

JEP LMC: Lexical Retrieval Manuscript Analyses

Abhilasha Kumar

May 29, 2018

1 Reading the File

2 State Prime Accuracy Figures

Experiment 1

```
> exp1_fig_stateprime_acc = Rmisc::summarySE(exp_1_stateprime_acc,
+                                             measurevar = "Trials",
+                                             groupvars = c("AgeGroup", "PrimeCondition", "Question.RESP", "A
> exp1_fig_stateprime_acc = arrange(exp1_fig_stateprime_acc,
+                                   desc(AgeGroup))
> exp1_fig_stateprime_acc$Accuracy = as.factor(as.character(exp1_fig_stateprime_acc$Accu
> exp1_fig_stateprime_acc_young = exp1_fig_stateprime_acc %>% filter(AgeGroup == "Young")
> exp1_fig_stateprime_acc_old = exp1_fig_stateprime_acc %>% filter(AgeGroup == "Old")
> library(ggplot2)
> library(ggthemes)
> stateprime_1_acc_young = exp1_fig_stateprime_acc_young %>%
+   mutate(State = factor(Question.RESP, levels = unique(Question.RESP),
+                     labels = c("Know", "Dont Know", "Other", "TOT")),
+          PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                     labels = c("Both", "Phonological", "Semantic", "Unrelated")),
+          TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+                     labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+   ggplot(aes(x = PrimeType, y = Trials,
+             group=TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   facet_wrap(~State, nrow =1)+
+   scale_fill_wsj()+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("") +
+   ggtitle("E1: Young (Without Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
```

```

+       legend.title = element_text(face = "bold", size = rel(1)),
+       plot.title = element_text(hjust = .5),
+       strip.text.x = element_text(face = "bold", size = rel(1.4)))
> stateprime_1_acc_young
> stateprime_1_acc_old = exp1_fig_stateprime_acc_old %>%
+   mutate(State = factor(Question.RESP, levels = unique(Question.RESP),
+     labels = c("Know", "Dont Know", "Other", "TOT")),
+     PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+     labels = c("Both", "Phonological", "Semantic", "Unrelated")),
+     TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+     labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+ ggplot(aes(x = PrimeType, y = Trials,
+   group=TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+     width=.2, color = "gray26",
+     position = position_dodge(0.7))+
+   theme_few()+
+   facet_wrap(~State, nrow =1)+
+   scale_fill_wsjs()+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("") +
+   ggtitle("E1: Old (Without Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+     axis.title = element_text(face = "bold", size = rel(1)),
+     legend.title = element_text(face = "bold", size = rel(1)),
+     plot.title = element_text(hjust = .5),
+     strip.text.x = element_text(face = "bold", size = rel(1.4)))
> stateprime_1_acc_old

```

2.0.1 E1: Combined Plot

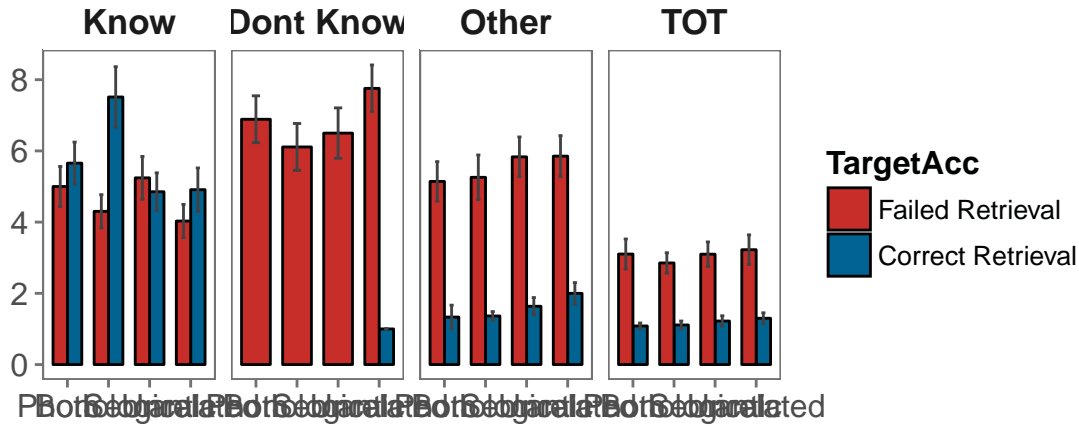
```

> library(grid)
> gridExtra::grid.arrange(stateprime_1_acc_young, stateprime_1_acc_old,
+   nrow = 2, ncol = 1,
+   top=textGrob("Raw Number of Retrieval States in E1",
+     gp=gpar(fontsize=20)))

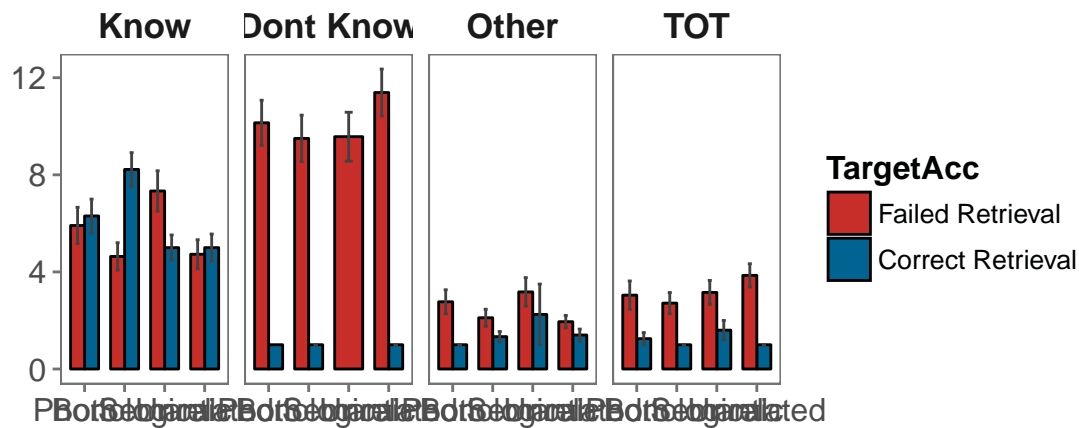
```

Raw Number of Retrieval States in E1

E1: Young (Without Instructions)



E1: Old (Without Instructions)



Experiment 2

```
> exp2_fig_stateprime_acc = Rmisc::summarySE(exp_2_stateprime_acc,
+                                           measurevar = "Trials",
+                                           groupvars = c("AgeGroup", "PrimeCondition", "Question.RESP", "A
+
> exp2_fig_stateprime_acc = arrange(exp2_fig_stateprime_acc,
+                                   desc(AgeGroup))
> exp2_fig_stateprime_acc$Accuracy = as.factor(as.character(exp2_fig_stateprime_acc$Accu
> exp2_fig_stateprime_acc_young = exp2_fig_stateprime_acc %>% filter(AgeGroup == "Young")
> exp2_fig_stateprime_acc_old = exp2_fig_stateprime_acc %>% filter(AgeGroup == "Old")
> library(ggplot2)
```

```

> library(ggthemes)
> stateprime_2_acc_young = exp2_fig_stateprime_acc_young %>%
+   mutate(State = factor(Question.RESP, levels = unique(Question.RESP),
+     labels = c("Know", "Dont Know", "Other", "TOT")),
+     PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+       labels = c("Both", "Phonological", "Semantic", "Unrelated")),
+     TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+       labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+ ggplot(aes(x = State, y = Trials,
+   group=TargetAcc, fill = TargetAcc))+
+ geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+ geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+   width=.2, color = "gray26",
+   position = position_dodge(0.7))+
+ theme_few()+
+ facet_wrap(~PrimeType, nrow =1)+
+ scale_fill_wsj()+
+ # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+ xlab("") + ylab("") +
+ ggtitle("E2: Young (With Instructions)") +
+ theme(axis.text = element_text(size = rel(1)),
+   axis.title = element_text(face = "bold", size = rel(1)),
+   legend.title = element_text(face = "bold", size = rel(1)),
+   plot.title = element_text(hjust = .5),
+   strip.text.x = element_text(face = "bold", size = rel(1.4)))
> stateprime_2_acc_young
> stateprime_2_acc_old = exp2_fig_stateprime_acc_old %>%
+   mutate(State = factor(Question.RESP, levels = unique(Question.RESP),
+     labels = c("Know", "Dont Know", "Other", "TOT")),
+     PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+       labels = c("Both", "Phonological", "Semantic", "Unrelated")),
+     TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+       labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+ ggplot(aes(x = State, y = Trials,
+   group=TargetAcc, fill = TargetAcc))+
+ geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+ geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+   width=.2, color = "gray26",
+   position = position_dodge(0.7))+
+ theme_few()+
+ facet_wrap(~PrimeType, nrow =1)+
+ scale_fill_wsj()+
+ # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+ xlab("") + ylab("") +
+ ggtitle("E2: Old (With Instructions)") +
+ theme(axis.text = element_text(size = rel(1)),
+   axis.title = element_text(face = "bold", size = rel(1)),
+   legend.title = element_text(face = "bold", size = rel(1)),

```

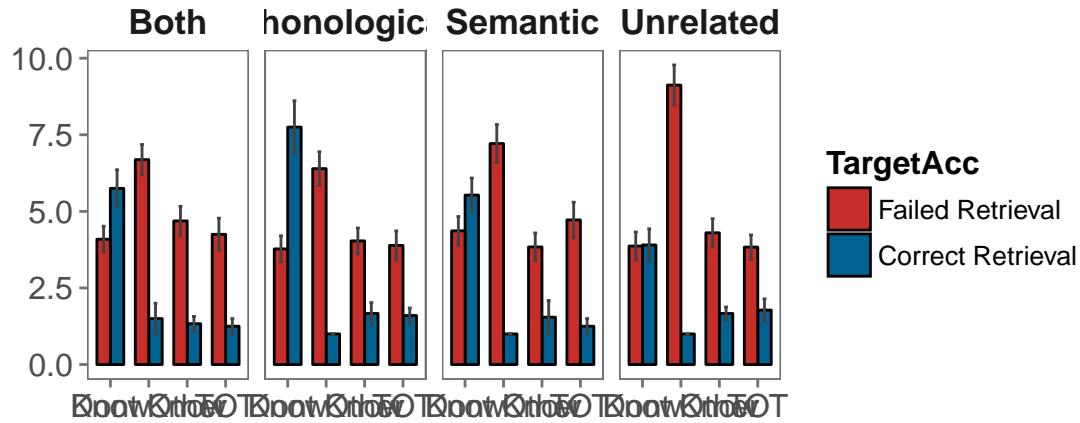
```
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> stateprime_2_acc_old
```

2.0.2 E2: Combined Plot

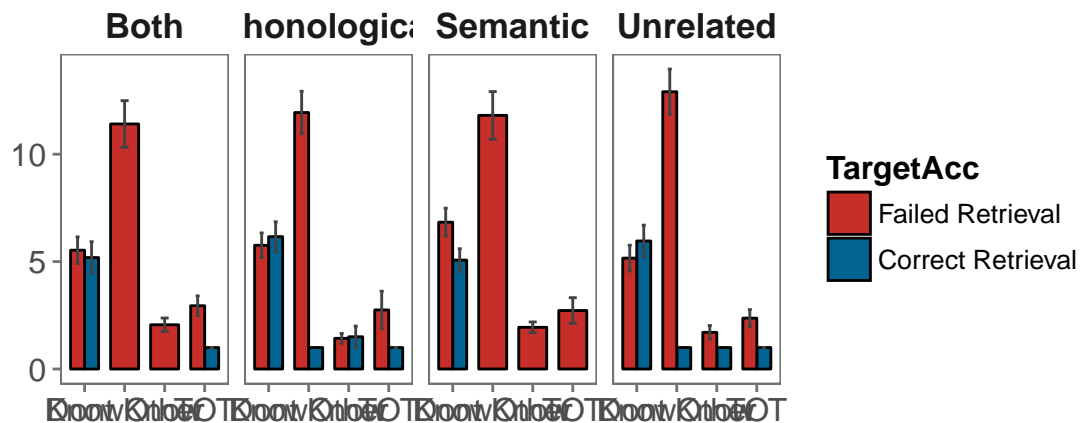
```
> library(grid)
> gridExtra::grid.arrange(stateprime_2_acc_young, stateprime_2_acc_old,
+                           nrow = 2, ncol = 1,
+                           top=textGrob("Raw Number of Retrieval States in E2",
+                                         gp=gpar(fontsize=20)))
```

Raw Number of Retrieval States in E2

E2: Young (With Instructions)



E2: Old (With Instructions)



Experiment 3

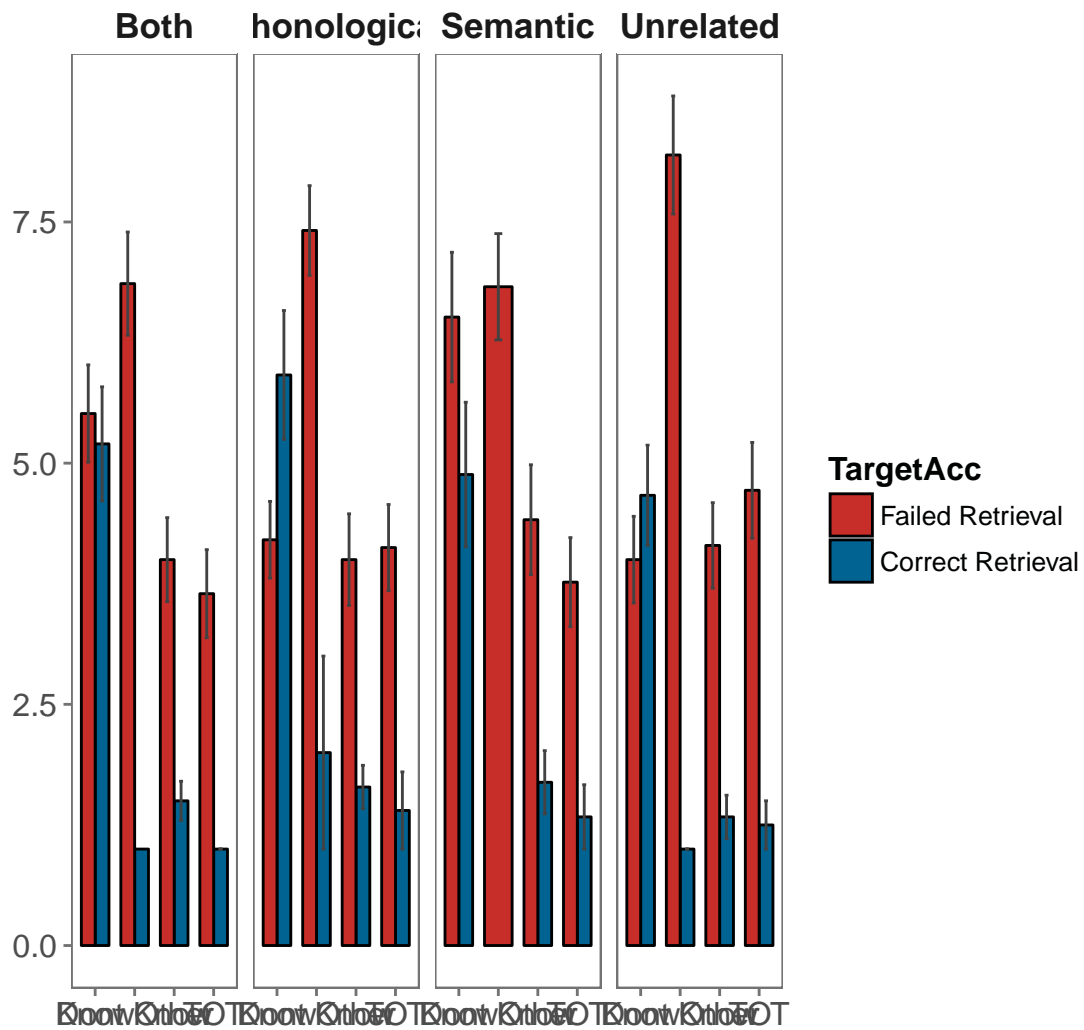
```
> exp3_fig_stateprime_acc = Rmisc::summarySE(exp_3_stateprime_acc,
+                                           measurevar = "Trials",
+                                           groupvars = c("AgeGroup", "PrimeCondition", "Question.RESP", "A
> exp3_fig_stateprime_acc$Accuracy = as.factor(as.character(exp3_fig_stateprime_acc$Accu
> library(ggplot2)
> library(ggthemes)
> stateprime_3_acc_young = exp3_fig_stateprime_acc %>%
+   mutate(State = factor(Question.RESP, levels = unique(Question.RESP),
+   labels = c("Know", "Dont Know", "Other", "TOT")),
```

```

+       PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+       labels = c("Both", "Phonological", "Semantic", "Unrelated")),
+       TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+       labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+ ggplot(aes(x = State, y = Trials,
+       group=TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+       width=.2, color = "gray26",
+       position = position_dodge(0.7))+
+   theme_few()+
+   facet_wrap(~PrimeType, nrow =1)+
+   scale_fill_wsj()+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("") +
+   ggtitle("E3: Young Adults Only (48 ms)") +
+   theme(axis.text = element_text(size = rel(1)),
+       axis.title = element_text(face = "bold", size = rel(1)),
+       legend.title = element_text(face = "bold", size = rel(1)),
+       plot.title = element_text(hjust = .5),
+       strip.text.x = element_text(face = "bold", size = rel(1.4)))
> stateprime_3_acc_young
>

```

E3: Young Adults Only (48 ms)



3 Know: PrimeType and Target Accuracy

```
> exp_1_knowacc = exp_1_stateprime_acc %>% filter(Question.RESP == "1")
> exp_2_knowacc = exp_2_stateprime_acc %>% filter(Question.RESP == "1")
> exp_3_knowacc = exp_3_stateprime_acc %>% filter(Question.RESP == "1")
```

3.1 Experiment 1

```
> ## HLM on trials
```



```

>
> library(lme4)
> contrasts(exp_1_knowacc$PrimeCondition) = contr.treatment(4, base = 3)
> exp_1_knowacc$Accuracy = as.factor(exp_1_knowacc$Accuracy)
> e1_know_hlm = lmer(data = exp_1_knowacc,
+                   Trials ~ AgeGroup*PrimeCondition*Accuracy +
+                   (1|Subject))
> summary(e1_know_hlm)

```

```

Linear mixed model fit by REML ['lmerMod']
Formula: Trials ~ AgeGroup * PrimeCondition * Accuracy + (1 | Subject)
Data: exp_1_knowacc

```

```
REML criterion at convergence: 2921.7
```

```
Scaled residuals:
```

Min	1Q	Median	3Q	Max
-1.8534	-0.6950	-0.1191	0.5296	3.1589

```
Random effects:
```

Groups	Name	Variance	Std.Dev.
Subject	(Intercept)	2.86	1.691
Residual		10.83	3.291

```
Number of obs: 550, groups: Subject, 73
```

```
Fixed effects:
```

	Estimate	Std. Error	t value
(Intercept)	7.3052	0.6406	11.404
AgeGroupYoung	-2.0620	0.8834	-2.334
PrimeCondition1	-1.4712	0.8060	-1.825
PrimeCondition2	-2.8533	0.8126	-3.511
PrimeCondition4	-2.7104	0.8137	-3.331
Accuracy1	-2.3937	0.8071	-2.966
AgeGroupYoung:PrimeCondition1	1.1784	1.1154	1.057
AgeGroupYoung:PrimeCondition2	1.8346	1.1334	1.619
AgeGroupYoung:PrimeCondition4	1.3926	1.1343	1.228
AgeGroupYoung:Accuracy1	1.8862	1.1249	1.677
PrimeCondition1:Accuracy1	2.8652	1.1273	2.542
PrimeCondition2:Accuracy1	6.1640	1.1319	5.445
PrimeCondition4:Accuracy1	2.8145	1.1414	2.466
AgeGroupYoung:PrimeCondition1:Accuracy1	-1.7061	1.5792	-1.080
AgeGroupYoung:PrimeCondition2:Accuracy1	-2.4481	1.5931	-1.537
AgeGroupYoung:PrimeCondition4:Accuracy1	-1.3887	1.6024	-0.867

```
> car::Anova(e1_know_hlm)
```

```
Analysis of Deviance Table (Type II Wald chisquare tests)
```

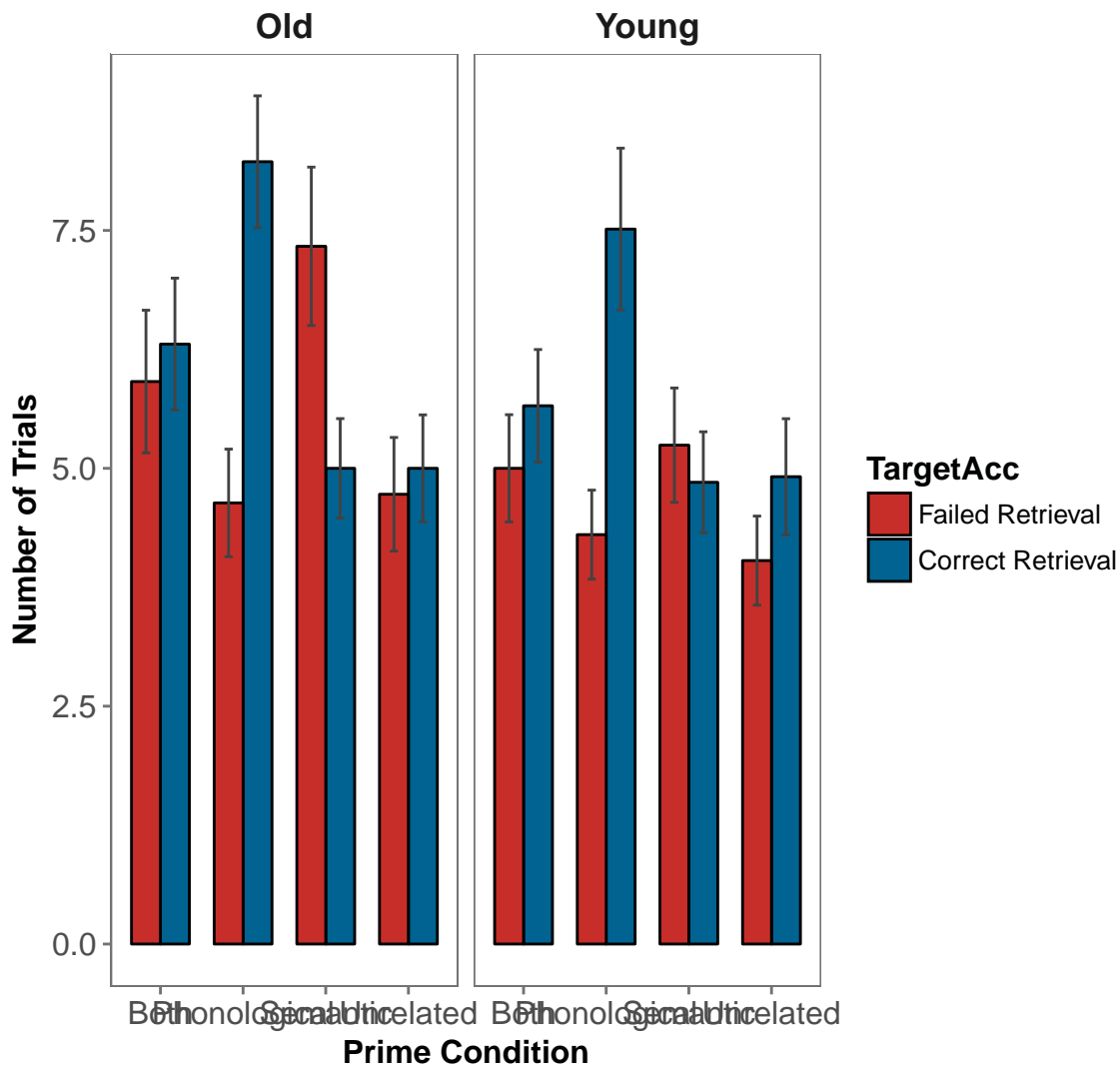
Response: Trials

	Chisq	Df	Pr(>Chisq)
AgeGroup	2.1831	1	0.139531
PrimeCondition	15.6097	3	0.001363 **
Accuracy	8.4850	1	0.003581 **
AgeGroup:PrimeCondition	0.9372	3	0.816441
AgeGroup:Accuracy	0.7839	1	0.375963
PrimeCondition:Accuracy	38.7657	3	1.946e-08 ***
AgeGroup:PrimeCondition:Accuracy	2.4898	3	0.477146

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> ## plotting
> library(ggplot2)
> library(ggthemes)
> e1_know_data = Rmisc::summarySE(exp_1_knowacc,
+                               measurevar = "Trials",
+                               groupvars = c("AgeGroup", "PrimeCondition", "Accuracy"))
> e1_know_plot = e1_know_data %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Both", "Phonological", "Semantic", "Unrelated")),
+          TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+                             labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+   ggplot(aes(x = PrimeType, y = Trials,
+              group=TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color = "black")+
+   geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_wsj()+
+   facet_wrap(~AgeGroup)+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Prime Condition") + ylab("Number of Trials") +
+   ggtitle("E1: Know Responses in Young and Old Adults (Without Instructions)")
+
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e1_know_plot
```

now Responses in Young and Old Adults (Without Instructions)



3.2 Experiment 1 Collapsed Prime

```
> ## HLM on trials
> exp_1_knowacc2 = exp_1_state_acc %>% filter(Question.RESP == "1")
> library(lme4)
> exp_1_knowacc2$Accuracy = as.factor(exp_1_knowacc2$Accuracy)
> e1_know_hlm = lmer(data = exp_1_knowacc2,
+                   Trials ~ AgeGroup*Accuracy +
+                   (1|Subject))
> summary(e1_know_hlm)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: Trials ~ AgeGroup * Accuracy + (1 | Subject)
Data: exp_1_knowacc2
```

```
REML criterion at convergence: 1136.8
```

```
Scaled residuals:
```

Min	1Q	Median	3Q	Max
-1.6223	-0.7381	-0.1749	0.5987	4.0582

```
Random effects:
```

Groups	Name	Variance	Std.Dev.
Subject	(Intercept)	4.278e-14	2.068e-07
Residual		1.586e+02	1.259e+01

```
Number of obs: 146, groups: Subject, 73
```

```
Fixed effects:
```

	Estimate	Std. Error	t value
(Intercept)	20.8889	2.0991	9.951
AgeGroupYoung	-3.3483	2.9485	-1.136
Accuracy1	3.0833	2.9686	1.039
AgeGroupYoung:Accuracy1	0.8086	4.1698	0.194

```
Correlation of Fixed Effects:
```

	(Intr)	AgGrpY	Accrc1
AgeGroupYng	-0.712		
Accuracy1	-0.707	0.503	
AgGrpYng:A1	0.503	-0.707	-0.712

```
> car::Anova(e1_know_hlm)
```

```
Analysis of Deviance Table (Type II Wald chisquare tests)
```

```
Response: Trials
```

	Chisq	Df	Pr(>Chisq)
AgeGroup	1.9940	1	0.15792
Accuracy	2.8077	1	0.09381
AgeGroup:Accuracy	0.0376	1	0.84625

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## percents
```

```
>
```

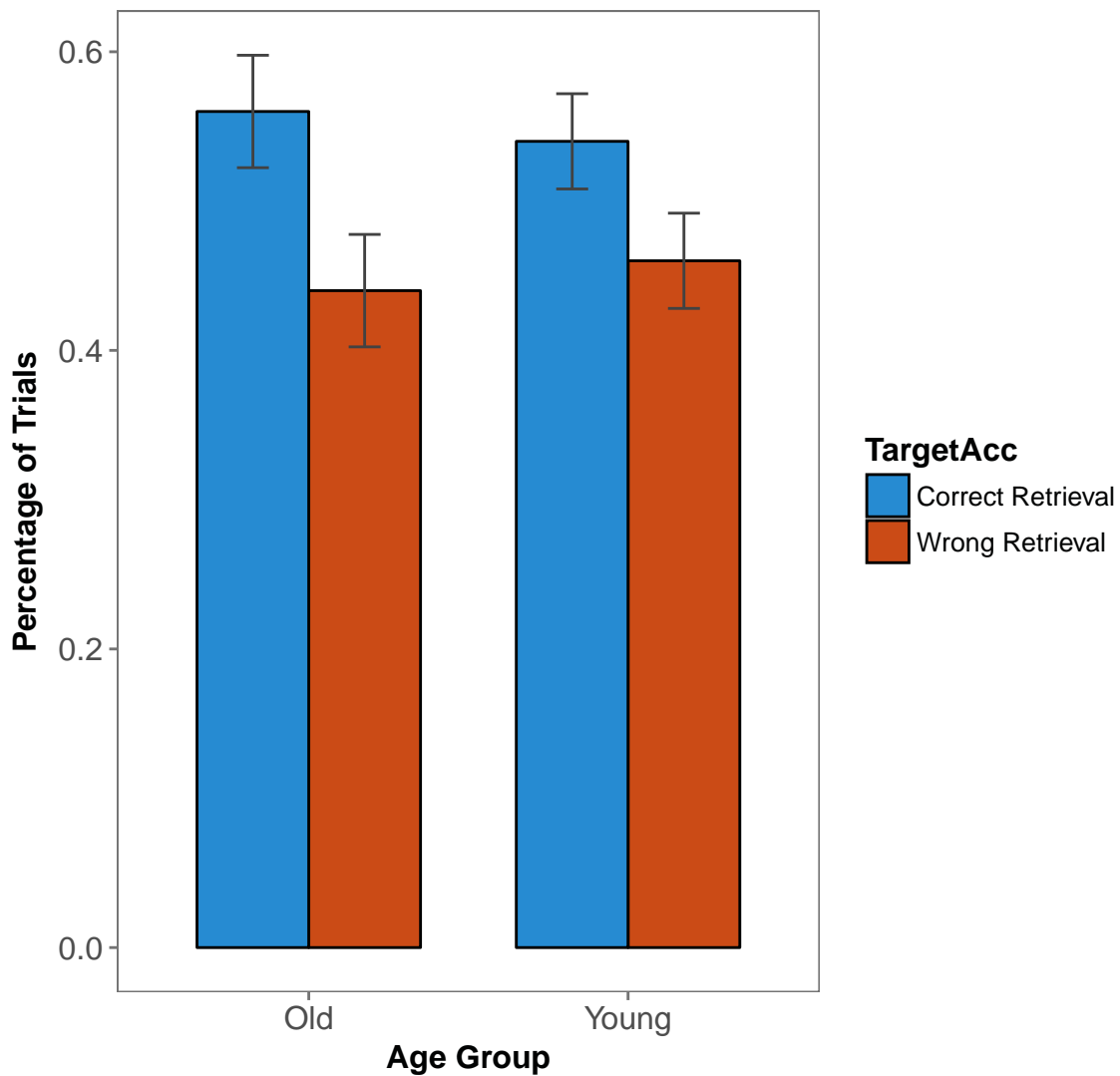
```
> exp1_knowacc_percent = spread(exp_1_knowacc2, Accuracy, Trials)
> exp1_knowacc_percent$`0` = ifelse(is.na(exp1_knowacc_percent$`0`),
+                                   0, exp1_knowacc_percent$`0`)
> exp1_knowacc_percent$`1` = ifelse(is.na(exp1_knowacc_percent$`1`),
```

```

+                                0, exp1_knowacc_percent$`1`)
> exp1_knowacc_percent$total = exp1_knowacc_percent$`0` + exp1_knowacc_percent$`1`
> exp1_knowacc_percent$pcorrect = exp1_knowacc_percent$`1`/exp1_knowacc_percent$total
> exp1_knowacc_percent$pwrong = exp1_knowacc_percent$`0`/exp1_knowacc_percent$total
> exp1_knowacc_long = exp1_knowacc_percent %>% gather(Type,
+                                     Percent,
+                                     pcorrect:pwrong)%>%
+   arrange(Subject)
> ## plotting
> library(ggplot2)
> library(ggthemes)
> e1_know_data = Rmisc::summarySE(exp1_knowacc_long,
+                                measurevar = "Percent",
+                                groupvars = c( "AgeGroup", "Type"))
> e1_know_data$Percent = round(e1_know_data$Percent, 2)
> e1_know_plot = e1_know_data %>%
+   mutate(TargetAcc = factor(Type, levels = unique(Type),
+                               labels = c("Correct Retrieval", "Wrong Retrieval")))%>%
+   ggplot(aes(x = AgeGroup, y = Percent,
+              group=TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_solarized()+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Age Group") + ylab("Percentage of Trials") +
+   ggtitle("E1: Know Responses in Young and Old Adults (Without Instructions)")
+
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e1_know_plot

```

now Responses in Young and Old Adults (Without Instructions)



3.3 Experiment 2

```
> ### EXPERIMENT 2
>
> contrasts(exp_2_knowacc$PrimeCondition) = contr.treatment(4, base = 3)
> e2_know_hlm = lmer(data = exp_2_knowacc, Trials ~ PrimeCondition*Accuracy +
+                   (1|Subject))
> summary(e2_know_hlm)
```

```
Linear mixed model fit by REML ['lmerMod']
```

```
Formula: Trials ~ PrimeCondition * Accuracy + (1 | Subject)
Data: exp_2_knowacc
```

REML criterion at convergence: 2546.2

Scaled residuals:

Min	1Q	Median	3Q	Max
-2.1487	-0.6850	-0.1042	0.5970	3.2616

Random effects:

Groups	Name	Variance	Std.Dev.
Subject	(Intercept)	2.874	1.695
Residual		8.853	2.975

Number of obs: 493, groups: Subject, 65

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	5.4792	0.4304	12.730
PrimeCondition1	-0.6792	0.5265	-1.290
PrimeCondition2	-0.9015	0.5380	-1.675
PrimeCondition4	-1.0417	0.5354	-1.945
Accuracy	-0.2416	0.5360	-0.451
PrimeCondition1:Accuracy	0.8871	0.7497	1.183
PrimeCondition2:Accuracy	2.5611	0.7594	3.372
PrimeCondition4:Accuracy	0.4343	0.7709	0.563

Correlation of Fixed Effects:

	(Intr)	PrmCn1	PrmCn2	PrmCn4	Accrcy	PrC1:A	PrC2:A
PrimeCndtn1	-0.622						
PrimeCndtn2	-0.608	0.497					
PrimeCndtn4	-0.611	0.499	0.489				
Accuracy	-0.612	0.500	0.490	0.492			
PrmCndtn1:A	0.437	-0.702	-0.350	-0.351	-0.715		
PrmCndtn2:A	0.431	-0.352	-0.709	-0.346	-0.706	0.504	
PrmCndtn4:A	0.426	-0.348	-0.340	-0.696	-0.696	0.498	0.490

```
> car::Anova(e2_know_hlm)
```

Analysis of Deviance Table (Type II Wald chisquare tests)

Response: Trials

	Chisq	Df	Pr(>Chisq)
PrimeCondition	10.4190	3	0.015321 *
Accuracy	7.4677	1	0.006282 **
PrimeCondition:Accuracy	12.9769	3	0.004687 **

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> e2_know_hlm_age = lmer(data = exp_2_knowacc, Trials ~ AgeGroup*PrimeCondition*Accuracy
+ (1|Subject))
> summary(e2_know_hlm_age)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: Trials ~ AgeGroup * PrimeCondition * Accuracy + (1 | Subject)
Data: exp_2_knowacc
```

```
REML criterion at convergence: 2507.6
```

```
Scaled residuals:
```

Min	1Q	Median	3Q	Max
-1.9299	-0.6912	-0.1097	0.6052	2.9701

```
Random effects:
```

Groups	Name	Variance	Std.Dev.
Subject	(Intercept)	2.767	1.663
Residual		8.506	2.917

Number of obs: 493, groups: Subject, 65

```
Fixed effects:
```

	Estimate	Std. Error	t value
(Intercept)	6.7050	0.6101	10.990
AgeGroupYoung	-2.3414	0.8449	-2.771
PrimeCondition1	-1.1737	0.7427	-1.580
PrimeCondition2	-1.1812	0.7620	-1.550
PrimeCondition4	-1.6047	0.7477	-2.146
Accuracy	-1.7736	0.7633	-2.323
AgeGroupYoung:PrimeCondition1	0.9010	1.0330	0.872
AgeGroupYoung:PrimeCondition2	0.5085	1.0558	0.482
AgeGroupYoung:PrimeCondition4	1.0162	1.0503	0.967
AgeGroupYoung:Accuracy	2.8995	1.0523	2.756
PrimeCondition1:Accuracy	1.4298	1.0556	1.355
PrimeCondition2:Accuracy	2.3312	1.0725	2.174
PrimeCondition4:Accuracy	2.1614	1.0978	1.969
AgeGroupYoung:PrimeCondition1:Accuracy	-0.9357	1.4709	-0.636
AgeGroupYoung:PrimeCondition2:Accuracy	0.5390	1.4900	0.362
AgeGroupYoung:PrimeCondition4:Accuracy	-3.1485	1.5149	-2.078

```
> car::Anova(e2_know_hlm_age)
```

```
Analysis of Deviance Table (Type II Wald chisquare tests)
```

```
Response: Trials
```

	Chisq	Df	Pr(>Chisq)
AgeGroup	2.0842	1	0.1488347
PrimeCondition	11.4279	3	0.0096235 **


```

Accuracy                8.0898    1    0.0044515  **
AgeGroup:PrimeCondition    3.2889    3    0.3491918
AgeGroup:Accuracy         15.0859    1    0.0001027  ***
PrimeCondition:Accuracy    13.6292    3    0.0034559  **
AgeGroup:PrimeCondition:Accuracy  6.8366    3    0.0772906  .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> ## only older adults
> library(lmerTest)
> exp_2_knowacc_old = exp_2_knowacc %>% filter(AgeGroup == "Old")
> e2_know_hlm_old = lmer(data = exp_2_knowacc_old, Trials ~ PrimeCondition*Accuracy +
+ (1|Subject))
> summary(e2_know_hlm_old)

```

Linear mixed model fit by REML t-tests use Satterthwaite approximations to degrees of freedom [lmerMod]

Formula: Trials ~ PrimeCondition * Accuracy + (1 | Subject)
Data: exp_2_knowacc_old

REML criterion at convergence: 1237.1

Scaled residuals:

Min	1Q	Median	3Q	Max
-1.75096	-0.69565	0.00181	0.57269	2.70216

Random effects:

Groups	Name	Variance	Std.Dev.
Subject	(Intercept)	3.514	1.875
Residual		9.207	3.034

Number of obs: 239, groups: Subject, 32

Fixed effects:

	Estimate	Std. Error	df	t value	Pr(> t)
(Intercept)	6.6975	0.6475	158.3600	10.344	<2e-16 ***
PrimeCondition1	-1.1663	0.7727	199.2600	-1.509	0.1328
PrimeCondition2	-1.1878	0.7928	199.8000	-1.498	0.1357
PrimeCondition4	-1.6007	0.7779	198.8700	-2.058	0.0409 *
Accuracy	-1.7740	0.7944	200.4500	-2.233	0.0266 *
PrimeCondition1:Accuracy	1.4303	1.0984	199.5000	1.302	0.1944
PrimeCondition2:Accuracy	2.3402	1.1159	199.4100	2.097	0.0372 *
PrimeCondition4:Accuracy	2.1394	1.1425	200.5400	1.873	0.0626 .

