

ModelReportAll

Fiona Zhu

4/4/2020

Load the packages

```
options(mc.cores = parallel::detectCores()-1)
rstan_options (auto_write=TRUE)

#customize theme
theme_new <- theme_bw() +
  theme(panel.border = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.line = element_line(colour = "black"),
        strip.background = element_rect(color = "white", fill = "white"),
        panel.grid = element_blank())
```

Define the Rstan models and functions to plot

Definition of the function to merge rstan result data

To further uncover the underlying structure of the prior, we hypothesize that in addition to two local priors, there is a general global prior. Participants may combine both local and global priors for the final reproduction. There are multiple possibilities for integrating those local and global priors with the sensory inputs. For examples, the sensory input could first integrate with the local prior, and then integrate with the global prior (a hierarchical local-global model, see Figure 7B in D1 proposal).

Our Hypothesis

- H2: A hierarchical local-global model In H2, global and local priors are integrated first. That means local prior is integrated with sensory input firstly, then global prior integrates with sensory input.

The sensory input (D_s) first integrates with the local prior (P_L) to a posterior (D_L), which further integrates with the global prior (P_G) to generate a final posterior for reproduction (D_r).

- H3: Global prior (the dual integration model)

integration of local priors firstly, then integration of the global prior

Both local and global priors independently integrate with the sensory inputs to generate two posteriors (D_L) and (D_G), the latter two are combined together for reproduction (D_r).

compute parameters for all models

```
# function to run models on lrz cluster parallely
runModel<- function(AllExpData, modelname, rltnname){
  expList <- unique(AllExpData$Exp)
  subList <- unique(AllExpData$NSub)
  # H2_V1 There were 135 divergent transitions after warmup.
  sub_exp_dat <- list()
  for (expName in expList) {
    data_exp <- AllExpData %>% dplyr::filter(valid == 1 & Exp == expName)
    data_exp$model <- modelname
```

```

sub_exp <- split(data_exp, data_exp$NSub) # split data for each subject
for (subNo in subList) {
  sub_exp_dat <- list.append(sub_exp_dat, sub_exp[[subNo]])
}
}

rlt <- runModelcluster(sub_exp_dat)
saveRDS(rlt, file = paste0(getwd(), "/", rstanmodelPath, "/rlt/rlt_", rltname, ".rds"))
}

# execute the model running
modellist_V3 <- c('H2_V3', 'H3_V3')

modellist <- modellist_V3
if (runModels == TRUE){
  for(model in modellist){
    runModel(AllExpData, model, model)
  }
}

```

Merge the Result data

To preprocess the model result data, and merge different model version data together.

```

#function to merge the model result data

funMergeMRlt <- function(modellist, path, version){
  rltfilename <- {}
  rltfilename <- c(rltfilename, paste0("rlt_", modellist, ".rds"))
  AllDat_Bayparlist <- {}
  AllDat_predY_mix <- {}
  AllDat_predY_s <- {}
  AllDat_predY_l <- {}
  modelResultAll <- list()
  merge.data.all <- {}
  merge.data <- readRDS(paste0(getwd(), "/", path, "/rlt/", rltfilename[1]))
  for (i in 1:length(merge.data)){
    modelResultAll <- list.append(modelResultAll, merge.data[[i]])
    AllDat_Bayparlist <- rbind(AllDat_Bayparlist, merge.data[[i]]$Baypar)
    AllDat_predY_mix <- rbind(AllDat_predY_mix, merge.data[[i]]$PredY_mix_list)
    AllDat_predY_s <- rbind(AllDat_predY_s, merge.data[[i]]$PredY_s_list)
    AllDat_predY_l <- rbind(AllDat_predY_l, merge.data[[i]]$PredY_l_list)
  }

  if (length(rltfilename) >= 2) {
    for (i in 2:length(rltfilename)){
      new.data = readRDS(paste0(getwd(), "/", path, "/rlt/", rltfilename[i]))
      for (i in 1:length(new.data)){
        modelResultAll <- list.append(modelResultAll, new.data[[i]])
        AllDat_Bayparlist <- rbind(AllDat_Bayparlist, new.data[[i]]$Baypar)
        AllDat_predY_mix <- rbind(AllDat_predY_mix, new.data[[i]]$PredY_mix_list)
        AllDat_predY_s <- rbind(AllDat_predY_s, new.data[[i]]$PredY_s_list)
        AllDat_predY_l <- rbind(AllDat_predY_l, new.data[[i]]$PredY_l_list)
      }
    }
  }
}

```

```

}
write.csv(AllDat_Bayparlist, paste0(getwd(), "/", path, "/rlt/AllDat_Bayparlist_", version, ".csv"))
write.csv(AllDat_predY_mix, paste0(getwd(), "/", path, "/rlt/AllDat_predY_mix_", version, ".csv"))
write.csv(AllDat_predY_s, paste0(getwd(), "/", path, "/rlt/AllDat_predY_s_", version, ".csv"))
write.csv(AllDat_predY_l, paste0(getwd(), "/", path, "/rlt/AllDat_predY_l_", version, ".csv"))
saveRDS(modelResultAll, file = paste0(getwd(), "/", path, "/rlt/modelResultAll_", version, ".rds"))
}

```

load result data (V3)

Analysis on the Rstan model parameters

Parameters

```

m_Baypar <- dplyr::group_by(AllDat_Bayparlist3, Nsub, Exp, model) %>%
  dplyr::summarize(m_wf1 = mean(wf1), m_wf2 = mean(wf2),
                  m_wf3 = mean(wf3), m_mu1 = mean(mu1),
                  m_mu2 = mean(mu2), m_mu3 = mean(mu3),
                  m_sig_m_square = mean(sig_m_square))
m_Baypar

## # A tibble: 64 x 10
## # Groups:   Nsub, Exp [32]
##   Nsub Exp  model m_wf1 m_wf2 m_wf3 m_mu1 m_mu2 m_mu3 m_sig_m_square
##   <dbl> <chr> <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     1 Exp1  H2_V3 0.216 0.403 1.35 0.607 1.71 0.779 0.116
## 2     1 Exp1  H3_V3 0.215 0.402 1.38 0.607 1.71 0.770 0.116
## 3     1 Exp2  H2_V3 0.285 0.650 1.09 0.675 1.14 1.10 0.117
## 4     1 Exp2  H3_V3 0.305 0.644 1.08 0.679 1.13 1.12 0.117
## 5     2 Exp1  H2_V3 0.283 0.440 1.18 0.549 1.52 1.59 0.0915
## 6     2 Exp1  H3_V3 0.283 0.442 1.18 0.549 1.52 1.59 0.0914
## 7     2 Exp2  H2_V3 0.0908 0.220 1.29 0.654 1.43 1.40 0.137
## 8     2 Exp2  H3_V3 0.0906 0.221 1.29 0.653 1.43 1.39 0.136
## 9     3 Exp1  H2_V3 0.215 0.506 1.51 0.588 1.15 1.36 0.126
## 10    3 Exp1  H3_V3 0.215 0.507 1.52 0.587 1.15 1.35 0.126
## # ... with 54 more rows

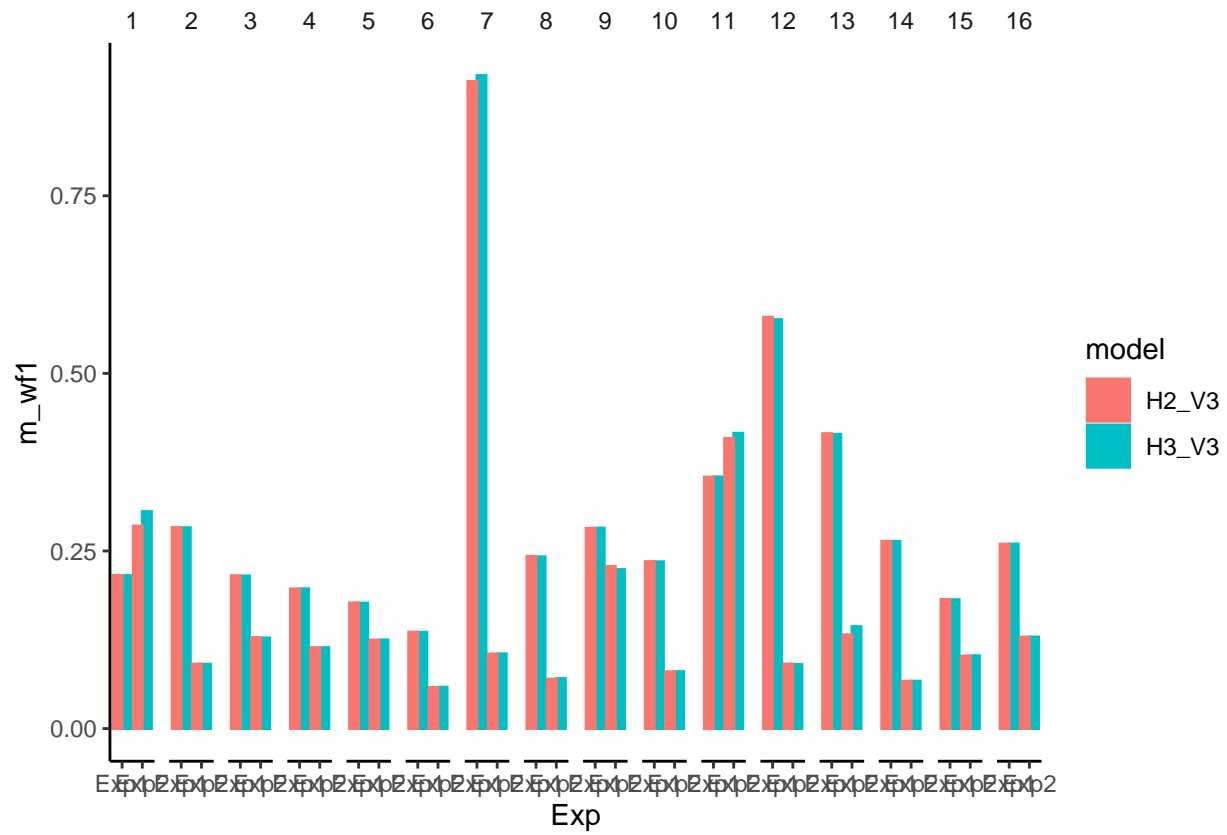
```

