The Role of Variability in Learning Transfer: A Similarity-Based Computational Approach

Thomas E. Gorman

Submitted to the faculty of the University Graduate School in partial fulfillment of the requirements for the degree Doctor of Philosophy in the Department of Psychology and Brain Sciences and the Cognitive Science Program, Indiana University Indiana University

Accepted by the Graduate Faculty, Indiana University, in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

______________________________ Robert L. Goldstone, PhD

______________________________ Robert Nosofsky, PhD

______________________________ Peter Todd, PhD

______________________________ Mike Jones, PhD

Acknowledgements

Abstract

This dissertation seeks to explore the cognitive underpinnings that govern the generalization of learning, focusing specifically on the role of variability during training in shaping subsequent transfer performance. A comprehensive review of the existing literature is presented, emphasizing the methodological complications associated with disentangling the confounding effects of similarity. Through a series of experiments involving several novel visuomotor tasks, this work investigates whether and how variability in training conditions affects performance in novel tasks. To theoretically account for the empirical outcomes, I employ both instance-based and connectionist computational models, both of which incorporate similarity-based mechanisms. These models serve to account for the extent to which variability influences the learners' generalization gradient, and also explain how training variation can produce both beneficial and deleterious outcomes.

Table of contents

List of Figures

List of Tables

```r
pacman::p_load(tidyr,papaja, knitr, tinytex, RColorBrewer, kableExtra, cowplot, patchwork)
source('../Functions/IGAS_ProcessFunctions.R')
```

$$e^{(c\cdot|x-800|)}$$

$$e^{(c\cdot|x-400|)}$$

$$e^{(c\cdot|x-800|)}$$

$+$

$$e^{(c\cdot|x-400|)}$$

$$e^{(c_{varied}\cdot|x-800|)}$$

$e^{(c_{varied}\cdot|x-800|)}$ $+e^{(c_{varied}\cdot|x-400|)}$ $e^{(c_{varied}\cdot|x-800|)} + e^{(c_{varied}\cdot|x-400|)}$

$e^{(c\cdot|x-800|)} + e^{(c\cdot|x-400|)}$,

```r
#

p=2
c<- .000002

trainingTestingDifference=2000;
cvaried=.00002
cconstant=.0005
simdat <- data.frame(x=rep(seq(200,1000),3),condit=c(rep("varied",1602),rep("constant",801)),
                     train.position=c(rep(400,801),rep(800,801),rep(600,801)),c=.0002,p=2) %>%
                     mutate(c2=ifelse(condit=="varied",cvaried,cconstant),
                            genGauss=exp(-c*(abs((x-train.position)^p))),
                            genGaussDist=exp(-c*(trainingTestingDifference+abs((x-train.positio
                            genGauss2=exp(-c2*(abs((x-train.position)^p))),
                            genGaussDist2=exp(-c2*(trainingTestingDifference+abs((x-train.posit
                            ) %>%
  group_by(x,condit) %>%
  summarise(genGauss=mean(genGauss),genGauss2=mean(genGauss2),genGaussDist=mean(genGaussDist),g


#plot(x,exp(c*(trainingTestingDifference+abs(x-800)))+exp(c*(trainingTestingDifference+abs(x-40

colorVec=c("darkblue","darkred")
plotSpecs <- list(geom_line(alpha=.7),scale_color_manual(values=colorVec),
                  geom_vline(alpha=.55,xintercept = c(400,800),color=colorVec[2]),
                  geom_vline(alpha=.55,xintercept = c(600),color=colorVec[1]),
```

```r
                  ylim(c(0,1.05)),
                  #xlim(c(250,950)),
                  scale_x_continuous(breaks=seq(200,1000,by=200)),
                  xlab("Test Stimulus"),
                  annotate(geom="text",x=455,y=1.05,label="Varied",size=3.0),
                  annotate(geom="text",x=455,y=.97,label="Training",size=3.0),
                  annotate(geom="text",x=662,y=1.05,label="Constant",size=3.0),
                  annotate(geom="text",x=657,y=.97,label="Training",size=3.0),
                  annotate(geom="text",x=855,y=1.05,label="Varied",size=3.0),
                  annotate(geom="text",x=855,y=.97,label="Training",size=3.0),
                  theme(panel.border = element_rect(colour = "black", fill=NA, size=1),
                        legend.position="none"))

ip1 <- simdat  %>% ggplot(aes(x,y=genGauss,group=condit,col=condit))+plotSpecs+ylab("Amount of
ip2 <- simdat %>%  ggplot(aes(x,y=genGauss2,group=condit,col=condit))+plotSpecs+ylab("")+ggtitl
ip3 <- simdat  %>% ggplot(aes(x,y=genGaussDist,group=condit,col=condit))+plotSpecs+ylab("Amoun
  ggtitle("Added distance due to context, 1c")+theme(plot.margin = margin(0, 0, 0, 1))
ip4 <- simdat %>%  ggplot(aes(x,y=genGaussDist2,group=condit,col=condit))+plotSpecs+ylab("")+
  ggtitle("Added distance due to context, 2c")+theme(plot.margin = margin(0, 0, 0, 1))
# gridExtra::grid.arrange(ip1,ip2,ip3,ip4,ncol=2)

gtitle="Figure 9."
title = ggdraw()+draw_label(gtitle,fontface = 'bold',x=0,hjust=0,size=11)+theme(plot.margin = m
plot_grid(title,NULL,ip1,ip2,ip3,ip4,ncol=2,rel_heights=c(.1,.8,.8))
```

Figure 9.