```

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Correlation of Fixed Effects:

	(Intr)	PrmCn1	PrmCn2	PrmCn4	Accrcy	PrC1:A	PrC2:A
PrimeCndtn1	-0.618						

```

PrimeCndtn2 -0.601 0.503
PrimeCndtn4 -0.612 0.513 0.498
Accuracy -0.602 0.505 0.492 0.499
PrmCndtn1:A 0.435 -0.704 -0.356 -0.361 -0.723
PrmCndtn2:A 0.427 -0.358 -0.711 -0.354 -0.710 0.514
PrmCndtn4:A 0.420 -0.352 -0.339 -0.683 -0.696 0.503 0.492

```

```
> car::Anova(e2_know_hlm_old)
```

Analysis of Deviance Table (Type II Wald chisquare tests)

Response: Trials

	Chisq	Df	Pr(>Chisq)
PrimeCondition	1.7447	3	0.6270
Accuracy	0.5845	1	0.4445
PrimeCondition:Accuracy	5.3347	3	0.1489

```

> ## only young adults
> exp_2_knowacc_young = exp_2_knowacc %>% filter(AgeGroup == "Young")
> e2_know_hlm_young = lmer(data = exp_2_knowacc_young, Trials ~ PrimeCondition*Accuracy
+ (1|Subject))
> summary(e2_know_hlm_young)

```

Linear mixed model fit by REML t-tests use Satterthwaite approximations to degrees of freedom [lmerMod]

Formula: Trials ~ PrimeCondition * Accuracy + (1 | Subject)
Data: exp_2_knowacc_young

REML criterion at convergence: 1267.9

Scaled residuals:

Min	1Q	Median	3Q	Max
-2.0332	-0.6085	-0.1436	0.6315	3.1273

Random effects:

Groups	Name	Variance	Std.Dev.
Subject	(Intercept)	2.080	1.442
Residual		7.842	2.800

Number of obs: 254, groups: Subject, 33

Fixed effects:

	Estimate	Std. Error	df	t value	Pr(> t)
(Intercept)	4.3636	0.5483	189.5000	7.958	1.55e-13 ***
PrimeCondition1	-0.2727	0.6894	213.8200	-0.396	0.6928
PrimeCondition2	-0.6665	0.7016	214.7000	-0.950	0.3432
PrimeCondition4	-0.5816	0.7081	215.0300	-0.821	0.4124
Accuracy	1.1292	0.6954	214.3100	1.624	0.1059
PrimeCondition1:Accuracy	0.4936	0.9834	214.2800	0.502	0.6162

```

PrimeCondition2:Accuracy    2.8652      0.9929 215.2100    2.886    0.0043 **
PrimeCondition4:Accuracy   -0.9981      1.0019 215.5600   -0.996    0.3203
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
      (Intr) PrmCn1 PrmCn2 PrmCn4 Accrcy PrC1:A PrC2:A
PrimeCndtn1 -0.629
PrimeCndtn2 -0.618  0.491
PrimeCndtn4 -0.612  0.487  0.480
Accuracy    -0.623  0.496  0.489  0.484
PrmCndtn1:A  0.441 -0.701 -0.345 -0.343 -0.707
PrmCndtn2:A  0.436 -0.347 -0.708 -0.339 -0.702  0.495
PrmCndtn4:A  0.433 -0.344 -0.341 -0.708 -0.695  0.492  0.488

```

```
> car::Anova(e2_know_hlm_young)
```

```
Analysis of Deviance Table (Type II Wald chisquare tests)
```

```
Response: Trials
```

```

              Chisq Df Pr(>Chisq)
PrimeCondition    14.047  3    0.002842 **
Accuracy          24.128  1    9.014e-07 ***
PrimeCondition:Accuracy 15.942  3    0.001166 **

```

```

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

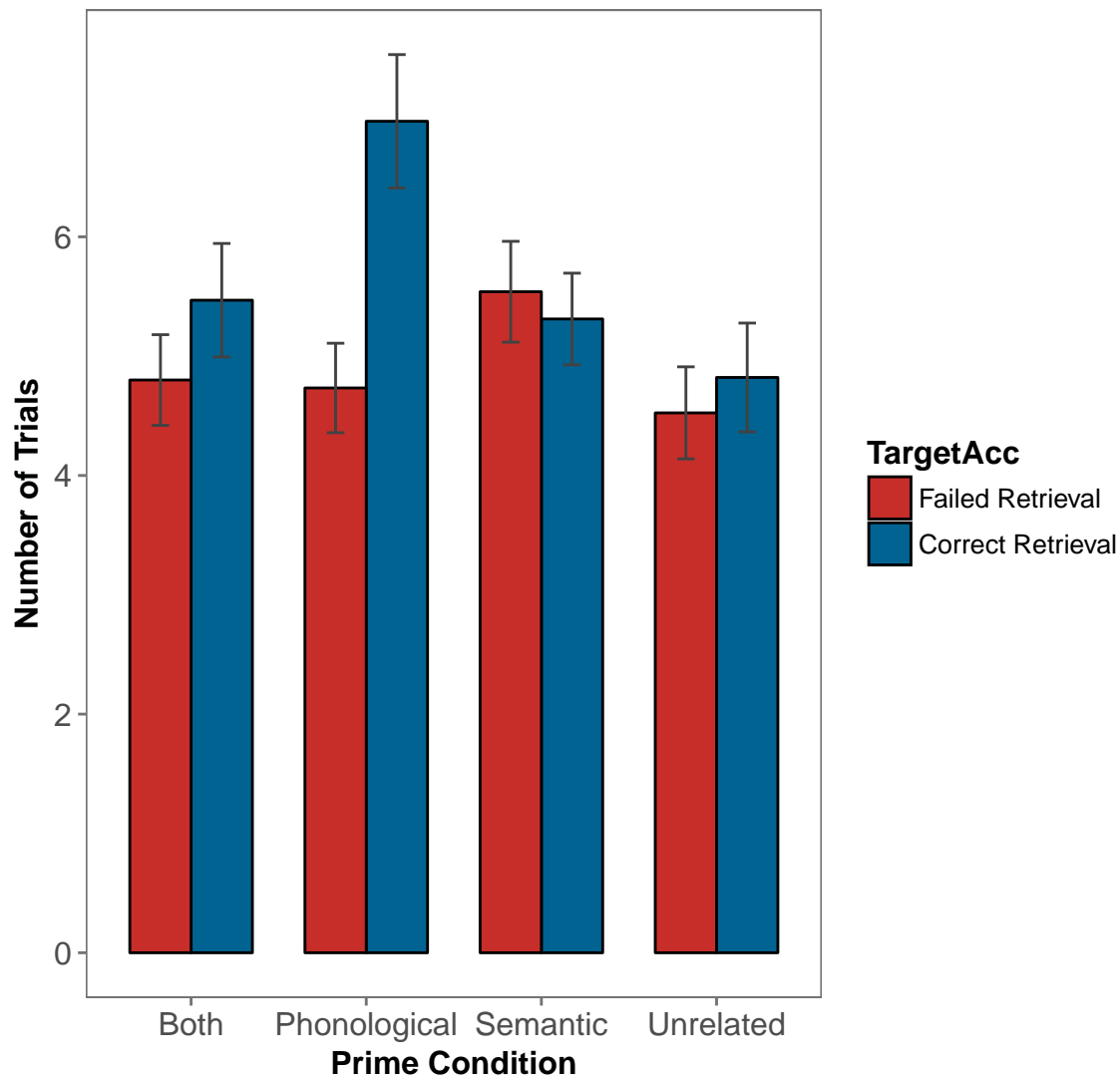
```

> #sjPlot::plot_model(e2_know_hlm, type = "int")
>
> ## plotting
>
> e2_know_data = Rmisc::summarySE(exp_2_knowacc,
+                               measurevar = "Trials",
+                               groupvars = c("PrimeCondition", "Accuracy"))
> e2_know_plot = e2_know_data %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Both", "Phonological", "Semantic", "Unrelated")),
+          TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+                             labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+   ggplot(aes(x = PrimeType, y = Trials,
+              group=TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_wsj()+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+

```

```
+ xlab("Prime Condition") + ylab("Number of Trials") +
+ ggtitle("E2: Know Responses in Young and Old Adults (With Instructions)") +
+ theme(axis.text = element_text(size = rel(1)),
+       axis.title = element_text(face = "bold", size = rel(1)),
+       legend.title = element_text(face = "bold", size = rel(1)),
+       plot.title = element_text(hjust = .5),
+       strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e2_know_plot
```

Know Responses in Young and Old Adults (With Instructions)



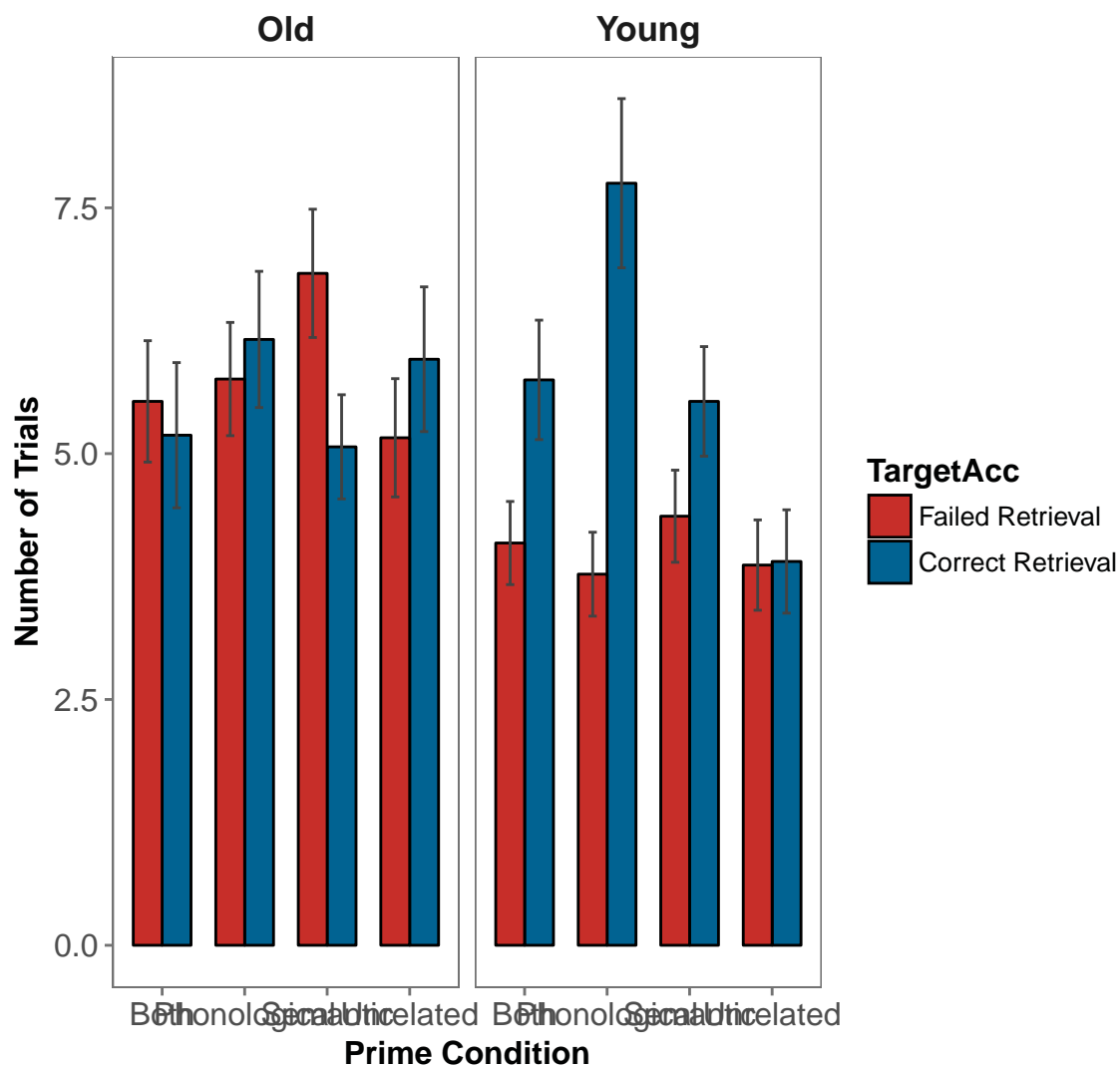
```
> e2_know_data_age = Rmisc::summarySE(exp_2_knowacc,
+                                     measurevar = "Trials",
```

```

+           groupvars = c("AgeGroup", "PrimeCondition", "Accuracy"))
> e2_know_plot_age = e2_know_data_age %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+     labels = c("Both", "Phonological", "Semantic", "Unrelated")),
+     TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+     labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+   ggplot(aes(x = PrimeType, y = Trials,
+     group=TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+     width=.2, color = "gray26",
+     position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_wsj()+
+   facet_wrap(~AgeGroup)+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Prime Condition") + ylab("Number of Trials") +
+   ggtitle("E2: Know Responses in Young and Old Adults (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+     axis.title = element_text(face = "bold", size = rel(1)),
+     legend.title = element_text(face = "bold", size = rel(1)),
+     plot.title = element_text(hjust = .5),
+     strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e2_know_plot_age

```

Know Responses in Young and Old Adults (With Instructions)



3.4 Experiment 2 Collapsed Prime

```
> ## HLM on trials
> exp_2_knowacc2 = exp_2_state_acc %>% filter(Question.RESP == "1")
> library(lme4)
> exp_2_knowacc2$Accuracy = as.factor(exp_2_knowacc2$Accuracy)
> e2_know_hlm = lmer(data = exp_2_knowacc2,
+                   Trials ~ AgeGroup*Accuracy +
+                   (1|Subject))
> summary(e2_know_hlm)
```

```

Linear mixed model fit by REML t-tests use Satterthwaite approximations to
degrees of freedom [lmerMod]
Formula: Trials ~ AgeGroup * Accuracy + (1 | Subject)
Data: exp_2_knowacc2

REML criterion at convergence: 979.1

Scaled residuals:
    Min       1Q   Median       3Q      Max
-1.4485 -0.7203 -0.1295  0.5245  2.2902

Random effects:
   Groups      Name      Variance Std.Dev.
   Subject (Intercept) 22.84      4.779
   Residual              103.52     10.175
Number of obs: 130, groups: Subject, 65

Fixed effects:
              Estimate Std. Error      df t value Pr(>|t|)
(Intercept)      22.156      1.987 122.010   11.150   <2e-16 ***
AgeGroupYoung     -6.641      2.789 122.010    -2.381    0.0188 *
Accuracy1         -1.750      2.544  63.000    -0.688    0.4940
AgeGroupYoung:Accuracy1  8.356      3.570  63.000     2.341    0.0224 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
      (Intr) AgGrpY Accrc1
AgeGroupYng -0.713
Accuracy1    -0.640  0.456
AgGrpYng:A1  0.456 -0.640 -0.713

```

```
> car::Anova(e2_know_hlm)
```

```

Analysis of Deviance Table (Type II Wald chisquare tests)

Response: Trials
              Chisq Df Pr(>Chisq)
AgeGroup      1.3212  1    0.25038
Accuracy      1.9501  1    0.16258
AgeGroup:Accuracy 5.4788  1    0.01925 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> ## percents
>
> exp2_knowacc_percent = spread(exp_2_knowacc2, Accuracy, Trials)

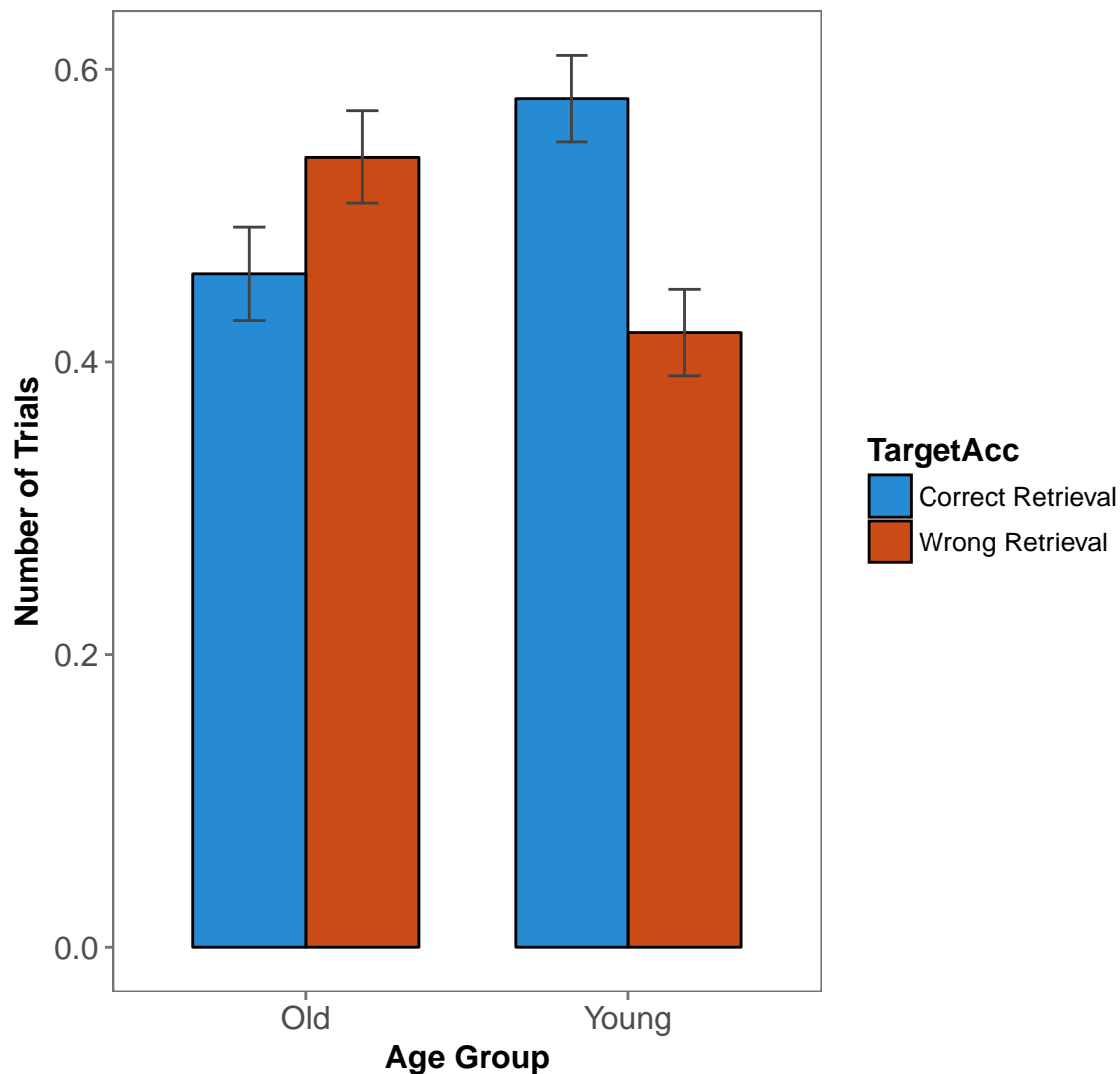
```

```

> exp2_knowacc_percent$`0` = ifelse(is.na(exp2_knowacc_percent$`0`),
+                                   0, exp2_knowacc_percent$`0`)
> exp2_knowacc_percent$`1` = ifelse(is.na(exp2_knowacc_percent$`1`),
+                                   0, exp2_knowacc_percent$`1`)
> exp2_knowacc_percent$total = exp2_knowacc_percent$`0` + exp2_knowacc_percent$`1`
> exp2_knowacc_percent$pcorrect = exp2_knowacc_percent$`1`/exp2_knowacc_percent$total
> exp2_knowacc_percent$pwrong = exp2_knowacc_percent$`0`/exp2_knowacc_percent$total
> exp2_knowacc_long = exp2_knowacc_percent %>% gather(Type,
+                                                     Percent,
+                                                     pcorrect:pwrong)%>%
+   arrange(Subject)
> ## plotting
> library(ggplot2)
> library(ggthemes)
> e2_know_data = Rmisc::summarySE(exp2_knowacc_long,
+                                 measurevar = "Percent",
+                                 groupvars = c( "AgeGroup", "Type"))
> e2_know_data$Percent = round(e2_know_data$Percent, 2)
> e2_know_plot = e2_know_data %>%
+   mutate(TargetAcc = factor(Type, levels = unique(Type),
+                               labels = c("Correct Retrieval", "Wrong Retrieval")))%>%
+   ggplot(aes(x = AgeGroup, y = Percent,
+             group=TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_solarized()+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Age Group") + ylab("Number of Trials") +
+   ggtitle("E2: Know Responses in Young and Old Adults (Without Instructions)")
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e2_know_plot

```


now Responses in Young and Old Adults (Without Instructions)



3.5 Experiment 3

```
> contrasts(exp_3_knowacc$PrimeCondition) = contr.treatment(4, base = 3)
> e3_know_hlm = lmer(data = exp_3_knowacc, Trials ~ PrimeCondition*Accuracy +
+                  (1|Subject))
> summary(e3_know_hlm)
```

```
Linear mixed model fit by REML t-tests use Satterthwaite approximations to
degrees of freedom [lmerMod]
Formula: Trials ~ PrimeCondition * Accuracy + (1 | Subject)
```

```

Data: exp_3_knowacc

REML criterion at convergence: 1405.7

Scaled residuals:
    Min       1Q   Median       3Q      Max
-2.7868 -0.6504 -0.1098  0.6154  4.2753

Random effects:
 Groups      Name          Variance Std.Dev.
 Subject    (Intercept)  3.066     1.751
 Residual                8.574     2.928
Number of obs: 275, groups: Subject, 36

Fixed effects:
              Estimate Std. Error      df t value Pr(>|t|)
(Intercept)      6.4956    0.5752  185.9600   11.292 < 2e-16 ***
PrimeCondition1   -1.0015    0.7010  232.4600   -1.429  0.154442
PrimeCondition2   -2.3782    0.7068  232.8900   -3.364  0.000897 ***
PrimeCondition4   -2.6208    0.7068  232.8900   -3.708  0.000261 ***
Accuracy          -1.6786    0.7068  232.8900   -2.375  0.018372 *
PrimeCondition1:Accuracy  1.3658    0.9963  233.0100    1.371  0.171738
PrimeCondition2:Accuracy  3.4885    1.0005  233.2800    3.487  0.000584 ***
PrimeCondition4:Accuracy  2.3321    1.0088  233.6000    2.312  0.021664 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
      (Intr) PrmCn1 PrmCn2 PrmCn4 Accrcy PrC1:A PrC2:A
PrimeCndtn1 -0.609
PrimeCndtn2 -0.605  0.496
PrimeCndtn4 -0.605  0.496  0.494
Accuracy    -0.605  0.496  0.494  0.494
PrmCndtn1:A  0.430 -0.705 -0.351 -0.351 -0.711
PrmCndtn2:A  0.427 -0.350 -0.708 -0.350 -0.708  0.503
PrmCndtn4:A  0.424 -0.348 -0.347 -0.702 -0.702  0.499  0.498

> car::Anova(e3_know_hlm)

Analysis of Deviance Table (Type II Wald chisquare tests)

Response: Trials
              Chisq Df Pr(>Chisq)
PrimeCondition    9.3749  3  0.024701 *
Accuracy          0.0956  1  0.757140
PrimeCondition:Accuracy 13.0918  3  0.004442 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

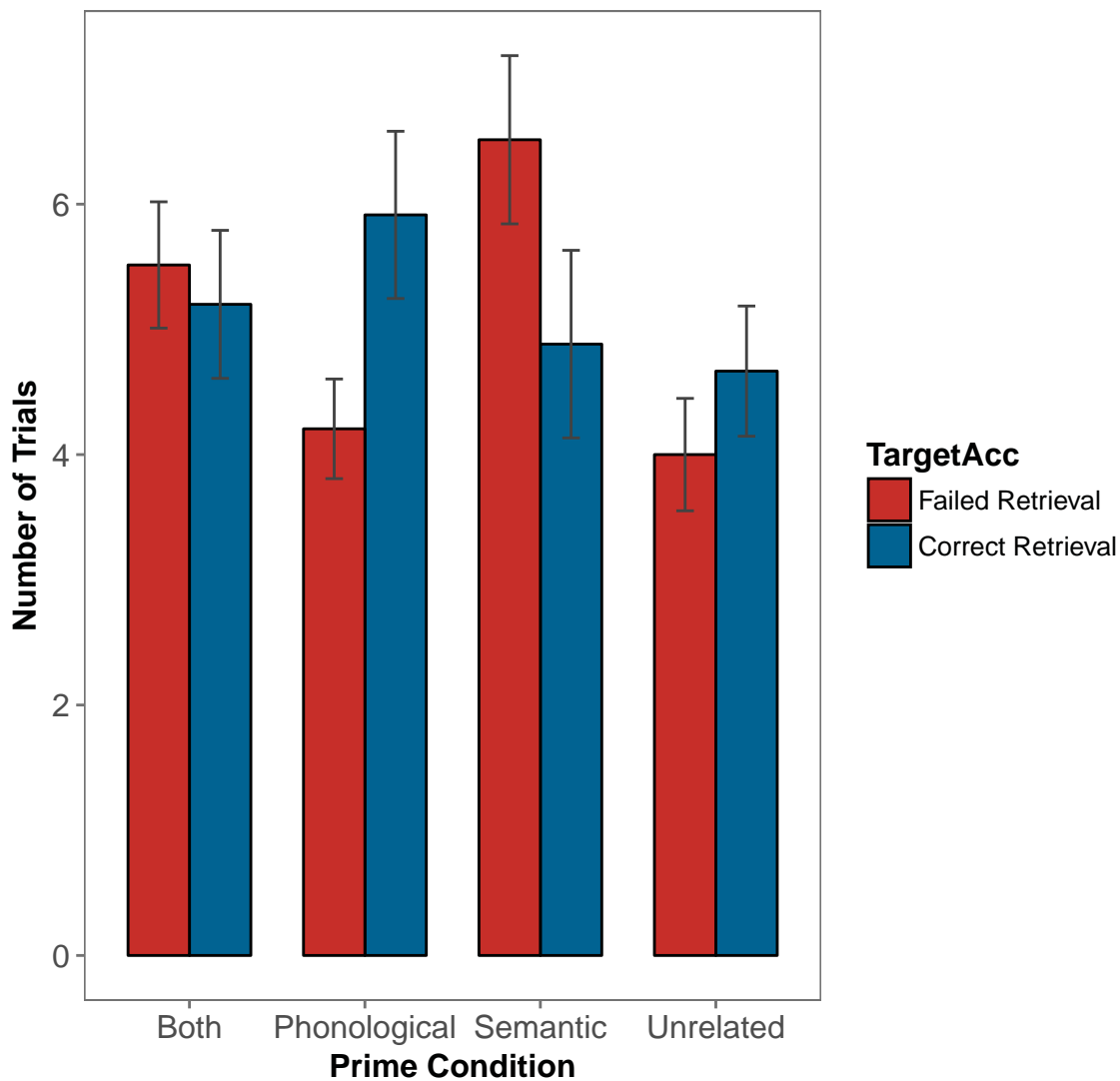
```

```

> #sjPlot::plot_model(e3_know_hlm, type = "int")
>
> ## plotting
>
> e3_know_data = Rmisc::summarySE(exp_3_knowacc,
+                               measurevar = "Trials",
+                               groupvars = c("PrimeCondition", "Accuracy"))
> e3_know_plot = e3_know_data %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Both", "Phonological", "Semantic", "Unrelated")),
+          TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+                             labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+   ggplot(aes(x = PrimeType, y = Trials,
+              group=TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_wsjs()+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Prime Condition") + ylab("Number of Trials") +
+   ggtitle("E3: Know Responses in Young Adults Only (48 ms)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e3_know_plot

```

E3: Know Responses in Young Adults Only (48 ms)



3.6 Experiment 3 Collapsed Prime

```
> ## HLM on trials
> exp_3_knowacc2 = exp_3_state_acc %>% filter(Question.RESP == "1")
> library(lme4)
> exp_3_knowacc2$Accuracy = as.factor(exp_3_knowacc2$Accuracy)
> e3_know_hlm = lmer(data = exp_3_knowacc2,
+                   Trials ~ Accuracy +
+                   (1|Subject))
> summary(e3_know_hlm)
```

```

Linear mixed model fit by REML t-tests use Satterthwaite approximations to
degrees of freedom [lmerMod]
Formula: Trials ~ Accuracy + (1 | Subject)
Data: exp_3_knowacc2

REML criterion at convergence: 540.5

Scaled residuals:
    Min       1Q   Median       3Q      Max
-1.3391 -0.6635 -0.1869  0.4674  4.1060

Random effects:
    Groups    Name      Variance Std.Dev.
Subject (Intercept) 24.17    4.916
Residual              97.44    9.871
Number of obs: 72, groups: Subject, 36

Fixed effects:
              Estimate Std. Error    df t value Pr(>|t|)
(Intercept)    19.444      1.838  67.340   10.580 4.44e-16 ***
Accuracy1        0.250      2.327  35.000    0.107  0.915
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
      (Intr)
Accuracy1 -0.633

```