wf in models

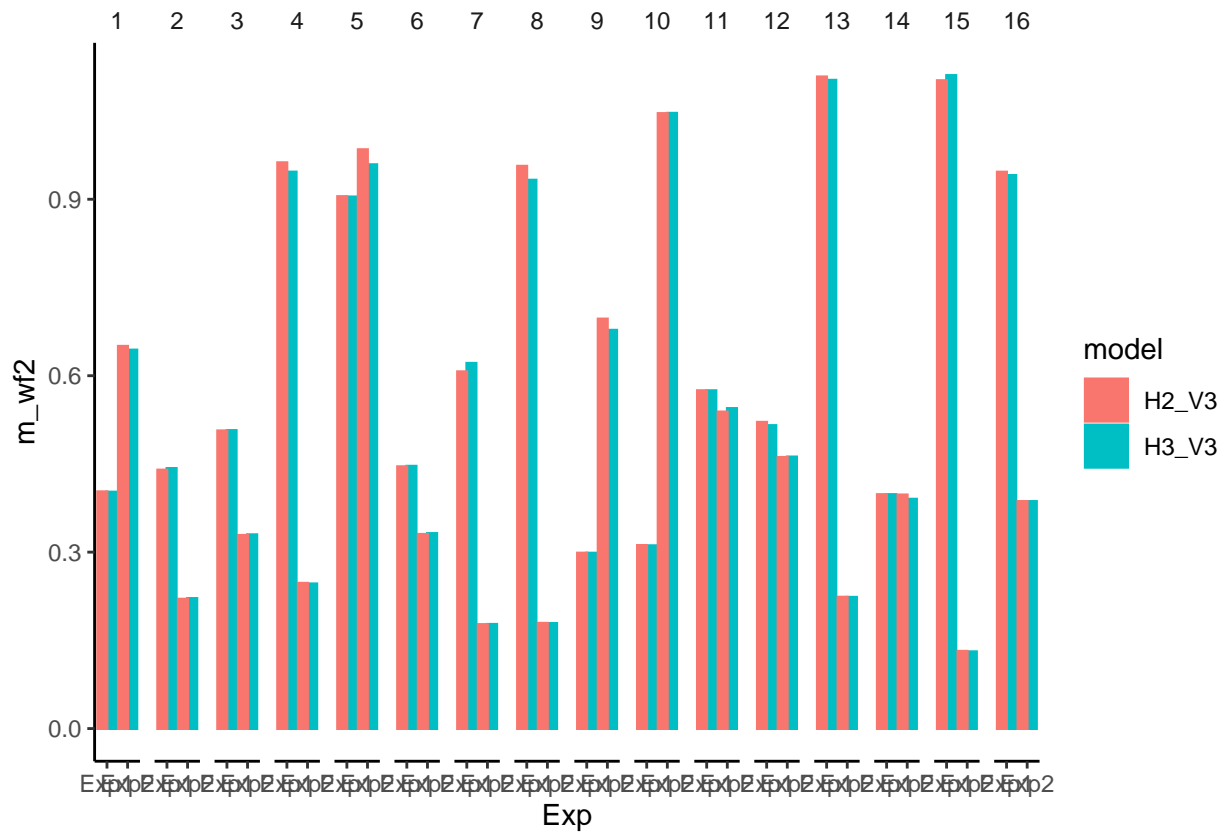
```

ggplot(m_Baypar, aes(x = Exp, y = m_wf1, color = model, fill = model, group = model)) +
  geom_bar(stat = "identity",
           position = position_dodge()) +
  facet_grid(~Nsub) +
  theme_new

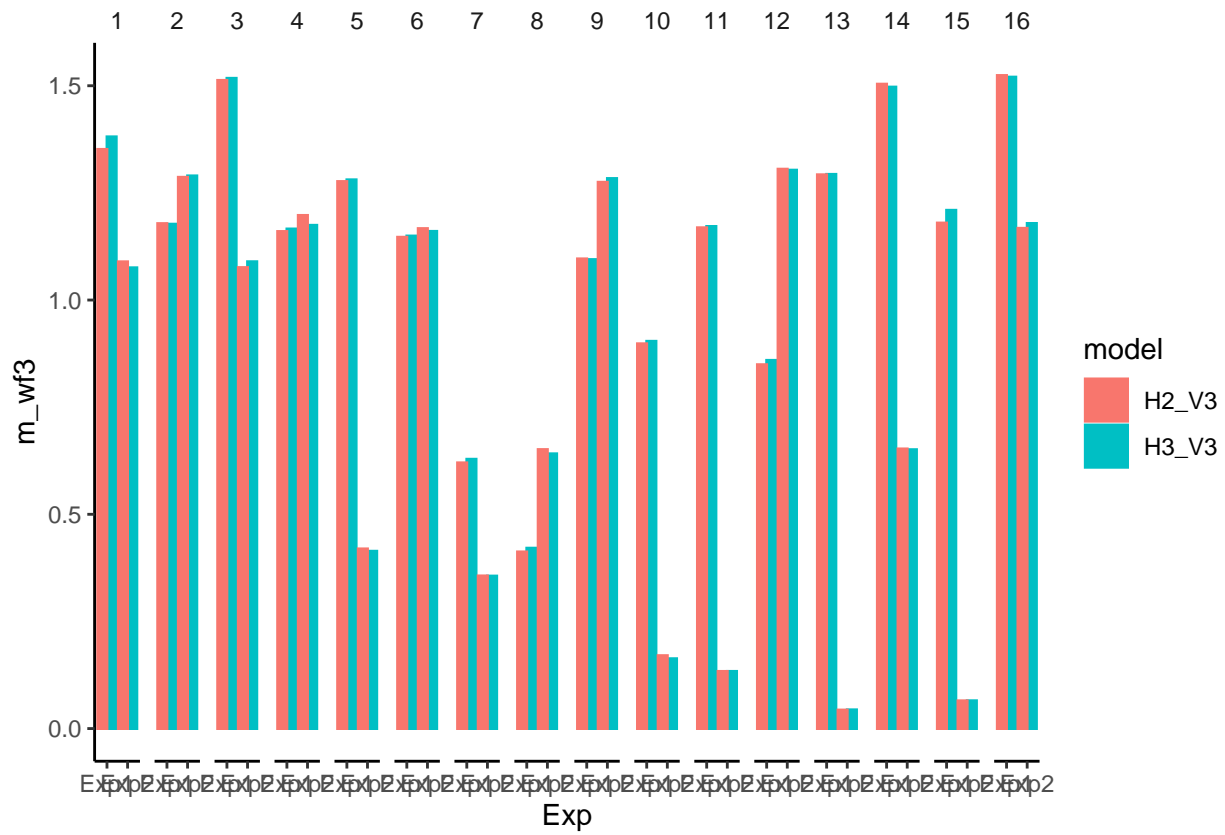
```



```
ggplot(m_Baypar, aes(x = Exp, y = m_wf2, color = model, fill = model, group = model)) +
  geom_bar(stat = "identity",
           position = position_dodge()) +
  facet_grid(~Nsub) +
  theme_new
```