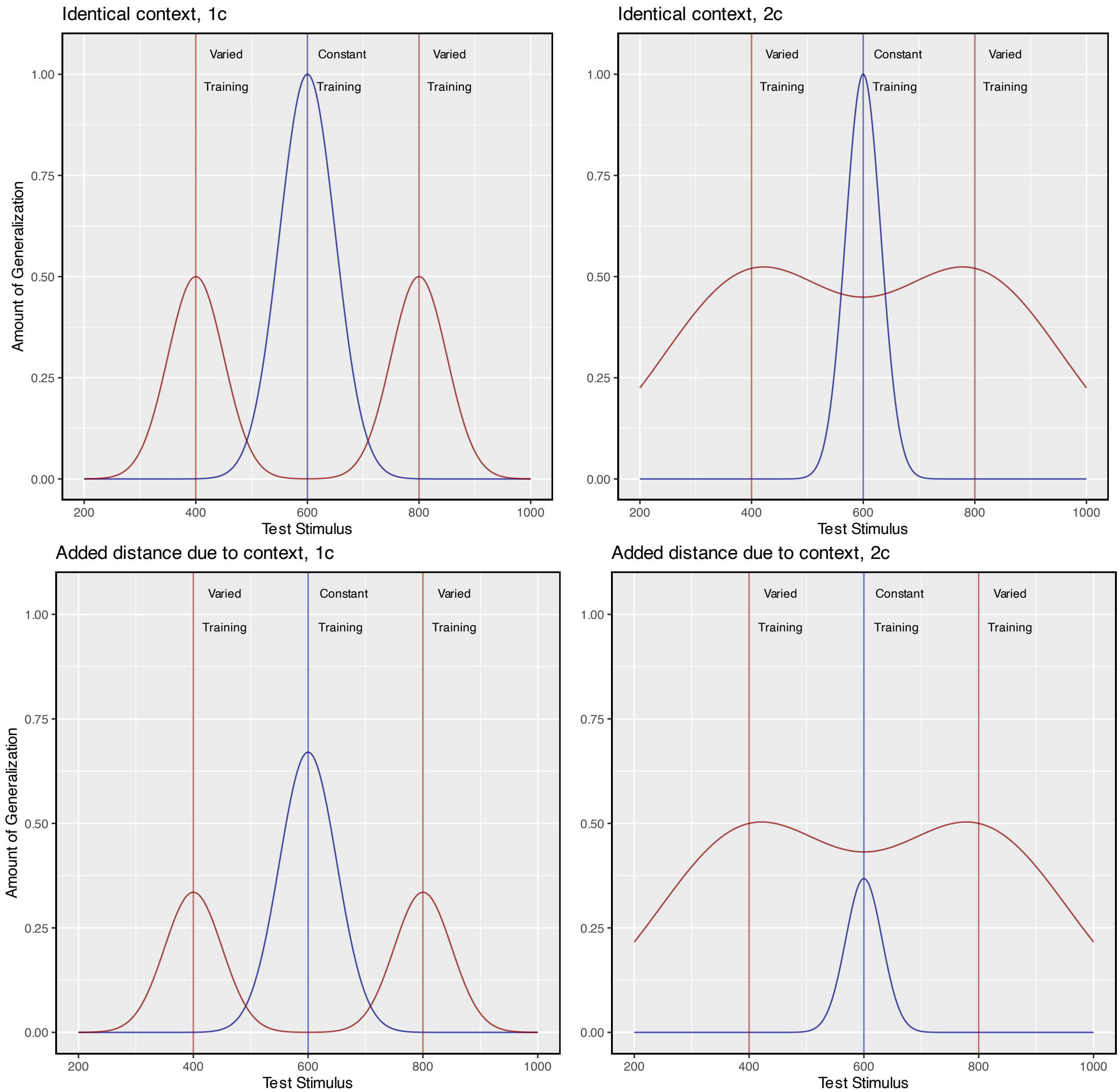

Figure 1: A simple model depicting the necessity of both of two separately fit generalization parameters, c, and a positive distance between training and testing contexts, in order for an instance model to predict a pattern of varied training from stimuli 400 and 800 outperforming constant training from position 600 at a test position of 600. For the top left panel, in which the generalization model assumes a single c value (-.008) for both varied and constant conditions, and identical contexts across training and testing, the equation which generates the varied condition is - Amount of Generalization $= e^{(c \cdot |x-800|)} + e^{(c \cdot |x-400|)}$, whereas the constant group generalization is generated from $2 \cdot e^{(c \cdot |x-600|)}$. For the top right panel, the c constants in the original equations are different for the 2 conditions, with $c = -.002$ for the varied condition, and $c = -.008$ for the constant condition. The bottom two panels are generated from identical equations to those immediately above, except for the addition of extra distance (100 units) to reflect the assumption of some change in context between training and testing conditions. Thus, the generalization model for the varied condition in the bottom-right panel is of the form - Amount of Generalization $= e^{(c_{varied} \cdot |x-800|)} + e^{(c_{varied} \cdot |x-400|)}$.

Main body