```
> car::Anova(e3_know_hlm)
```

```
Analysis of Deviance Table (Type II Wald chisquare tests)
```

```

Response: Trials
      Chisq Df Pr(>Chisq)
Accuracy 0.0115  1      0.9144

```

```

> ## percents
>
> exp3_knowacc_percent = spread(exp_3_knowacc2, Accuracy, Trials)
> exp3_knowacc_percent$`0` = ifelse(is.na(exp3_knowacc_percent$`0`),
+                                   0, exp3_knowacc_percent$`0`)
> exp3_knowacc_percent$`1` = ifelse(is.na(exp3_knowacc_percent$`1`),
+                                   0, exp3_knowacc_percent$`1`)
> exp3_knowacc_percent$total = exp3_knowacc_percent$`0` + exp3_knowacc_percent$`1`
> exp3_knowacc_percent$pcorrect = exp3_knowacc_percent$`1`/exp3_knowacc_percent$total
> exp3_knowacc_percent$pwrong = exp3_knowacc_percent$`0`/exp3_knowacc_percent$total
> exp3_knowacc_long = exp3_knowacc_percent %>% gather(Type,

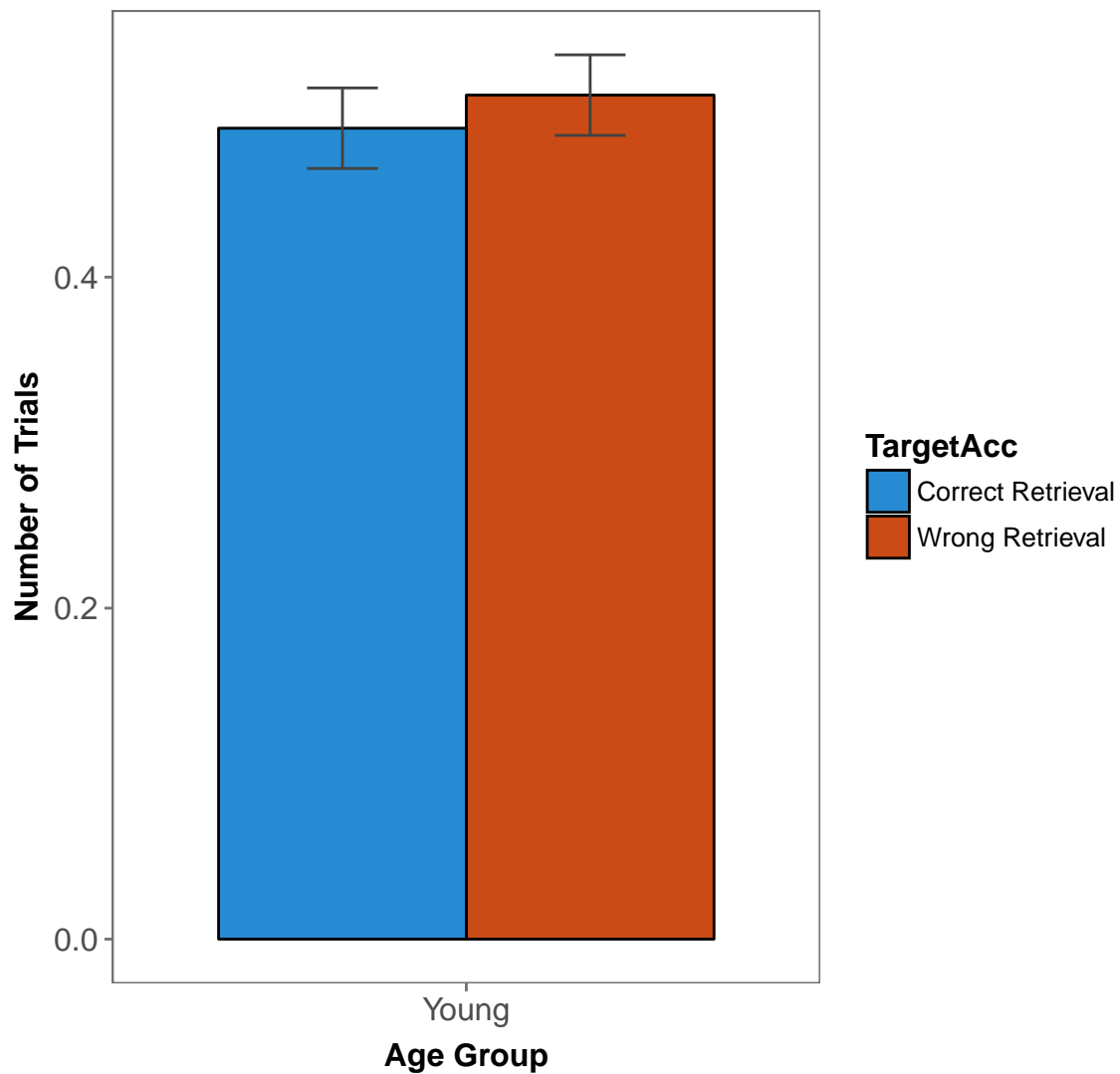
```

```

+                                     Percent,
+                                     pcorrect:pwrong)%>%
+   arrange(Subject)
> ## plotting
> library(ggplot2)
> library(ggthemes)
> e3_know_data = Rmisc::summarySE(exp3_knowacc_long,
+                                 measurevar = "Percent",
+                                 groupvars = c("AgeGroup", "Type"))
> e3_know_data$Percent = round(e3_know_data$Percent, 2)
> e3_know_plot = e3_know_data %>%
+   mutate(TargetAcc = factor(Type, levels = unique(Type),
+                               labels = c("Correct Retrieval", "Wrong Retrieval")))%>%
+   ggplot(aes(x = AgeGroup, y = Percent,
+             group=TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_solarized()+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Age Group") + ylab("Number of Trials") +
+   ggtitle("E2: Know Responses in Young and Old Adults (Without Instructions)")
+
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e3_know_plot

```

now Responses in Young and Old Adults (Without Instructions)



4 Know: Across E1 E2 E3

```
> ## all data is in data_stateprime_acc
>
> ## only know responses in young and old
>
> data_stateprime_acc$Experiment = ifelse(data_stateprime_acc$StudyNo == "2" |
+                                         data_stateprime_acc$StudyNo == "4",
+                                         "No Instructions",
+                                         ifelse(data_stateprime_acc$StudyNo == "5" |
```

```

+           data_stateprime_acc$StudyNo == "6",
+           "With Instructions", "Young 48 ms"))
> data_stateprime_acc$Experiment = as.factor(data_stateprime_acc$Experiment)
> stateprime_young_know_ru = data_stateprime_acc %>%
+   filter(AgeGroup == "Young" & Question.RESP == "1" &
+           PrimeCondition %in% c("R", "U"))
> stateprime_old_know_ru = data_stateprime_acc %>%
+   filter(AgeGroup == "Old" & Question.RESP == "1" &
+           PrimeCondition %in% c("R", "U"))
>
>

```

4.1 Young HLM and Plot

```

> ## hlm on young-know
> contrasts(stateprime_young_know_ru$PrimeCondition) =
+   contr.treatment(4, base = 3)
> contrasts(stateprime_young_know_ru$Experiment) =
+   contr.treatment(3, base = 2)
> library(lmerTest)
> library(lme4)
> hlm_young_know = lmer(data = stateprime_young_know_ru,
+   Trials ~ Experiment*PrimeCondition*Accuracy +
+   (1|Subject))
> summary(hlm_young_know)

```

Linear mixed model fit by REML t-tests use Satterthwaite approximations to degrees of freedom [lmerMod]

Formula: Trials ~ Experiment * PrimeCondition * Accuracy + (1 | Subject)

Data: stateprime_young_know_ru

REML criterion at convergence: 2048.5

Scaled residuals:

Min	1Q	Median	3Q	Max
-2.1249	-0.6868	-0.1754	0.4797	4.7281

Random effects:

Groups	Name	Variance	Std.Dev.
Subject	(Intercept)	1.505	1.227
Residual		9.070	3.012

Number of obs: 400, groups: Subject, 106

Fixed effects:

	Estimate	Std. Error	df	t value
(Intercept)	4.3636	0.5661	366.9000	7.709
Experiment1	0.8796	0.7786	366.9000	1.130


```

Experiment3                2.1304      0.7888 367.9000      2.701
PrimeConditionU            -0.5424      0.7618 291.1000     -0.712
Accuracy                   1.1548      0.7479 287.6000      1.544
Experiment1:PrimeConditionU -0.7369      1.0505 291.4000     -0.701
Experiment3:PrimeConditionU -2.0203      1.0530 290.9000     -1.919
Experiment1:Accuracy        -1.5835      1.0362 289.0000     -1.528
Experiment3:Accuracy        -2.7931      1.0430 289.1000     -2.678
PrimeConditionU:Accuracy    -1.0528      1.0780 293.7000     -0.977
Experiment1:PrimeConditionU:Accuracy 2.4012      1.4908 292.8000      1.611
Experiment3:PrimeConditionU:Accuracy 3.3732      1.4963 294.1000      2.254

                                Pr(>|t|)
(Intercept)                1.2e-13 ***
Experiment1                  0.25933
Experiment3                  0.00723 **
PrimeConditionU              0.47702
Accuracy                     0.12366
Experiment1:PrimeConditionU   0.48358
Experiment3:PrimeConditionU   0.05602 .
Experiment1:Accuracy          0.12755
Experiment3:Accuracy          0.00783 **
PrimeConditionU:Accuracy      0.32955
Experiment1:PrimeConditionU:Accuracy 0.10832
Experiment3:PrimeConditionU:Accuracy 0.02491 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
(Intr) Exprm1 Exprm3 PrmCnU Accrcy Ex1:PCU Ex3:PCU Exp1:A Exp3:A
Experiment1 -0.727
Experiment3 -0.718 0.522
PrimeCndtnU -0.637 0.463 0.457
Accuracy    -0.649 0.472 0.466 0.484
Exprmn1:PCU 0.462 -0.636 -0.332 -0.725 -0.351
Exprmn3:PCU 0.461 -0.335 -0.644 -0.723 -0.350 0.525
Exprmnt1:Ac 0.469 -0.644 -0.336 -0.350 -0.722 0.478 0.253
Exprmnt3:Ac 0.465 -0.338 -0.650 -0.347 -0.717 0.252 0.489 0.518
PrmCndtnU:A 0.450 -0.327 -0.323 -0.708 -0.695 0.514 0.512 0.502 0.498
Expr1:PCU:A -0.326 0.448 0.234 0.512 0.503 -0.706 -0.371 -0.695 -0.360
Expr3:PCU:A -0.324 0.236 0.453 0.510 0.501 -0.370 -0.705 -0.361 -0.698
PrCU:A E1:PCU:
Experiment1
Experiment3
PrimeCndtnU
Accuracy
Exprmn1:PCU
Exprmn3:PCU
Exprmnt1:Ac
Exprmnt3:Ac

```

```
PrmCndtnU:A
Expr1:PCU:A -0.723
Expr3:PCU:A -0.720 0.521
```

```
> car::Anova(hlm_young_know)
```

```
Analysis of Deviance Table (Type II Wald chisquare tests)
```

```
Response: Trials
```

	Chisq	Df	Pr(>Chisq)
Experiment	1.5074	2	0.4706268
PrimeCondition	11.6825	1	0.0006309 ***
Accuracy	0.1376	1	0.7107259
Experiment:PrimeCondition	1.2277	2	0.5412559
Experiment:Accuracy	2.4599	2	0.2923087
PrimeCondition:Accuracy	2.3254	1	0.1272813
Experiment:PrimeCondition:Accuracy	5.3435	2	0.0691300 .

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> sjPlot::plot_model(hlm_young_know, type= "int")
> stateprime_young_know_e2e3 = stateprime_young_know_ru %>%
+   filter(Experiment != "No Instructions")
> hlm_young_know_e2e3 = lmer(data = stateprime_young_know_e2e3,
+   Trials ~ Experiment*PrimeCondition*Accuracy +
+   (1|Subject))
> summary(hlm_young_know_e2e3)
```

```
Linear mixed model fit by REML t-tests use Satterthwaite approximations to
degrees of freedom [lmerMod]
```

```
Formula: Trials ~ Experiment * PrimeCondition * Accuracy + (1 | Subject)
```

```
Data: stateprime_young_know_e2e3
```

```
REML criterion at convergence: 1337.5
```

```
Scaled residuals:
```

Min	1Q	Median	3Q	Max
-2.2023	-0.6596	-0.1701	0.4564	4.7476

```
Random effects:
```

Groups	Name	Variance	Std.Dev.
Subject	(Intercept)	1.652	1.285
Residual		8.819	2.970

```
Number of obs: 262, groups: Subject, 69
```

```
Fixed effects:
```

	Estimate	Std. Error	df
(Intercept)	4.3636	0.5633	236.8100

```

ExperimentYoung 48 ms                2.1288      0.7849 237.5500
PrimeConditionU                      -0.5463      0.7514 189.8700
Accuracy                            1.1536      0.7376 187.6300
ExperimentYoung 48 ms:PrimeConditionU -2.0207      1.0386 189.7400
ExperimentYoung 48 ms:Accuracy        -2.7927      1.0287 188.5900
PrimeConditionU:Accuracy              -1.0460      1.0634 191.5200
ExperimentYoung 48 ms:PrimeConditionU:Accuracy 3.3689      1.4760 191.8100
t value Pr(>|t|)
(Intercept)                          7.746 2.76e-13 ***
ExperimentYoung 48 ms                  2.712 0.00717 **
PrimeConditionU                       -0.727 0.46805
Accuracy                              1.564 0.11948
ExperimentYoung 48 ms:PrimeConditionU -1.946 0.05319 .
ExperimentYoung 48 ms:Accuracy         -2.715 0.00725 **
PrimeConditionU:Accuracy               -0.984 0.32653
ExperimentYoung 48 ms:PrimeConditionU:Accuracy 2.282 0.02356 *

```

```

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Correlation of Fixed Effects:

```

(Intr) ExY48m PrmCnU Accrcy ExY48m:PCU EY48m:A PrCU:A
ExprmntY48m -0.718
PrimeCndtnU -0.631 0.453
Accuracy -0.643 0.462 0.484
ExpY48m:PCU 0.457 -0.638 -0.723 -0.350
ExprmY48m:A 0.461 -0.644 -0.347 -0.717 0.489
PrmCndtnU:A 0.446 -0.320 -0.708 -0.695 0.512 0.498
EY48m:PCU:A -0.321 0.449 0.510 0.501 -0.705 -0.698 -0.720

```

```
> car::Anova(hlm_young_know_e2e3)
```

Analysis of Deviance Table (Type II Wald chisquare tests)

Response: Trials

```

Chisq Df Pr(>Chisq)
Experiment      1.4642 1 0.2262709
PrimeCondition  11.6201 1 0.0006524 ***
Accuracy        0.0221 1 0.8817086
Experiment:PrimeCondition 0.2244 1 0.6356940
Experiment:Accuracy 2.4523 1 0.1173500
PrimeCondition:Accuracy 0.9076 1 0.3407588
Experiment:PrimeCondition:Accuracy 5.2094 1 0.0224652 *

```

```

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

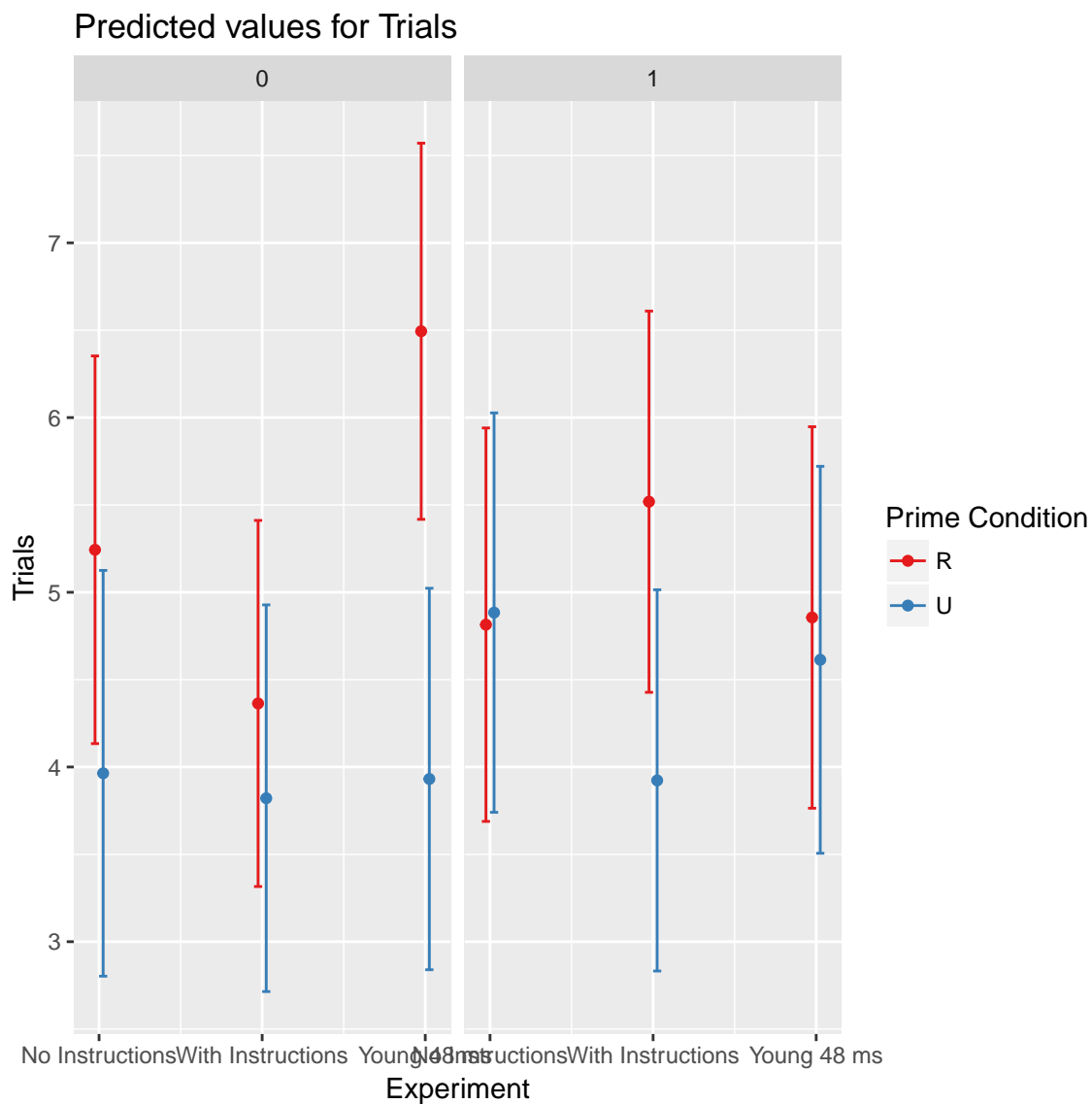
> sjPlot::plot_model(hlm_young_know_e2e3, type= "int")
> ## plotting young data
>

```

```

> young_know_data = Rmisc::summarySE(stateprime_young_know_ru,
+                                   measurevar = "Trials",
+                                   groupvars = c("Experiment" ,
+                                                "PrimeCondition", "Accuracy"))
> young_know_plot = young_know_data %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Semantic", "Unrelated")),
+          TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+                             labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+   ggplot(aes(x = PrimeType, y = Trials,
+              group=TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_wsj()+
+   facet_wrap(~Experiment)+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Prime Condition") + ylab("Number of Trials") +
+   ggtitle("Young Adults: Know Responses in E1, E2 and E3") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> young_know_plot

```



4.2 Old HLM and Plot

```
> ## hlm on old-know
> contrasts(stateprime_old_know_ru$PrimeCondition) =
+   contr.treatment(4, base = 3)
> hlm_old_know = lmer(data = stateprime_old_know_ru,
+   Trials ~ Experiment*PrimeCondition*Accuracy +
+   (1|Subject))
> summary(hlm_old_know)
```

```

Linear mixed model fit by REML t-tests use Satterthwaite approximations to
degrees of freedom [lmerMod]
Formula: Trials ~ Experiment * PrimeCondition * Accuracy + (1 | Subject)
Data: stateprime_old_know_ru

REML criterion at convergence: 1315.6

Scaled residuals:
    Min       1Q   Median       3Q      Max
-1.6132 -0.7257 -0.0348  0.5891  3.3797

Random effects:
    Groups      Name              Variance Std.Dev.
Subject (Intercept)    1.763      1.328
Residual                10.804      3.287
Number of obs: 249, groups: Subject, 68

Fixed effects:

```

	Estimate	Std. Error
(Intercept)	7.3234	0.6159
ExperimentWith Instructions	-0.5335	0.8930
PrimeConditionU	-2.6557	0.8132
Accuracy	-2.3493	0.8065
ExperimentWith Instructions:PrimeConditionU	1.0135	1.1711
ExperimentWith Instructions:Accuracy	0.5738	1.1793
PrimeConditionU:Accuracy	2.7044	1.1406
ExperimentWith Instructions:PrimeConditionU:Accuracy	-0.3082	1.6831

	df	t value	Pr(> t)
(Intercept)	231.8100	11.890	< 2e-16
ExperimentWith Instructions	230.9300	-0.597	0.55080
PrimeConditionU	181.5400	-3.266	0.00131
Accuracy	180.3900	-2.913	0.00403
ExperimentWith Instructions:PrimeConditionU	178.9800	0.865	0.38794
ExperimentWith Instructions:Accuracy	181.5400	0.487	0.62717
PrimeConditionU:Accuracy	180.4300	2.371	0.01879
ExperimentWith Instructions:PrimeConditionU:Accuracy	182.0500	-0.183	0.85489


```

(Intercept) ***
ExperimentWith Instructions
PrimeConditionU **
Accuracy **
ExperimentWith Instructions:PrimeConditionU
ExperimentWith Instructions:Accuracy
PrimeConditionU:Accuracy *
ExperimentWith Instructions:PrimeConditionU:Accuracy
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
Correlation of Fixed Effects:
              (Intr) ExprWI PrmCnU Accrcy ExWI:PCU ExWI:A PrCU:A
ExprmntWthI -0.690
PrimeCndtnU -0.660  0.455
Accuracy     -0.665  0.459  0.504
ExprmWI:PCU  0.458 -0.662 -0.694 -0.350
ExprmntWI:A  0.455 -0.659 -0.345 -0.684  0.501
PrmCndtnU:A  0.471 -0.325 -0.713 -0.707  0.495  0.484
ExpWI:PCU:A -0.319  0.463  0.483  0.479 -0.697 -0.701 -0.678
```

```
> car::Anova(hlm_old_know)
```

```
Analysis of Deviance Table (Type II Wald chisquare tests)
```

```
Response: Trials
```

	Chisq	Df	Pr(>Chisq)
Experiment	0.1043	1	0.746703
PrimeCondition	4.9814	1	0.025621 *
Accuracy	3.8470	1	0.049834 *
Experiment:PrimeCondition	1.0578	1	0.303723
Experiment:Accuracy	0.2522	1	0.615503
PrimeCondition:Accuracy	9.3365	1	0.002246 **
Experiment:PrimeCondition:Accuracy	0.0335	1	0.854687

```
---
```

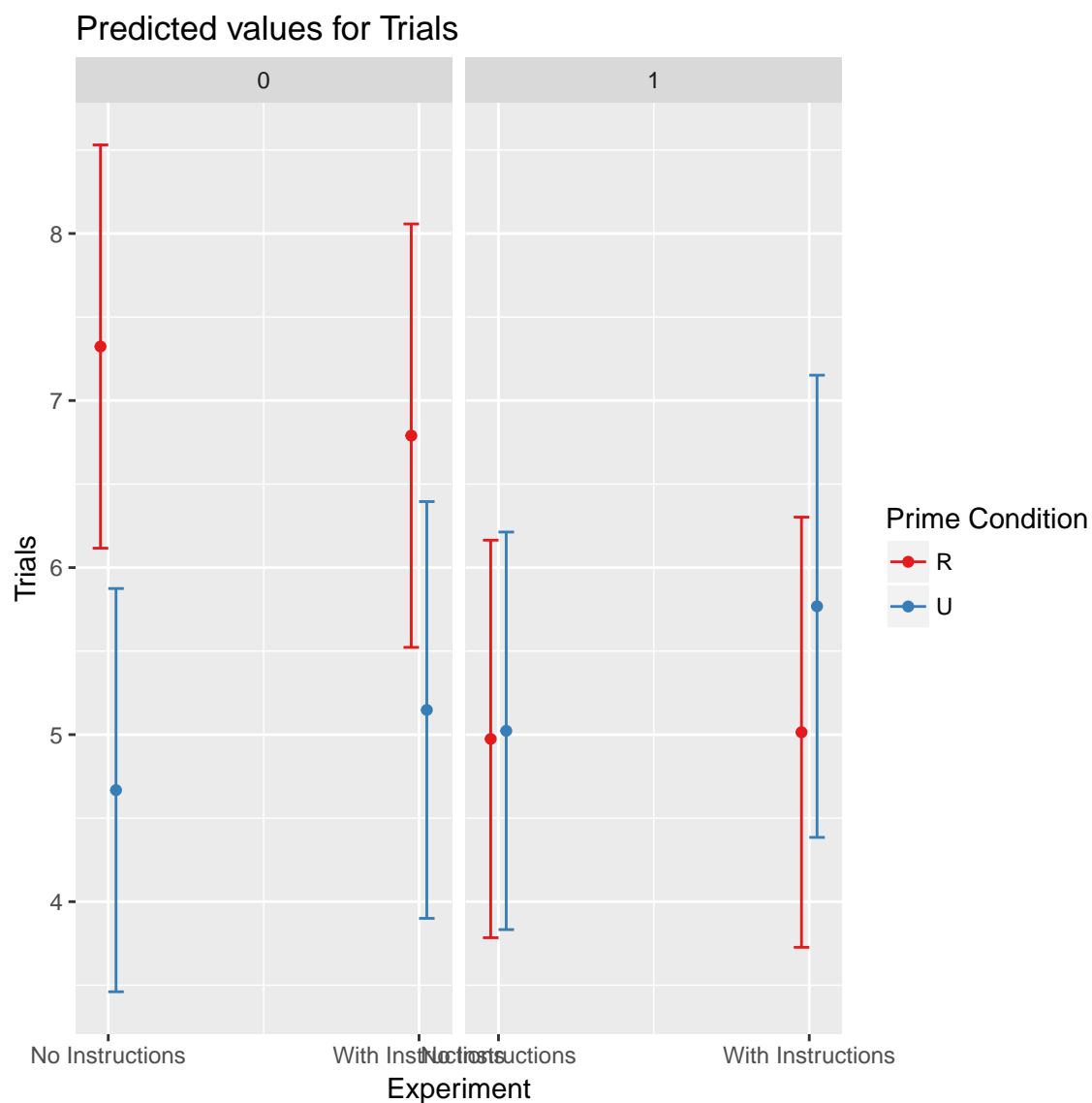
```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> sjPlot::plot_model(hlm_old_know, type= "int")
> stateprime_old_know_ru1 = stateprime_old_know_ru %>% filter(Subject != "702")
> old_know_data = Rmisc::summarySE(stateprime_old_know_ru1,
+                                 measurevar = "Trials",
+                                 groupvars = c("Experiment" ,
+                                               "PrimeCondition", "Accuracy"))
> old_know_plot = old_know_data %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Semantic", "Unrelated")),
+          TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+                             labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+   ggplot(aes(x = PrimeType, y = Trials,
+             group=TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_wsj()+
+   facet_wrap(~Experiment)+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
```

```

+   xlab("Prime Condition") + ylab("Number of Trials") +
+   ggtitle("Old Adults: Know Responses in E1 and E2") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> old_know_plot

```



```

> ### ADDING ZEROS WHERE NEEDED
>

```

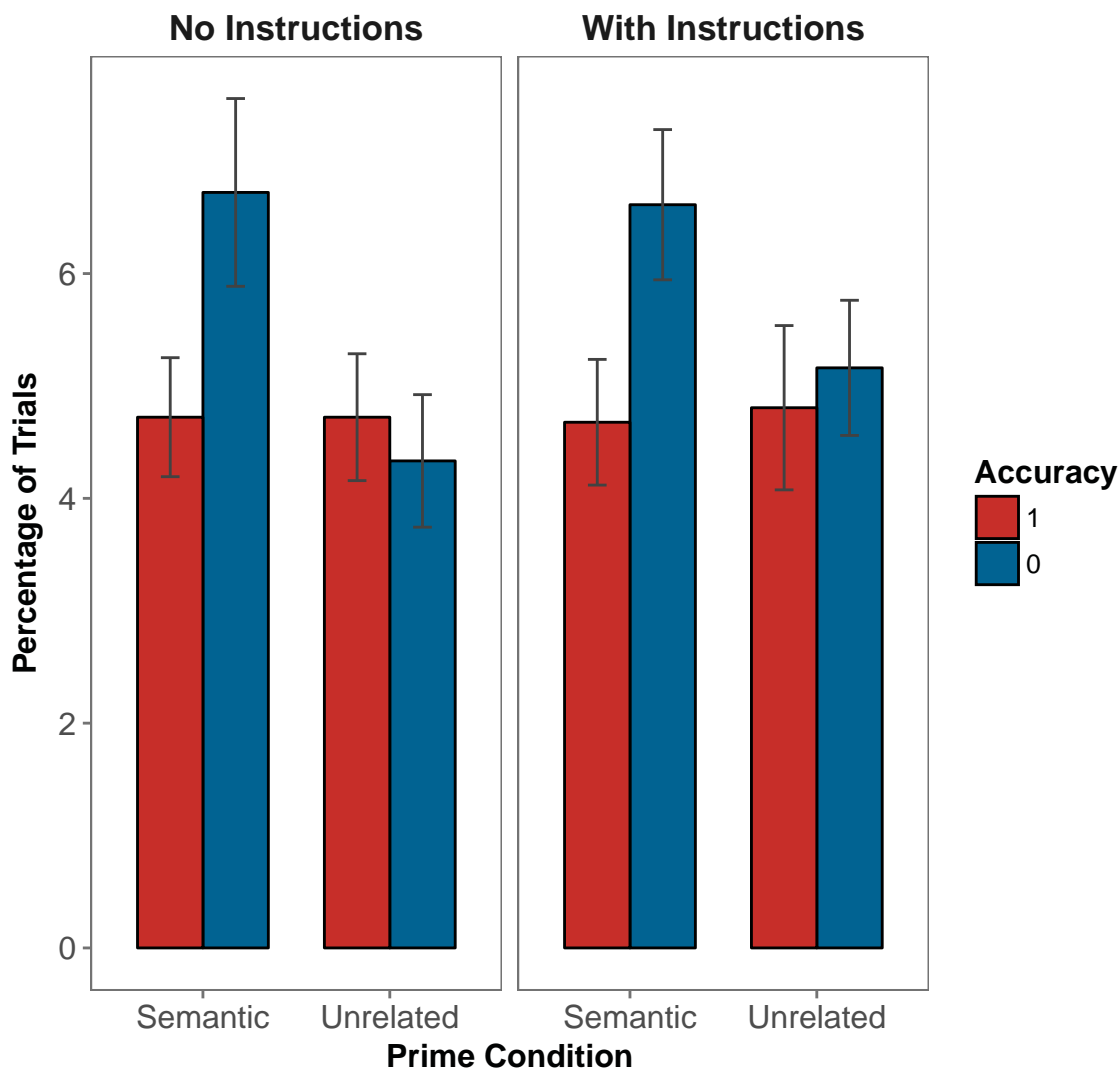


```

> know_old_ru_trials = stateprime_old_know_ru1
> old_know_ru_trials = know_old_ru_trials
> oldtotaltrials_trials = spread(old_know_ru_trials, Accuracy, Trials)
> oldtotaltrials_trials$`0` = ifelse(is.na(oldtotaltrials_trials$`0`),
+                                   0, oldtotaltrials_trials$`0`)
> oldtotaltrials_trials$`1` = ifelse(is.na(oldtotaltrials_trials$`1`),
+                                   0, oldtotaltrials_trials$`1`)
> long_oldpercent_trials <- oldtotaltrials_trials %>% gather(Type,
+                                   Trials,
+                                   `0`:`1`)%>%
+   arrange(Subject, PrimeCondition)
> long_oldpercent_trials$Type = as.factor(long_oldpercent_trials$Type)
> old_know_data_all_trials = Rmisc::summarySE(long_oldpercent_trials,
+                                   measurevar = "Trials",
+                                   groupvars = c("Experiment" ,
+                                                 "PrimeCondition", "Type"))
> old_know_data_all_trials$Accuracy = factor(old_know_data_all_trials$Type,
+                                   levels(old_know_data_all_trials$Type)[c(2,1)])
> old_know_plot_all_trials = old_know_data_all_trials %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                                   labels = c("Semantic", "Unrelated")))%>%
+   ggplot(aes(x = PrimeType, y = Trials,
+               group= Accuracy, fill= Accuracy))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_wsj()+
+   facet_wrap(~Experiment)+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Prime Condition") + ylab("Percentage of Trials") +
+   ggtitle("Old Adults: Know Responses in E1 and E2 ") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> old_know_plot_all_trials

```

Old Adults: Know Responses in E1 and E2



5 Know ANOVA

```
> stateprime_young_know_ru_complete = stateprime_young_know_ru %>%
+   filter(!Subject %in% c(14,17, 24,26,28,30,44,68,67,72,79,80,85,86,90,95,164, 170,1
+ stateprime_young_know_ru_complete$Subject = as.factor(stateprime_young_know_ru_comple
+ know_aov = aov(data = stateprime_young_know_ru_complete, Trials ~ Experiment*PrimeCon
+   Error(Subject/(PrimeCondition*Accuracy)))
+ summary(know_aov)
```

Error: Subject

```

      Df Sum Sq Mean Sq F value Pr(>F)
Experiment  2    48.6   24.30   1.648  0.199
Residuals  82  1208.6   14.74

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition      1   130.9   130.94   24.311 4.22e-06 ***
Experiment:PrimeCondition  2    12.6     6.31    1.172   0.315
Residuals          82   441.7     5.39

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:Accuracy
      Df Sum Sq Mean Sq F value Pr(>F)
Accuracy      1     2.1    2.144   0.154  0.695
Experiment:Accuracy  2    18.7    9.330   0.671  0.514
Residuals      82  1139.4   13.896

Error: Subject:PrimeCondition:Accuracy
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition:Accuracy      1    23.3    23.30   2.238  0.139
Experiment:PrimeCondition:Accuracy  2    37.3    18.65   1.792  0.173
Residuals          82   853.6   10.41

```

```

> onlye2e3 = stateprime_young_know_ru_complete %>% filter(Experiment != "Young 48 ms")
> know_aov2 = aov(data = onlye2e3, Trials ~ Experiment*PrimeCondition*Accuracy +
+               Error(Subject/(PrimeCondition*Accuracy)))
> summary(know_aov2)

```

```

Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
Experiment  1    21.3   21.25   1.593  0.212
Residuals  54   720.3   13.34

