	0.0000000
IT	1345.879	144.64185	161.237934	132.478842	19.27433	0.0000000
ADSIT	1348.714	142.29377	164.072877	130.485213	23.06652	0.0000000
ADS	1349.706	145.13624	165.065051	133.323101	23.10634	0.0000000
ADSIT_ADSIT	1351.571	154.69385	166.930496	140.429304	31.91506	0.0000000
ADSITIC_ADSIC	1352.972	154.55405	168.331385	140.054704	30.65579	0.0000000
ADSITIC	1353.603	147.38084	168.962571	134.581071	28.54365	0.0000000
ADSIC	1353.939	147.29717	169.298374	134.821036	26.91553	0.0000000
ADSITIC_ADSIC_ADSIT	1355.353	156.53444	170.712149	142.268677	33.80858	0.0000000
ADSITIC_ADSIT_ITIC_ADSIC	1355.442	155.55820	170.800602	140.950680	34.03861	0.0000000

```
plot(comparison)
```



The three models that stand out differ in including CBS as a predictor. Moreover the final model includes an interaction between treatment group and CBS. Adding a further interaction between CBS and IC takes us too far. *WAIC*-based weighing assigns the weight of 83% to the final model, and the standard errors for the difference in WAIC for the top three models is fairly low, so we will employ the top model (Final) in further analyses.

x	x	x	x	x	x
0.0414089	0.1562753	-0.2037740	0.2796946	228.1593	0.9980513
0.1979211	0.0568494	0.1086423	0.2907669	768.4216	1.0013137
0.2484756	0.1546028	0.0036955	0.4942282	526.1494	0.9995921
-0.1097667	0.1014546	-0.2678764	0.0628934	418.0876	0.9980791
-0.0042060	0.0051643	-0.0122338	0.0040408	361.1195	0.9983620
-0.5142899	0.0414267	-0.5776731	-0.4463925	722.3116	0.9991926
-0.0919148	0.1245614	-0.2965873	0.1059034	671.6673	0.9998084
-0.5302840	0.1088835	-0.7023272	-0.3593766	823.7157	0.9980583
-0.0212337	0.1630179	-0.2603815	0.2295494	219.0190	0.9980137
0.1482970	0.1857528	-0.1246834	0.4367149	304.7859	0.9980345
-0.0686005	0.1776783	-0.3448157	0.2099344	295.3412	0.9980482
-0.0037630	0.3161707	-0.5084313	0.4981378	829.2326	1.0011451
-0.0117801	0.0256771	-0.0529960	0.0290304	488.1809	0.9994090
0.0118155	0.0185521	-0.0192521	0.0412303	612.3810	0.9986804
0.7945882	0.0277774	0.7535960	0.8434971	693.5845	0.9982674