Following the procedure used by Mcdaniel et al. (2009), we will assess the ability of both ALM and EXAM to account for the empirical data when fitting the models to 1) only the training data, and 2) both training and testing data. Models will be fit directly to the trial by trial data of each individual participants, both by minimizing the root-mean squared deviation (RMSE), and by maximizing log likelihood. Because ALM has been shown to do poorly at accounting for human patterns extrapolation (DeLosh et al., 1997), we will also fit the extended EXAM version of the model, which operates identically to ALM during training, but includes a linear extrapolation mechanism for generating novel responses during testing.

quarto pandoc –citeproc –pdf-engine xelatex -t pdf
–bibliography=../Assets/Bib/Dissertation.bib
–standalone
-f markdown igas_e1.pdf.md
-o refer-test.pdf

quarto render igas_e1.qmd –citeproc –pdf-engine xelatex -t pdf
–bibliography=../Assets/Bib/Dissertation.bib
–standalone
-o refer-test.pdf

Appendix

```
# print(getwd())
# here::set_here(path='..')
# print(getwd())
source(here::here("Functions", "packages.R"))
```

```
test <- readRDS(here::here("data/e1_08-21-23.rds")) |> filter(expMode2 == "Test")  |>
  select(id,condit,bandInt,vb,vx,dist,sdist,bandType,tOrder)
```

Posterior Predictive Distributions

```
dist_pred <-
  posterior_predict(e1_distBMM, ndraws = 500) |>
  array_branch(margin=1) |>
   map_dfr(
    function(yrep_iter) {
      test  |>
        mutate(dist_pred = yrep_iter)
    },
    .id = 'iter'
  ) |>
  mutate(iter = as.numeric(iter))
```

```
dist_pred  |>
  filter(iter < 100) %>%
  ggplot(aes(dist_pred, group = iter)) +
  geom_line(alpha = .03, stat = 'density', color = 'blue') +
  geom_density(data = test,
               aes(dist,col=vb),
               inherit.aes = FALSE,
               size = 0.7) + # 1
  facet_grid(condit ~ vb) +
  xlab('Deviation')
```