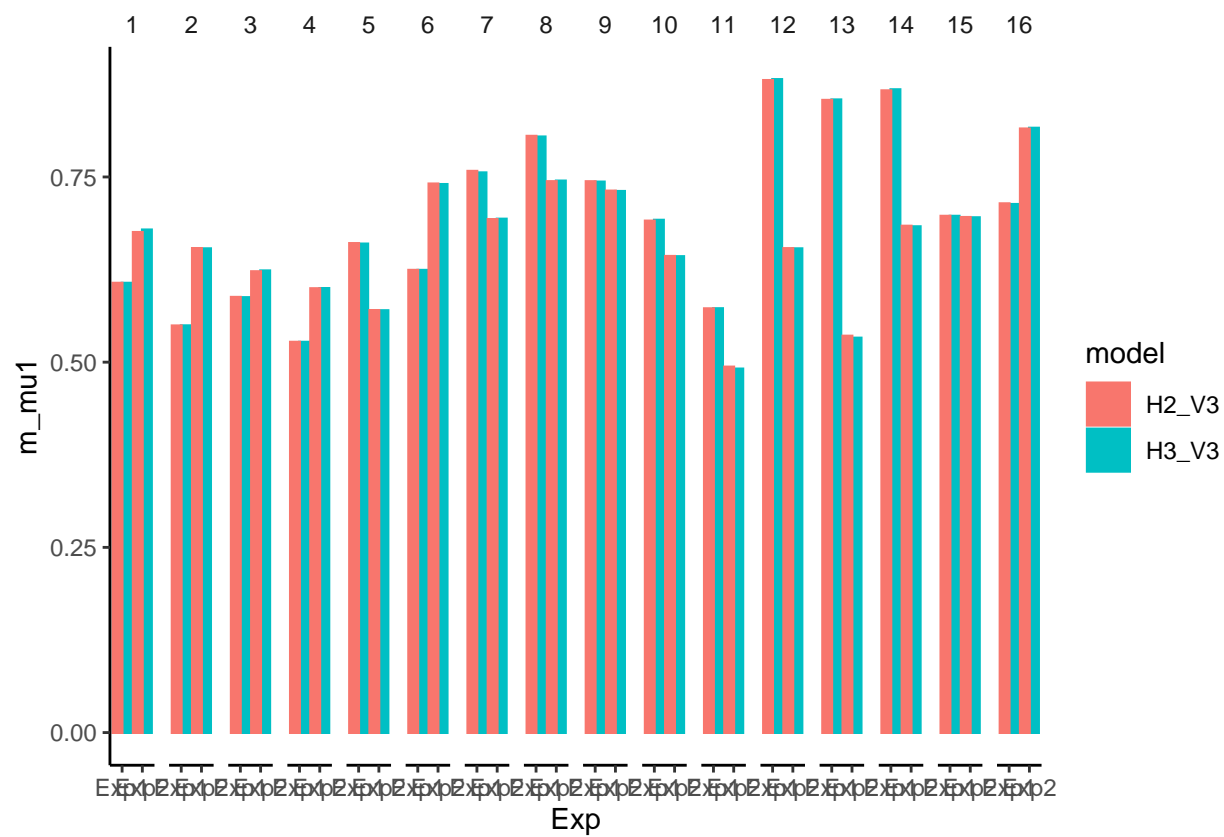


```
ggplot(m_Baypar, aes(x = Exp, y = m_wf3, color = model, fill = model, group = model)) +
  geom_bar(stat = "identity",
           position = position_dodge()) +
  facet_grid(~Nsub) +
  theme_new
```

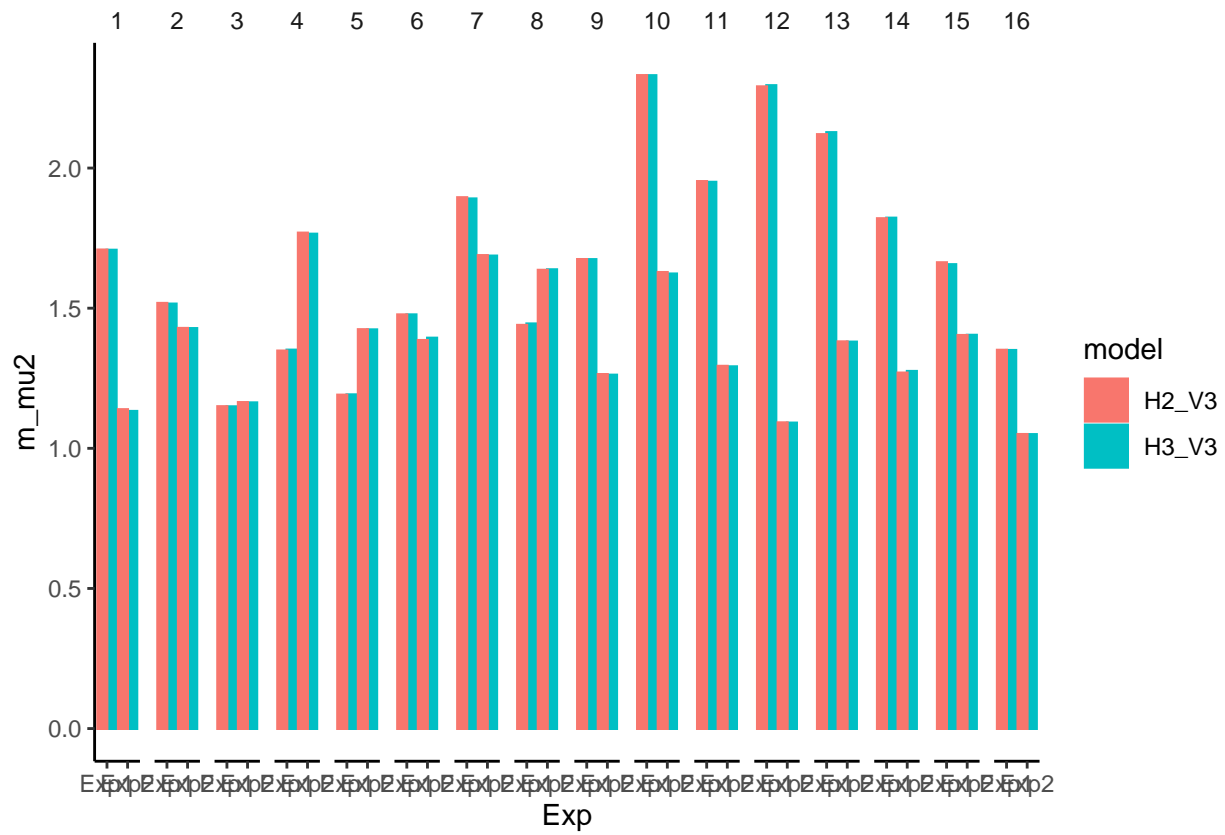


means of the prior

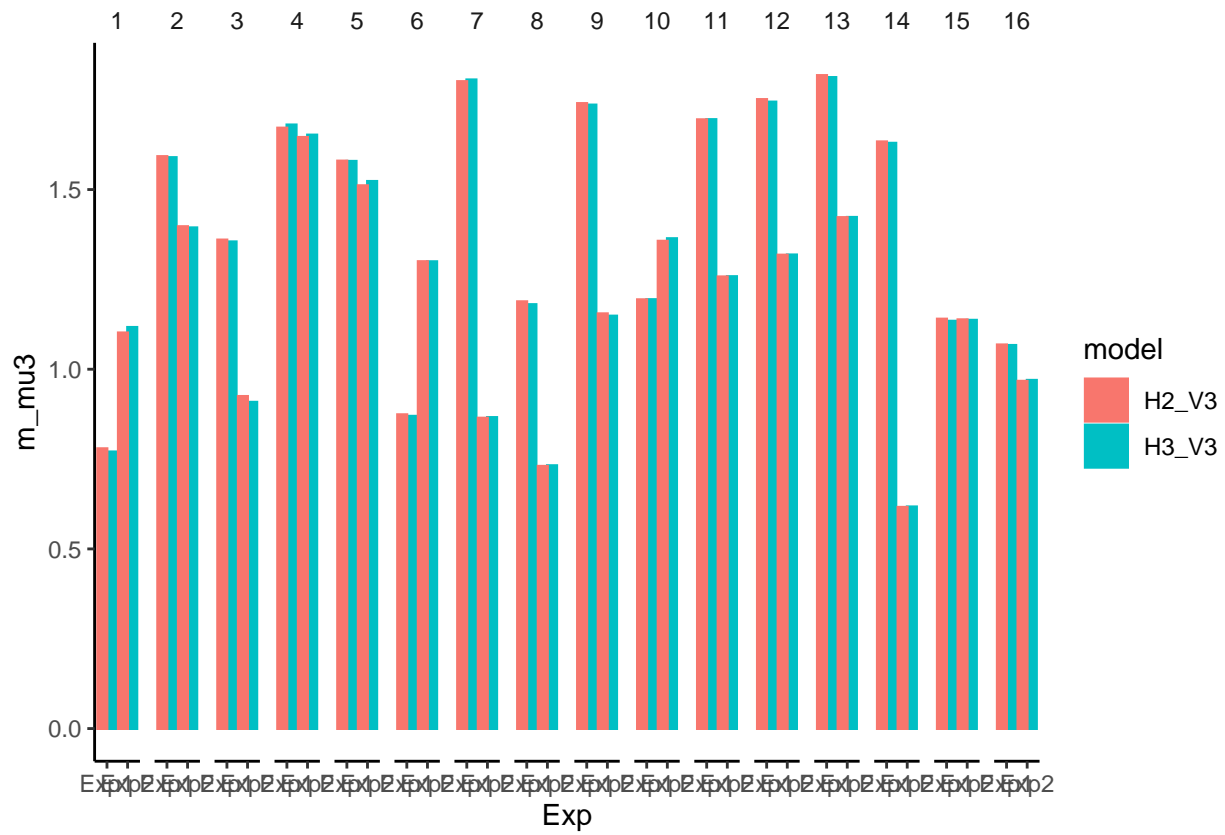
```
ggplot(m_Baypar, aes(x = Exp, y = m_mu1, color = model, fill = model, group = model)) +
  geom_bar(stat = "identity",
    position = position_dodge()) +
  facet_grid(~Nsub) +
  theme_new
```



```
ggplot(m_Baypar, aes(x = Exp, y = m_mu2, color = model, fill = model, group = model)) +
  geom_bar(stat = "identity",
           position = position_dodge()) +
  facet_grid(~Nsub) +
  theme_new
```