5 Inspecting the model and effect sizes

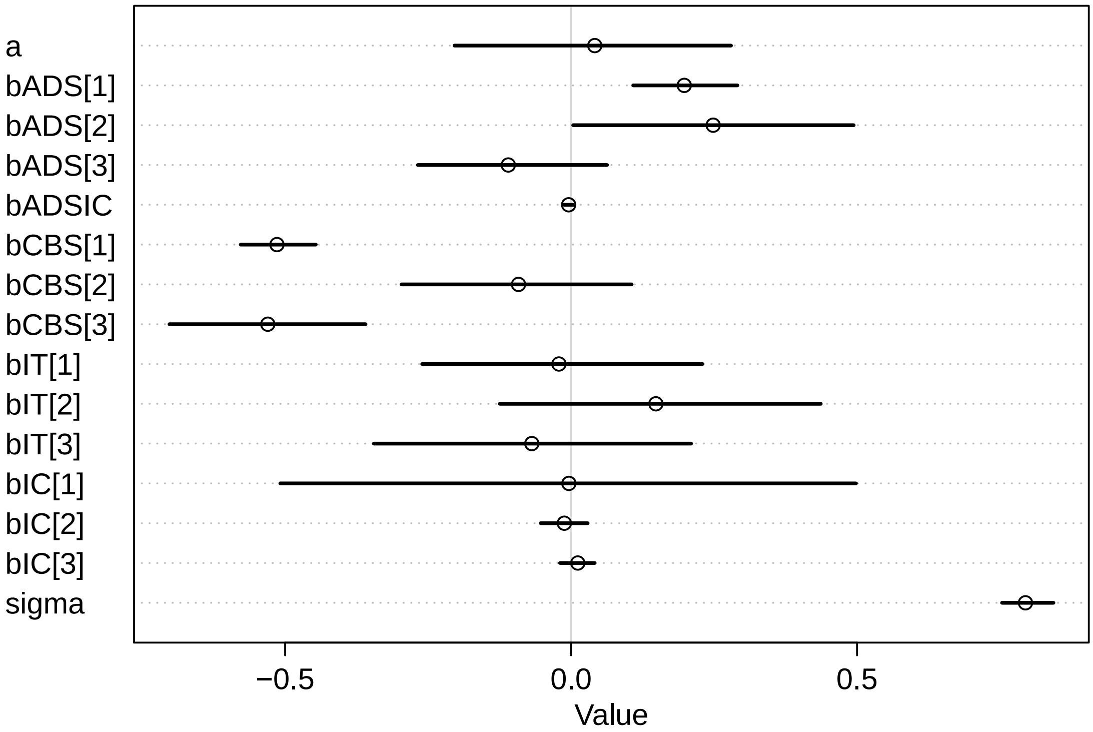
We start by using the Final model formula to build a model, this time using Hamiltionian Monte Carlo. We leave the code commented out and load a pre-compiled model for computational convenience

```
# FinalHMC <- ulam(
#   alist(
#     AdiffS ~ dnorm( mu, sigma ),
#     mu <- a + bADS[groupID] * ADS +  bIT[groupID] +
#     bIC[groupID] * IC + bADSIC * ADS * IC+
#     bCBS[groupID] *CBS,
#     a ~ dnorm (0,0.3),
#     bADS[groupID] ~ dnorm(0,.3),
#     bADSIC ~ dnorm(0,.3),
#     bCBS[groupID] ~ dnorm(0,.3),
#     bIT[groupID] ~ dnorm(0,.3),
#     bIC[groupID] ~ dnorm(0,.3),
#     sigma  ~ dexp(1)
#   ),
#   data = summaries
# )
# saveRDS(FinalHMC, file = "models/FinalHMC.rds")

FinalHMC <- readRDS(file = "models/FinalHMC.rds")
```

First, let's take a look at the best model coefficients.

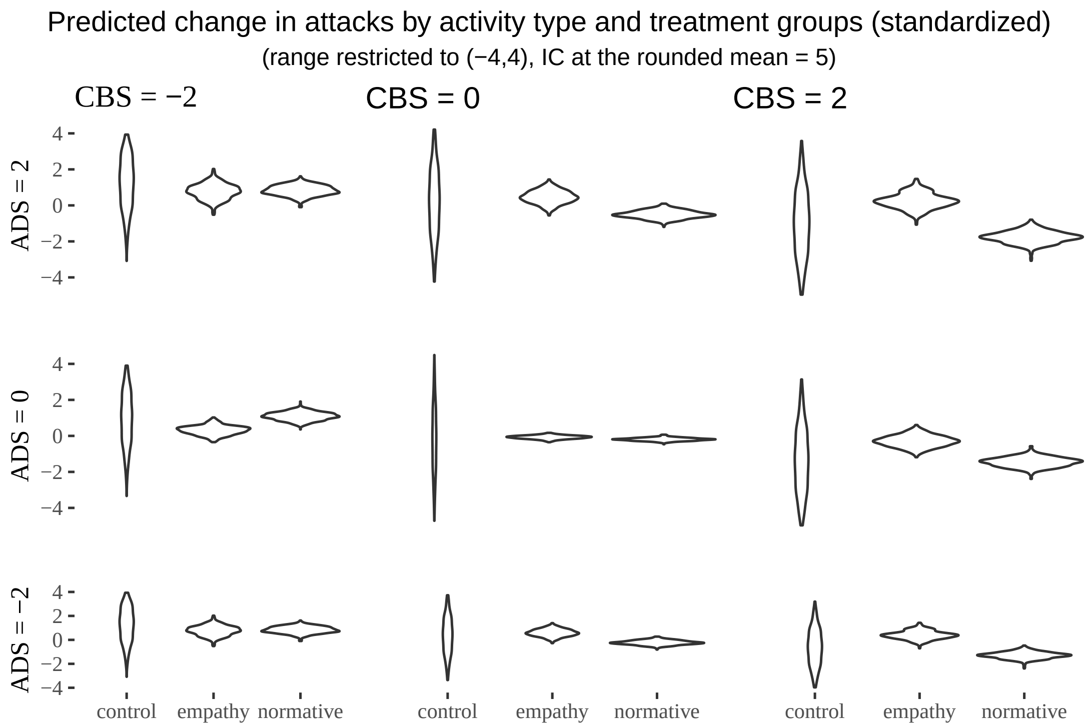
```
precis(FinalHMC, depth = 2)
```

```
plot(precis(FinalHMC, depth = 2))
```