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition      1    55.0   55.00   9.285 0.00357 **
Experiment:PrimeCondition  1     2.4    2.36   0.399 0.53044
Residuals          54   319.9     5.92

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:Accuracy
      Df Sum Sq Mean Sq F value Pr(>F)
Accuracy      1    13.5   13.50   1.161  0.286
Experiment:Accuracy  1     0.5    0.54   0.046  0.830
Residuals      54   628.2   11.63

```

```
Error: Subject:PrimeCondition:Accuracy
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeCondition:Accuracy	1	2.8	2.790	0.301	0.585
Experiment:PrimeCondition:Accuracy	1	22.5	22.504	2.431	0.125
Residuals	54	500.0	9.258		

```
>
```

6 Know Percent Rel Unrel

6.1 Young

```
> know_young_ru_percent = read.csv("young_know_ru.csv",
+                                 header = TRUE, sep = ",")
> young_know_ru = know_young_ru_percent[, -1]
> youngtotaltrials = spread(young_know_ru, Accuracy, Trials)
> youngtotaltrials$`0` = ifelse(is.na(youngtotaltrials$`0`), 0, youngtotaltrials$`0`)
> youngtotaltrials$`1` = ifelse(is.na(youngtotaltrials$`1`), 0, youngtotaltrials$`1`)
> youngtotaltrials$total = youngtotaltrials$`0` + youngtotaltrials$`1`
> youngtotaltrials$PercentCorrect = youngtotaltrials$`1`/youngtotaltrials$total
> youngtotaltrials$PercentIncorrect = youngtotaltrials$`0`/youngtotaltrials$total
> ## remove NA trials
>
> #totaltrials = totaltrials %>% filter(!(is.na(Rpercent) & is.na(Upercent)))
>
> ## convert back to long
>
> # long_youngpercent <- totaltrials %>% gather(PrimeCondition,
> #                                           Percent,
> #                                           PercentCorrect:PercentIncorrect)%>%
> #   arrange(Subject)
>
> youngtotaltrials$Subject = as.factor(youngtotaltrials$Subject)
> anova_youngpercent = aov(data = youngtotaltrials,
+                           PercentCorrect ~ Experiment*PrimeCondition +
+                           Error(Subject/PrimeCondition))
> summary(anova_youngpercent)
```

```
Error: Subject
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Experiment	2	0.127	0.06329	0.882	0.417
Residuals	103	7.394	0.07179		

```
Error: Subject:PrimeCondition
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeCondition	1	0.256	0.25569	3.937	0.0499 *

```

Experiment:PrimeCondition    2    0.187 0.09333    1.437 0.2424
Residuals                   103    6.690 0.06495
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

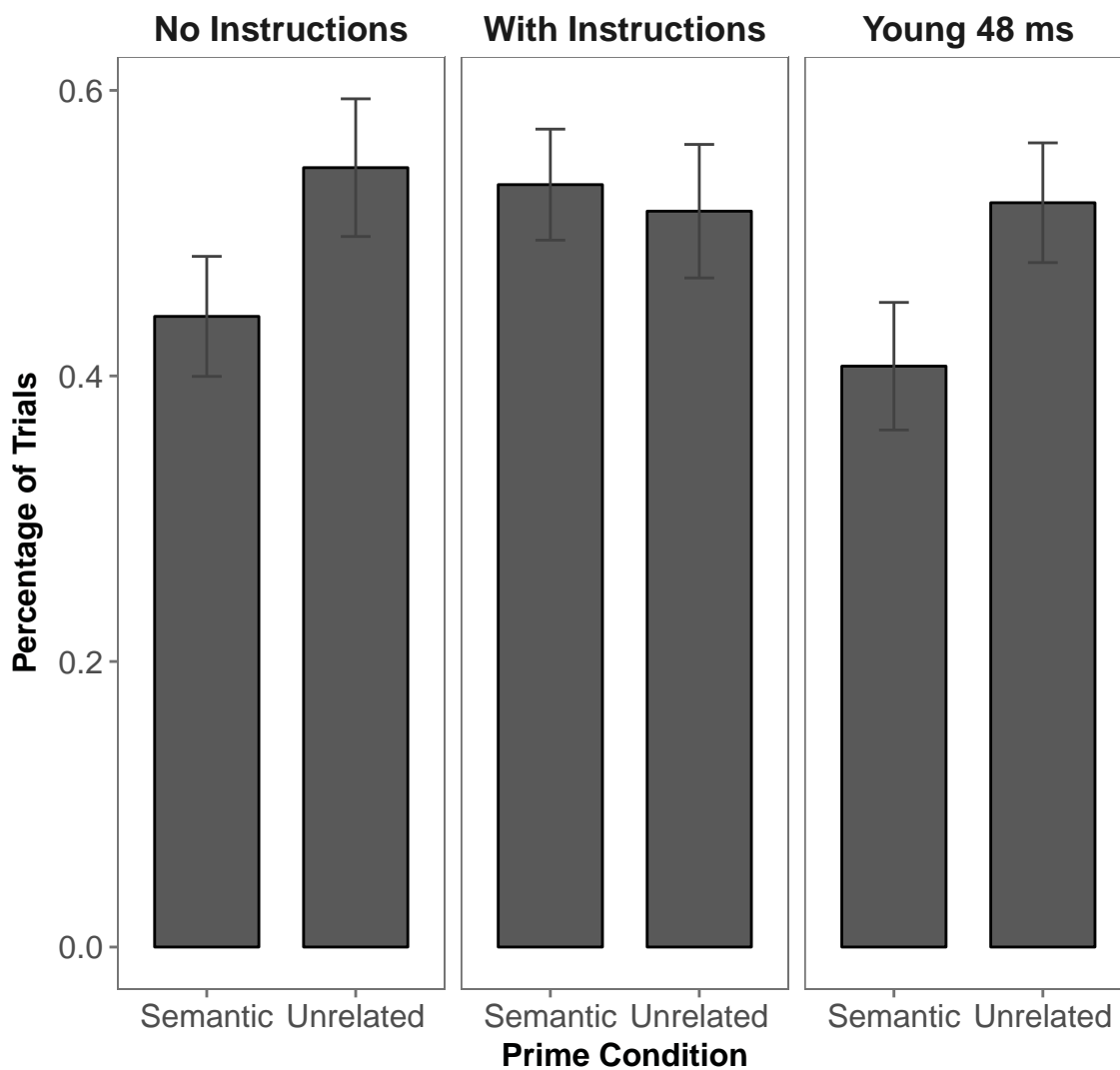
```

```

> young_know_data = Rmisc::summarySE(youngtotaltrials,
+                                     measurevar = "PercentCorrect",
+                                     groupvars = c("Experiment" ,
+                                                  "PrimeCondition"))
> young_know_plot = young_know_data %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Semantic", "Unrelated")))%>%
+   ggplot(aes(x = PrimeType, y = PercentCorrect))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=PercentCorrect - se, ymax=PercentCorrect + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_wsj()+
+   facet_wrap(~Experiment)+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Prime Condition") + ylab("Percentage of Trials") +
+   ggtitle("Young Adults: Correct Know Responses in E1, E2 and E3") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> young_know_plot

```

Young Adults: Correct Know Responses in E1, E2 and E3



6.2 Young Correct Incorrect

```
> know_young_ru_percent = read.csv("young_know_ru.csv",
+                                 header = TRUE, sep = ",")
> young_know_ru = know_young_ru_percent[, -1]
> youngtotaltrials = spread(young_know_ru, Accuracy, Trials)
> youngtotaltrials$`0` = ifelse(is.na(youngtotaltrials$`0`), 0, youngtotaltrials$`0`)
> youngtotaltrials$`1` = ifelse(is.na(youngtotaltrials$`1`), 0, youngtotaltrials$`1`)
> youngtotaltrials$total = youngtotaltrials$`0` + youngtotaltrials$`1`
> youngtotaltrials$PercentCorrect = youngtotaltrials$`1`/youngtotaltrials$total
```

```

> youngtotaltrials$PercentIncorrect = youngtotaltrials$`0`/youngtotaltrials$total
> ## convert back to long
>
> long_youngpercent <- youngtotaltrials %>% gather(Type,
+                                     Percent,
+                                     PercentCorrect:PercentIncorrect)%>%
+   arrange(Subject, PrimeCondition)
> long_youngpercent$Type = as.factor(long_youngpercent$Type)
> long_youngpercent$Subject = as.factor(long_youngpercent$Subject)
> ## correct and incorrect anova
> anova_youngpercent_all = aov(data = long_youngpercent,
+                               Percent ~ Experiment*PrimeCondition*Type +
+                               Error(Subject/(PrimeCondition*Type)))
> summary(anova_youngpercent_all)

```

Error: Subject

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Experiment	2	2.400e-30	1.202e-30	1.915	0.153
Residuals	103	6.467e-29	6.278e-31		

Error: Subject:PrimeCondition

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeCondition	1	3.32e-29	3.319e-29	3.417	0.0674 .
Experiment:PrimeCondition	2	6.82e-29	3.410e-29	3.511	0.0335 *
Residuals	103	1.00e-27	9.710e-30		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:Type

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Type	1	0.019	0.01896	0.132	0.717
Experiment:Type	2	0.253	0.12659	0.882	0.417
Residuals	103	14.788	0.14357		

Error: Subject:PrimeCondition:Type

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeCondition:Type	1	0.511	0.5114	3.937	0.0499 *
Experiment:PrimeCondition:Type	2	0.373	0.1867	1.437	0.2424
Residuals	103	13.380	0.1299		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

> ## correct and incorrect
> young_know_data_all = Rmisc::summarySE(long_youngpercent,
+   measurevar = "Percent",
+   groupvars = c("Experiment",
+                 "PrimeCondition", "Type"))
> young_know_data_all$Accuracy = factor(young_know_data_all$Type,

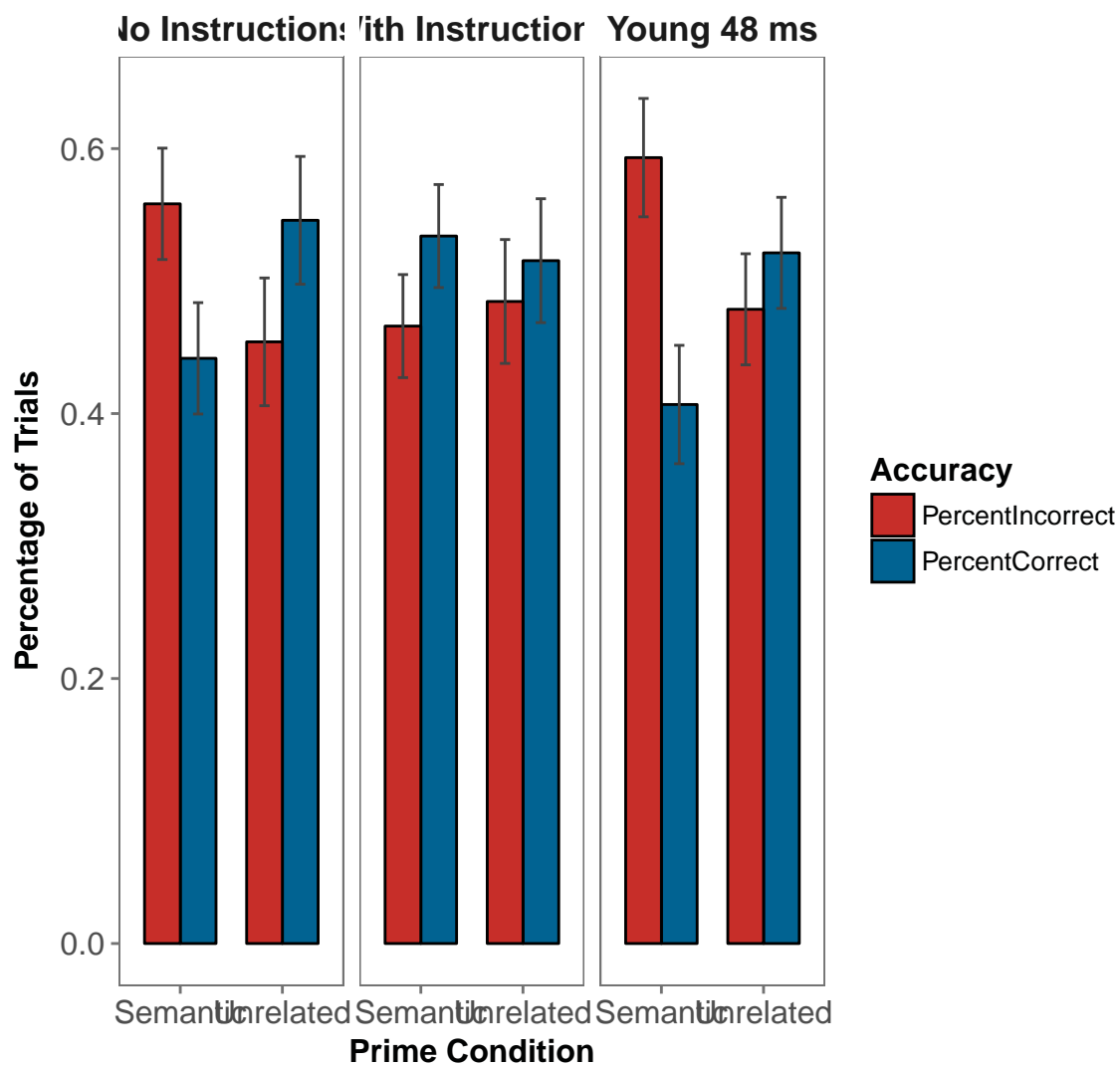
```

```

+           levels(young_know_data_all$Type)[c(2,1)])
> young_know_plot_all = young_know_data_all %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+     labels = c("Semantic", "Unrelated")))%>%
+   ggplot(aes(x = PrimeType, y = Percent,
+     group = Accuracy, fill = Accuracy))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+     width=.2, color = "gray26",
+     position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_wsj()+
+   facet_wrap(~Experiment)+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Prime Condition") + ylab("Percentage of Trials") +
+   ggtitle("Young Adults: Know Responses in E1, E2 and E3") +
+   theme(axis.text = element_text(size = rel(1)),
+     axis.title = element_text(face = "bold", size = rel(1)),
+     legend.title = element_text(face = "bold", size = rel(1)),
+     plot.title = element_text(hjust = .5),
+     strip.text.x = element_text(face = "bold", size = rel(1.4)))
> young_know_plot_all

```


Young Adults: Know Responses in E1, E2 and E3



6.3 Old

```
> know_old_ru_percent = read.csv("old_know_ru.csv",
+                               header = TRUE, sep = ",")
> old_know_ru = know_old_ru_percent[, -1]
> oldtotaltrials = spread(old_know_ru, Accuracy, Trials)
> oldtotaltrials$`0` = ifelse(is.na(oldtotaltrials$`0`), 0, oldtotaltrials$`0`)
> oldtotaltrials$`1` = ifelse(is.na(oldtotaltrials$`1`), 0, oldtotaltrials$`1`)
> oldtotaltrials$total = oldtotaltrials$`0` + oldtotaltrials$`1`
> oldtotaltrials$PercentCorrect = oldtotaltrials$`1`/oldtotaltrials$total
```

```

> oldtotaltrials$PercentIncorrect = oldtotaltrials$`0`/oldtotaltrials$total
> #totaltrials = totaltrials %>% filter(!(is.na(Rpercent) & is.na(Upercent)))
>
>
> ## convert back to long
>
> # long_oldpercent <- totaltrials %>% gather(PrimeCondition,
> #                                           Percent, Rpercent:Upercent)%>%
> #   arrange(Subject)
>
>
> ## Subject 702 does not have U know trials at all
>
>
> oldtotaltrials = oldtotaltrials %>% filter(Subject != "702")
> oldtotaltrials$`0` = as.numeric(as.character(oldtotaltrials$`0`))
> oldtotaltrials$`1` = as.numeric(as.character(oldtotaltrials$`1`))
> oldtotaltrials$Subject = as.factor(oldtotaltrials$Subject)
> anova_oldpercent = aov(data = oldtotaltrials,
+                         PercentCorrect ~ Experiment*PrimeCondition +
+                         Error(Subject/PrimeCondition))
> summary(anova_oldpercent)

```

Error: Subject

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Experiment	1	0.234	0.2344	2.114	0.151
Residuals	65	7.207	0.1109		

Error: Subject:PrimeCondition

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeCondition	1	0.049	0.04888	1.005	0.320
Experiment:PrimeCondition	1	0.046	0.04608	0.947	0.334
Residuals	65	3.163	0.04866		

```

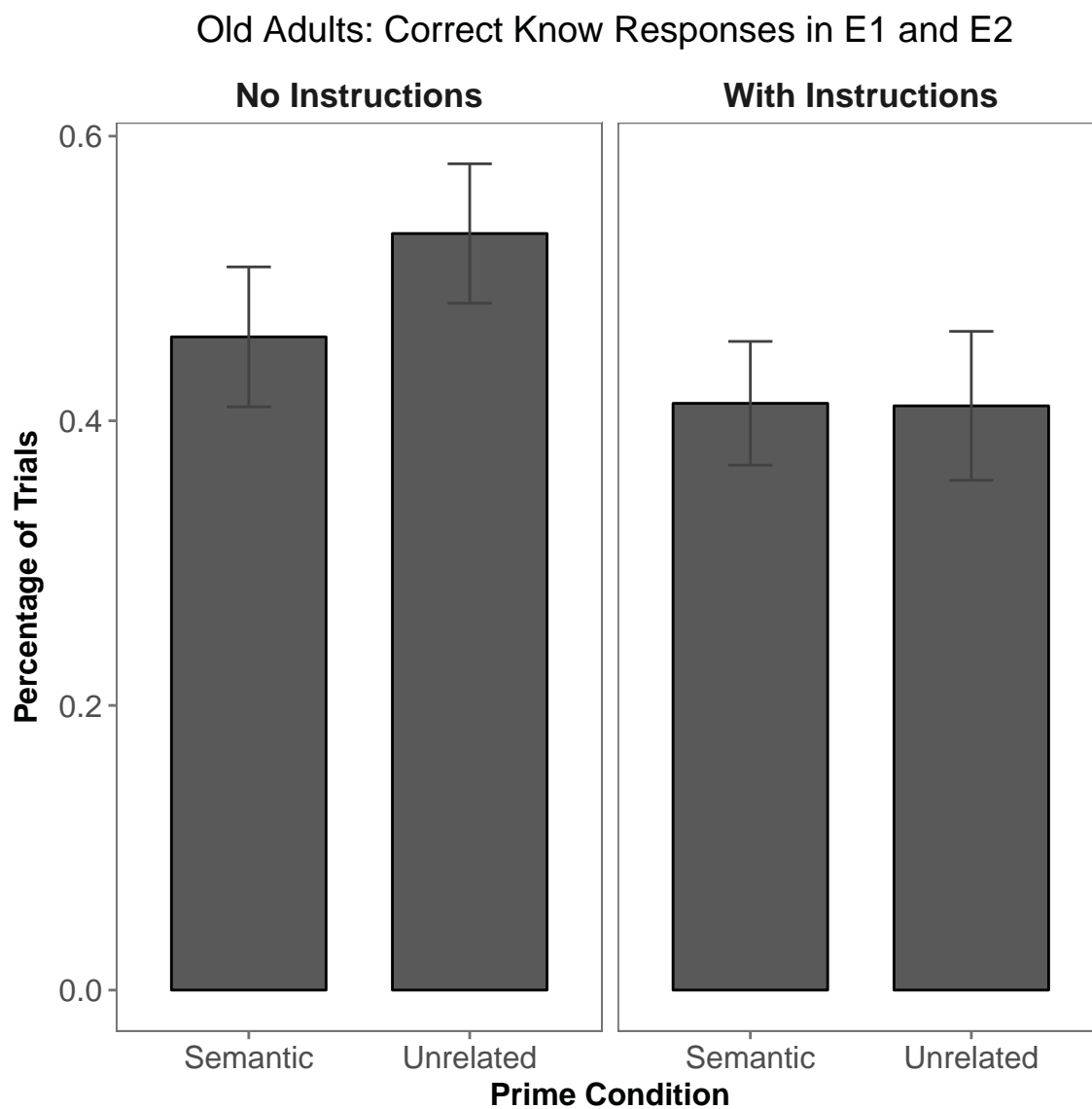
> old_know_data = Rmisc::summarySE(oldtotaltrials,
+                                 measurevar = "PercentCorrect",
+                                 groupvars = c("Experiment",
+                                               "PrimeCondition"))
> old_know_plot = old_know_data %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Semantic", "Unrelated")))%>%
+   ggplot(aes(x = PrimeType, y = PercentCorrect))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=PercentCorrect - se, ymax=PercentCorrect + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_ws()

```

```

+ facet_wrap(~Experiment)+
+ # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+ xlab("Prime Condition") + ylab("Percentage of Trials") +
+ ggtitle("Old Adults: Correct Know Responses in E1 and E2 ") +
+ theme(axis.text = element_text(size = rel(1)),
+       axis.title = element_text(face = "bold", size = rel(1)),
+       legend.title = element_text(face = "bold", size = rel(1)),
+       plot.title = element_text(hjust = .5),
+       strip.text.x = element_text(face = "bold", size = rel(1.4)))
> old_know_plot

```



6.4 Old Correct Incorrect

```
> know_old_ru_percent = read.csv("old_know_ru.csv",
+                               header = TRUE, sep = ",")
> old_know_ru = know_old_ru_percent[, -1]
> oldtotaltrials = spread(old_know_ru, Accuracy, Trials)
> oldtotaltrials$`0` = ifelse(is.na(oldtotaltrials$`0`), 0, oldtotaltrials$`0`)
> oldtotaltrials$`1` = ifelse(is.na(oldtotaltrials$`1`), 0, oldtotaltrials$`1`)
> oldtotaltrials$total = oldtotaltrials$`0` + oldtotaltrials$`1`
> oldtotaltrials$PercentCorrect = oldtotaltrials$`1`/oldtotaltrials$total
> oldtotaltrials$PercentIncorrect = oldtotaltrials$`0`/oldtotaltrials$total
> #totaltrials = totaltrials %>% filter(!(is.na(Rpercent) & is.na(Upercent)))
> ## convert back to long
>
> long_oldpercent <- oldtotaltrials %>% gather(Type,
+                               Percent,
+                               PercentCorrect:PercentIncorrect)%>%
+   arrange(Subject, PrimeCondition)
> long_oldpercent$Type = as.factor(long_oldpercent$Type)
> ## Subject 702 does not have U know trials at all
>
>
> long_oldpercent = long_oldpercent %>% filter(Subject != "702")
> long_oldpercent$Subject = as.factor(long_oldpercent$Subject)
> ## correct and incorrect anova
> anova_oldpercent_all = aov(data = long_oldpercent,
+                             Percent ~ Experiment*PrimeCondition*Type +
+                             Error(Subject/(PrimeCondition*Type)))
> summary(anova_oldpercent_all)
```

```
Error: Subject
      Df    Sum Sq   Mean Sq F value Pr(>F)
Experiment  1 9.760e-31 9.761e-31   2.821 0.0978 .
Residuals 65 2.249e-29 3.460e-31
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition
      Df    Sum Sq   Mean Sq F value Pr(>F)
PrimeCondition  1 8.000e-31 7.982e-31   0.338 0.563
Experiment:PrimeCondition  1 1.150e-30 1.152e-30   0.487 0.488
Residuals      65 1.537e-28 2.365e-30

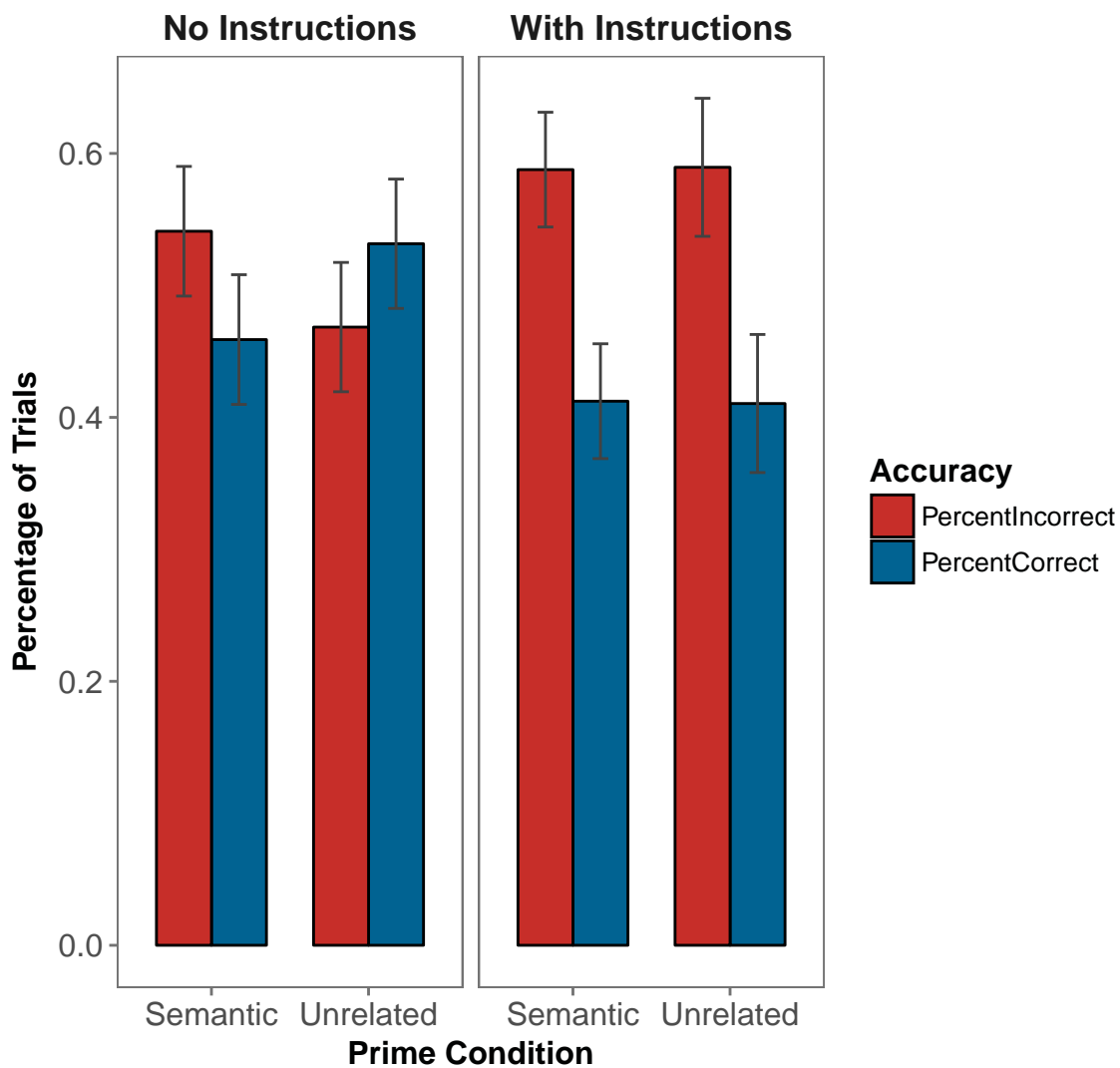
Error: Subject:Type
      Df Sum Sq Mean Sq F value Pr(>F)
Type    1  0.508  0.5081   2.291 0.135
Experiment:Type  1  0.469  0.4688   2.114 0.151
Residuals     65 14.415  0.2218
```

Error: Subject:PrimeCondition:Type

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeCondition:Type	1	0.098	0.09776	1.005	0.320
Experiment:PrimeCondition:Type	1	0.092	0.09216	0.947	0.334
Residuals	65	6.326	0.09732		

```
> old_know_data_all = Rmisc::summarySE(long_oldpercent,
+                                     measurevar = "Percent",
+                                     groupvars = c("Experiment" ,
+                                                  "PrimeCondition", "Type"))
> old_know_data_all$Accuracy = factor(old_know_data_all$Type,
+                                     levels(old_know_data_all$Type)[c(2,1)])
> old_know_plot_all = old_know_data_all %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Semantic", "Unrelated")))%>%
+   ggplot(aes(x = PrimeType, y = Percent,
+             group= Accuracy, fill= Accuracy))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_wsj()+
+   facet_wrap(~Experiment)+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Prime Condition") + ylab("Percentage of Trials") +
+   ggtitle("Old Adults: Know Responses in E1 and E2 ") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> old_know_plot_all
```

Old Adults: Know Responses in E1 and E2



6.5 Age Differences

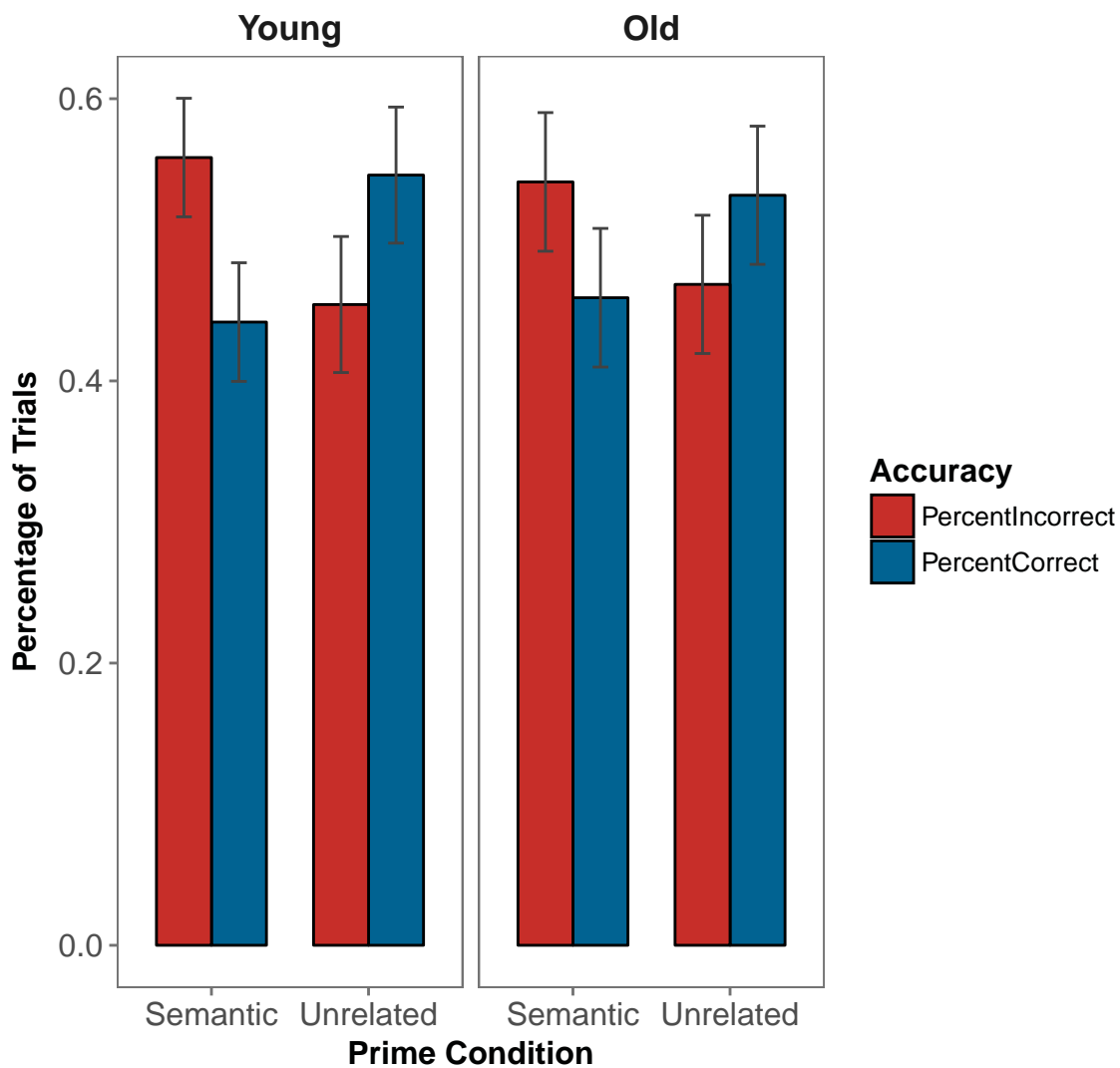
```
> long_oldyoung = rbind(long_youngpercent, long_oldpercent)
> long_e1 = long_oldyoung %>% filter(Experiment == "No Instructions")
> long_e2 = long_oldyoung %>% filter(Experiment == "With Instructions")
> ### Experiment 1
> long_e1_rmisc = Rmisc::summarySE(long_e1,
+                                 measurevar = "Percent",
+                                 groupvars = c("AgeGroup",
+                                              "PrimeCondition", "Type"))
```