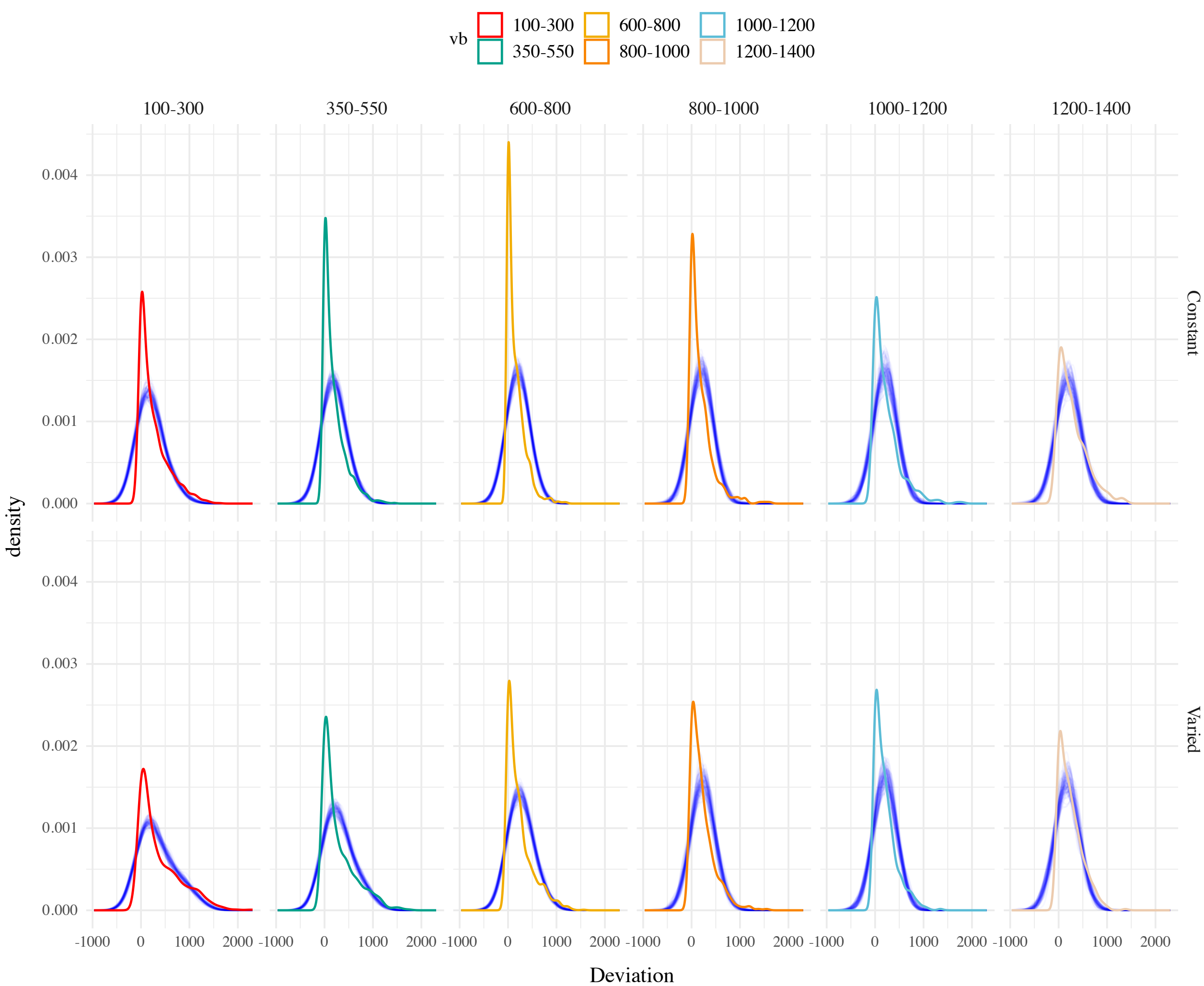

Figure 2: Posterior Predictive distributions for Absolute Deviance. Posterior Draws in Blue, colored lines are empirical data.

```r
vx_pred <-
  posterior_predict(e1_vxBMM, ndraws = 500) |>
  array_branch(margin=1) |>
   map_dfr(
    function(yrep_iter) {
      test  |>
        mutate(vx_pred = yrep_iter)
    },
    .id = 'iter'
  ) |>
  mutate(iter = as.numeric(iter))



vx_pred  |>
  filter(iter < 100) %>%
  ggplot(aes(vx_pred, group = iter)) +
  geom_line(alpha = .03, stat = 'density', color = 'blue') +
  geom_density(data = test,
               aes(vx,col=vb),
               inherit.aes = FALSE,
               size = 0.7) + # 1
  facet_grid(condit ~ vb) +
  xlab('Vx')
```