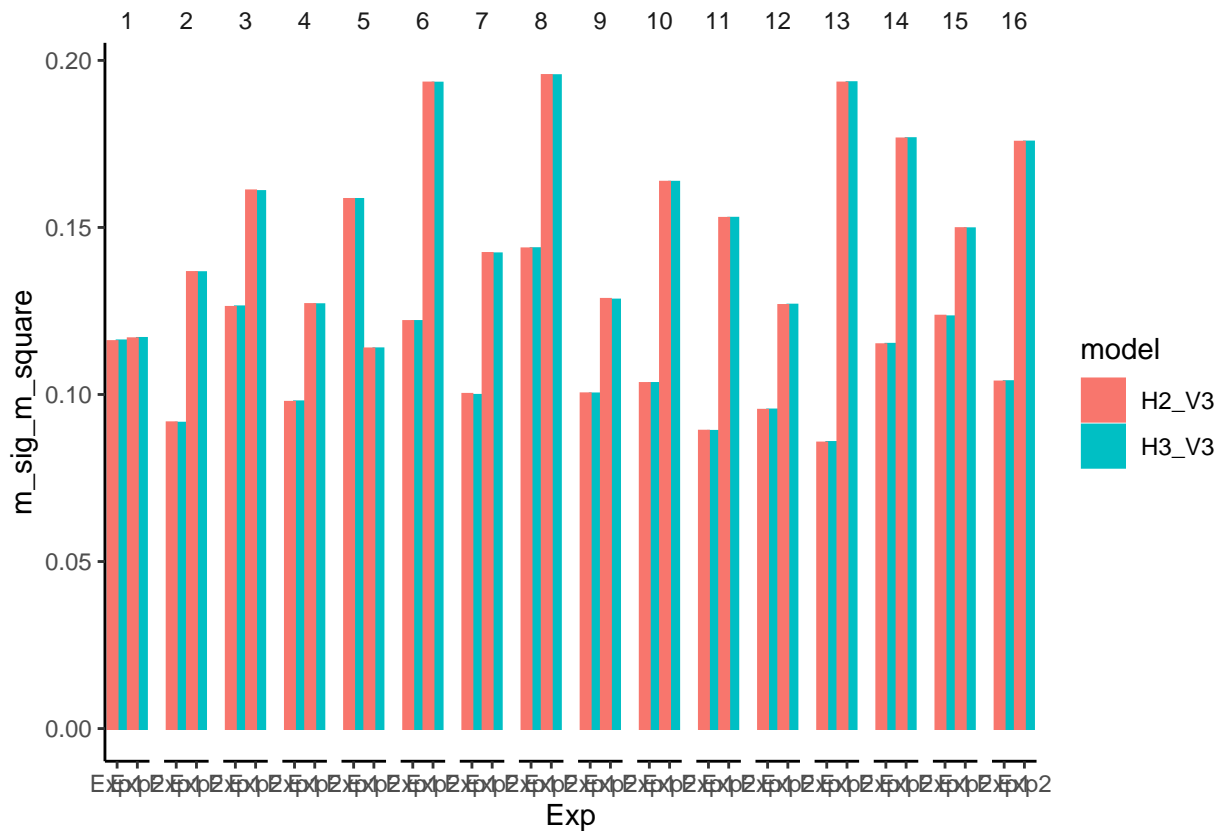


```
ggplot(m_Baypar, aes(x = Exp, y = m_mu3, color = model, fill = model, group = model)) +
  geom_bar(stat = "identity",
           position = position_dodge()) +
  facet_grid(~Nsub) +
  theme_new
```