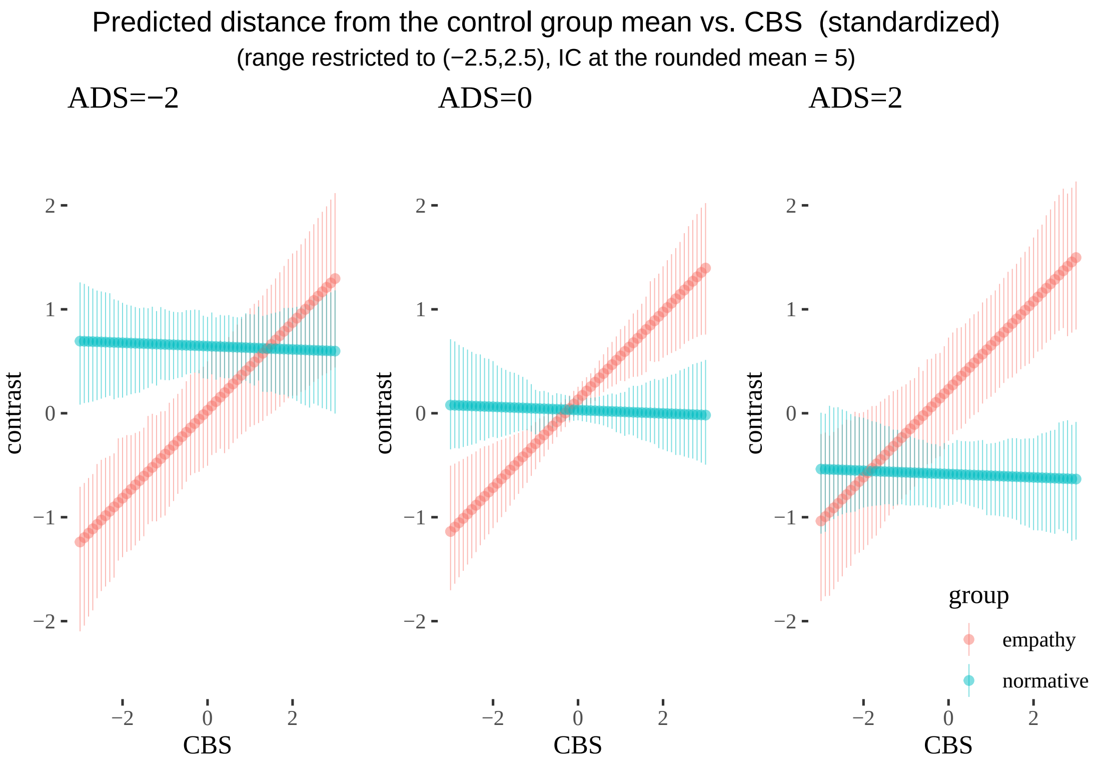
These, however, are notoriously hard to interpret in models with interactions. For this reason, it is better to plot predicted effects for various combinations of predictors.

```
visGroup <- function (model, ADS, CBS, xmin =2, ymax = -3)
{
groupID <- 1:3
IC <- 5
data <- expand.grid(ADS = ADS,groupID = groupID, CBS = CBS, IC =  IC)
posterior <- extract.samples(model, n = 1e5)
mu <- link( model, data=data )
colnames(mu) <- levels(summaries$group)
muLong <- melt(mu)
colnames(muLong) <- c("id", "group", "AdiffS")
means <-  round(apply(mu , 2 , mean ), 2)
mu_HPDI <- round(apply( mu , 2 , HPDI ),2)
means <- as.data.frame(means)
means$group <- rownames(means)
rownames(means) <- NULL
meansDisp <- cbind(means,t(as.data.frame(mu_HPDI)))
meansDisp <- meansDisp[,c(1,3,4)]

plot <- ggplot(muLong)+geom_violin(aes(x = group, y = AdiffS), alpha = 0.2)+
  xlab("")+
  labs(title = paste("ADS=", ADS, ", CBS=",  CBS,  sep = ""))+
  theme_tufte()+ylim(c(-4,4))
#+   annotation_custom(tableGrob(meansDisp), xmin=xmin,  ymax=ymax)
return(plot)
}


visGroupA2C_2 <- visGroup(model = FinalHMC, ADS = 2,CBS = -2)
visGroupA2C0 <- visGroup(model = FinalHMC, ADS = 2,CBS = 0 )
visGroupA2C2 <- visGroup(model = FinalHMC, ADS = 2,CBS = 2)

visGroupA0C_2 <- visGroup(model = FinalHMC, ADS = 0,CBS = -2 )
visGroupA0C0 <- visGroup(model = FinalHMC, ADS = 0,CBS = 0 )
visGroupA0C2 <-  visGroup(model = FinalHMC, ADS = 0,CBS = 2)

visGroupA2C_2 <-  visGroup(model = FinalHMC, ADS = 2,CBS = -2 )
visGroupA2C0 <- visGroup(model = FinalHMC, ADS = 2,CBS = 0 )
visGroupA2C2 <- visGroup(model = FinalHMC, ADS = 2,CBS = 2 )

visGroupJoint <- ggarrange(visGroupA2C_2+removeX + ggtitle("CBS = -2")+ylab("ADS = 2") , visGroupA2C0+th
          visGroupA0C_2+removeX+ylab("ADS = 0")+ggtitle(""), visGroupA0C0+theme_void()+ggtitle(""), visG
          visGroupA2C_2+ylab("ADS = -2")+ggtitle(""), visGroupA2C0+removeY+ggtitle(""), visGroupA2C2+rem


visGroupJoint2 <- annotate_figure(visGroupJoint,
  top = text_grob("(range restricted to (-4,4), IC at the rounded mean = 5)",
                  size = 10))
visGroupJoint3 <- annotate_figure(visGroupJoint2,
  top = text_grob("Predicted change in attacks by activity type and treatment groups (standardized)",
                  size = 12))

visGroupJoint3
```