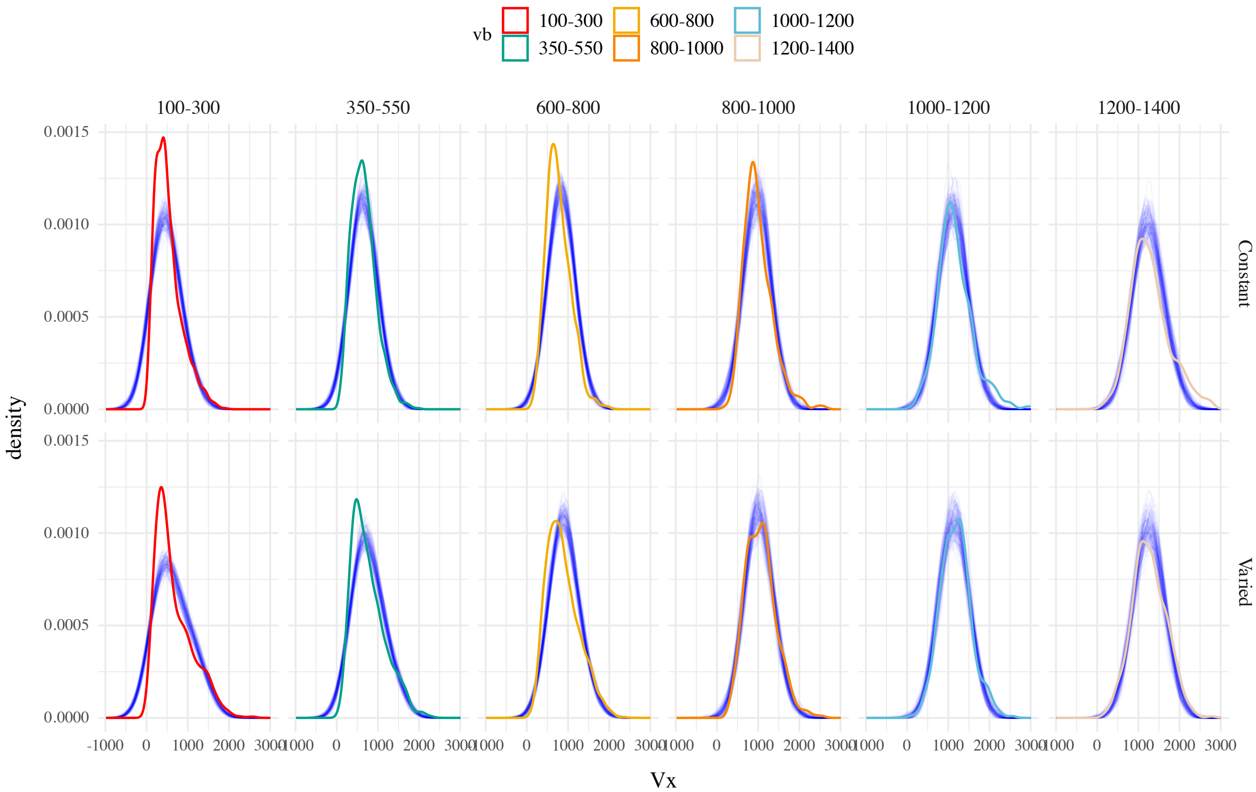

Figure 3: Posterior Predictive distributions for Vx. Posterior Draws in Blue, colored lines are empirical data.

Empirical vs. Predicted

```
{
vx_pred  |>
  filter(iter < 100)  |> group_by(id,condit,vb,iter) |>
  summarise(vx_pred=mean(vx_pred)) %>%
  ggplot(aes(x=vb,y=vx_pred,fill=condit)) +
  geom_flat_violin( position = position_nudge(x = 0.1, y = 0),
                    adjust = 1.5,
                    trim = FALSE, alpha = .5, colour = NA) +
  # geom_point(aes(x = as.numeric(vb) - 0.15, y = vx_pred, colour = vb),
  #             position = position_jitter(width = 0.05, height = 0),
  #             size = 1, shape = 20) +
  geom_boxplot(aes(x = vb, y = vx_pred, fill = condit),
               outlier.shape = NA,
               alpha = 0.5,
               width = 0.1,
               colour = "black") +
  geom_hline(yintercept = 0,
             linetype = 'dashed',
             color = 'red',
```

```
            size = 0.4) +
  coord_flip() + ggtitle("Predicted Vx")
} / {
vx_pred  |>
  filter(iter < 2)  |> group_by(id,condit,vb) |>
  summarise(vx=mean(vx)) %>%
  ggplot(aes(x=vb,y=vx,fill=condit)) +
  geom_flat_violin( position = position_nudge(x = 0.1, y = 0),
                   adjust = 1.5,
                   trim = FALSE,
                   alpha = .5,
                   colour = NA) +
  geom_point(aes(x = as.numeric(vb) - 0.15,col=condit),
             # position = position_jitter(width = 0.05),
             position = position_jitter(width = 0.05, height = 0),
             size = 1,
             shape = 20) +
  geom_boxplot(
              outlier.shape = NA,
              alpha = 0.5,
              width = 0.1,
              colour = "black") +
  geom_hline(yintercept = 0,
             linetype = 'dashed',
             color = 'red',
             size = 0.4) +
  coord_flip() + ggtitle("Empirical Vx")
}
```