motor noise

```
ggplot(m_Baypar, aes(x = Exp, y = m_sig_m_square, color = model, fill = model, group = model)) +
  geom_bar(stat = "identity",
    position = position_dodge()) +
  facet_grid(~Nsub) +
  theme_new
```



```
mm_Baypar <- m_Baypar %>% dplyr::group_by(Exp, model) %>%
  dplyr::summarize(m_wf1 = mean(m_wf1), m_wf2 = mean(m_wf2),
                  m_wf3 = mean(m_wf3), m_mu1 = mean(m_mu1),
                  m_mu2 = mean(m_mu2), m_mu3 = mean(m_mu3),
                  m_sig_m_square = mean(m_sig_m_square))

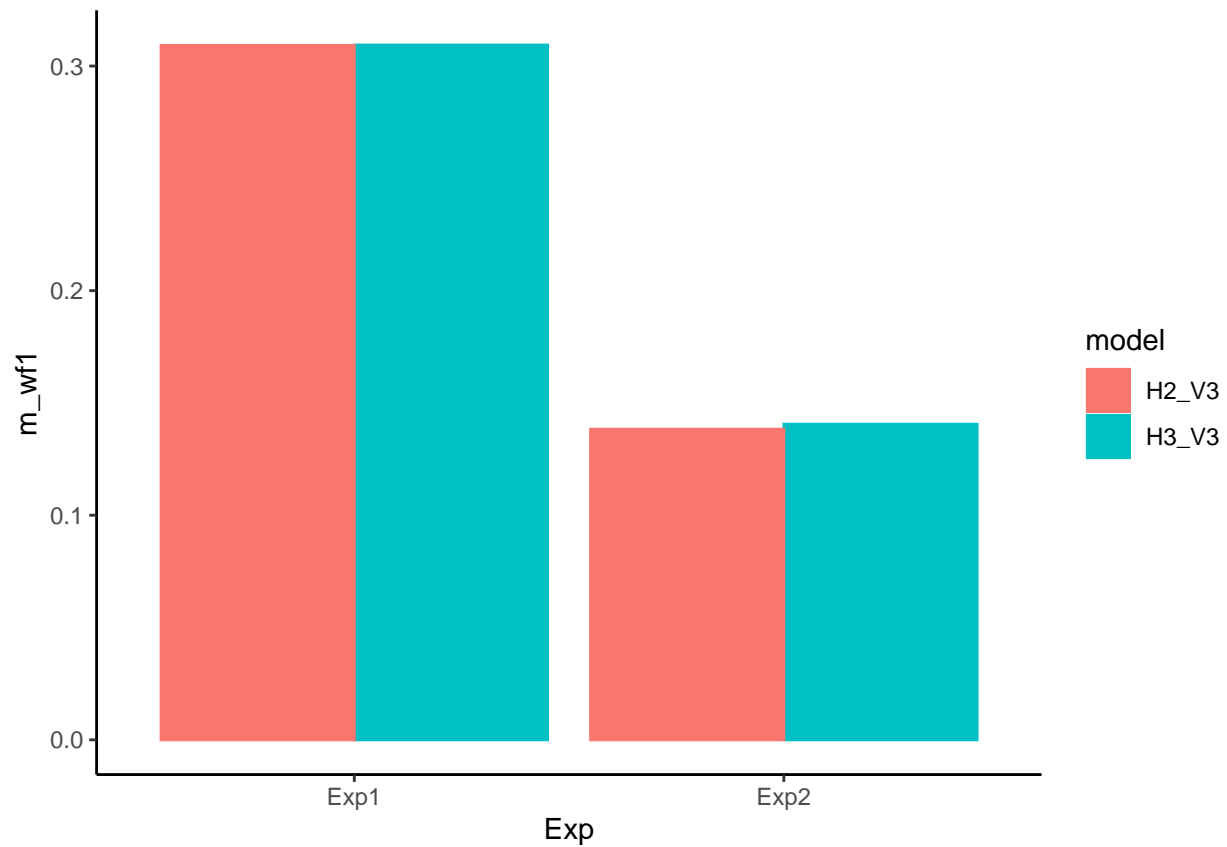
mm_Baypar

## # A tibble: 4 x 9
## # Groups:   Exp [2]
##   Exp  model m_wf1 m_wf2 m_wf3 m_mu1 m_mu2 m_mu3 m_sig_m_square
##   <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 Exp1  H2_V3 0.309 0.655 1.14  0.696 1.68 1.43 0.111
## 2 Exp1  H3_V3 0.309 0.653 1.14  0.696 1.68 1.43 0.111
## 3 Exp2  H2_V3 0.138 0.437 0.753 0.659 1.38 1.17 0.153
## 4 Exp2  H3_V3 0.141 0.434 0.751 0.659 1.37 1.17 0.153

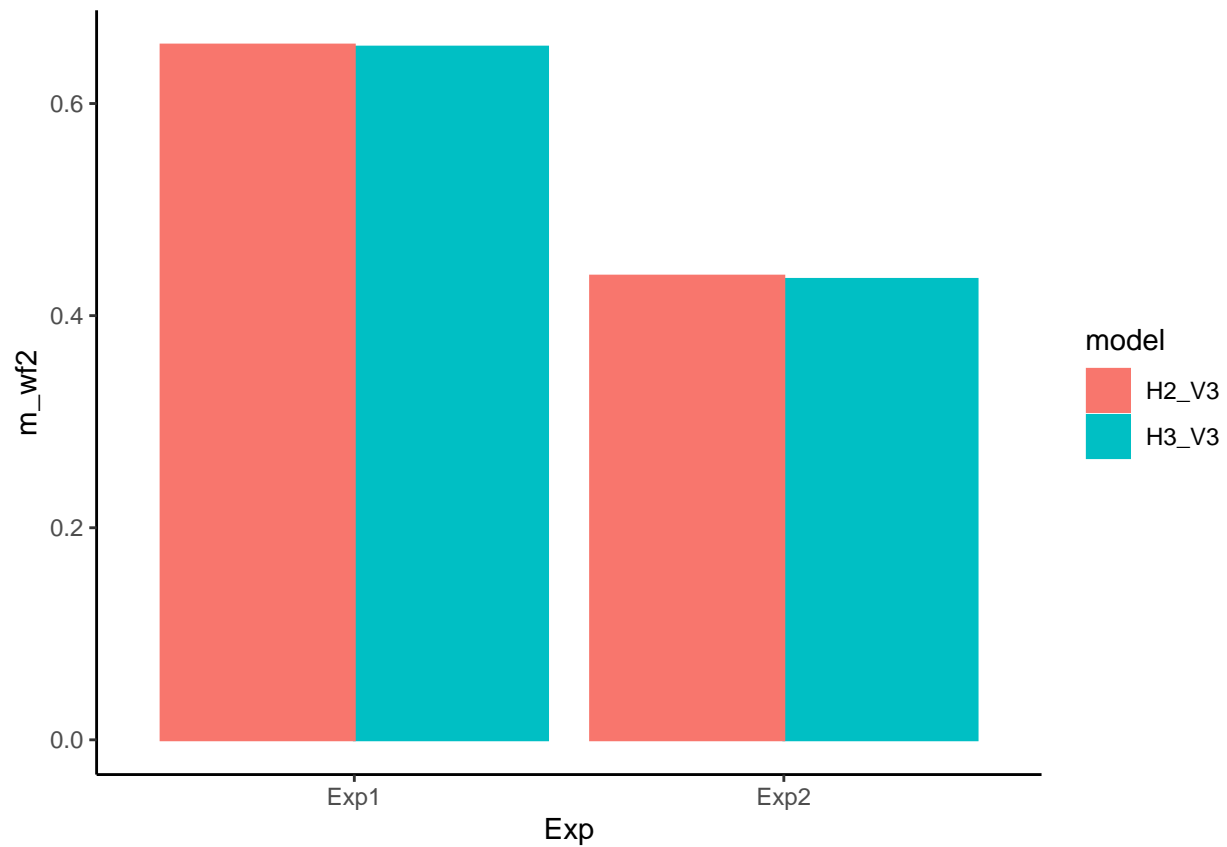
fig_m_wf1 <- ggplot(mm_Baypar, aes(x = Exp, y = m_wf1, color = model, fill = model)) +
  geom_bar(stat = "identity",
           position = position_dodge()) + theme(legend.position="bottom")+
  theme_new

ggsave(file.path('figures', 'fig_m_wf1.png'), fig_m_wf1, width = 7, height = 5)

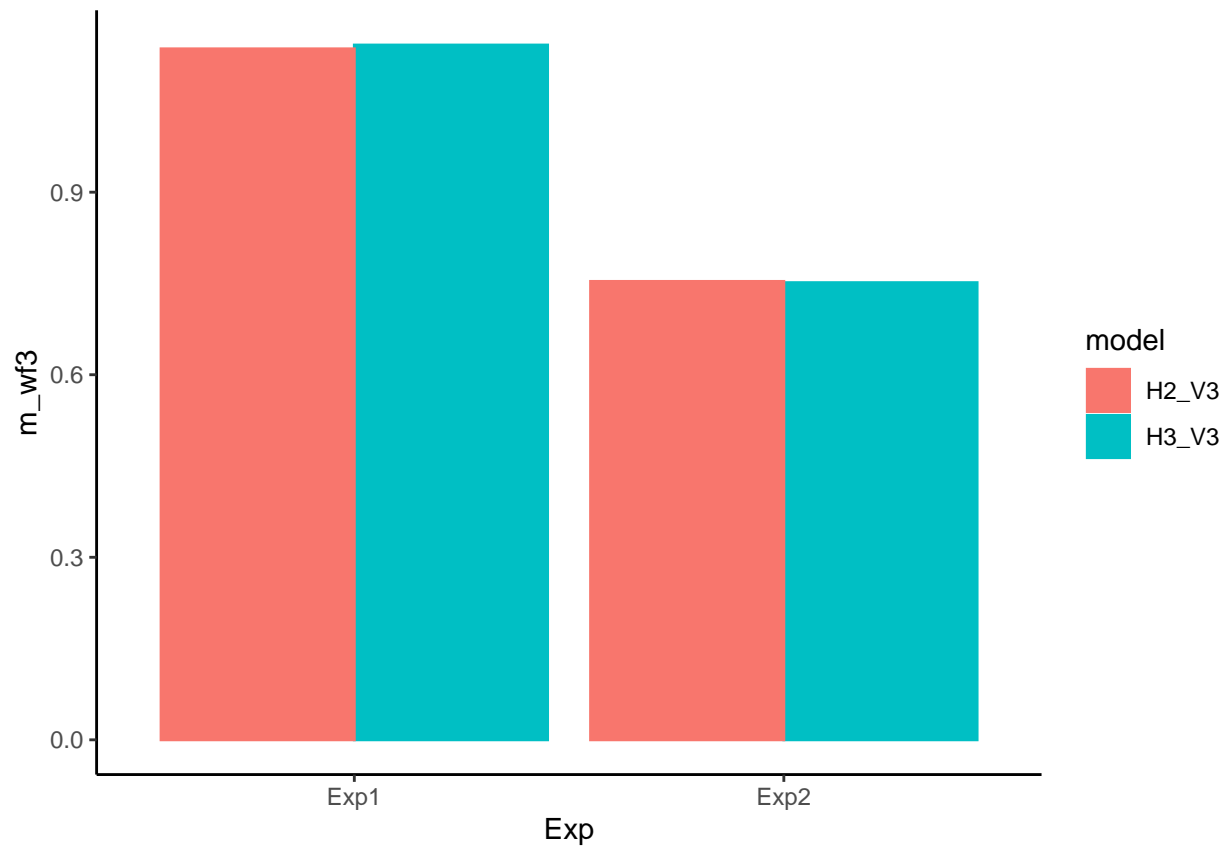
fig_m_wf1
```