```

> long_e1_rmisc$Accuracy = factor(long_e1_rmisc$Type,
+                               levels(long_e1_rmisc$Type)[c(2,1)])
> long_e1_rmisc_plot = long_e1_rmisc %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Semantic", "Unrelated")))%>%
+   ggplot(aes(x = PrimeType, y = Percent,
+             group= Accuracy, fill= Accuracy))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_wsj()+
+   facet_wrap(~AgeGroup)+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Prime Condition") + ylab("Percentage of Trials") +
+   ggtitle("Know Responses in E1 ") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> long_e1_rmisc_plot

```

Know Responses in E1



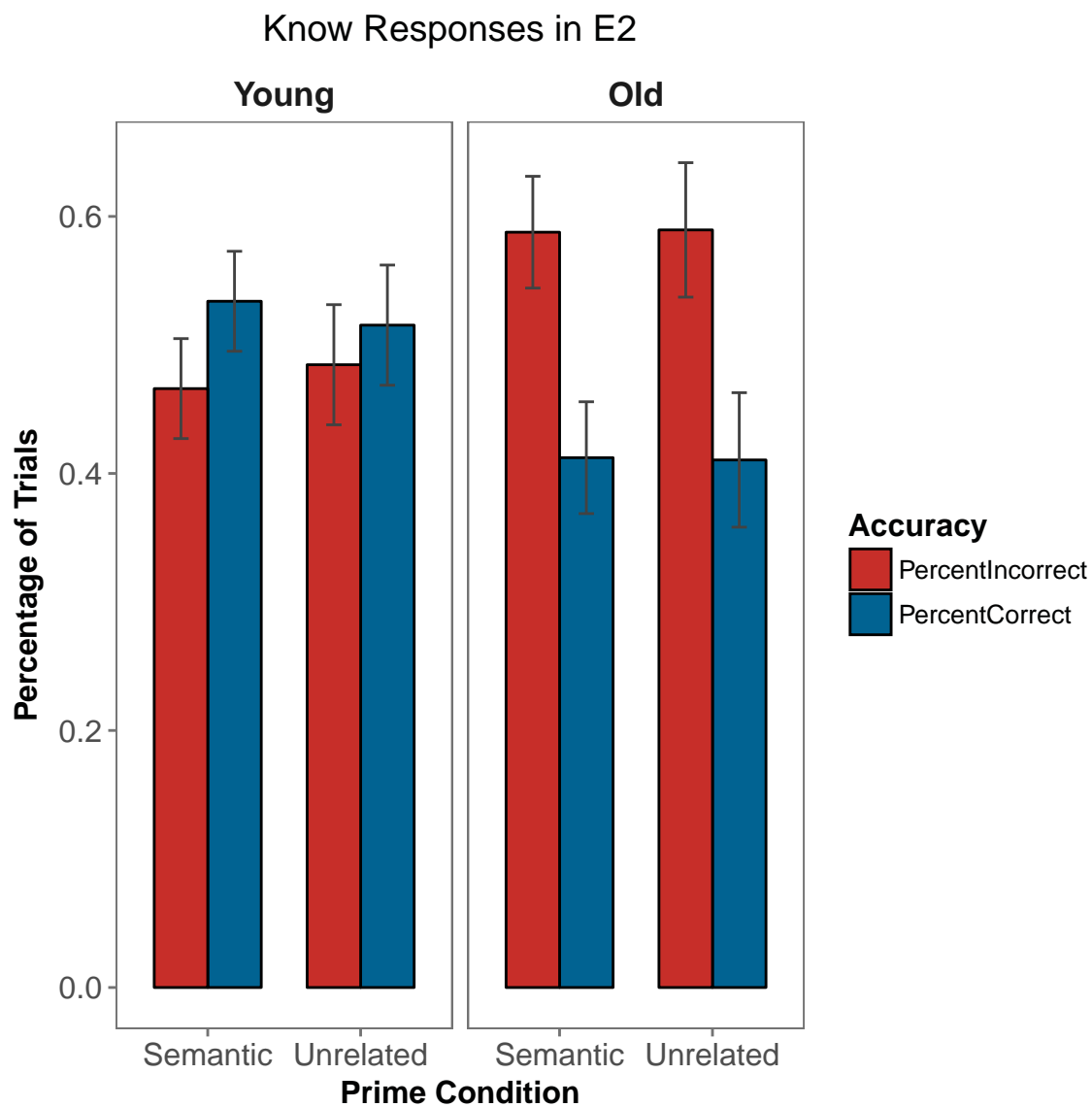
```
> ### Experiment 2
> long_e2_rmisc = Rmisc::summarySE(long_e2,
+                                 measurevar = "Percent",
+                                 groupvars = c("AgeGroup",
+                                              "PrimeCondition", "Type"))
> long_e2_rmisc$Accuracy = factor(long_e2_rmisc$Type,
+                                 levels(long_e2_rmisc$Type)[c(2,1)])
> long_e2_rmisc_plot = long_e2_rmisc %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Semantic", "Unrelated")))%>%
+   ggplot(aes(x = PrimeType, y = Percent,
```



```

+       group= Accuracy, fill= Accuracy))+
+ geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+     width=.2, color = "gray26",
+     position = position_dodge(0.7))+
+ theme_few()+
+   scale_fill_wsj()+
+   facet_wrap(~AgeGroup)+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("Prime Condition") + ylab("Percentage of Trials") +
+   ggtitle("Know Responses in E2 ") +
+   theme(axis.text = element_text(size = rel(1)),
+     axis.title = element_text(face = "bold", size = rel(1)),
+     legend.title = element_text(face = "bold", size = rel(1)),
+     plot.title = element_text(hjust = .5),
+     strip.text.x = element_text(face = "bold", size = rel(1.4)))
> long_e2_rmisc_plot

```



6.5.1 HLMS

```
> long_e1_hlm = lmer(data = long_e1, Percent ~ AgeGroup*PrimeCondition*Type +
+                   (1|Subject))
> car::Anova(long_e1_hlm)
```

Analysis of Deviance Table (Type II Wald chisquare tests)

Response: Percent

	Chisq	Df	Pr(>Chisq)
AgeGroup	0.0000	1	1.000000

```

PrimeCondition      0.0000  1  1.000000
Type                0.1083  1  0.742059
AgeGroup:PrimeCondition 0.0000  1  1.000000
AgeGroup:Type       0.0020  1  0.964385
PrimeCondition:Type  7.0686  1  0.007845 **
AgeGroup:PrimeCondition:Type 0.2241  1  0.635930
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> long_e2_hlm = lmer(data = long_e2, Percent ~ AgeGroup*PrimeCondition*Type +
+                   (1|Subject))
> car::Anova(long_e2_hlm)

```

Analysis of Deviance Table (Type II Wald chisquare tests)

```

Response: Percent

              Chisq Df Pr(>Chisq)
AgeGroup      0.0000  1  1.0000000
PrimeCondition 0.0000  1  1.0000000
Type          3.5245  1  0.0604677 .
AgeGroup:PrimeCondition 0.0000  1  1.0000000
AgeGroup:Type    12.3983  1  0.0004297 ***
PrimeCondition:Type  0.1053  1  0.7455727
AgeGroup:PrimeCondition:Type 0.0682  1  0.7939437
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

6.6 State Accuracy Figures

Experiment 1

```

> exp1_fig_state_acc = Rmisc::summarySE(exp1_state_acc,
+                                       measurevar = "Trials",
+                                       groupvars = c("AgeGroup", "Question.RESP", "Accuracy"))
> exp1_fig_state_acc = arrange(exp1_fig_state_acc,
+                               desc(AgeGroup))
> exp1_fig_state_acc$Accuracy = as.factor(as.character(exp1_fig_state_acc$Accuracy))
> library(ggplot2)
> library(ggthemes)
> state_1_acc = exp1_fig_state_acc %>% mutate(State = factor(Question.RESP,
+                                                             levels = unique(Question.RESP),
+                                                             labels = c("Know", "Dont Know", "Other", "TOT")),
+                                           Age = factor(AgeGroup, levels = unique(AgeGroup),
+                                           labels = c("Young", "Old")),
+                                           TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+                                           labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+ ggplot(aes(x = State, y = Trials,

```

```

+       group = TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+     width=.2, color = "gray26",
+     position = position_dodge(0.7))+
+   theme_few()+
+   facet_wrap(~Age)+
+   scale_fill_wsj()+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("") +
+   ggtitle("E1: Young vs. Old (Without Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+     axis.title = element_text(face = "bold", size = rel(1)),
+     legend.title = element_text(face = "bold", size = rel(1)),
+     plot.title = element_text(hjust = .5),
+     strip.text.x = element_text(face = "bold", size = rel(1.4)))
+
>

```

Experiment 2

```

> exp2_fig_state_acc = Rmisc::summarySE(exp_2_state_acc,
+   measurevar = "Trials",
+   groupvars = c("AgeGroup", "Question.RESP", "Accuracy"))
> exp2_fig_state_acc = arrange(exp2_fig_state_acc,
+   desc(AgeGroup))
> exp2_fig_state_acc$Accuracy = as.factor(as.character(exp2_fig_state_acc$Accuracy))
> library(ggplot2)
> library(ggthemes)
> state_2_acc = exp2_fig_state_acc %>% mutate(State = factor(Question.RESP,
+   levels = unique(Question.RESP),
+   labels = c("Know", "Dont Know", "Other", "TOT")),
+   Age = factor(AgeGroup, levels = unique(AgeGroup),
+   labels = c("Young", "Old")),
+   TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+   labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+   ggplot(aes(x = State, y = Trials,
+     group = TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+     width=.2, color = "gray26",
+     position = position_dodge(0.7))+
+   theme_few()+
+   facet_wrap(~Age)+
+   scale_fill_wsj()+
+   # scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("") +
+   ggtitle("E2: Young vs. Old (With Instructions)") +

```

```
+ theme(axis.text = element_text(size = rel(1)),
+       axis.title = element_text(face = "bold", size = rel(1)),
+       legend.title = element_text(face = "bold", size = rel(1)),
+       plot.title = element_text(hjust = .5),
+       strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

Experiment 3

```
> exp3_fig_state_acc = Rmisc::summarySE(exp_3_state_acc,
+                                       measurevar = "Trials",
+                                       groupvars = c("AgeGroup", "Question.RESP", "Accuracy"))
> exp3_fig_state_acc = arrange(exp3_fig_state_acc,
+                               desc(AgeGroup))
> exp3_fig_state_acc$Accuracy = as.factor(as.character(exp3_fig_state_acc$Accuracy))
> library(ggplot2)
> library(ggthemes)
> state_3_acc= exp3_fig_state_acc %>% mutate(State = factor(Question.RESP,
+                                                           levels = unique(Question.RESP),
+                                                           labels = c("Know", "Dont Know", "Other", "TOT")),
+                                           Age = factor(AgeGroup, levels = unique(AgeGroup),
+                                           labels = c("Young")),
+                                           TargetAcc = factor(Accuracy, levels = unique(Accuracy),
+                                           labels = c("Failed Retrieval", "Correct Retrieval")))%>%
+ ggplot(aes(x = State, y = Trials,
+             group = TargetAcc, fill = TargetAcc))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   facet_wrap(~Age)+
+   scale_fill_wsj()+
+   #scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("") +
+   ggtitle("E3: Young(48 ms)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

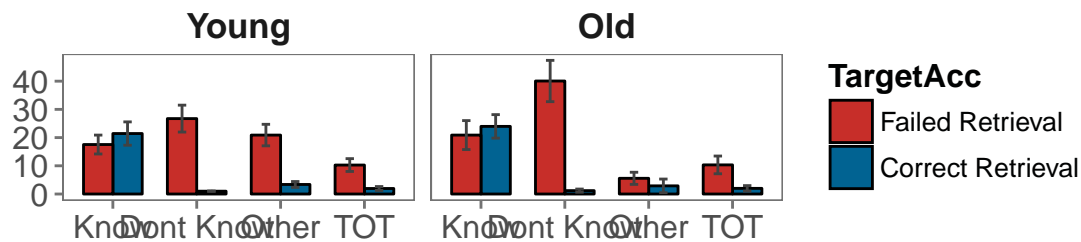
Combined

```
> library(grid)
> gridExtra::grid.arrange(state_1_acc, state_2_acc, state_3_acc,
+                           nrow = 3, ncol = 1,
```

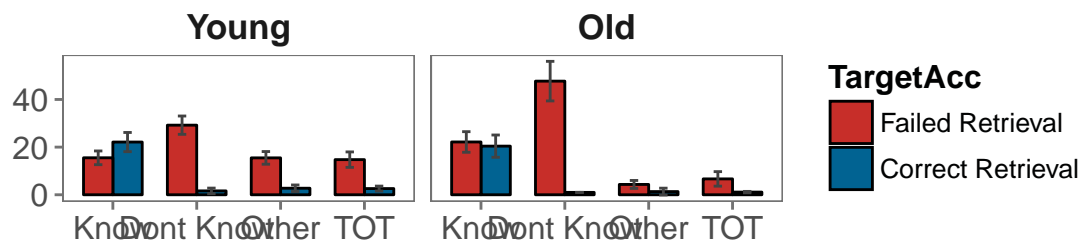
```
+ top=textGrob("Retrieval States Across Experiments E1, E2, E3",
+ gp=gpar(fontsize=20))
```

Retrieval States Across Experiments E1, E2, E3

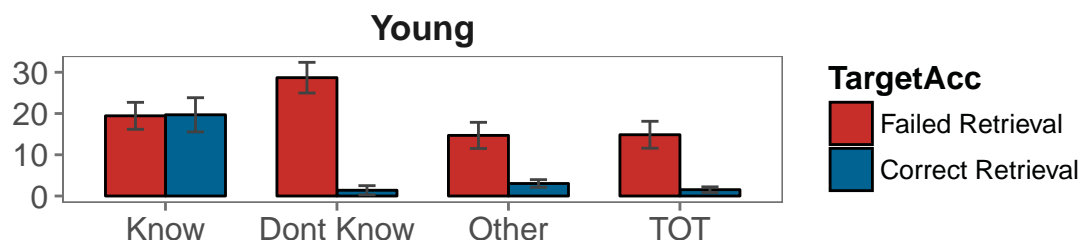
E1: Young vs. Old (Without Instructions)



E2: Young vs. Old (With Instructions)



E3: Young(48 ms)



7 Conditional TOT Analysis

```
> j <- read.csv("MainJulieagg_5studies.csv", header = TRUE, sep = ",")
> j <- subset(j, j$value.Subject != 198 & j$value.Subject != 95)
> j_condTOT = j[,c(2,3,4,5,95:103)]
> j_condTOT$value.Subject = as.factor(j_condTOT$value.Subject)
> library(tidyr)
```

```

> library(dplyr)
> condTOTprime <- j_condTOT %>%
+   gather(PrimeState, Proportion,
+         condpropTOT_r, condpropTOT_p, condpropTOT_b, condpropTOT_u) %>%
+   separate(PrimeState, c('State', 'Prime'), sep = "_") %>%
+   arrange(value.Subject)
> colnames(condTOTprime) = c("AgeGroup", "Subject", "StudyNo", "PrimeInstruction", "cond",
+                             "r_TOT", "p_TOT", "b_TOT", "u_TOT", "State",
+                             "PrimeCondition", "Proportion")
> condTOTprime$AgeGroup <- as.factor(condTOTprime$AgeGroup)
> condTOTprime$Subject <- as.factor(condTOTprime$Subject)
> condTOTprime$StudyNo <- as.factor(condTOTprime$StudyNo)
> condTOTprime$PrimeInstruction <- as.factor(condTOTprime$PrimeInstruction)
> condTOTprime$PrimeCondition <- as.factor(condTOTprime$PrimeCondition)
> condTOTprime$Proportion <- as.numeric(as.character(condTOTprime$Proportion))
> condTOT_exp1 = j_condTOT %>% filter(value.StudyNo == '2' | value.StudyNo == '4')
> condTOT_exp2 = j_condTOT %>% filter(value.StudyNo == '5' | value.StudyNo == '6')
> condTOT_exp3 = j_condTOT %>% filter(value.StudyNo == '1')
> condTOTprime_exp1 = condTOTprime %>% filter(StudyNo == '2' | StudyNo == '4')
> condTOTprime_exp2 = condTOTprime %>% filter(StudyNo == '5' | StudyNo == '6')
> condTOTprime_exp3 = condTOTprime %>% filter(StudyNo == '1')
>

```

7.1 Experiment 1

7.1.1 CondTOT: Young vs Old

```

> e1_condTOT_aov = aov(data = condTOT_exp1, condTOTprop ~ value.AgeGroup)
> summary(e1_condTOT_aov)

```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
value.AgeGroup	1	0.0032	0.003247	0.104	0.748
Residuals	70	2.1829	0.031185		

7.1.2 CondTOT: Age x PrimeType

```

> condTOTprime_exp1[179,12 ] = 0
> e1_condTOTprime_aov = aov(data = condTOTprime_exp1, Proportion ~ AgeGroup*PrimeCondition
+                             Error(Subject/PrimeCondition))
> summary(e1_condTOTprime_aov)

```

Error: Subject

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup	1	0.027	0.02724	0.227	0.635
Residuals	70	8.403	0.12004		

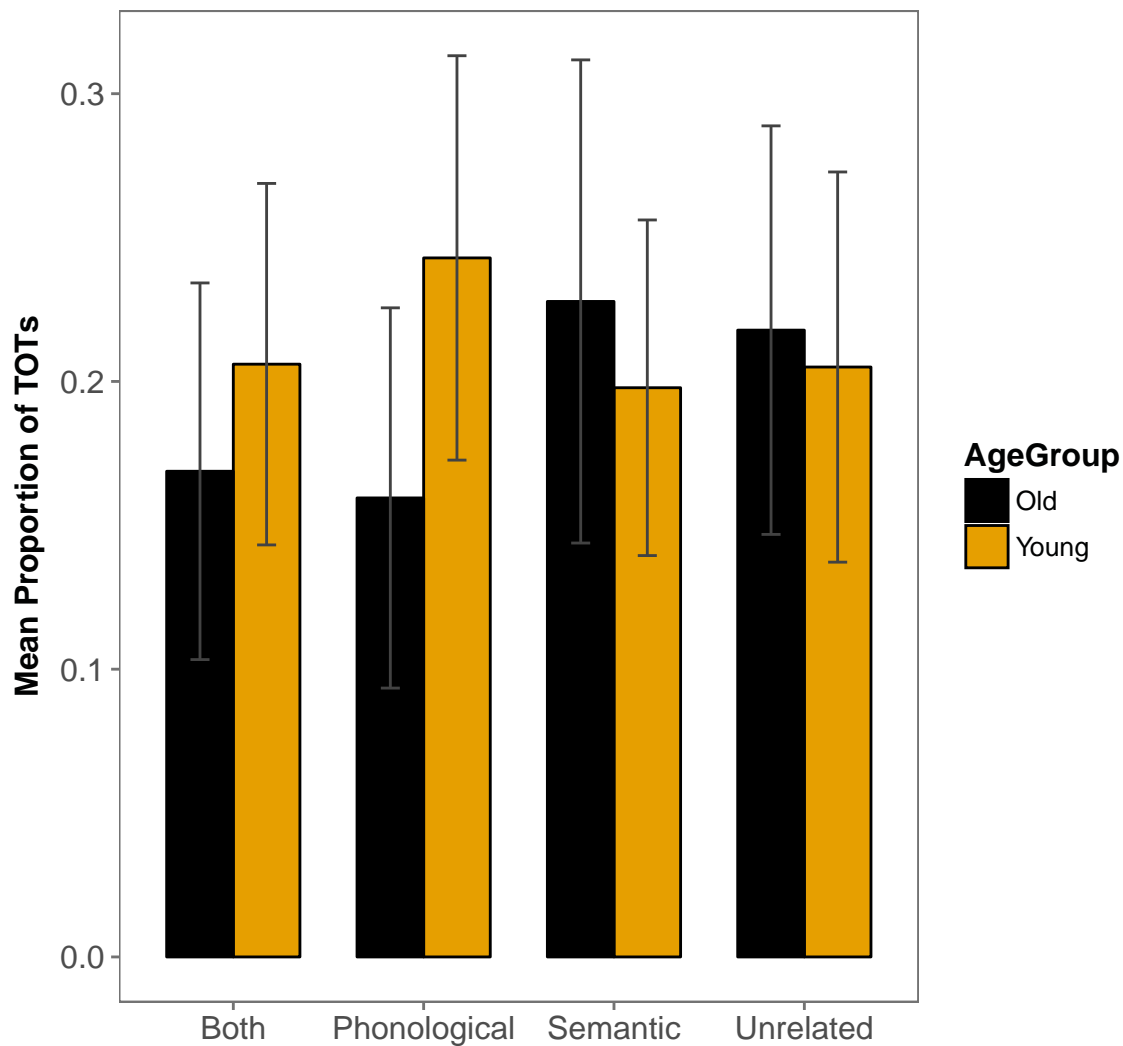
Error: Subject:PrimeCondition

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeCondition	3	0.0298	0.00992	0.673	0.570
AgeGroup:PrimeCondition	3	0.1421	0.04736	3.211	0.024 *
Residuals	210	3.0971	0.01475		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> exp1_fig_condTOT = Rmisc::summarySE(condTOTprime_exp1,
+                                     measurevar = "Proportion",
+                                     groupvars = c("AgeGroup", "PrimeCondition"))
> library(ggplot2)
> library(ggthemes)
> exp1_fig_condTOT = exp1_fig_condTOT %>% mutate(PrimeType = factor(PrimeCondition,
+                                     levels = unique(PrimeCondition),
+                                     labels = c("Both", "Phonological",
+                                     "Semantic", "Unrelated")))%>%
+ ggplot(aes(x = PrimeType, y = Proportion,
+                                     fill = AgeGroup, group = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color = "black")+
+   geom_errorbar(aes(ymin=Proportion - ci, ymax=Proportion + ci),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_colorblind()+
+   xlab("") + ylab("Mean Proportion of TOTs") +
+   ggtitle("E1: Young and Old Adults (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> exp1_fig_condTOT
```


E1: Young and Old Adults (With Instructions)



7.2 Experiment 2

7.2.1 CondTOT: Young vs Old

```
> e2_condTOT_aov = aov(data = condTOT_exp2, condTOTprop ~ value.AgeGroup)
> summary(e2_condTOT_aov)
```

```

      Df Sum Sq Mean Sq F value    Pr(>F)
value.AgeGroup  1   0.263   0.26297    11.11 0.00145 **
Residuals    62   1.467   0.02366
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmmeans)
> library(multcomp)
> exp2_target_lsm = lsmmeans::lsmmeans(e2_condTOT_aov, c("value.AgeGroup"))
> prime_effect = cld(exp2_target_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE)
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

contrast	estimate	SE	df	t.ratio	p.value
Young - Old	0.1282018	0.0384561	62	3.333719	0.0014496

7.2.2 CondTOT: Age x PrimeType

```
> e2_condTOTprime_aov = aov(data = condTOTprime_exp2, Proportion ~ AgeGroup*PrimeCondition
+                               Error(Subject/PrimeCondition))
> summary(e2_condTOTprime_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  1.027  1.0270   10.65 0.00179 **
Residuals 62  5.976  0.0964
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3  0.0573  0.019111   1.984  0.118
AgeGroup:PrimeCondition  3  0.0181  0.006047   0.628  0.598
Residuals      186  1.7914  0.009631
```

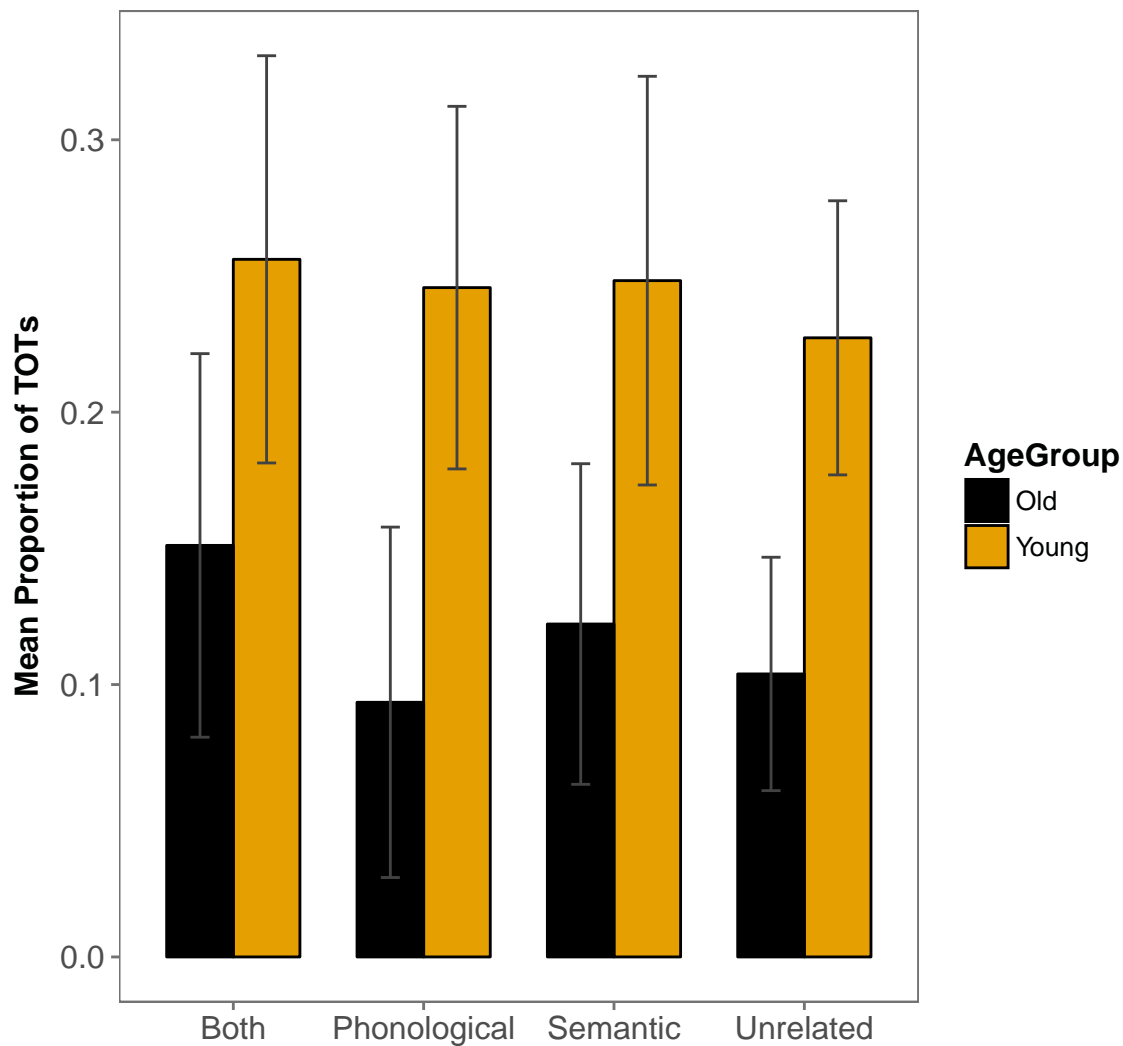
```
> exp2_fig_condTOT = Rmisc::summarySE(condTOTprime_exp2,
+                                     measurevar = "Proportion",
+                                     groupvars = c("AgeGroup", "PrimeCondition"))
> library(ggplot2)
> library(ggthemes)
> exp2_fig_condTOT = exp2_fig_condTOT %>% mutate(PrimeType = factor(PrimeCondition,
+                                     levels = unique(PrimeCondition),
+                                     labels = c("Both", "Phonological",
+                                                 "Semantic", "Unrelated")))%>%
+ ggplot(aes(x = PrimeType, y = Proportion,
+            fill = AgeGroup, group = AgeGroup))+
+ geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+ geom_errorbar(aes(ymin=Proportion - ci, ymax=Proportion + ci),
```

```

+           width=.2, color = "gray26",
+           position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_colorblind()+
+   xlab("") + ylab("Mean Proportion of TOTs") +
+   ggtitle("E2: Young and Old Adults (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> exp2_fig_condTOT

```

E2: Young and Old Adults (With Instructions)



7.3 Experiment 3

7.3.1 CondTOT:PrimeType

```
> e3_condTOTprime_aov = aov(data = condTOTprime_exp3, Proportion ~ PrimeCondition +
+                               Error(Subject/PrimeCondition))
> summary(e3_condTOTprime_aov)
```

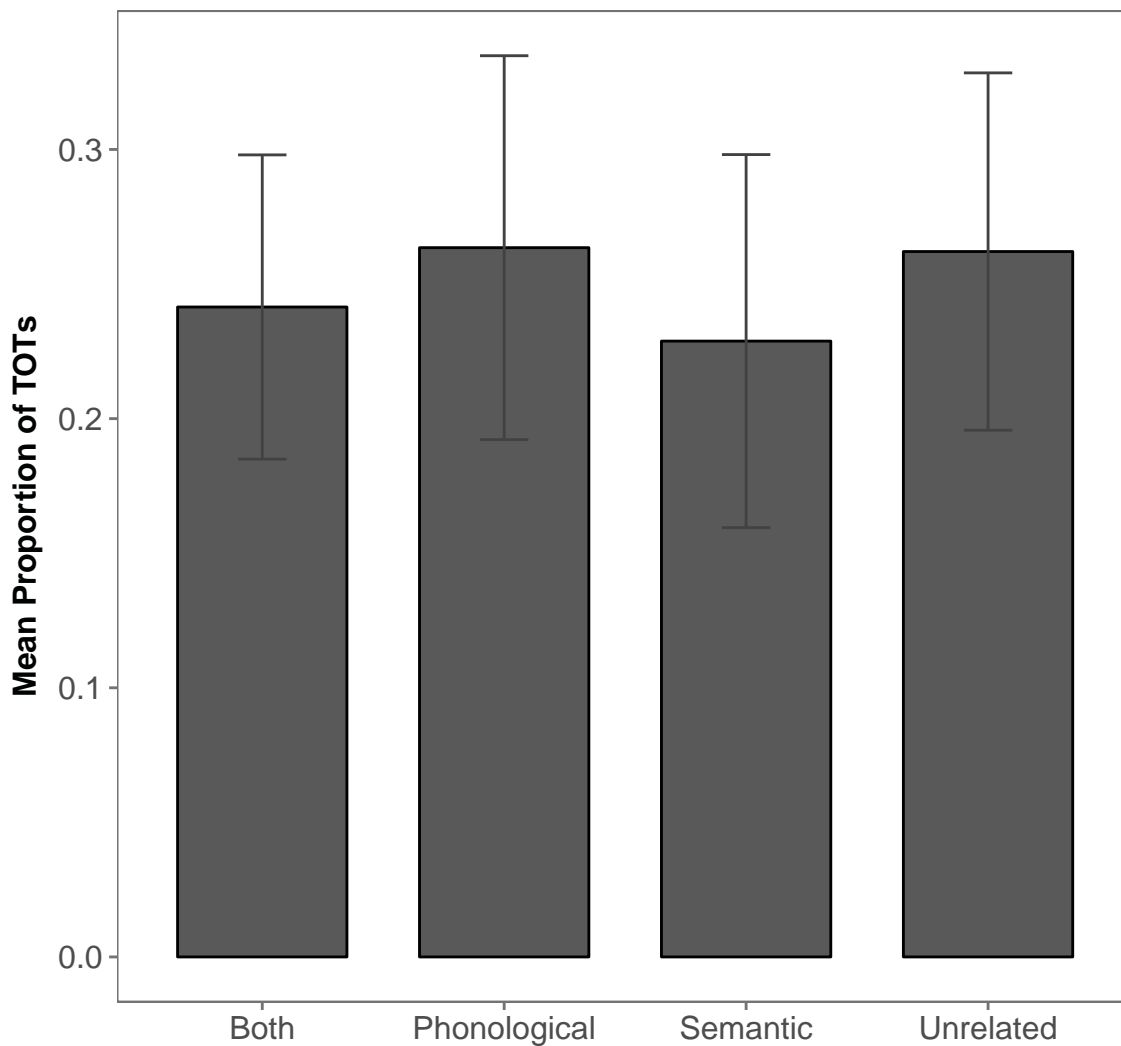
```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 35  4.385   0.1253
```

Error: Subject:PrimeCondition

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeCondition	3	0.0305	0.010153	1.108	0.349
Residuals	105	0.9625	0.009167		

```
> exp3_fig_condTOT = Rmisc::summarySE(condTOTprime_exp3,
+                                     measurevar = "Proportion",
+                                     groupvars = c("PrimeCondition"))
> library(ggplot2)
> library(ggthemes)
> exp3_fig_condTOT = exp3_fig_condTOT %>% mutate(PrimeType = factor(PrimeCondition,
+                                     levels = unique(PrimeCondition),
+                                     labels = c("Both", "Phonological",
+                                     "Semantic", "Unrelated")))%>%
+ ggplot(aes(x = PrimeType, y = Proportion))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Proportion - ci, ymax=Proportion + ci),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_colorblind()+
+   xlab("") + ylab("Mean Proportion of TOTs") +
+   ggtitle("E2: Young and Old Adults (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> exp3_fig_condTOT
```

E2: Young and Old Adults (With Instructions)



8 TOT for No Responses

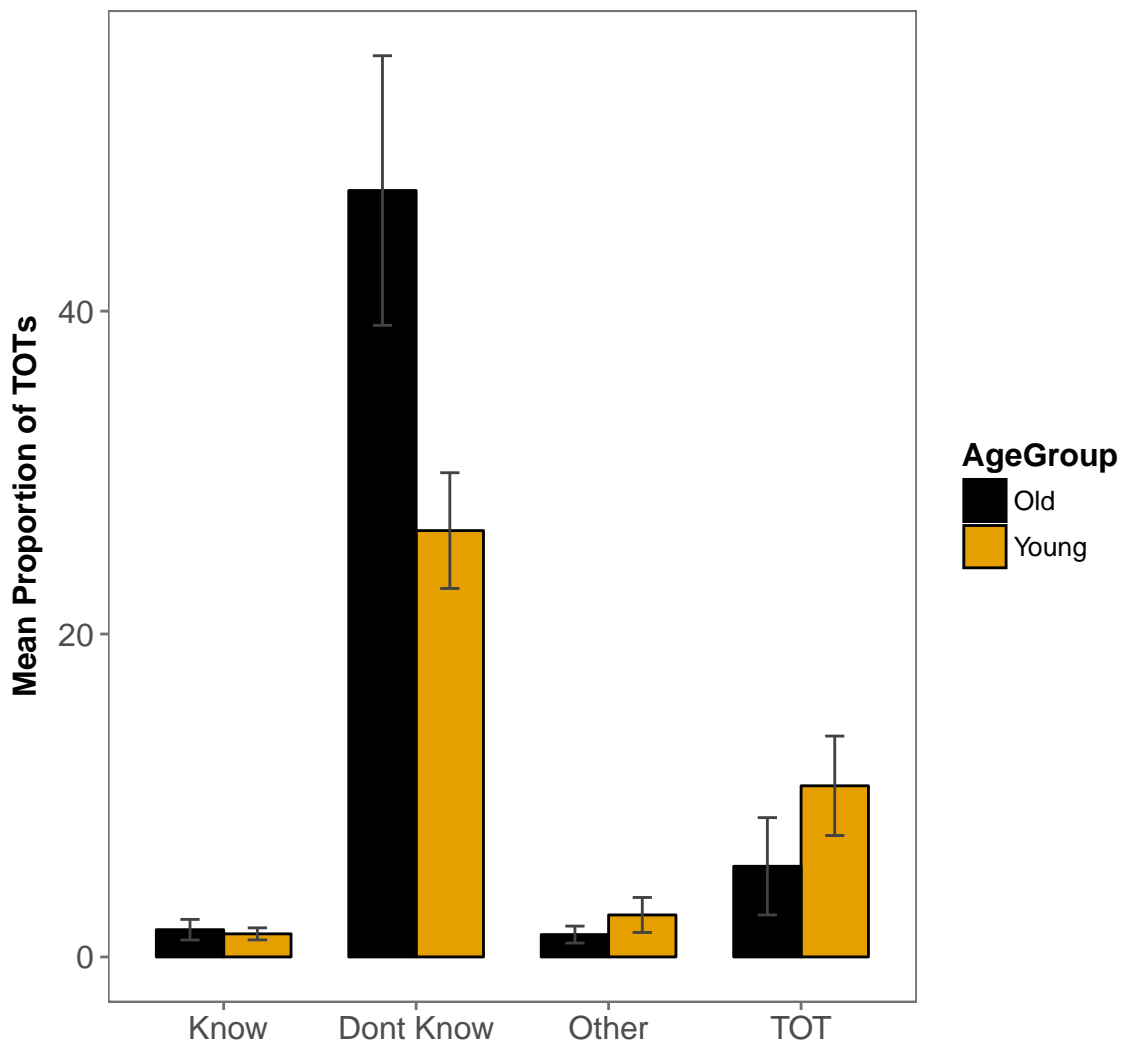
```
> statedata <- read.csv("Julie_Main5Studies.csv", header = TRUE, sep = ",")
> statedata <- subset(statedata, statedata$Subject != 198 & statedata$Subject != 95)
> statedata_TOT = statedata %>% filter(FreeResp %in% c("0", "", "1", "2", "3",
+ "4", "9", "20", "40",
+ "{-}", "{-}{SPACE}",
+ "{SHIFT}", "{SPACE}"))
> ## now we are looking only at trials in which the participant did not respond at all
>
```

```

> NoResp_exp1 = statedata_TOT %>% filter(StudyNo == '2' | StudyNo == '4')
> NoResp_exp2 = statedata_TOT %>% filter(StudyNo == '5' | StudyNo == '6')
> NoResp_exp3 = statedata_TOT %>% filter(StudyNo == '1')
> NoResp_exp2_agg = group_by(NoResp_exp2, Subject, AgeGroup, Question.RESP) %>%
+   summarise(Trials = n())
> exp2_fig_noresp= Rmisc::summarySE(NoResp_exp2_agg,
+                                   measurevar = "Trials",
+                                   groupvars = c("AgeGroup", "Question.RESP"))
> library(ggplot2)
> library(ggthemes)
> exp2_fig_noresp_plot = exp2_fig_noresp %>%
+   mutate (RetrievalState = factor(Question.RESP, levels = unique(Question.RESP),
+                                   labels = c(" Know", "Dont Know", "Other", "TOT")))%>%
+   ggplot(aes(x = RetrievalState, y = Trials,
+              group = AgeGroup, fill = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_colorblind()+
+   xlab("") + ylab("Mean Proportion of TOTs") +
+   ggtitle("E2: Young and Old Adults (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> exp2_fig_noresp_plot