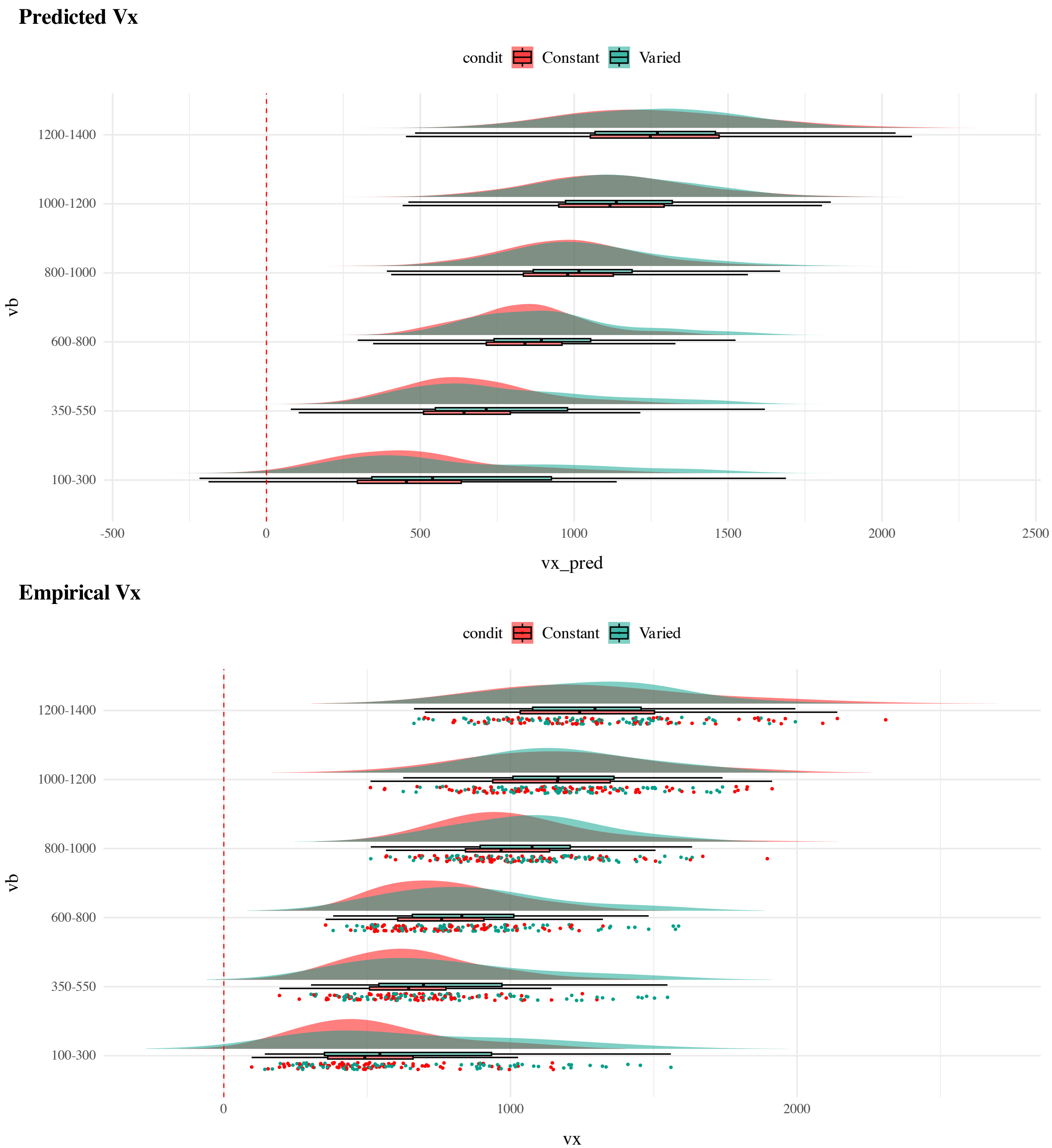

Figure 4: Bayesian Mixed Model predictions vs. Empirical Predictions - X velocity

Different Aggregations

```
epId <- dist_pred  |>
  filter(iter < 2)  |> group_by(id,condit,vb) |>
  summarise(dist=median(dist)) |>
  ggplot(aes(x=vb,y=dist,fill=condit)) +
```

```
  geom_flat_violin(aes(fill=condit), position = position_nudge(x = 0.1, y = 0),
                   adjust = 1.5,trim = FALSE, alpha = .5, colour = NA) +
  geom_point(aes(x = as.numeric(vb) - 0.15, col=condit),
             position = position_jitter(width = 0.05, height = 0),
             size = 1, shape = 20, alpha=.7) +
  geom_boxplot(aes(x=vb,y=dist,fill=condit),
               outlier.shape = NA,
               alpha = 0.5, width = 0.1) +
  geom_hline(yintercept = 0,
             linetype = 'dashed',
             color = 'red',
             size = 0.4) +
  coord_flip() + ggtitle("Empirical Deviation - Subject level averaging")



epId
```