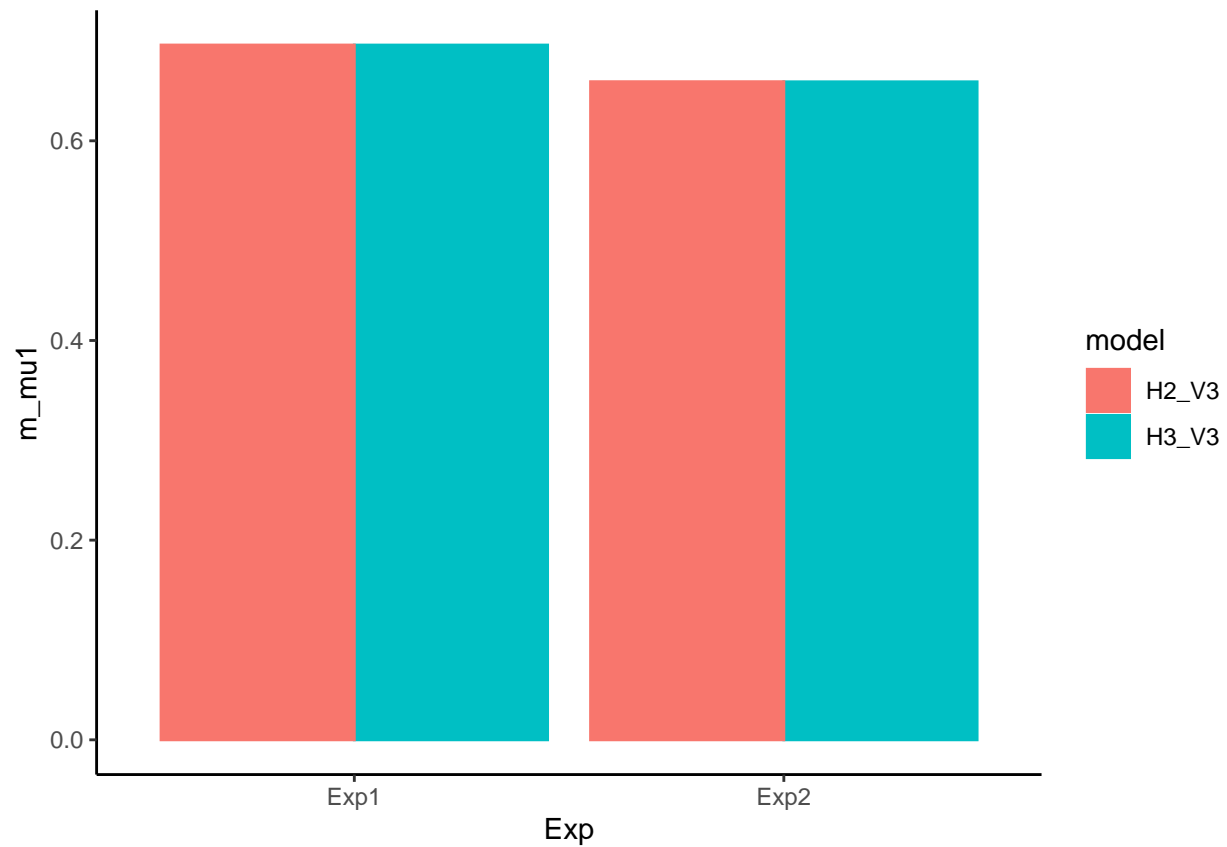
```
fig_m_wf2 <- ggplot(mm_Baypar, aes(x = Exp, y = m_wf2, color = model, fill = model)) +  
  geom_bar(stat = "identity",  
           position = position_dodge()) + theme(legend.position="bottom")+  
  theme_new  
  
ggsave(file.path('figures','fig_m_wf2.png'), fig_m_wf2, width = 7, height = 5)  
  
fig_m_wf2
```



```
fig_m_wf3 <- ggplot(mm_Baypar, aes(x = Exp, y = m_wf3, color = model, fill = model)) +  
  geom_bar(stat = "identity",  
           position = position_dodge()) + theme(legend.position="bottom")+  
  theme_new  
  
ggsave(file.path('figures','fig_m_wf3.png'), fig_m_wf3, width = 7, height = 5)  
  
fig_m_wf3
```



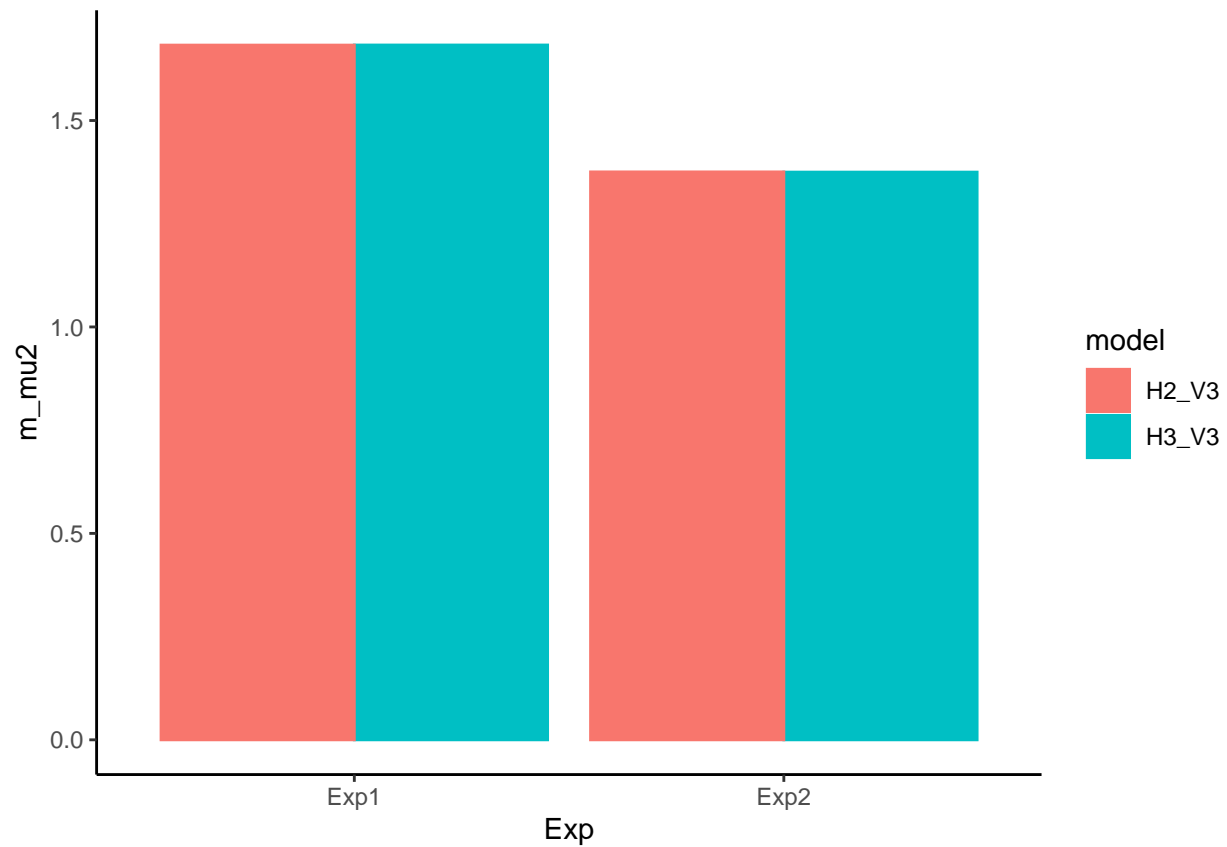
```
fig_m_mu1 <- ggplot(mm_Baypar, aes(x = Exp, y = m_mu1, color = model, fill = model)) +  
  geom_bar(stat = "identity",  
           position = position_dodge()) + theme(legend.position="bottom")+  
  theme_new  
  
ggsave(file.path('figures','fig_m_mu1.png'), fig_m_mu1, width = 7, height = 5)  
  
fig_m_mu1
```



```
fig_m_mu2 <- ggplot(mm_Baypar, aes(x = Exp, y = m_mu2, color = model, fill = model)) +
  geom_bar(stat = "identity",
           position = position_dodge()) + theme(legend.position="bottom")+
  theme_new

ggsave(file.path('figures', 'fig_m_mu2.png'), fig_m_mu2, width = 7, height = 5)

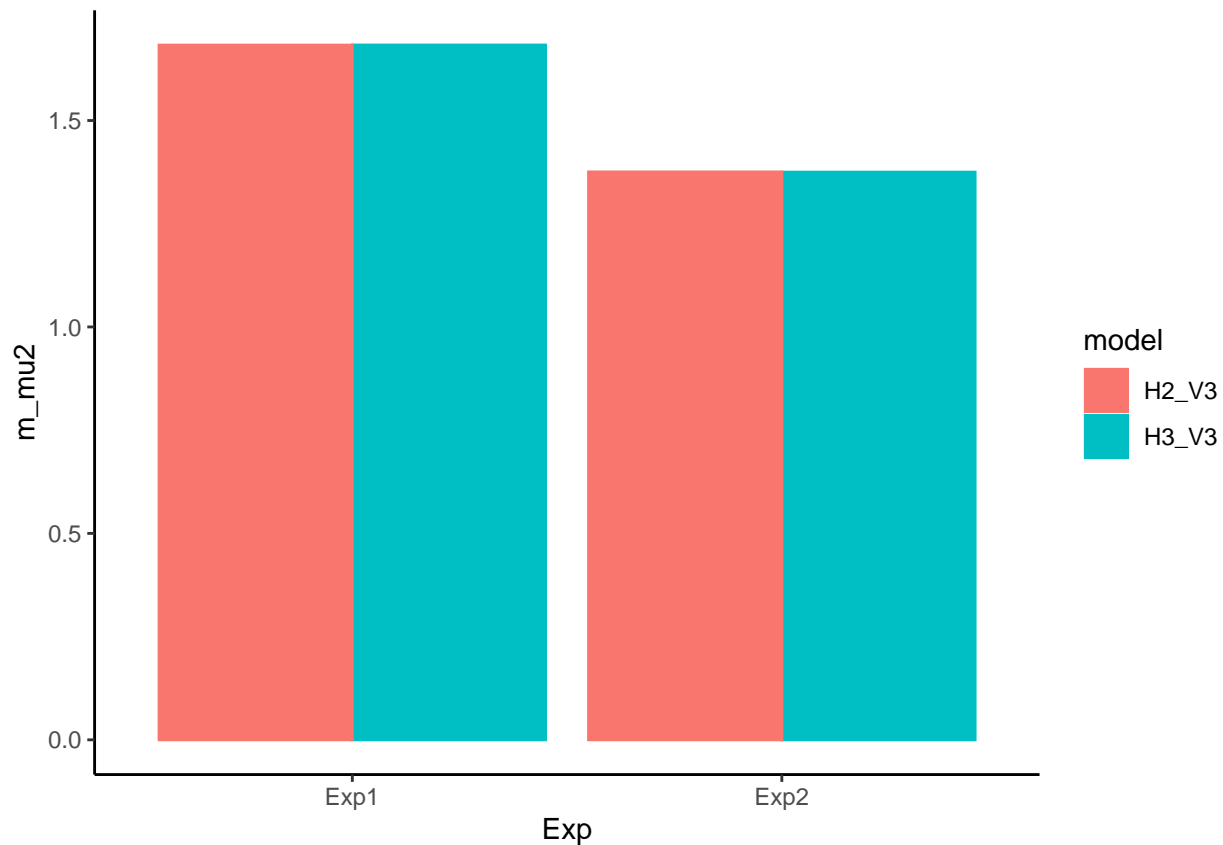
fig_m_mu2
```



```
fig_m_mu3 <- ggplot(mm_Baypar, aes(x = Exp, y = m_mu3, color = model, fill = model)) +
  geom_bar(stat = "identity",
           position = position_dodge()) + theme(legend.position="bottom")+
  theme_new

ggsave(file.path('figures','fig_m_mu3.png'), fig_m_mu3, width = 7, height = 5)

fig_m_mu2
```