```

E2: Young and Old Adults (With Instructions)



```
> e2_noresp_TOT = NoResp_exp2_agg %>% filter(Question.RESP == "4")
> y_TOT = e2_noresp_TOT %>% filter(AgeGroup == "Young")
> o_TOT = e2_noresp_TOT %>% filter(AgeGroup == "Old")
> t.test(y_TOT$Trials, o_TOT$Trials)
```

Welch Two Sample t-test

```
data: y_TOT$Trials and o_TOT$Trials
t = 2.3737, df = 53.888, p-value = 0.02121
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
```



```

0.7742541 9.1949767
sample estimates:
mean of x mean of y
10.600000  5.615385

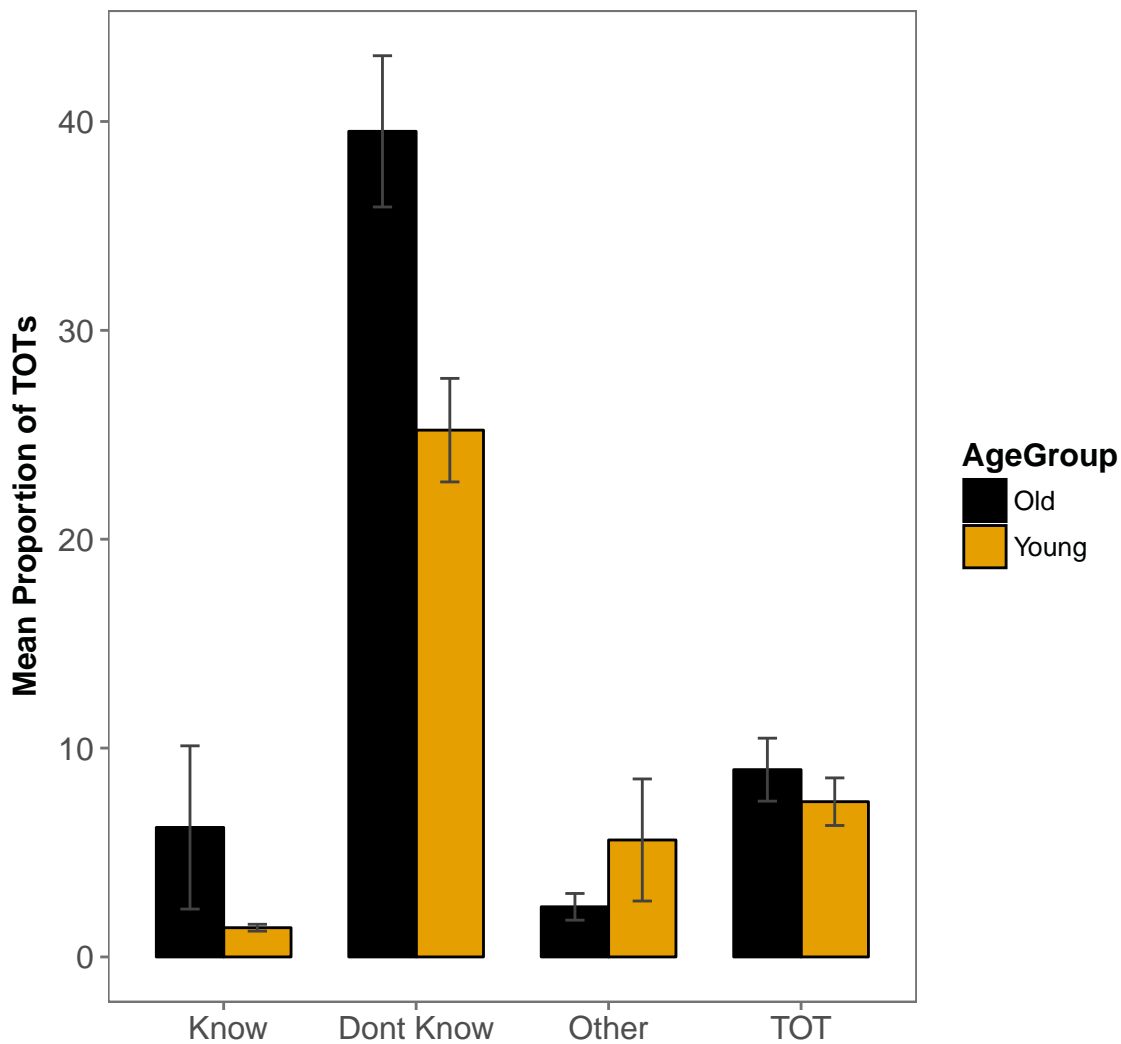
```

```

> ## E1
>
> NoResp_exp1_agg = group_by(NoResp_exp1, Subject, AgeGroup, Question.RESP) %>%
+   summarise(Trials = n())
> exp1_fig_noresp = Rmisc::summarySE(NoResp_exp1_agg,
+                                     measurevar = "Trials",
+                                     groupvars = c("AgeGroup", "Question.RESP"))
> library(ggplot2)
> library(ggthemes)
> exp1_fig_noresp_plot = exp1_fig_noresp %>%
+   mutate (RetrievalState = factor(Question.RESP, levels = unique(Question.RESP),
+                                     labels = c(" Know", "Dont Know", "Other", "TOT")))%>%
+   ggplot(aes(x = RetrievalState, y = Trials,
+               group = AgeGroup, fill = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_colorblind()+
+   xlab("") + ylab("Mean Proportion of TOTs") +
+   ggtitle("E1: Young and Old Adults (No Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> exp1_fig_noresp_plot

```

E1: Young and Old Adults (No Instructions)



```
> ## OA more TOTs than YA when not producing a response: test this AOV
>
> e1_noresp_TOT = NoResp_exp1_agg %>% filter(Question.RESP == "4")
> y_TOT = e1_noresp_TOT %>% filter(AgeGroup == "Young")
> o_TOT = e1_noresp_TOT %>% filter(AgeGroup == "Old")
> t.test(y_TOT$Trials, o_TOT$Trials) ## no difference
```

Welch Two Sample t-test

```
data: y_TOT$Trials and o_TOT$Trials
t = -0.81147, df = 53.981, p-value = 0.4207
```

```

alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-5.321736  2.255069
sample estimates:
mean of x mean of y
7.433333  8.966667

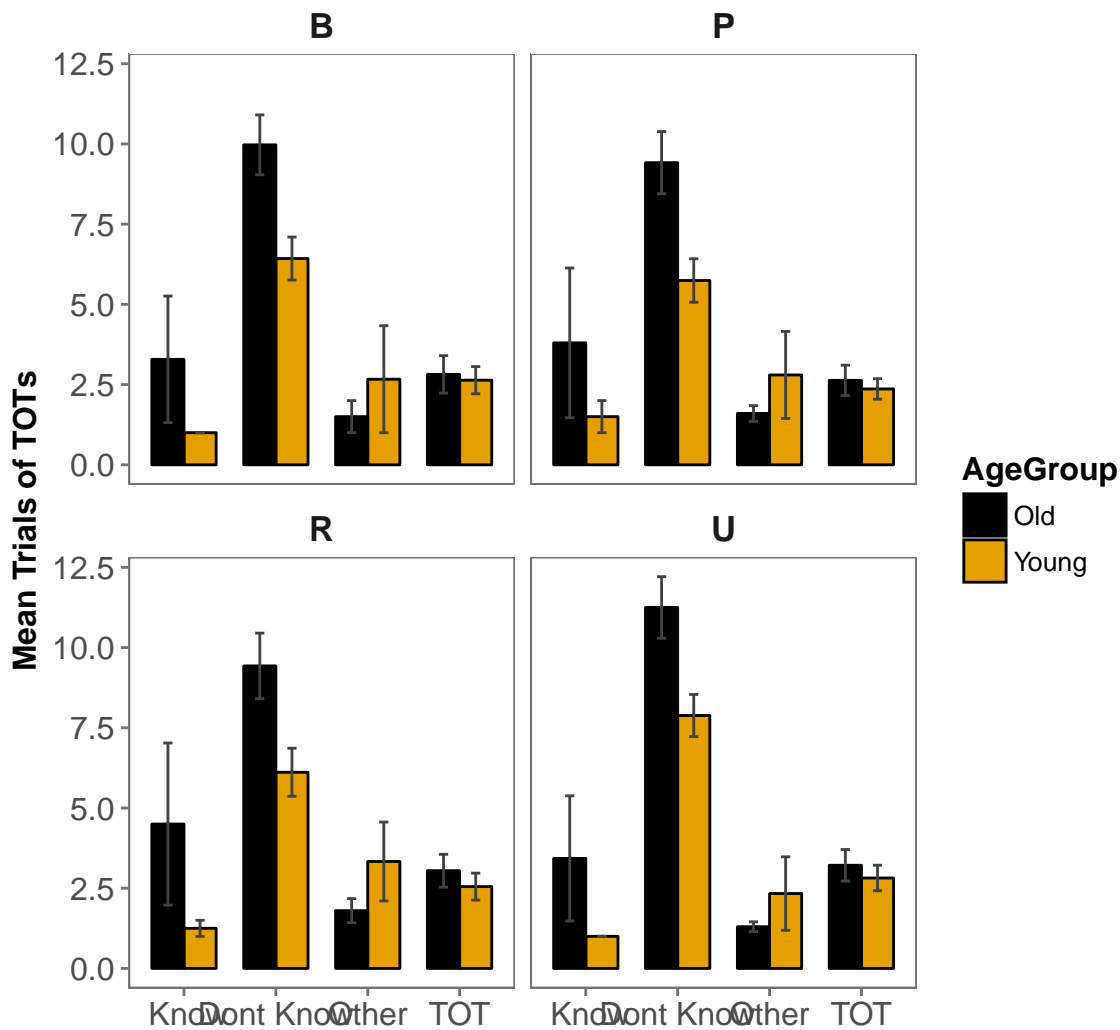
```

```

> ## Does this vary with prime type? ## nope
>
> NoResp_exp1_prime_agg = group_by(NoResp_exp1, Subject, AgeGroup, PrimeCondition, Question)
+ summarise(Trials = n())
> exp1_fig_norespprime = Rmisc::summarySE(NoResp_exp1_prime_agg,
+ measurevar = "Trials",
+ groupvars = c("AgeGroup", "PrimeCondition", "Question.RESP"))
> library(ggplot2)
> library(ggthemes)
> exp1_fig_norespprime_plot = exp1_fig_norespprime %>%
+ mutate (RetrievalState = factor(Question.RESP, levels = unique(Question.RESP),
+ labels = c(" Know", "Dont Know", "Other", "TOT")))%>%
+ ggplot(aes(x = RetrievalState, y = Trials,
+ group = AgeGroup, fill = AgeGroup))+
+ geom_bar(stat = "identity", position = "dodge", width = 0.7, color = "black")+
+ geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+ width=.2, color = "gray26",
+ position = position_dodge(0.7))+
+ theme_few()+
+ facet_wrap(~PrimeCondition)+
+ scale_fill_colorblind()+
+ xlab("") + ylab("Mean Trials of TOTs") +
+ ggtitle("E1: Young and Old Adults (No Instructions)") +
+ theme(axis.text = element_text(size = rel(1)),
+ axis.title = element_text(face = "bold", size = rel(1)),
+ legend.title = element_text(face = "bold", size = rel(1)),
+ plot.title = element_text(hjust = .5),
+ strip.text.x = element_text(face = "bold", size = rel(1.4)))
> exp1_fig_norespprime_plot

```

E1: Young and Old Adults (No Instructions)



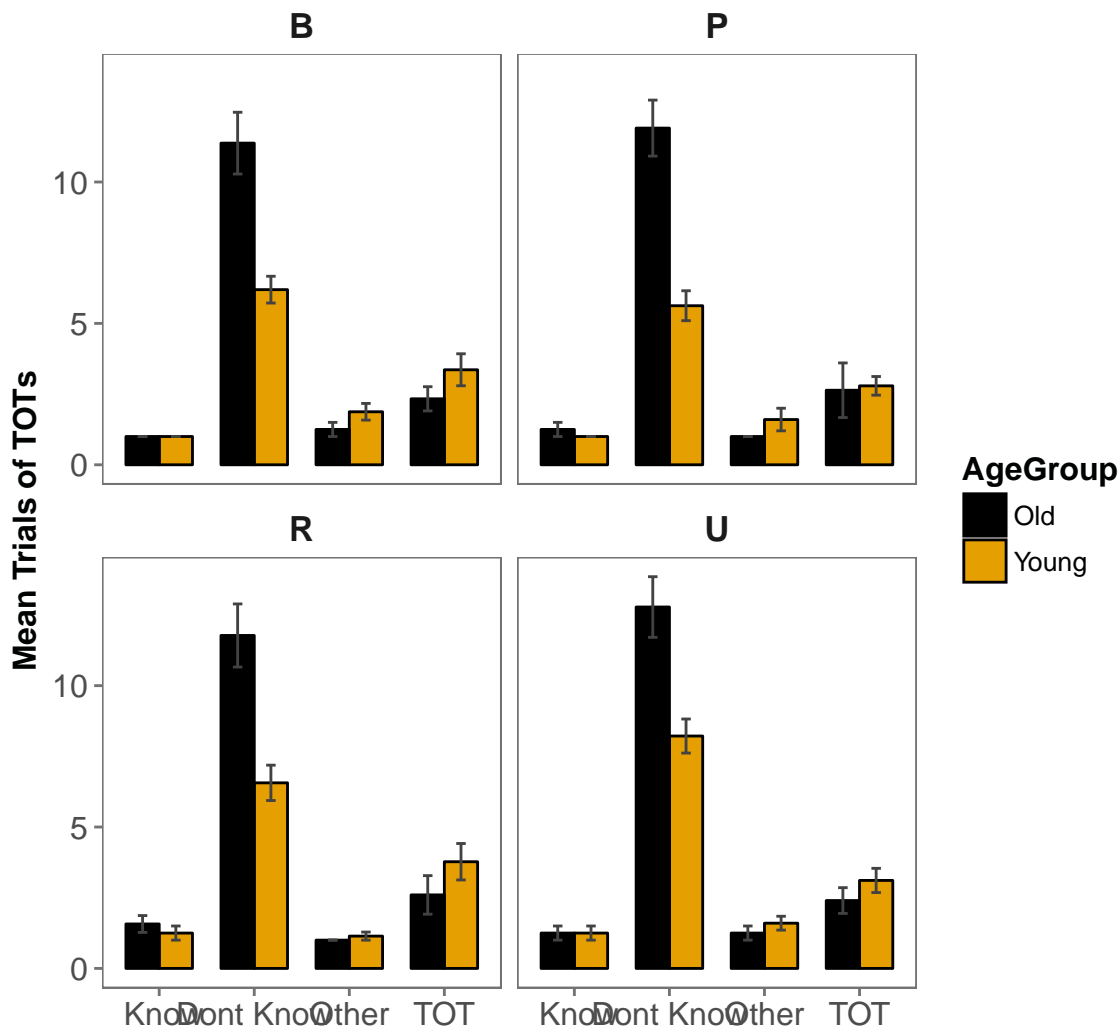
```
> ## E2
> NoResp_exp2_prime_agg = group_by(NoResp_exp2, Subject, AgeGroup, PrimeCondition, Question.RESP)
+ summarise(Trials = n())
> exp2_fig_norespprime = Rmisc::summarySE(NoResp_exp2_prime_agg,
+ measurevar = "Trials",
+ groupvars = c("AgeGroup", "PrimeCondition", "Question.RESP"))
> library(ggplot2)
> library(ggthemes)
> exp2_fig_norespprime_plot = exp2_fig_norespprime %>%
+ mutate (RetrievalState = factor(Question.RESP, levels = unique(Question.RESP),
+ labels = c(" Know", "Dont Know", "Other", "TOT")))%>%
```

```

+ ggplot(aes(x = RetrievalState, y = Trials,
+           group = AgeGroup, fill = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - se, ymax=Trials + se),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   facet_wrap(~PrimeCondition)+
+   scale_fill_colorblind()+
+   xlab("") + ylab("Mean Trials of TOTs") +
+   ggtitle("E2: Young and Old Adults (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> exp2_fig_norespprime_plot

```

E2: Young and Old Adults (With Instructions)



9 TOT: Split by Target Accuracy

```
> ## remove all incorrect TOT: when McAcc was 0
> statedata <- read.csv("Julie_Main5Studies.csv", header = TRUE, sep = ",")
> statedata <- subset(statedata, statedata$Subject != 198 & statedata$Subject != 95)
> statedata$TOTmeasure = ifelse(statedata$Question.RESP == "4" &
+                             statedata$McAcc == "1", "correctTOT",
+                             ifelse(statedata$Question.RESP == "4" &
+                                     statedata$McAcc == "0", "incorrectTOT",
+                                     ifelse(statedata$Question.RESP == "1" &
```

```

+         statedata$Accuracy == "0","incorrectKnow",
+         ifelse(statedata$Question.RESP == "2","dontknow",
+         ifelse(statedata$Question.RESP == "3"&
+         statedata$Accuracy == "0","incorrectOther","NA")))))
> age_statedata = group_by(statedata, AgeGroup,
+         ExperimentName, Subject, TOTmeasure) %>%
+   summarise(Trials = n())
> age_statedata_wide = spread(age_statedata, TOTmeasure, Trials)
> age_statedata_wide$correctTOT = ifelse(is.na(age_statedata_wide$correctTOT),0,
+         age_statedata_wide$correctTOT)
> age_statedata_wide$incorrectTOT = ifelse(is.na(age_statedata_wide$incorrectTOT),0, age
> age_statedata_wide$incorrectKnow = ifelse(is.na(age_statedata_wide$incorrectKnow),0,
age_statedata_wide$incorrectKnow)
> age_statedata_wide$dontknow = ifelse(is.na(age_statedata_wide$dontknow),0,
+         age_statedata_wide$dontknow)
> age_statedata_wide$incorrectOther = ifelse(is.na(age_statedata_wide$incorrectOther),0,
age_statedata_wide$incorrectOther)
> age_statedata_wide = mutate(age_statedata_wide,
+         propTOT = correctTOT/(correctTOT + dontknow +
+         incorrectKnow + incorrectTOT +
+         incorrectOther))
> exp1_age_TOT = age_statedata_wide %>% filter(ExperimentName == "tot extended prime")
> exp2_age_TOT = age_statedata_wide %>% filter(ExperimentName == "tot not the prime")
> e1_TOT_aov = aov(data = exp1_age_TOT, propTOT ~ AgeGroup)
> summary(e1_TOT_aov)

```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup	1	0.0012	0.001197	0.122	0.728
Residuals	70	0.6868	0.009812		

```

> e2_TOT_aov = aov(data = exp2_age_TOT, propTOT ~ AgeGroup)
> summary(e2_TOT_aov)

```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup	1	0.1306	0.13056	13.12	0.000589 ***
Residuals	62	0.6168	0.00995		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

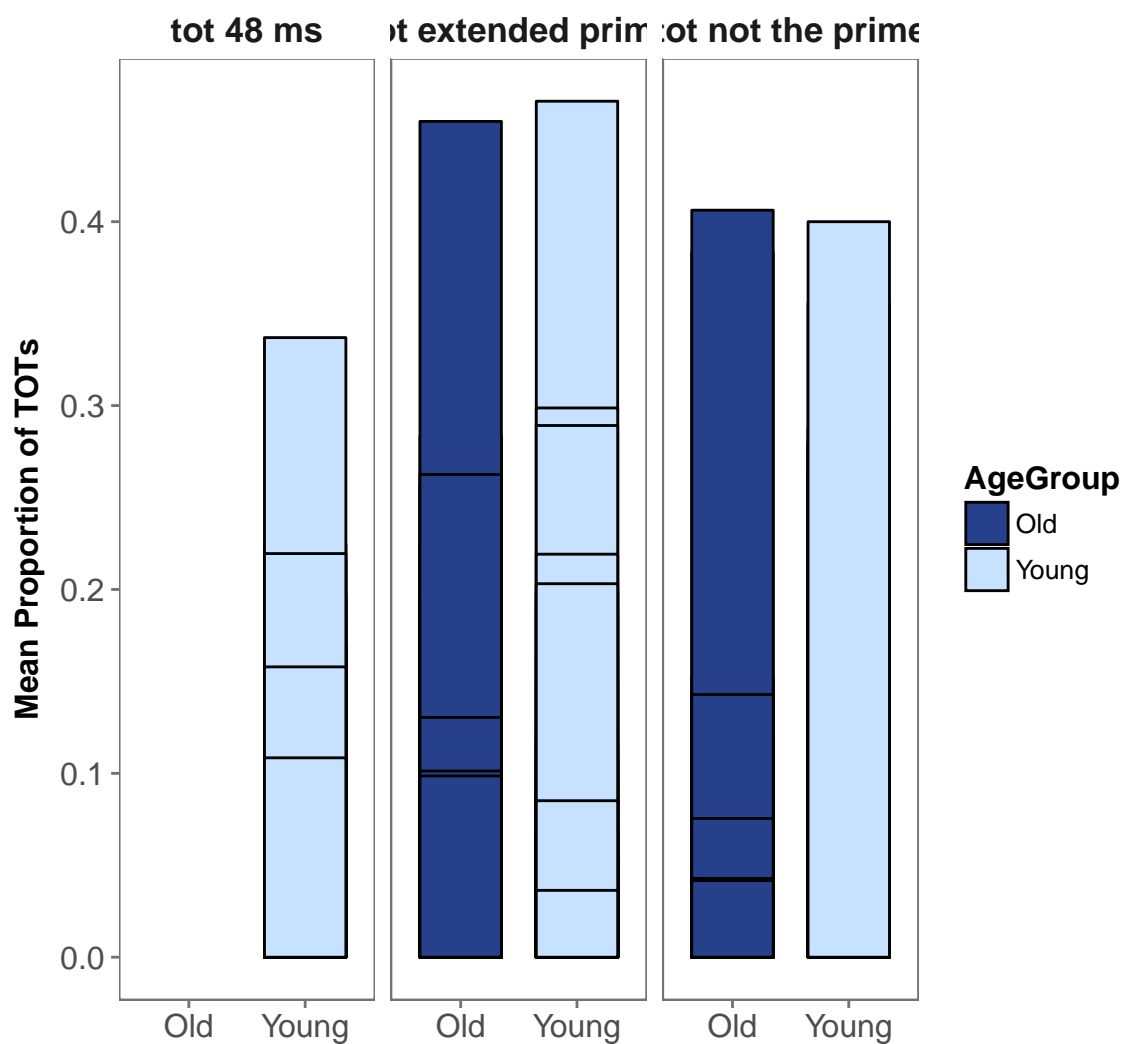
> ## plotting this proportion ## remove subject from dply code
>
> successTOT_plot = age_statedata_wide %>%
+   ggplot(aes(x = AgeGroup, y = propTOT,
+             group = AgeGroup, fill = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7,
+           color= "black")+
+   theme_few()+
+   facet_wrap(~ExperimentName)+

```

```

+   scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("Mean Proportion of TOTs") +
+   ggtitle("") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> successTOT_plot
> ## again, OA lower than YA
>
> burkeTOT = statedata %>% filter(!(Question.RESP == "4" & McAcc == "0"))
>
> ## now we want proportion of correct TOTs as a function of all other unsuccessful retr
>
>
>

```

10 Percent State Analysis

```
> j <- read.csv("MainJulieagg_5studies.csv", header = TRUE, sep = ",")
> j <- subset(j, j$value.Subject != 198 & j$value.Subject != 95)
> #j_statepercent = j[,c(2,3,4,5,104:119)] # use for state percents
>
> j_statepercent = j[,c(2,3,4,5,120:135)] # use for prime percents
> j_statepercent$value.Subject = as.factor(j_statepercent$value.Subject)
> library(tidyr)
> library(dplyr)
```

```

> # use comments for state wise percent
> # statepercent <- j_statepercent %>%
> #   gather(StatePrime, Percent,
> #         know_r_percent, know_p_percent, know_b_percent, know_u_percent,
> #         dontknow_r_percent, dontknow_p_percent,
> #         dontknow_b_percent, dontknow_u_percent,
> #         other_r_percent, other_p_percent, other_b_percent, other_u_percent,
> #         TOT_r_percent, TOT_p_percent, TOT_b_percent, TOT_u_percent) %>%
> #   separate(StatePrime, c('State', 'Prime'), sep = "_") %>%
> #   arrange(value.Subject)
>
> # use below for prime wise percent
> statepercent <- j_statepercent %>%
+   gather(StatePrime, Percent,
+         r_know_new, r_dontknow_new, r_other_new, r_TOT_new,
+         p_know_new, p_dontknow_new, p_other_new, p_TOT_new,
+         b_know_new, b_dontknow_new, b_other_new, b_TOT_new,
+         u_know_new, u_dontknow_new, u_other_new, u_TOT_new) %>%
+   separate(StatePrime, c('Prime', 'State'), sep = "_") %>%
+   arrange(value.Subject)
> # state wise percent
> # colnames(statepercent) = c("AgeGroup", "Subject", "StudyNo", "PrimeInstruction", "State", "Percent")
>
> ## prime wise percent
> colnames(statepercent) = c("AgeGroup", "Subject", "StudyNo", "PrimeInstruction", "PrimeCondition", "State", "Percent")
> statepercent$AgeGroup <- as.factor(statepercent$AgeGroup)
> statepercent$Subject <- as.factor(statepercent$Subject)
> statepercent$StudyNo <- as.factor(statepercent$StudyNo)
> statepercent$PrimeInstruction <- as.factor(statepercent$PrimeInstruction)
> statepercent$PrimeCondition <- as.factor(statepercent$PrimeCondition)
> statepercent$State <- as.factor(statepercent$State)
> statepercent$Percent <- as.numeric(as.character(statepercent$Percent))
> for(i in 1:nrow(statepercent)){
+   if(is.na(statepercent[i,7])) {
+     print(i)
+     statepercent[i,7] = 0
+   }
+   else
+     statepercent[i,7] = statepercent[i,7]
+ }
> statepercent_exp1 = statepercent %>% filter(StudyNo == '2' | StudyNo == '4')
> statepercent_exp2 = statepercent %>% filter(StudyNo == '5' | StudyNo == '6')
> statepercent_exp3 = statepercent %>% filter(StudyNo == '1')
>

```

10.1 Experiment 1

10.1.1 MANOVA

```
> ## we want to do a manova on our data for exp 1
> ## first need to convert each state to wide format
> e1_data_wide <- spread(statepercent_exp1, State, Percent)
> ## grouping
>
> e1_wide_agg = group_by(e1_data_wide, AgeGroup, PrimeCondition) %>%
+   summarise_at(vars(dontknow, know, other, TOT), mean)
> output1 <- manova(cbind(dontknow, know,
+   other, TOT)~AgeGroup*PrimeCondition, data = e1_data_wide )
> summary.aov(output1)
```

```
Response dontknow :
              Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup      1  1.2800  1.28000    31.1289 5.694e-08 ***
PrimeCondition 3   0.2651  0.08837     2.1491  0.09429  .
AgeGroup:PrimeCondition 3   0.0078  0.00259     0.0631  0.97928
Residuals    280 11.5134  0.04112
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Response know :
              Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup      1  0.2006  0.200556    4.7449 0.030220 *
PrimeCondition 3   0.6211  0.207037    4.8982 0.002468 **
AgeGroup:PrimeCondition 3   0.0139  0.004630    0.1095 0.954488
Residuals    280 11.8350  0.042268
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Response other :
              Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup      1  1.9668  1.96681   141.4984 <2e-16 ***
PrimeCondition 3   0.0496  0.01652     1.1888  0.3143
AgeGroup:PrimeCondition 3   0.0108  0.00361     0.2594  0.8546
Residuals    280  3.8920  0.01390
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Response TOT :
              Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup      1  0.03125  0.0312500    3.0356 0.08255  .
PrimeCondition 3   0.04553  0.0151759    1.4742 0.22177
AgeGroup:PrimeCondition 3   0.02571  0.0085685    0.8323 0.47703
Residuals    280  2.88244  0.0102944
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

10.1.2 overall

```
> e1_all_aov = aov(data = statepercent_exp1,
+                   Percent ~ AgeGroup*State*PrimeCondition +
+                   Error(Subject/(State*PrimeCondition)))
> summary(e1_all_aov)
```

Error: Subject

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup	1	5.430e-29	5.428e-29	3.975	0.0501 .
Residuals	70	9.558e-28	1.365e-29		

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Error: Subject:State

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
State	3	20.351	6.784	62.62	< 2e-16 ***
AgeGroup:State	3	3.479	1.160	10.70	1.41e-06 ***
Residuals	210	22.750	0.108		

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Error: Subject:PrimeCondition

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeCondition	3	1.030e-29	3.420e-30	0.672	0.570
AgeGroup:PrimeCondition	3	9.600e-30	3.205e-30	0.630	0.596
Residuals	210	1.068e-27	5.087e-30		

Error: Subject:State:PrimeCondition

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
State:PrimeCondition	9	0.981	0.10904	9.316	2.38e-13 ***
AgeGroup:State:PrimeCondition	9	0.058	0.00647	0.552	0.836
Residuals	630	7.373	0.01170		

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

10.1.3 know

```
> e1_know = statepercent_exp1 %>% filter(State == "know")
> e1_know_aov = aov(data = e1_know,
+                   Percent ~ AgeGroup*PrimeCondition +
+                   Error(Subject/PrimeCondition))
> summary(e1_know_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup 1 0.201 0.2006 1.636 0.205
Residuals 70 8.580 0.1226

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition 3 0.621 0.20704 13.359 5.16e-08 ***
AgeGroup:PrimeCondition 3 0.014 0.00463 0.299 0.826
Residuals 210 3.255 0.01550
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e1_know_aov,
+                               c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = "AgeGroup")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
1	r - u	Old	0.0955556	0.0293429	210	3.256515	0.0071461
2	b - u	Old	0.1133333	0.0293429	210	3.862378	0.0008559
4	p - u	Old	0.1366667	0.0293429	210	4.657573	0.0000335
8	b - u	Young	0.0811111	0.0293429	210	2.764251	0.0313385
10	p - u	Young	0.1133333	0.0293429	210	3.862378	0.0008559

```
> target_p = e1_know %>% filter(PrimeCondition == "p")
> target_r = e1_know %>% filter(PrimeCondition == "r")
> target_b = e1_know %>% filter(PrimeCondition == "b")
> target_u = e1_know %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_r$Percent
t = -3.4694, df = 71, p-value = 0.0008909
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.12247907 -0.03307648
sample estimates:
mean of the differences
 -0.07777778
```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = -4.9679, df = 71, p-value = 4.506e-06
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.1362439 -0.0582005
sample estimates:
mean of the differences
 -0.09722222
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = -6.1735, df = 71, p-value = 3.698e-08
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.16537312 -0.08462688
sample estimates:
mean of the differences
 -0.125
```

```
> ## old diff in know semantic and know unrelated
>
> old_semantic = e1_know %>% filter(PrimeCondition == "r" & AgeGroup == "Old")
> old_unrel = e1_know %>% filter(PrimeCondition == "u" & AgeGroup == "Old")
> t.test(old_semantic$Percent, old_unrel$Percent, paired = TRUE)
```

Paired t-test

```
data: old_semantic$Percent and old_unrel$Percent
t = 3.361, df = 35, p-value = 0.001889
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.03783746 0.15327365
sample estimates:
mean of the differences
 0.09555556
```

10.1.4 dont know

```
> e1_dontknow = statepercent_exp1 %>% filter(State == "dontknow")
> e1_dontknow_aov = aov(data = e1_dontknow,
+       Percent ~ AgeGroup*PrimeCondition +
+       Error(Subject/PrimeCondition))
> summary(e1_dontknow_aov)
```