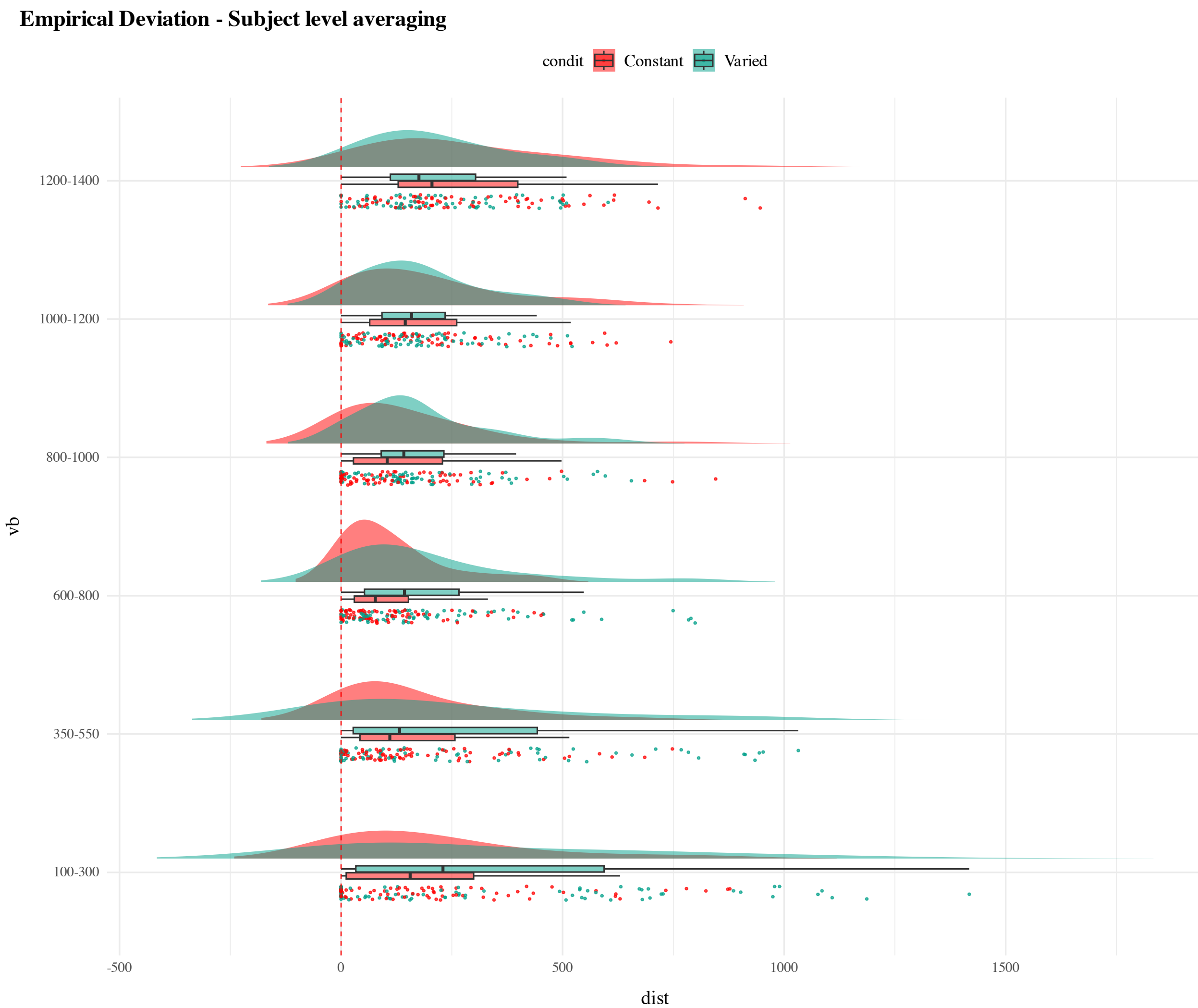

Figure 5: E1. Distribution of Vx at Participant level

```
epTrial <- dist_pred  |>
  filter(iter < 2)  |> group_by(id,condit,vb) |>
  ggplot(aes(x=vb,y=dist,fill=condit)) +
  geom_flat_violin(aes(fill=condit), position = position_nudge(x = 0.1, y = 0),
                   adjust = 1.5,trim = FALSE, alpha = .5, colour = NA) +
  geom_point(aes(x = as.numeric(vb) - 0.15, col=condit),
             position = position_jitter(width = 0.05, height = 0),
             size = .5, shape = 20, alpha=.7) +
  geom_boxplot(aes(x=vb,y=dist,fill=condit),
               outlier.shape = NA,
               alpha = 0.5, width = 0.1) +
  geom_hline(yintercept = 0,
             linetype = 'dashed',
             color = 'red',
             size = 0.4) +
  coord_flip() + ggtitle("Empirical Deviation - Raw Trial") +
```

```
  theme(axis.title.y=element_blank(),
        axis.text.y=element_blank())

epTrial
```

Figure 6: E1. Distribution of Vx at Trial level

```
new_data_grid=map_dfr(1, ~data.frame(unique(test[,c("id","condit","bandInt")])))

cSamp <- e1_distBMM  |>
  emmeans("condit",by="bandInt",at=list(bandInt=c(100,350,600,800,1000,1200)),
          epred = TRUE, re_formula = NA) |>
  pairs() |> gather_emmeans_draws()  |>
  group_by(contrast, .draw,bandInt) |> summarise(value=mean(.value), n=n())
```

```
 ameBand <- cSamp |> ggplot(aes(x=value,y="")) +
  stat_halfeye() +
  geom_vline(xintercept=0,alpha=.4)+
  facet_wrap(~bandInt,ncol=1) + labs(x="Marginal Effect (Constant - Varied)", y= NULL)+
  ggtitle("Average Marginal Effect")

bothConditGM <- e1_distBMM %>%
  epred_draws(newdata = new_data_grid,ndraws = 2000, re_formula = NA) |>
  ggplot(aes(x=.epred,y="Mean",fill=condit)) +
  stat_halfeye() +facet_wrap(~bandInt, ncol = 1) +
  labs(x="Predicted Deviation", y=NULL)+
  ggtitle("Grand Means") +theme(legend.position = "bottom")

(bothConditGM | ameBand) + plot_layout(widths=c(2,1.0))
```