Prediction results

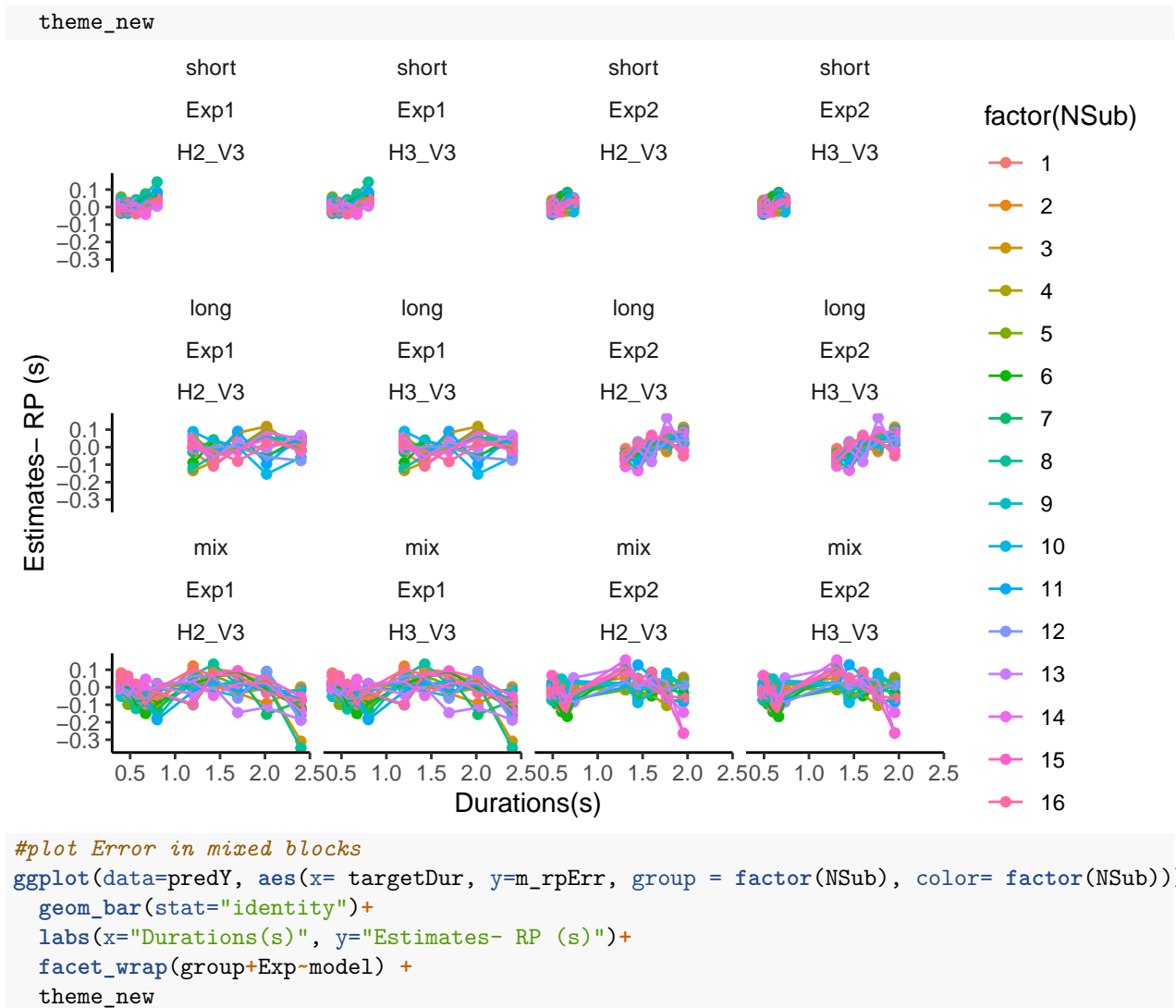
```
predY <- group_by(AllDat_predY, targetDur, Exp, NSub, model, group) %>%
  summarize(m_RP = mean(RP), n = n(),
            sd_RP = sd(RP) / sqrt(n-1),
            m_predY = mean(predY),
            sd_predY = sd(predY) / sqrt(n-1))
```

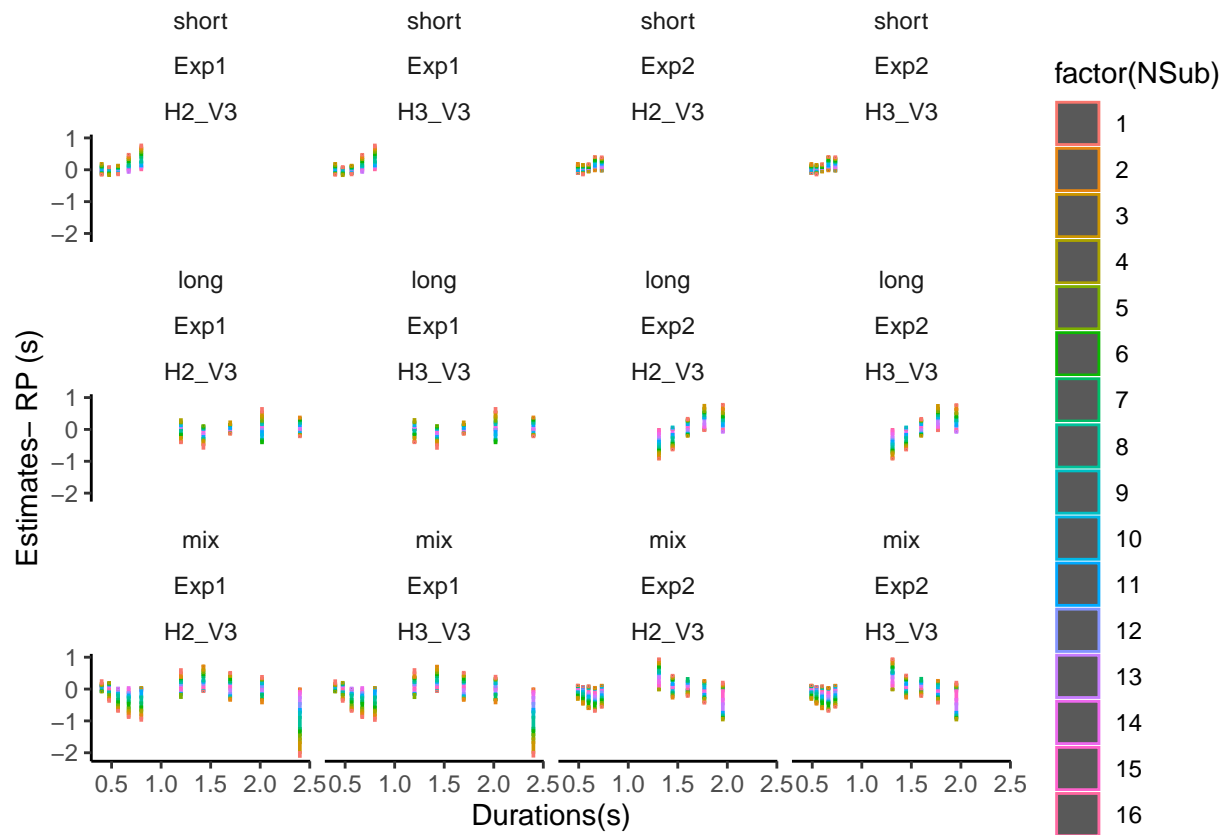
```
predY$m_rpErr = predY$m_predY - predY$m_RP
predY$m_relativeErr = predY$m_rpErr / predY$targetDur
```

```
m_predY <- predY%>%
  group_by(targetDur, Exp, model, group)%>%
  summarize(
    n = n(),
    m_m_predY = mean(m_predY),
    se_m_predY = sd(m_predY) / sqrt(n - 1),
    m_m_RP = mean(m_RP),
    se_m_RP = sd(m_RP) / sqrt(n-1)
  )
```