To gain more clarity, let’s look at predicted contrasts, here understood as distances from the control group mean, by activity types, first versus CBS, then versus ADS.

```
visContrastsCBS <- function(model = FinalHMC, ADS = ADS , IC =  5,
                            CBS = seq(-3,3,by  = 0.1))
  {
  groupID <- 1:3
  data <- expand.grid(ADS, groupID, IC , CBS)
  colnames(data) <- c("ADS", "groupID", "IC", "CBS")
  posterior <- extract.samples(model, n = 1e5)
  link( model, data=data )
  mu <- link( model, data=data )
  means <-  round(apply(mu , 2 , mean ), 4)
  HPDIs <- round(apply( mu , 2 , HPDI ),4)
  visContrast <- cbind(data,means,t(as.data.frame(HPDIs)))

  ones <- 3 * (1:(nrow(visContrast)/3))-2
  twos <- 3 * (1:(nrow(visContrast)/3))-1
  threes <- 3 * (1:(nrow(visContrast)/3))

  colnames(visContrast)[c(6,7)] <- c("low", "high")
  contrast <- numeric(nrow(visContrast))
  cLow <- numeric(nrow(visContrast))
  cHigh <- numeric(nrow(visContrast))
  for(i in threes){
  contrast[i] <- visContrast$means[i] - visContrast$means[i-2]
  }
  for(i in twos){
  contrast[i] <- visContrast$means[i] - visContrast$means[i-1]
  }
  visContrast$contrast <- contrast
  visContrast$shift <-  visContrast$contrast - visContrast$means
  for(i in ones){
  visContrast$shift[i] <- 0
  }
  visContrast$cLow <- visContrast$low + visContrast$shift
  visContrast$cHigh <- visContrast$high + visContrast$shift

  visContrast$group = rep(c("control", "empathy", "normative"),
                          nrow(visContrast)/3)

  visContrastTreatment <- visContrast[groupID !=1,]

  return(ggplot(visContrastTreatment, aes(x = CBS, y = contrast, color = group ))+
            geom_pointrange(mapping =
        aes(ymin = cLow, ymax = cHigh), size = .2, alpha = .5) +theme_tufte())
}


visContrastCBSJoint <- ggarrange(visContrastsCBS(FinalHMC,ADS = -2)+
      ggtitle("ADS=-2")+ylim(c(-2.5,2.5))+ scale_color_discrete(guide=FALSE),
          visContrastsCBS(FinalHMC,ADS = 0)+ggtitle("ADS=0")+
        ylim(c(-2.5,2.5))+ scale_color_discrete(guide=FALSE),
        visContrastsCBS(FinalHMC,ADS = 2)+ggtitle("ADS=2")+
        ylim(c(-2.5,2.5))+ theme(legend.position = c(0.75, 0.1)), ncol = 3)

visContrastCBSJoint2 <- annotate_figure(visContrastCBSJoint,
      top = text_grob("(range restricted to (-2.5,2.5), IC at the rounded mean = 5)",
                                size = 10))
visContrastCBSJoint3 <- annotate_figure(visContrastCBSJoint2,
top = text_grob("Predicted distance from the control group mean vs. CBS  (standardized)",
                                size = 12))

visContrastCBSJoint3
```



References