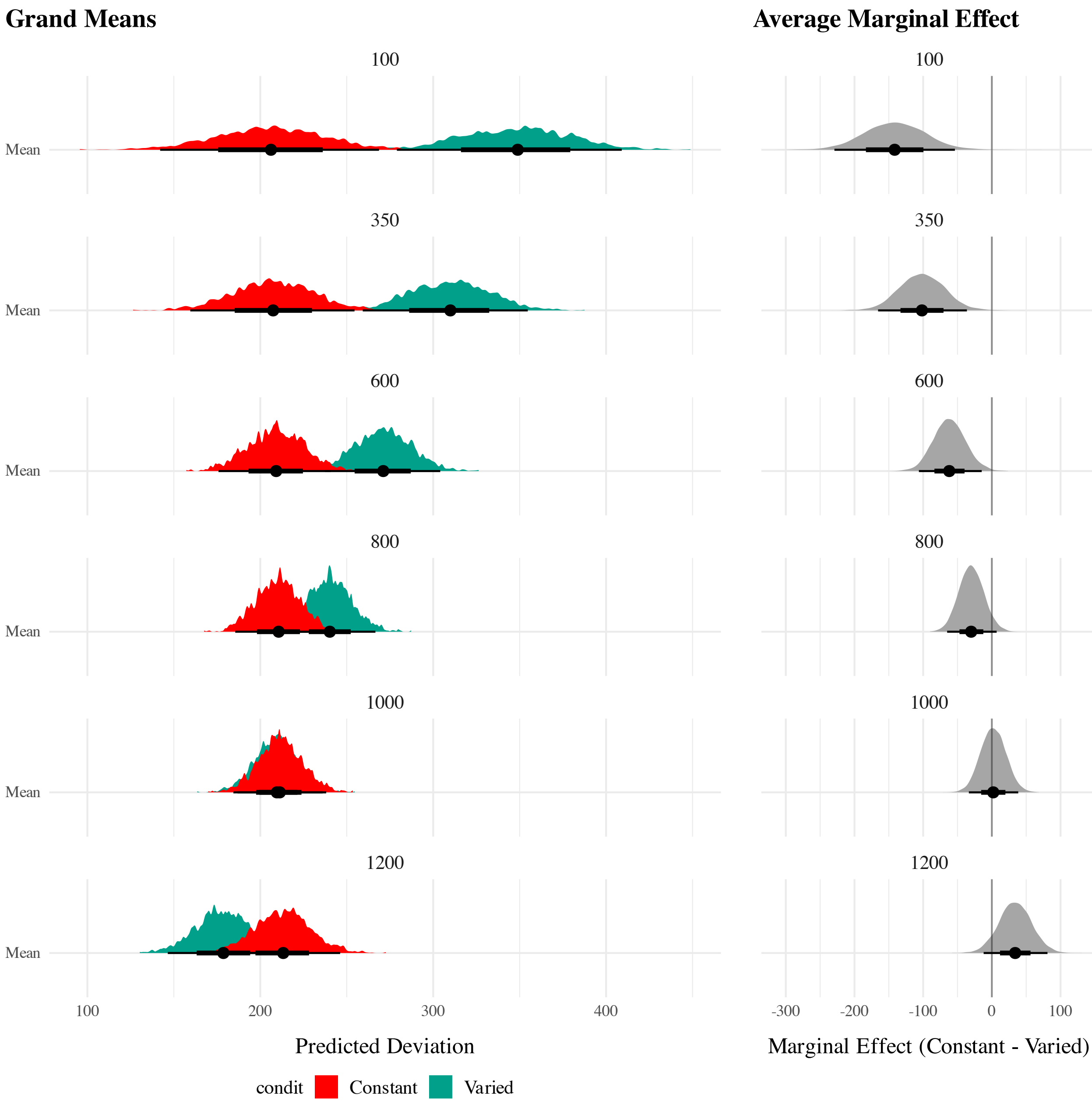

Figure 7: E1. Predicted Means Per Condition and Band, and Average Marginal Effect (Constant - Varied)

References

DeLosh, E. L., McDaniel, M. A., & Busemeyer, J. R. (1997). Extrapolation: The Sine Qua Non for Abstraction in Function Learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *23*(4), 19. https://doi.org/10.1037/0278-7393.23.4.968

Mcdaniel, M. A., Dimperio, E., Griego, J. A., & Busemeyer, J. R. (2009). Predicting transfer performance: A comparison of competing function learning models. *Journal of Experimental Psychology. Learning, Memory, and Cognition*, *35*, 173–195. https://doi.org/10.1037/a0013982