Error: Subject

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup	1	1.280	1.2800	9.417	0.00306 **
Residuals	70	9.514	0.1359		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeCondition	3	0.2651	0.08837	9.283	8.61e-06 ***
AgeGroup:PrimeCondition	3	0.0078	0.00259	0.272	0.845
Residuals	210	1.9991	0.00952		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e1_dontknow_aov,
+       c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+       adjust = "tukey", details = TRUE, by = "AgeGroup")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
4	u - r	Old	0.0866667	0.0229971	210	3.768595	0.0012136
5	u - p	Old	0.0766667	0.0229971	210	3.333758	0.0055511
6	u - b	Old	0.0633333	0.0229971	210	2.753974	0.0322399
10	u - r	Young	0.0722222	0.0229971	210	3.140496	0.0103363
11	u - p	Young	0.0611111	0.0229971	210	2.657343	0.0418744

```
> target_p = e1_dontknow %>% filter(PrimeCondition == "p")
> target_r = e1_dontknow %>% filter(PrimeCondition == "r")
> target_b = e1_dontknow %>% filter(PrimeCondition == "b")
> target_u = e1_dontknow %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```

data: target_u$Percent and target_r$Percent
t = 4.1572, df = 71, p-value = 8.878e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.03844905 0.10932873
sample estimates:
mean of the differences
      0.07388889

```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```

data: target_u$Percent and target_b$Percent
t = 3.8613, df = 71, p-value = 0.0002463
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.02606076 0.08171702
sample estimates:
mean of the differences
      0.05388889

```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```

data: target_u$Percent and target_p$Percent
t = 4.3341, df = 71, p-value = 4.735e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.04019578 0.10869311
sample estimates:
mean of the differences
      0.07444444

```

```

> target_y = e1_dontknow %>% filter(AgeGroup == "Young")
> target_o = e1_dontknow %>% filter(AgeGroup == "Old")
> t.test(target_y$Percent, target_o$Percent, paired = FALSE)

```

Welch Two Sample t-test

```

data: target_y$Percent and target_o$Percent
t = -5.5731, df = 258.87, p-value = 6.273e-08
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.18044440 -0.08622227
sample estimates:

```



```
mean of x mean of y
0.2688889 0.4022222
```

```
>
```

10.1.5 other

```
> e1_other = statepercent_exp1 %>% filter(State == "other")
> e1_other_aov = aov(data = e1_other,
+                     Percent ~ AgeGroup*PrimeCondition +
+                     Error(Subject/PrimeCondition))
> summary(e1_other_aov)
```

```
Error: Subject
```

```
      Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup  1   1.967   1.9668    51.85 5.37e-10 ***
Residuals 70   2.655   0.0379
```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Error: Subject:PrimeCondition
```

```
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3  0.0496  0.016524    2.806 0.0407 *
AgeGroup:PrimeCondition  3  0.0108  0.003606    0.612 0.6078
Residuals      210  1.2368  0.005890
```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e1_other_aov,
+                               c("AgeGroup","PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = "PrimeCondition")
> library(knitr)
> kable(subset(prime_effect$comparisons,prime_effect$comparisons$p.value < 0.05 ))
```

contrast	PrimeCondition	estimate	SE	df	t.ratio	p.value
Young - Old b		0.1633333	0.0277887	140.2586	5.877683	0e+00
Young - Old p		0.1466667	0.0277887	140.2586	5.277919	5e-07
Young - Old r		0.1711111	0.0277887	140.2586	6.157573	0e+00
Young - Old u		0.1800000	0.0277887	140.2586	6.477447	0e+00

```
> target_y = e1_other %>% filter(AgeGroup == "Young")
> target_o = e1_other %>% filter(AgeGroup == "Old")
> t.test(target_y$Percent, target_o$Percent, paired = FALSE)
```

Welch Two Sample t-test

```
data: target_y$Percent and target_o$Percent
t = 11.93, df = 232.16, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.1379820 0.1925736
sample estimates:
 mean of x mean of y
0.21888889 0.05361111
```

10.1.6 TOT

```
> e1_TOT = statepercent_exp1 %>% filter(State == "TOT")
> e1_TOT_aov = aov(data = e1_TOT,
+                 Percent ~ AgeGroup*PrimeCondition +
+                 Error(Subject/PrimeCondition))
> summary(e1_TOT_aov)
```

Error: Subject

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup	1	0.0312	0.03125	1.094	0.299
Residuals	70	1.9997	0.02857		

Error: Subject:PrimeCondition

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeCondition	3	0.0455	0.015176	3.610	0.0142 *
AgeGroup:PrimeCondition	3	0.0257	0.008569	2.038	0.1096
Residuals	210	0.8828	0.004204		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e1_TOT_aov,
+                               c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE, by = "AgeGroup")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
4	u - p	Old	0.0600000	0.0152819	210	3.926216	0.0006721
5	u - b	Old	0.0411111	0.0152819	210	2.690185	0.0383542

```
> target_o_u = e1_TOT %>% filter(AgeGroup == "Old" & PrimeCondition == "u")
> target_o_p = e1_TOT %>% filter(AgeGroup == "Old" & PrimeCondition == "p")
> target_o_b = e1_TOT %>% filter(AgeGroup == "Old" & PrimeCondition == "b")
> target_o_r = e1_TOT %>% filter(AgeGroup == "Old" & PrimeCondition == "r")
> t.test(target_o_u$Percent, target_o_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_o_u$Percent and target_o_p$Percent
t = 4.9651, df = 35, p-value = 1.783e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.03546745 0.08453255
sample estimates:
mean of the differences
      0.06
```

```
> t.test(target_o_u$Percent, target_o_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_o_u$Percent and target_o_r$Percent
t = 1.5792, df = 35, p-value = 0.1233
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-0.007930356 0.063485911
sample estimates:
mean of the differences
 0.02777778
```

```
> t.test(target_o_u$Percent, target_o_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_o_u$Percent and target_o_b$Percent
t = 2.4882, df = 35, p-value = 0.01775
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.007569415 0.074652807
sample estimates:
mean of the differences
 0.04111111
```

```
> target_u = e1_TOT %>% filter(PrimeCondition == "u")
> target_p = e1_TOT %>% filter(PrimeCondition == "p")
> target_b = e1_TOT %>% filter(PrimeCondition == "b")
> target_r = e1_TOT %>% filter(PrimeCondition == "r")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_r$Percent
t = 1.5348, df = 71, p-value = 0.1293
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.005484392  0.042151058
sample estimates:
mean of the differences
      0.01833333
```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = 2.3211, df = 71, p-value = 0.02316
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  0.003680573  0.048541650
sample estimates:
mean of the differences
      0.02611111
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = 3.6375, df = 71, p-value = 0.0005182
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  0.01531223  0.05246555
sample estimates:
mean of the differences
      0.03388889
```

```
> t.test(target_p$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_p$Percent and target_r$Percent
t = -1.3179, df = 71, p-value = 0.1918
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.039090415  0.007979304
sample estimates:
mean of the differences
 -0.01555556
```

```
> t.test(target_b$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_b$Percent and target_r$Percent
t = -0.88769, df = 71, p-value = 0.3777
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.025248324  0.009692769
sample estimates:
mean of the differences
 -0.007777778
```

```
>
```

10.1.7 plot

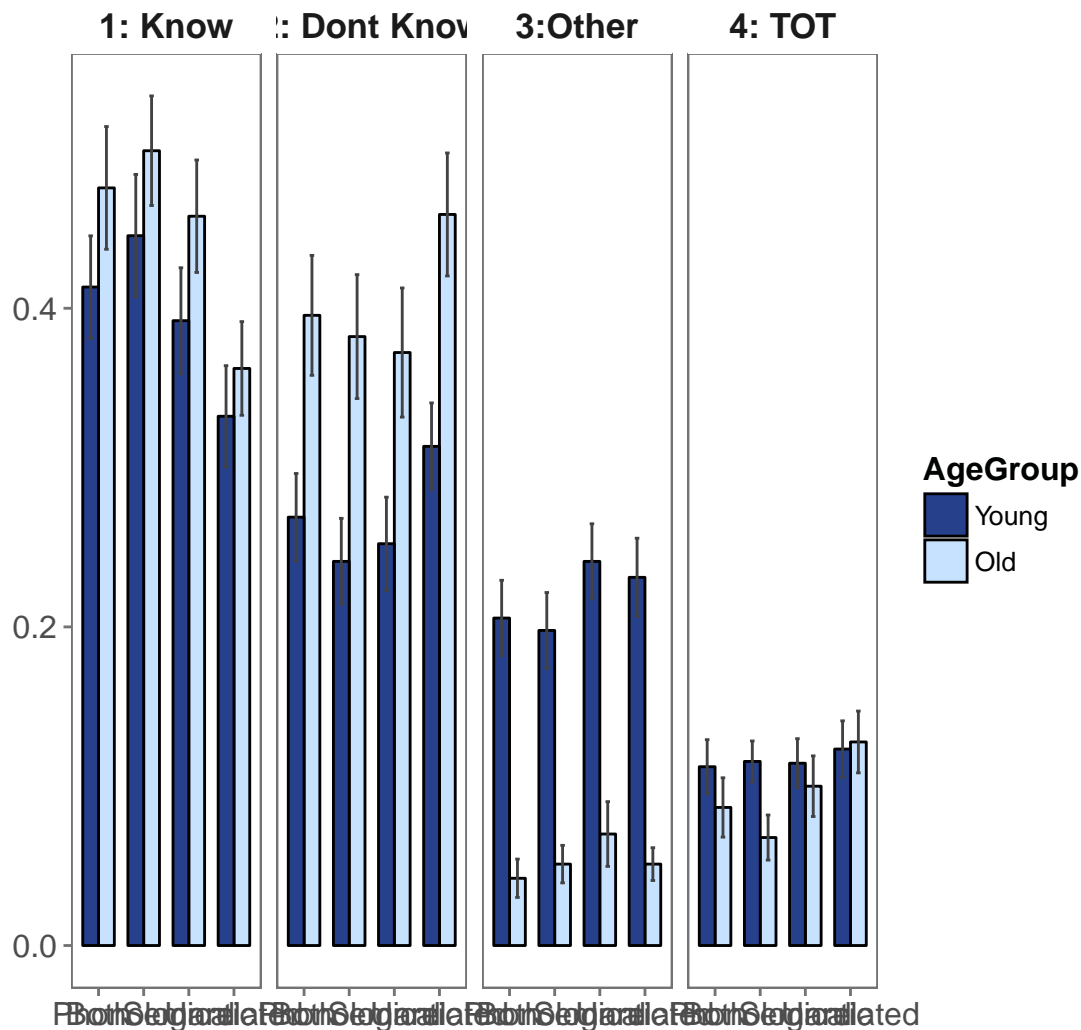
```
> exp1_statepercent = Rmisc::summarySE(statepercent_exp1,
+                                     measurevar = "Percent",
+                                     groupvars = c("State", "AgeGroup", "PrimeCondition"))
> exp1_statepercent$RetrievalState = factor(exp1_statepercent$State, levels(exp1_statepercent$State))
> exp1_statepercent$AgeGroup = factor(exp1_statepercent$AgeGroup, levels(exp1_statepercent$AgeGroup))
> #write.csv(exp1_statepercent, file = "exp1_statepercent.csv")
> exp1_statepercent = read.csv("exp1_statepercent.csv", sep = ",",
+                              header = TRUE)
> exp1_statepercent$AgeGroup = factor(exp1_statepercent$AgeGroup, levels(exp1_statepercent$AgeGroup))
> library(ggplot2)
> library(ggthemes)
> e1_percentplot = exp1_statepercent %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Both", "Phonological",
+                                         "Semantic", "Unrelated")),
+          R = factor(RetrievalState, levels = unique(RetrievalState),
+                     labels = c("1: Know", "2: Dont Know",
+                                 "3: Other", "4: TOT")))%>%
```

```

+
+ ggplot(aes(x = PrimeType, y = Percent,
+           group = AgeGroup, fill = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7,
+           color= "black")+
+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   facet_wrap(~R, nrow =1)+
+   scale_fill_manual(values = c( "royalblue4", "slategray1"))+
+   xlab("") + ylab("") +
+   ggtitle("E1: Young and Old Adults (No Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         axis.text.x = element_text(size = rel(1)),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e1_percentplot
>

```

E1: Young and Old Adults (No Instructions)



10.1.8 Know Only

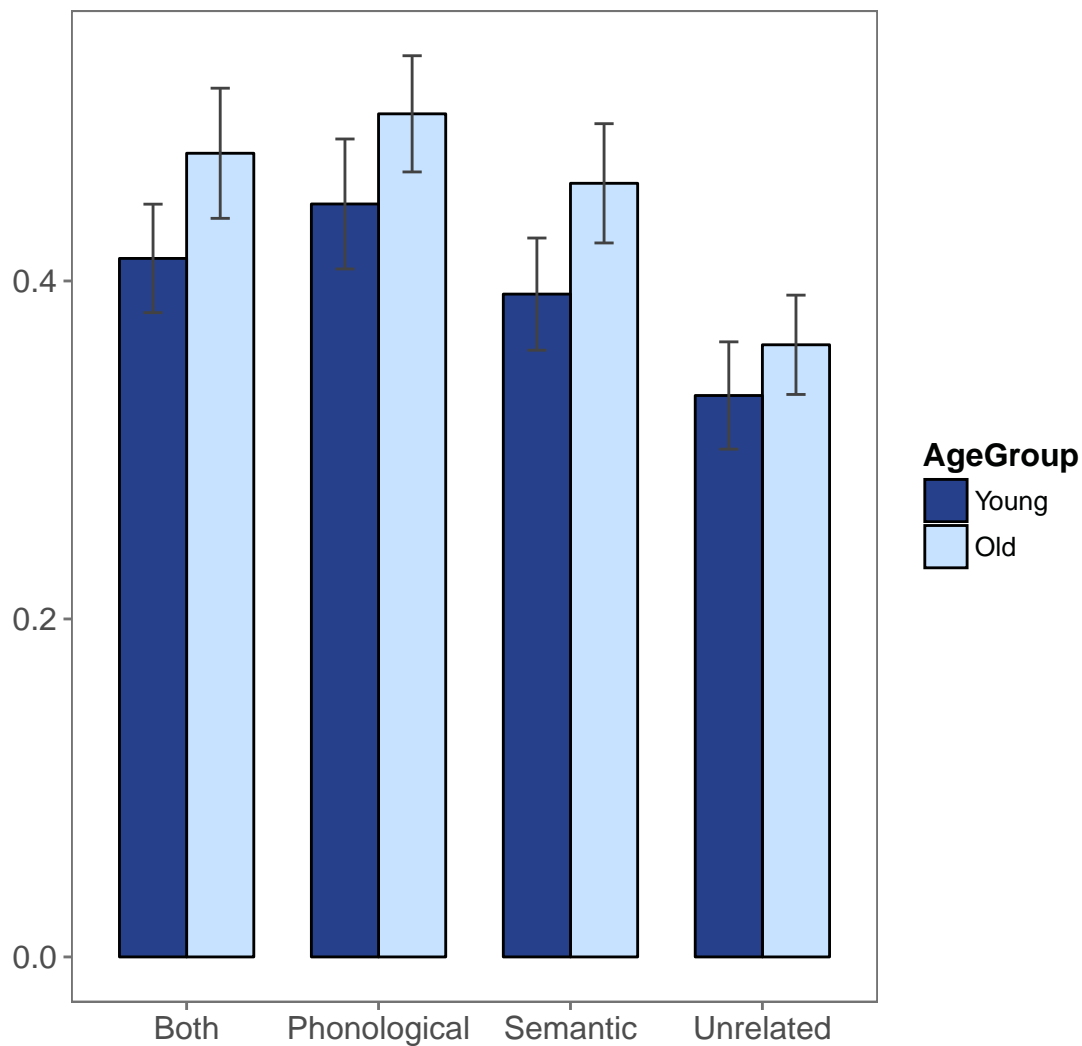
```
> exp1_statepercent_know = exp1_statepercent %>% filter(RetrievalState == "know")
> e1_percentplot_know = exp1_statepercent_know %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Both", "Phonological",
+                                         "Semantic", "Unrelated")))%>%
+   ggplot(aes(x = PrimeType, y = Percent,
+              group = AgeGroup, fill = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7,
+           color = "black")+
+   theme_minimal()
```

```

+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_manual(values = c( "royalblue4", "slategray1"))+
+   xlab("") + ylab("") +
+   ggtitle("E1: Young and Old Adults (No Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e1_percentplot_know

```


E1: Young and Old Adults (No Instructions)



10.2 Experiment 2

10.2.1 MANOVA

```
> ## we want to do a manova on our data for exp 1
> ## first need to convert each state to wide format
> e2_data_wide <- spread(statepercent_exp2, State, Percent)
> ## grouping
>
> e1_wide_agg = group_by(e2_data_wide, AgeGroup, PrimeCondition) %>%
+   summarise_at(vars(dontknow, know, other, TOT), mean)
```

```
> output2 <- manova(cbind(dontknow, know,
+ other, TOT)~AgeGroup*PrimeCondition, data = e2_data_wide )
> summary.aov(output2)
```

```
Response dontknow :
              Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup      1  2.1536  2.15356  55.7314 1.407e-12 ***
PrimeCondition 3   0.2640  0.08801   2.2775  0.08018 .
AgeGroup:PrimeCondition 3  0.0419  0.01397   0.3616  0.78079
Residuals    248  9.5831  0.03864
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Response know :
              Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup      1  0.1722  0.172225   4.4640  0.03562 *
PrimeCondition 3   0.3836  0.127875   3.3145  0.02064 *
AgeGroup:PrimeCondition 3  0.0649  0.021642   0.5609  0.64127
Residuals    248  9.5680  0.038581
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Response other :
              Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup      1  1.24881  1.24881 155.6184 <2e-16 ***
PrimeCondition 3   0.01657  0.00552   0.6882  0.5600
AgeGroup:PrimeCondition 3  0.00627  0.00209   0.2604  0.8539
Residuals    248  1.99015  0.00802
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Response TOT :
              Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup      1  0.58523  0.58523  55.7850 1.376e-12 ***
PrimeCondition 3   0.01645  0.00548   0.5227   0.6671
AgeGroup:PrimeCondition 3  0.00372  0.00124   0.1184   0.9493
Residuals    248  2.60170  0.01049
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

10.2.2 overall

```
> e2_all_aov = aov(data = statepercent_exp2,
+ Percent ~ AgeGroup*State*PrimeCondition +
+ Error(Subject/(State*PrimeCondition)))
> summary(e2_all_aov)
```

```
Error: Subject
      Df      Sum Sq    Mean Sq F value Pr(>F)
AgeGroup  1 2.190e-30 2.191e-30    1.58  0.213
Residuals 62 8.598e-29 1.387e-30

Error: Subject:State
      Df Sum Sq Mean Sq F value    Pr(>F)
State      3 20.89   6.962   69.44 < 2e-16 ***
AgeGroup:State  3  4.16   1.387   13.83 3.52e-08 ***
Residuals    186 18.65   0.100

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition
      Df      Sum Sq    Mean Sq F value Pr(>F)
PrimeCondition  3 4.000e-32 1.320e-32    0.024  0.995
AgeGroup:PrimeCondition  3 1.000e-31 3.470e-32    0.063  0.979
Residuals    186 1.022e-28 5.494e-31

Error: Subject:State:PrimeCondition
      Df Sum Sq Mean Sq F value    Pr(>F)
State:PrimeCondition  9  0.681 0.07563   8.284 1.27e-11 ***
AgeGroup:State:PrimeCondition  9  0.117 0.01298   1.422  0.175
Residuals    558  5.095 0.00913

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

10.2.3 know

```
> e2_know = statepercent_exp2 %>% filter(State == "know")
> e2_know_aov = aov(data = e2_know,
+                   Percent ~ AgeGroup*PrimeCondition +
+                   Error(Subjct/PrimeCondition))
> summary(e2_know_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  0.172   0.1722   1.425  0.237
Residuals 62  7.491   0.1208

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition  3 0.3836 0.12788  11.451 6.33e-07 ***
AgeGroup:PrimeCondition  3 0.0649 0.02164   1.938  0.125
Residuals    186 2.0771 0.01117

---
```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e2_know_aov,
+                               c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE, by = "AgeGroup")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
7	b - u	Young	0.09750	0.0264184	186	3.690605	0.0016599
8	r - u	Young	0.10375	0.0264184	186	3.927183	0.0006942
10	p - u	Young	0.14875	0.0264184	186	5.630539	0.0000004

```
> target_p = e2_know %>% filter(PrimeCondition == "p")
> target_r = e2_know %>% filter(PrimeCondition == "r")
> target_b = e2_know %>% filter(PrimeCondition == "b")
> target_u = e2_know %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_r$Percent
t = -3.9791, df = 63, p-value = 0.0001813
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.1136042 -0.0376458
sample estimates:
mean of the differences
 -0.075625
```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = -4.4743, df = 63, p-value = 3.277e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.10578469 -0.04046531
sample estimates:
mean of the differences
 -0.073125
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = -5.3881, df = 63, p-value = 1.129e-06
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.14394262 -0.06605738
sample estimates:
mean of the differences
      -0.105
```

```
> ## old diff in know semantic and know unrelated
>
> old_semantic = e2_know %>% filter(PrimeCondition == "r" & AgeGroup == "Old")
> old_unrel = e2_know %>% filter(PrimeCondition == "u" & AgeGroup == "Old")
> t.test(old_semantic$Percent, old_unrel$Percent, paired = TRUE)
```

Paired t-test

```
data: old_semantic$Percent and old_unrel$Percent
t = 2.3047, df = 31, p-value = 0.02804
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.0061851 0.1013149
sample estimates:
mean of the differences
      0.05375
```

10.2.4 dont know

```
> e2_dontknow = statepercent_exp2 %>% filter(State == "dontknow")
> e2_dontknow_aov = aov(data = e2_dontknow,
+                        Percent ~ AgeGroup*PrimeCondition +
+                        Error(Subject/PrimeCondition))
> summary(e2_dontknow_aov)
```

Error: Subject

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup	1	2.154	2.1536	16.55	0.000137 ***
Residuals	62	8.070	0.1302		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition

```

          Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition      3  0.2640  0.08801    10.820 1.38e-06 ***
AgeGroup:PrimeCondition  3  0.0419  0.01397     1.718    0.165
Residuals          186  1.5129  0.00813
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e2_dontknow_aov,
+                               c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE, by = "AgeGroup")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))

```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
4	u - b	Old	0.06125	0.0225467	186	2.716582	0.0360154
5	u - r	Old	0.06000	0.0225467	186	2.661142	0.0417764
10	u - b	Young	0.10500	0.0225467	186	4.656998	0.0000359
11	u - r	Young	0.10125	0.0225467	186	4.490677	0.0000730
12	u - p	Young	0.07250	0.0225467	186	3.215546	0.0082895

```

> target_p = e2_dontknow %>% filter(PrimeCondition == "p")
> target_r = e2_dontknow %>% filter(PrimeCondition == "r")
> target_b = e2_dontknow %>% filter(PrimeCondition == "b")
> target_u = e2_dontknow %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)

```

Paired t-test

```

data: target_u$Percent and target_r$Percent
t = 3.9738, df = 63, p-value = 0.0001846
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.03293392 0.09956608
sample estimates:
mean of the differences
      0.06625

```

```

> t.test(target_u$Percent, target_b$Percent, paired = TRUE)

```

Paired t-test

```

data: target_u$Percent and target_b$Percent

```

```
t = 6.0538, df = 63, p-value = 8.598e-08
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.05442948 0.10807052
sample estimates:
mean of the differences
      0.08125
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = 4.8548, df = 63, p-value = 8.3e-06
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.04228942 0.10146058
sample estimates:
mean of the differences
      0.071875
```

```
> target_y = e2_dontknow %>% filter(AgeGroup == "Young")
> target_o = e2_dontknow %>% filter(AgeGroup == "Old")
> t.test(target_y$Percent, target_o$Percent, paired = FALSE)
```

Welch Two Sample t-test

```
data: target_y$Percent and target_o$Percent
t = -7.4373, df = 207.49, p-value = 2.667e-12
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.2320625 -0.1348125
sample estimates:
mean of x mean of y
0.2940625 0.4775000
```

```
>
```

10.2.5 other

```
> e2_other = statepercent_exp2 %>% filter(State == "other")
> e2_other_aov = aov(data = e2_other,
+                   Percent ~ AgeGroup*PrimeCondition +
+                   Error(Subject/PrimeCondition))
> summary(e2_other_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup  1  1.249   1.2488   60.39 9.89e-11 ***
Residuals 62  1.282   0.0207
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition  3 0.0166  0.005523   1.451  0.229
AgeGroup:PrimeCondition  3 0.0063  0.002090   0.549  0.649
Residuals      186 0.7080  0.003806
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e2_other_aov,
+                               c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = "PrimeCondition")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

contrast	PrimeCondition	estimate	SE	df	t.ratio	p.value
Young - Old	b	0.14875	0.0223953	135.5897	6.642016	0e+00
Young - Old	p	0.13625	0.0223953	135.5897	6.083863	0e+00
Young - Old	r	0.12500	0.0223953	135.5897	5.581526	1e-07
Young - Old	u	0.14875	0.0223953	135.5897	6.642016	0e+00

```
> target_y = e2_other %>% filter(AgeGroup == "Young")
> target_o = e2_other %>% filter(AgeGroup == "Old")
> t.test(target_y$Percent, target_o$Percent, paired = FALSE)
```

Welch Two Sample t-test

```
data: target_y$Percent and target_o$Percent
t = 12.553, df = 173, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.1177235 0.1616515
sample estimates:
mean of x mean of y
0.1775000 0.0378125
```

```
>
```


10.2.6 TOT

```
> e2_TOT = statepercent_exp2 %>% filter(State == "TOT")
> e2_TOT_aov = aov(data = e2_TOT,
+                   Percent ~ AgeGroup*PrimeCondition +
+                   Error(Subject/PrimeCondition))
> summary(e2_TOT_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup  1  0.5852   0.5852    20.1 3.24e-05 ***
Residuals 62  1.8051   0.0291
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition  3  0.0165  0.005483    1.28  0.283
AgeGroup:PrimeCondition  3  0.0037  0.001242    0.29  0.833
Residuals      186  0.7966  0.004283
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e2_TOT_aov,
+                               c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE, by = "AgeGroup")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

```
|contrast|AgeGroup| estimate| SE| df| t.ratio| p.value|
|:-----|:-----|:-----|:--|:--|:-----|:-----|
```

```
> target_y = e2_TOT %>% filter(AgeGroup == "Young")
> target_o = e2_TOT %>% filter(AgeGroup == "Old")
> t.test(target_y$Percent, target_o$Percent, paired = FALSE)
```

Welch Two Sample t-test

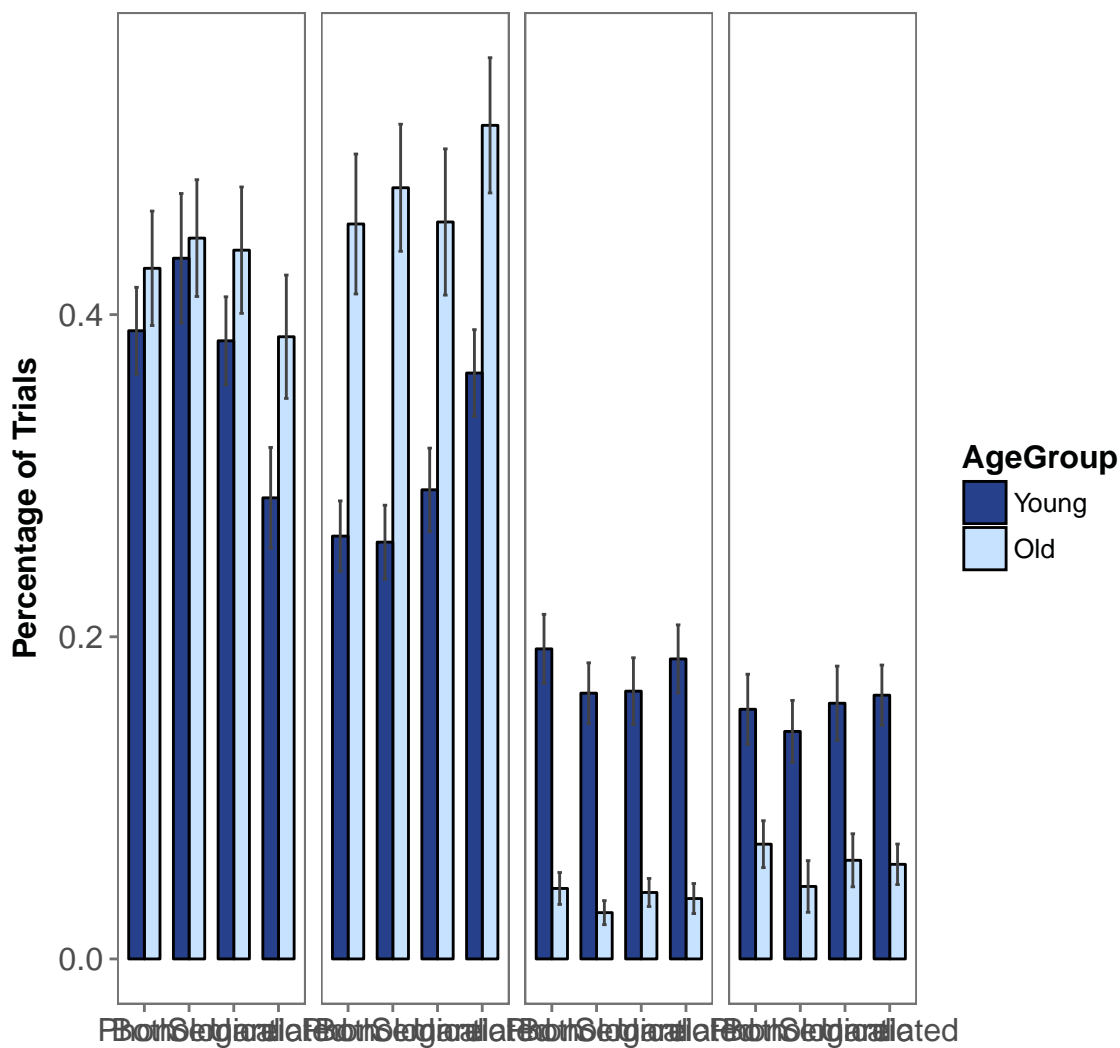
```
data: target_y$Percent and target_o$Percent
t = 7.5296, df = 231.5, p-value = 1.13e-12
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.07060291 0.12064709
sample estimates:
mean of x mean of y
0.1546875 0.0590625
```

```
>
```

10.2.7 plot

```
> exp2_statepercent = Rmisc::summarySE(statepercent_exp2,
+                                     measurevar = "Percent",
+                                     groupvars = c("State", "AgeGroup", "PrimeCondition"))
> exp2_statepercent$RetrievalState = factor(exp2_statepercent$State, levels(exp2_statepercent$State))
> exp2_statepercent$AgeGroup = factor(exp2_statepercent$AgeGroup, levels(exp2_statepercent$AgeGroup))
> #write.csv(exp2_statepercent, file = "exp2_statepercent.csv")
> exp2_statepercent = read.csv("exp2_statepercent.csv", sep = ",",
+                              header = TRUE)
> exp2_statepercent$AgeGroup = factor(exp2_statepercent$AgeGroup, levels(exp2_statepercent$AgeGroup))
> library(ggplot2)
> library(ggthemes)
> e2_percentplot = exp2_statepercent %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Both", "Phonological",
+                                         "Semantic", "Unrelated")),
+          RetrievalState = factor(RetrievalState, levels = unique(RetrievalState),
+                                  labels = c("1: Know", "2: Dont Know", "3: Other", "4: Other")))
> ggplot(aes(x = PrimeType, y = Percent,
+             group = AgeGroup, fill = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7,
+            color = "black")+
+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   facet_wrap(~RetrievalState, nrow = 1)+
+   scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("Percentage of Trials") +
+   ggtitle("E2: Young and Old Adults (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         axis.text.x = element_text(size = rel(1)),
+         strip.text.x = element_blank())
> e2_percentplot
>
```

E2: Young and Old Adults (With Instructions)



10.2.8 Know Only

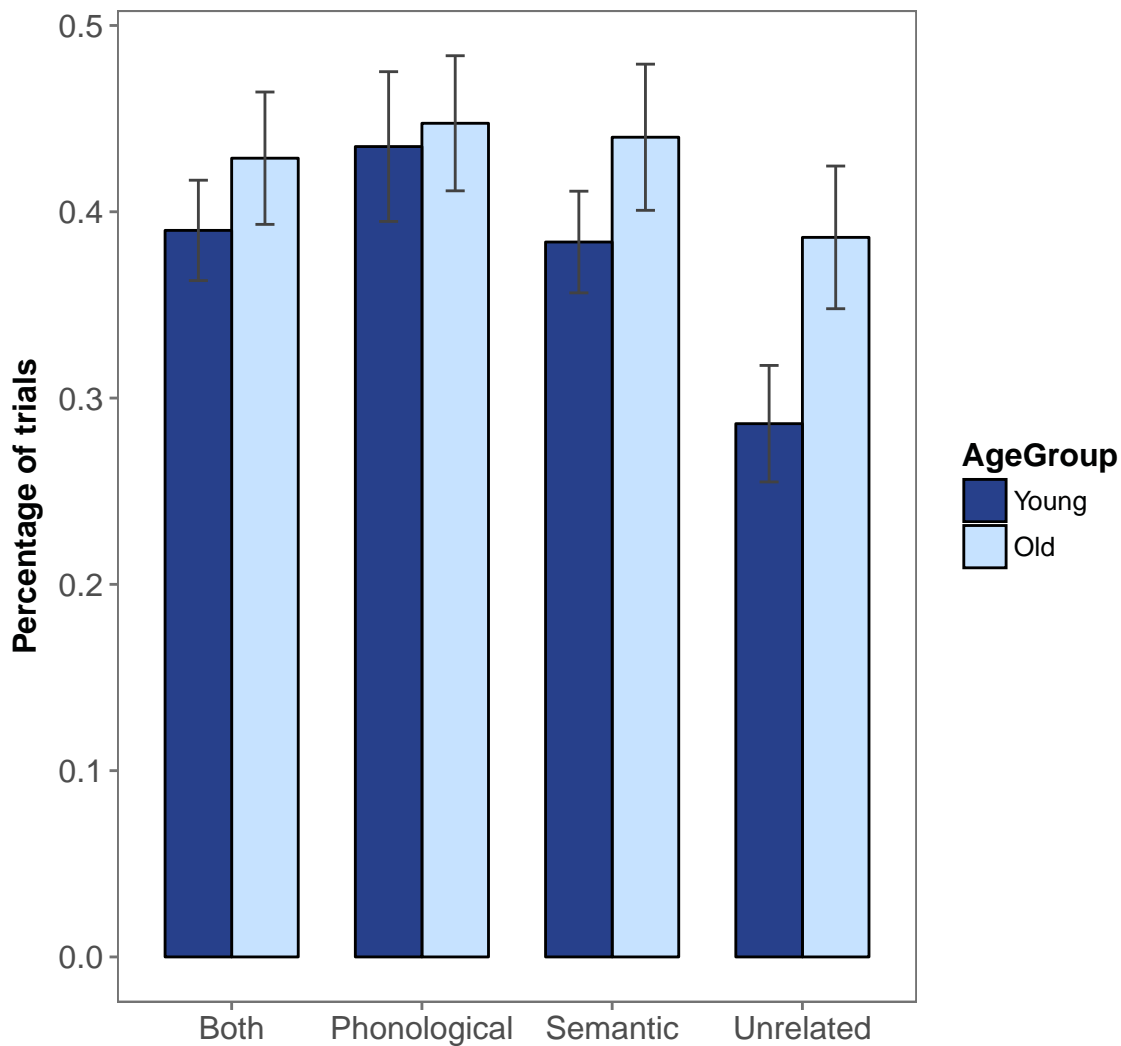
```
> exp2_statepercent_know = exp2_statepercent %>% filter(RetrievalState == "know")
> e2_percentplot_know = exp2_statepercent_know %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Both", "Phonological",
+                                         "Semantic", "Unrelated")))%>%
+   ggplot(aes(x = PrimeType, y = Percent,
+             group = AgeGroup, fill = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7,
+           color = "black")+
+   theme_minimal()
```

```

+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_manual(values = c( "royalblue4", "slategray1"))+
+   xlab("") + ylab("Percentage of trials") +
+   ggtitle("E2: Young and Old Adults (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e2_percentplot_know