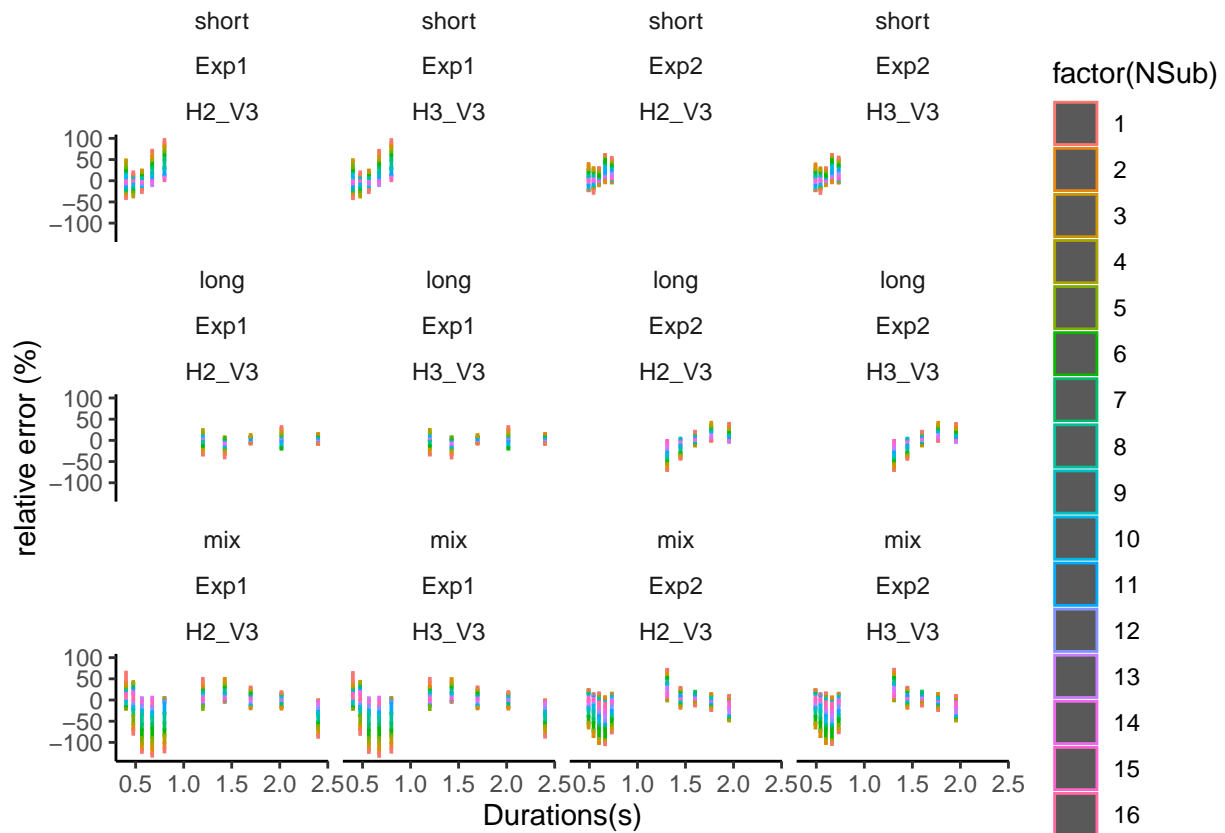
#plot Error in mixed blocks

```
ggplot(data=predY, aes(x= targetDur, y=m_rpErr, group = factor(NSub), color= factor(NSub))) +
  geom_point()+
  geom_line()+
  labs(x="Durations(s)", y="Estimates- RP (s)") +
  facet_wrap(group+Exp~ model)+
```



```
#plot relative Error for mixed blocks
fig_rerr_model <- ggplot(data=predY, aes(x= targetDur, y=m_relativeErr*100, group = factor(NSub), color=
  geom_bar(stat="identity")+
  labs(x="Durations(s)", y="relative error (%)")+
  facet_wrap(group+Exp~model) +
  theme_new
fig_rerr_model
```



```
ggsave(file.path('figures', 'fig_rerr_model.png'), fig_rerr_model, width = 7, height = 5)
```

```
#plot the average of the predicted Y under the mixed condition
```

```
fig_mpredY = ggplot(m_predY) +
```

```
  geom_point(aes(targetDur, m_m_predY, group = interaction(Exp, model), color = factor(Exp), shape = fa
```

```
  geom_line(aes(targetDur, m_m_predY, group = interaction(Exp, model), linetype = factor(model), color = f
```

```
  geom_line(aes(targetDur, m_m_RP, group = interaction(Exp, model), linetype = factor(model), color = f
```

```
  geom_point(aes(targetDur, m_m_RP, group = interaction(Exp, model), color = factor(Exp), shape = facto
```

```
  geom_abline(slope = 1, linetype = 2, size = 1, color = 'lightgray') + # add diagonal line
```

```
  facet_wrap(model~group) +
```

```
  guides(color = guide_legend(title = element_blank())) + # remove legend title
```

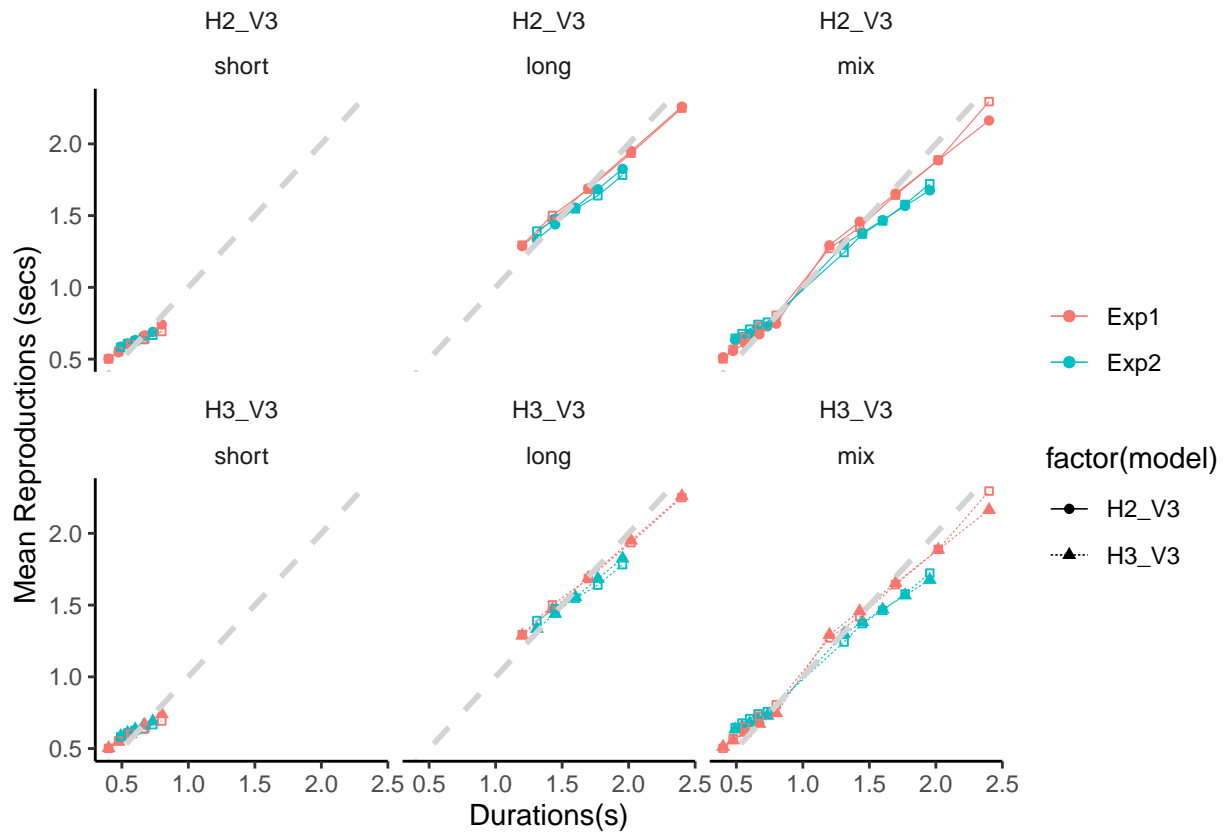
```
  theme_classic() +
```

```
  theme(strip.background = element_blank()) + # remove subtitle background
```

```
  labs(x = "Durations(s)", y = "Mean Reproductions (secs)", size = 15) + theme(legend.position = "bottom")
```

```
  theme_new
```

```
fig_mpredY
```



```
ggsave(file.path('figures', 'fig_mpredY_V3.png'), fig_mpredY, width = 7, height = 5)
```

```
predY$rpErr_squared <- predY$m_rpErr^2
m_predY_Err <- dplyr::group_by(predY, Exp, model, group) %>%
  dplyr::summarize(sum_rpErr = sum(rpErr_squared),
                  n = n())
```

```
fig_rpErr_model <- ggplot(m_predY_Err, aes(x = group, y = sum_rpErr, color = model, fill = model, group
  geom_bar(stat = "identity",
            position = position_dodge()) + theme(legend.position="bottom")+
  facet_wrap(Exp~.) +
  theme_new
```

```
ggsave(file.path('figures', 'fig_rpErr_model.png'), fig_rpErr_model, width = 7, height = 5)
```

```
fig_rpErr_model
```