```

E2: Young and Old Adults (With Instructions)



10.3 Experiment 3

10.3.1 MANOVA

```
> ## we want to do a manova on our data for exp 1
> ## first need to convert each state to wide format
> e3_data_wide <- spread(statepercent_exp3, State, Percent)
> ## grouping
>
> e3_wide_agg = group_by(e3_data_wide, AgeGroup, PrimeCondition) %>%
+   summarise_at(vars(dontknow, know, other, TOT), mean)
```

```
> output3 <- manova(cbind(dontknow, know,
+ other, TOT)~ PrimeCondition, data = e3_data_wide )
> summary.aov(output3)
```

```
Response dontknow :
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3 0.08756 0.029185  1.6243 0.1865
Residuals     140 2.51547 0.017968

Response know :
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3 0.2729 0.090974  2.3291 0.07709 .
Residuals     140 5.4684 0.039060
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Response other :
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3 0.00199 0.000663  0.0417 0.9886
Residuals     140 2.22600 0.015900

Response TOT :
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3 0.03689 0.012296  0.9118 0.437
Residuals     140 1.88791 0.013485
```

10.3.2 overall

```
> e3_all_aov = aov(data = statepercent_exp3,
+ Percent ~ State*PrimeCondition +
+ Error(Subject/(State*PrimeCondition)))
> summary(e3_all_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 35 1.123e-29 3.21e-31

Error: Subject:State
      Df Sum Sq Mean Sq F value Pr(>F)
State   3 5.464 1.8215 21.8 4.71e-11 ***
Residuals 105 8.772 0.0835
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3 1.141e-30 3.802e-31 1.88 0.138
```

```

Residuals      105 2.124e-29 2.023e-31

Error: Subject:State:PrimeCondition
              Df Sum Sq Mean Sq F value    Pr(>F)
State:PrimeCondition    9  0.399  0.04437    4.202 3.79e-05 ***
Residuals              315  3.326  0.01056
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

10.3.3 know

```

> e3_know = statepercent_exp3 %>% filter(State == "know")
> e3_know_aov = aov(data = e3_know,
+                   Percent ~ PrimeCondition +
+                   Error(Subject/PrimeCondition))
> summary(e3_know_aov)

```

```

Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 35  4.082  0.1166

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition    3 0.2729 0.09097    6.889 0.00028 ***
Residuals      105 1.3867 0.01321
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e3_know_aov,
+                               c("PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE)
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))

```

	contrast	estimate	SE	df	t.ratio	p.value
2	b - u	0.0944444	0.0270867	105	3.486740	0.0039328
4	r - u	0.1155556	0.0270867	105	4.266129	0.0002524

```

> target_p = e3_know %>% filter(PrimeCondition == "p")
> target_r = e3_know %>% filter(PrimeCondition == "r")
> target_b = e3_know %>% filter(PrimeCondition == "b")
> target_u = e3_know %>% filter(PrimeCondition == "u")

```

```
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_r$Percent
t = -4.174, df = 35, p-value = 0.0001881
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.17175765 -0.05935346
sample estimates:
mean of the differences
 -0.1155556
```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = -3.2616, df = 35, p-value = 0.002474
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.15322942 -0.03565947
sample estimates:
mean of the differences
 -0.09444444
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = -2.6248, df = 35, p-value = 0.01276
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.11822819 -0.01510515
sample estimates:
mean of the differences
 -0.06666667
```

10.3.4 dont know

```
> e3_dontknow = statepercent_exp3 %>% filter(State == "dontknow")
> e3_dontknow_aov = aov(data = e3_dontknow,
+                       Percent ~ PrimeCondition +
+                       Error(Subject/PrimeCondition))
> summary(e3_dontknow_aov)
```



```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 35  1.664  0.04754

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3  0.0876  0.029185  3.598  0.016 *
Residuals      105  0.8516  0.008111
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e3_dontknow_aov,
+                               c("PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE)
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	estimate	SE	df	t.ratio	p.value
4	u - r	0.0644444	0.0212275	105	3.035898	0.0157648

```
> target_p = e3_dontknow %>% filter(PrimeCondition == "p")
> target_r = e3_dontknow %>% filter(PrimeCondition == "r")
> target_b = e3_dontknow %>% filter(PrimeCondition == "b")
> target_u = e3_dontknow %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_r$Percent
t = 3.136, df = 35, p-value = 0.003461
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.02272621 0.10616268
sample estimates:
mean of the differences
 0.06444444
```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = 2.8858, df = 35, p-value = 0.006645
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01614348 0.09274541
sample estimates:
mean of the differences
      0.05444444
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = 1.9488, df = 35, p-value = 0.05936
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.001899482 0.093010593
sample estimates:
mean of the differences
      0.04555556
```

10.3.5 other

```
> e3_other = statepercent_exp3 %>% filter(State == "other")
> e3_other_aov = aov(data = e3_other,
+                   Percent ~ PrimeCondition +
+                   Error(Subject/PrimeCondition))
> summary(e3_other_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 35  1.615  0.04614

Error: Subject:PrimeCondition
      Df Sum Sq  Mean Sq F value Pr(>F)
PrimeCondition  3 0.0020 0.000663  0.114  0.952
Residuals     105 0.6112 0.005821
```

```
>
```

10.3.6 TOT

```
> e3_TOT = statepercent_exp3 %>% filter(State == "TOT")
> e3_TOT_aov = aov(data = e3_TOT,
+                   Percent ~ PrimeCondition +
```

```
+ Error(Subject/PrimeCondition))
> summary(e3_TOT_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 35  1.411  0.04032

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3  0.0369  0.01230  2.708 0.0489 *
Residuals      105  0.4767  0.00454
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e3_TOT_aov,
+                               c("PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE)
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	estimate	SE	df	t.ratio	p.value
1	u - r	0.0433333	0.0158817	105	2.728509	0.0369009

```
> target_p = e3_TOT %>% filter(PrimeCondition == "p")
> target_r = e3_TOT %>% filter(PrimeCondition == "r")
> target_b = e3_TOT %>% filter(PrimeCondition == "b")
> target_u = e3_TOT %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_r$Percent
t = 3.1114, df = 35, p-value = 0.003695
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01505952 0.07160715
sample estimates:
mean of the differences
 0.04333333
```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = 2.0797, df = 35, p-value = 0.04494
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.0007425523 0.0614796700
sample estimates:
mean of the differences
      0.03111111
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = 1.2259, df = 35, p-value = 0.2284
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-0.01239224 0.05017002
sample estimates:
mean of the differences
      0.01888889
```

10.3.7 plot

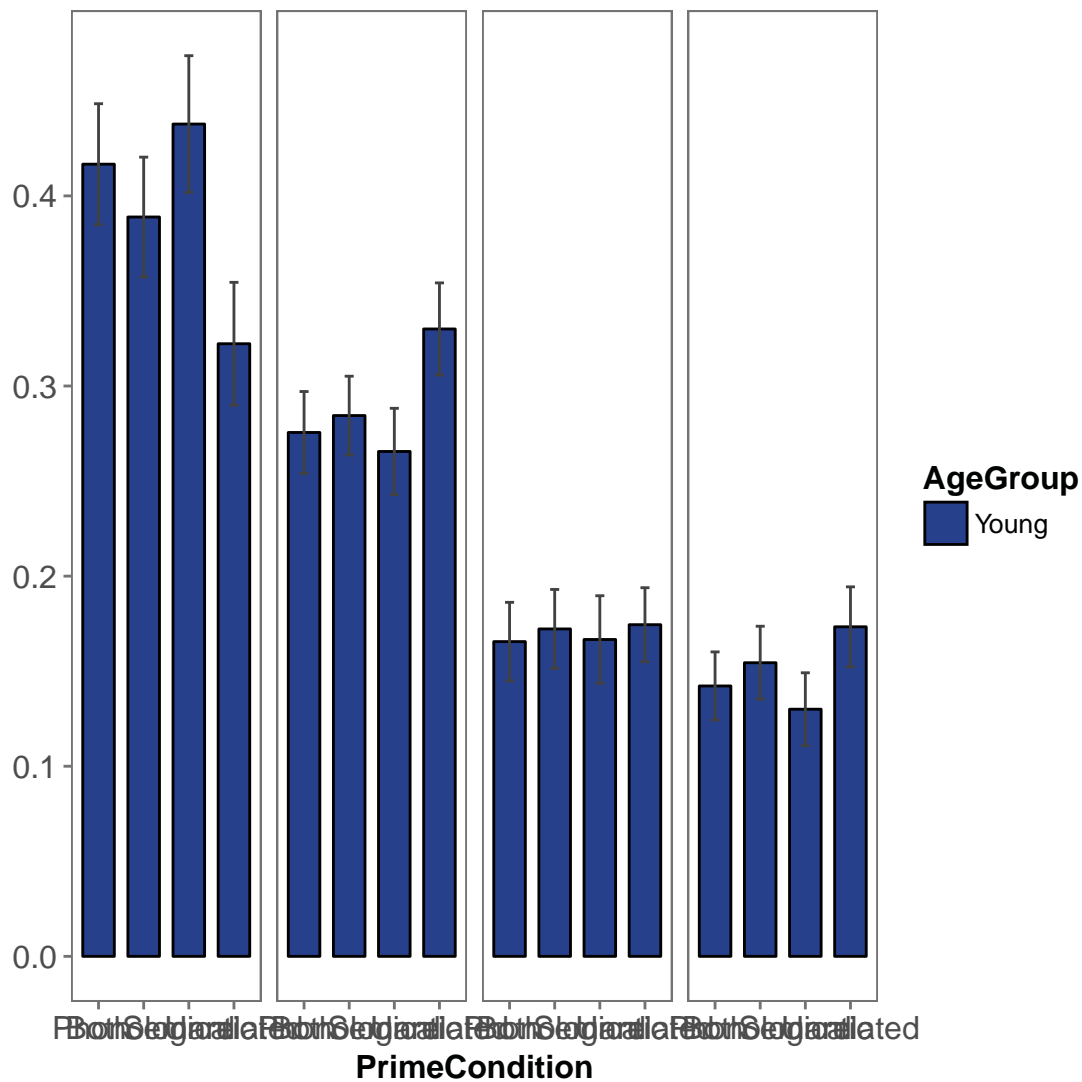
```
> exp3_statepercent = Rmisc::summarySE(statepercent_exp3,
+                                     measurevar = "Percent",
+                                     groupvars = c("State", "AgeGroup", "PrimeCondition"))
> exp3_statepercent$RetrievalState = factor(exp3_statepercent$State, levels(exp3_statepercent$State))
> #write.csv(exp3_statepercent, file = "exp3_statepercent.csv")
> exp3_statepercent = read.csv("exp3_statepercent.csv", sep = ",",
+                              header = TRUE)
> library(ggplot2)
> library(ggthemes)
> e3_percentplot = exp3_statepercent %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Both", "Phonological",
+                                         "Semantic", "Unrelated")),
+          RetrievalState = factor(RetrievalState, levels = unique(RetrievalState),
+                                  labels = c("1: Know", "2: Dont Know", "3: Other", "4: Other")))
> ggplot(aes(x = PrimeType, y = Percent,
+             group = AgeGroup, fill = AgeGroup)) +
+   geom_bar(stat = "identity", position = "dodge", width = 0.7,
+           color = "black") +
+   geom_errorbar(aes(ymin = Percent - se, ymax = Percent + se),
```

```

+         width=.2, color = "gray26",
+         position = position_dodge(0.7))+
+ theme_few()+
+ facet_wrap(~RetrievalState, nrow =1 )+
+   scale_fill_manual(values = c("royalblue4"))+
+   xlab("PrimeCondition") + ylab("") +
+   ggtitle("E3: Young Adults Only (48 ms)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         axis.text.x = element_text(size = rel(1)),
+         strip.text.x = element_blank())
> e3_percentplot
>

```

E3: Young Adults Only (48 ms)



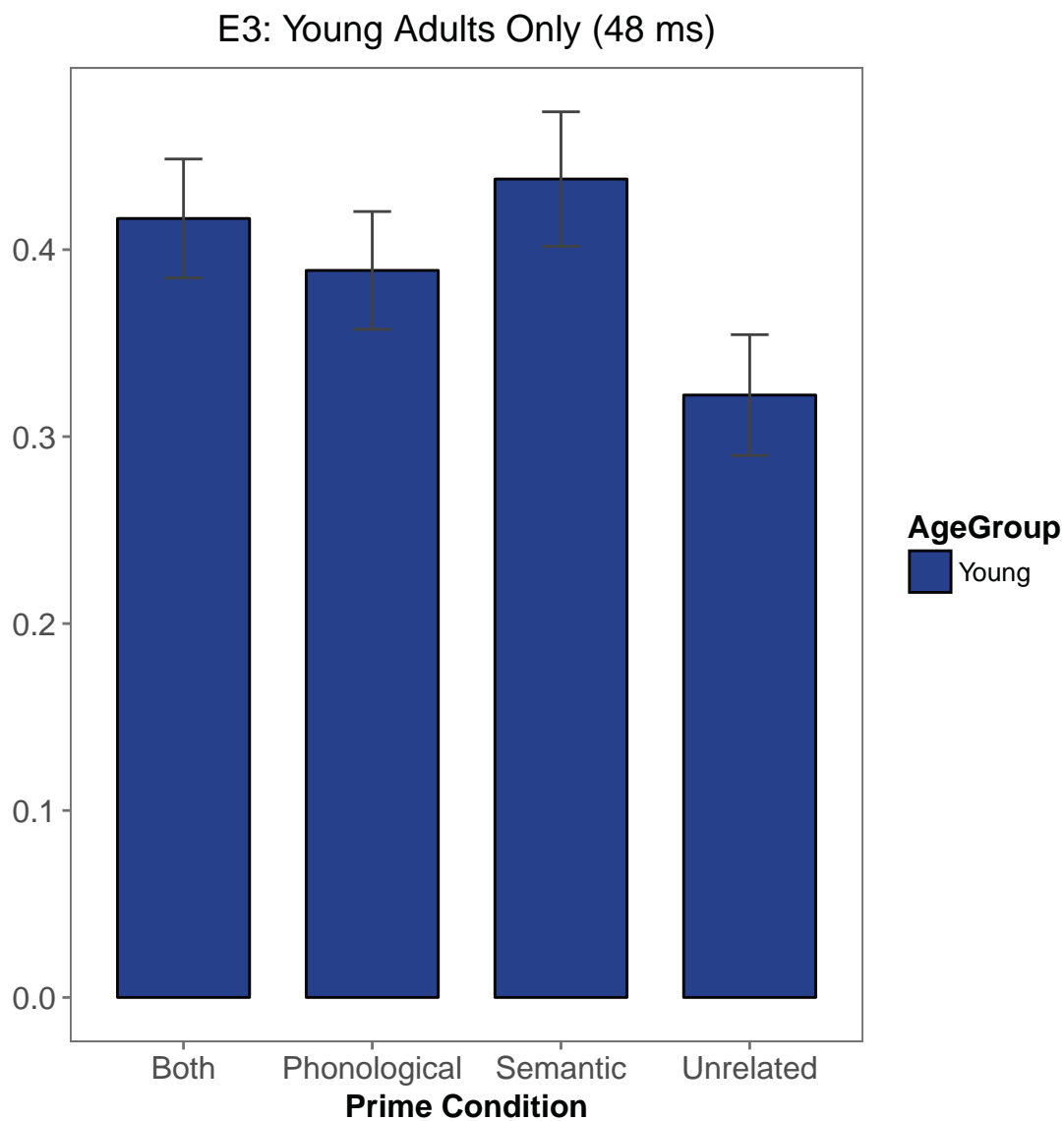
10.3.8 Know Only

```
> exp3_statepercent_know = exp3_statepercent %>% filter(RetrievalState == "know")
> e3_percentplot_know = exp3_statepercent_know %>%
+   mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                             labels = c("Both", "Phonological",
+                                         "Semantic", "Unrelated")))%>%
+   ggplot(aes(x = PrimeType, y = Percent,
+             group = AgeGroup, fill = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7,
+           color = "black")+
+   theme_minimal()
```

```

+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_manual(values = c( "royalblue4", "slategray1"))+
+   xlab("Prime Condition") + ylab("") +
+   ggtitle("E3: Young Adults Only (48 ms)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e3_percentplot_know

```

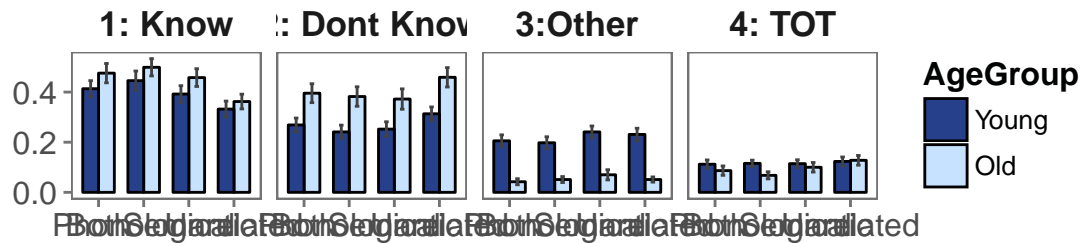


10.4 Combined Plot for State Percent

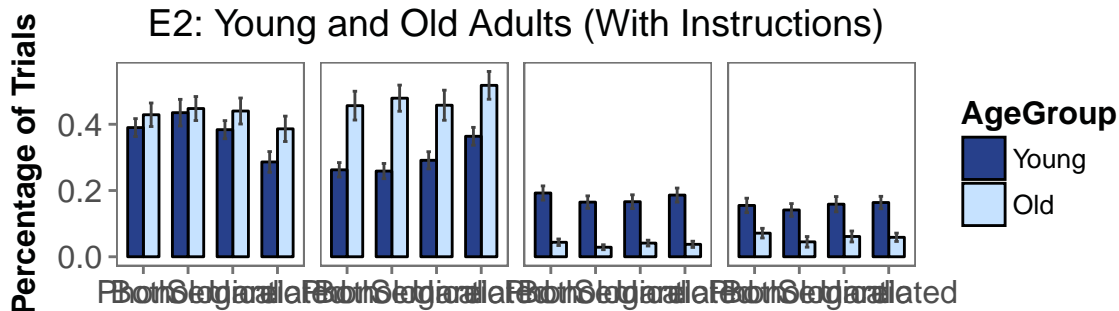
```
> library(grid)
> gridExtra::grid.arrange(e1_percentplot, e2_percentplot, e3_percentplot, nrow = 3, ncol = 1,
+                           top=textGrob("Percentage of Retrieval States Across Experiments 1, 2, 3",
+                                       gp=gpar(fontsize=20)))
```


Percentage of Retrieval States Across Experiments 1,

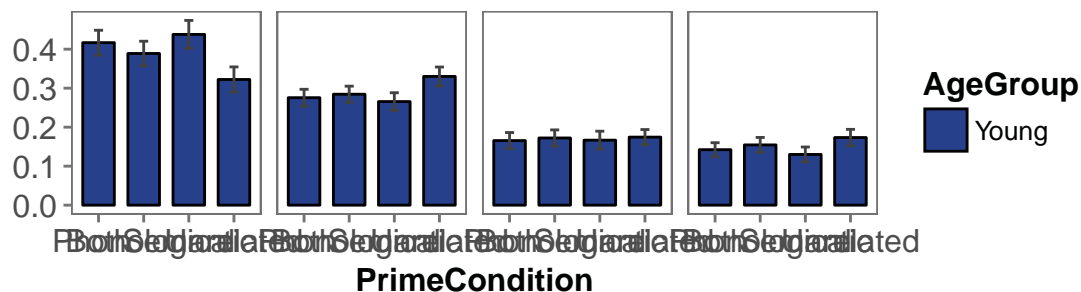
E1: Young and Old Adults (No Instructions)



E2: Young and Old Adults (With Instructions)



E3: Young Adults Only (48 ms)

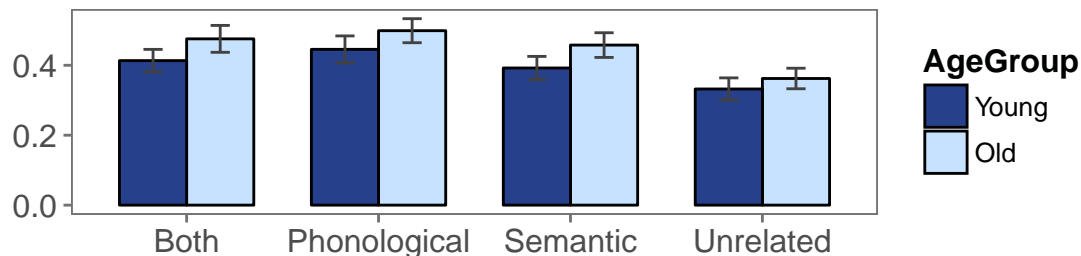


10.4.1 Combined Plot for Know Percent

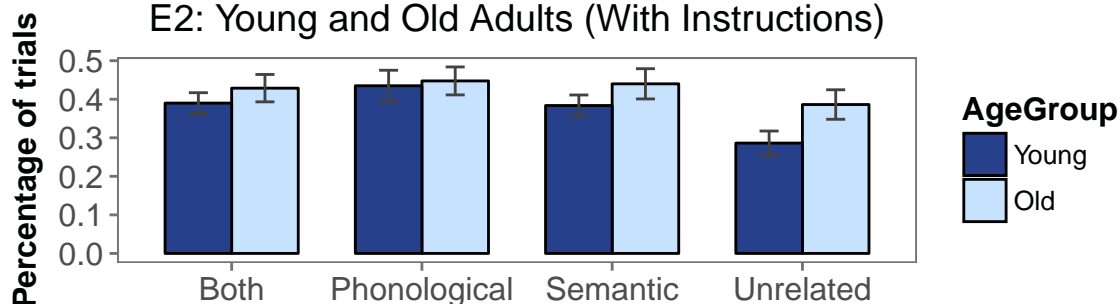
```
> library(grid)
> gridExtra::grid.arrange(e1_percentplot_know, e2_percentplot_know, e3_percentplot_know,
+                           top=textGrob("Percentage of Know responses Across Experiments 1, 2, 3",
+                                       gp=gpar(fontsize=18)))
```

Percentage of Know responses Across Experiments 1, 2

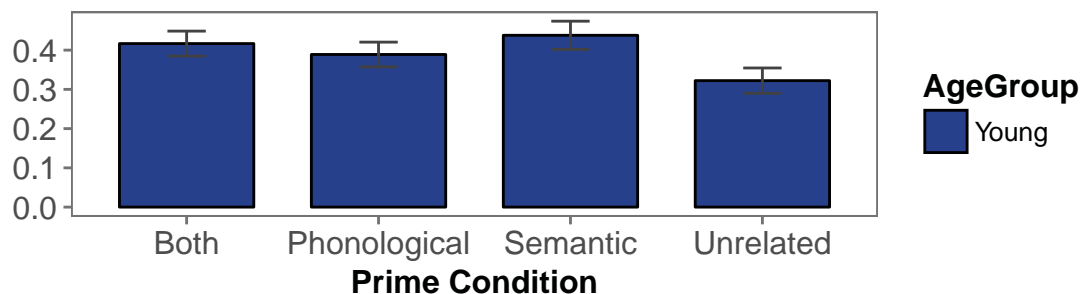
E1: Young and Old Adults (No Instructions)



E2: Young and Old Adults (With Instructions)



E3: Young Adults Only (48 ms)



11 Experiment 1: YA-OA No Instruction

```
> exp1_target = subset(final_j, final_j$StudyNo == '2' | final_j$StudyNo == '4')
> exp1_mcq = subset(final_mcq, final_mcq$StudyNo == '2' | final_mcq$StudyNo == '4')
> exp1_state = subset(final_statedata, final_statedata$StudyNo == '2' |
+                       final_statedata$StudyNo == '4')
> exp1_state_prime = subset(statedata_primetype_long, statedata_primetype_long$StudyNo ==
+                          statedata_primetype_long$StudyNo == '4')
> exp1_state_prime$PrimeCondition = as.factor(as.character(exp1_state_prime$PrimeCondition))
> exp1_state_prime$State = as.factor(as.character(exp1_state_prime$State))
```

```
> exp1_state_prime$Subject = as.factor(as.character(exp1_state_prime$Subject))
>
```

Exp 1: Target Accuracy

```
> ### TARGET RETRIEVAL ACCURACY
>
> exp1_target_aov = aov(data = exp1_target, Accuracy ~ AgeGroup*PrimeCondition +
+ Error (Subject/PrimeCondition))
> summary(exp1_target_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  0.000  0.00020    0.003  0.957
Residuals 70  4.877  0.06966

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition      3  0.7761  0.25871    21.765 2.59e-12 ***
AgeGroup:PrimeCondition  3  0.0117  0.00389     0.327    0.806
Residuals          210  2.4962  0.01189

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> exp1_target_lsm = lsmeans::lsmeans(exp1_target_aov, c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(exp1_target_lsm, alpha = 0.05,
+ adjust = "tukey", details = TRUE, by = c("AgeGroup"))
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
4	Phonological - Related	Old	0.1366667	0.0256977	210	5.318251	0.0000016
5	Phonological - Unrelated	Old	0.1366667	0.0256977	210	5.318251	0.0000016
6	Phonological - Both	Old	0.0833333	0.0256977	210	3.242836	0.0074688
10	Phonological - Related	Young	0.1255556	0.0256977	210	4.885873	0.0000121
11	Phonological - Unrelated	Young	0.1088889	0.0256977	210	4.237305	0.0001974
12	Phonological - Both	Young	0.0888889	0.0256977	210	3.459025	0.0036435

```
> ## specific t-tests
> target_p = exp1_target %>% filter(PrimeCondition == "Phonological")
> target_r = exp1_target %>% filter(PrimeCondition == "Related")
```

```
> target_b = expl_target %>% filter(PrimeCondition == "Both")
> target_u = expl_target %>% filter(PrimeCondition == "Unrelated")
> t.test(target_p$Accuracy, target_r$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_r$Accuracy
t = 7.0518, df = 71, p-value = 9.434e-10
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.09403883 0.16818339
sample estimates:
mean of the differences
      0.1311111
```

```
> t.test(target_p$Accuracy, target_b$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_b$Accuracy
t = 4.0628, df = 71, p-value = 0.0001235
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.04384895 0.12837327
sample estimates:
mean of the differences
      0.08611111
```

```
> t.test(target_p$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_u$Accuracy
t = 7.3454, df = 71, p-value = 2.72e-10
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.08944906 0.15610650
sample estimates:
mean of the differences
      0.1227778
```

```
> t.test(target_b$Accuracy, target_r$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_b$Accuracy and target_r$Accuracy
t = 2.8803, df = 71, p-value = 0.005249
```

```

alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01384772 0.07615228
sample estimates:
mean of the differences
      0.045

```

```
> t.test(target_b$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```

data: target_b$Accuracy and target_u$Accuracy
t = 2.233, df = 71, p-value = 0.02871
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.00392468 0.06940865
sample estimates:
mean of the differences
      0.03666667

```

```
> t.test(target_r$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```

data: target_r$Accuracy and target_u$Accuracy
t = -0.43074, df = 71, p-value = 0.668
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.04690961 0.03024294
sample estimates:
mean of the differences
     -0.008333333

```

```
>
>
```

Exp 1: Multiple Choice

```

> ## MULTIPLE CHOICE ACCURACY
> library(dplyr)
> exp1_mcq_acc = group_by(exp1_mcq, Subject, PrimeType, AgeGroup) %>%
+   summarise_at(vars(MCQAcc), mean)
> exp1_mcq_acc_aov = aov(data = exp1_mcq_acc, MCQAcc ~ AgeGroup*PrimeType +
+   Error(Subject/PrimeType))
> summary(exp1_mcq_acc_aov)

```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup 1 0.045 0.04500    0.62  0.434
Residuals 70 5.084 0.07262

Error: Subject:PrimeType
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeType 3 0.3792 0.12640    16.89 7.27e-10 ***
AgeGroup:PrimeType 3 0.0114 0.00381     0.51  0.676
Residuals 210 1.5718 0.00748

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ez::ezANOVA(data = exp1_mcq_acc, wid = .(Subject),
+             dv = .(MCQAcc), within = .(PrimeType),
+             between = .(AgeGroup))
```

```
$ANOVA
      Effect DFn DFd      F      p p<.05      ges
2      AgeGroup 1 70 0.6196261 4.338434e-01 0.006715950
3      PrimeType 3 210 16.8881223 7.271326e-10 * 0.053904473
4 AgeGroup:PrimeType 3 210 0.5096919 6.760176e-01 0.001716604

$`Mauchly's Test for Sphericity`
      Effect      W      p p<.05
3      PrimeType 0.8099912 0.01283039 *
4 AgeGroup:PrimeType 0.8099912 0.01283039 *

$`Sphericity Corrections`
      Effect      GGe      p[GG] p[GG]<.05      HFe      p[HF]
3      PrimeType 0.8785143 6.240475e-09 * 0.9160298 3.211211e-09
4 AgeGroup:PrimeType 0.8785143 6.522547e-01 0.9160298 6.599249e-01
p[HF]<.05
3      *
4
```

```
> exp1_mcqacc_lsm = lsmeans::lsmeans(exp1_mcq_acc_aov, c("AgeGroup", "PrimeType"))
> prime_effect = cld(exp1_mcqacc_lsm, alpha = 0.05,
+                   adjust = "tukey", details = TRUE, by = c("AgeGroup"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
2	u - r	Old	0.0588889	0.0203914	210	2.887931	0.0220951
4	p - r	Old	0.0777778	0.0203914	210	3.814248	0.0010248
5	p - b	Old	0.0622222	0.0203914	210	3.051398	0.0136071
8	u - r	Young	0.0855556	0.0203914	210	4.195673	0.0002336

10	p - r	Young	0.1088889	0.0203914	210	5.339947	0.0000014
11	p - b	Young	0.0644444	0.0203914	210	3.160377	0.0097114

```
> ## SPECIFIC T TESTS
>
> e1_mcq_p = exp1_mcq_acc %>% filter(PrimeType == "p")
> e1_mcq_r = exp1_mcq_acc %>% filter(PrimeType == "r")
> e1_mcq_b = exp1_mcq_acc %>% filter(PrimeType == "b")
> e1_mcq_u = exp1_mcq_acc %>% filter(PrimeType == "u")
> t.test(e1_mcq_p$MCQAcc, e1_mcq_r$MCQAcc, paired = TRUE)
```

Paired t-test

```
data: e1_mcq_p$MCQAcc and e1_mcq_r$MCQAcc
t = 5.8474, df = 71, p-value = 1.401e-07
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.06150678 0.12515989
sample estimates:
mean of the differences
      0.09333333
```

```
> t.test(e1_mcq_p$MCQAcc, e1_mcq_b$MCQAcc, paired = TRUE)
```

Paired t-test

```
data: e1_mcq_p$MCQAcc and e1_mcq_b$MCQAcc
t = 4.1686, df = 71, p-value = 8.53e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.03303921 0.09362746
sample estimates:
mean of the differences
      0.06333333
```

```
> t.test(e1_mcq_p$MCQAcc, e1_mcq_u$MCQAcc)
```

Welch Two Sample t-test

```
data: e1_mcq_p$MCQAcc and e1_mcq_u$MCQAcc
t = 0.85891, df = 136.93, p-value = 0.3919
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.02749250 0.06971473
sample estimates:
mean of x mean of y
0.7350000 0.7138889
```

```
> t.test(e1_mcq_r$MCQAcc, e1_mcq_u$MCQAcc, paired = TRUE)
```

Paired t-test

```
data: e1_mcq_r$MCQAcc and e1_mcq_u$MCQAcc
t = -4.8541, df = 71, p-value = 6.946e-06
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.10188914 -0.04255531
sample estimates:
mean of the differences
 -0.07222222
```

```
> ## MULTIPLE CHOICE ERRORS
>
> ## before we do ANOVA, we need to replace NAs with 0.
>
> for (i in 1: nrow(exp1_mcq)){
+   if(is.na(exp1_mcq[i,7])){
+     exp1_mcq[i,7] = 0
+   }
+ }
> exp1_mcq_aov = aov(data = exp1_mcq, Proportion ~ AgeGroup*PrimeType*ChosenPrime +
+   Error(Subject/(PrimeType*ChosenPrime)))
> summary(exp1_mcq_aov)
```

Error: Subject

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup	1	0.1648	0.16480	17.15	9.51e-05 ***
Residuals	70	0.6727	0.00961		

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Error: Subject:PrimeType

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeType	3	0.0099	0.003296	0.670	0.572
AgeGroup:PrimeType	3	0.0228	0.007595	1.543	0.204
Residuals	210	1.0338	0.004923		

Error: Subject:ChosenPrime

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
ChosenPrime	3	50.51	16.836	543.313	<2e-16 ***
AgeGroup:ChosenPrime	3	0.31	0.104	3.359	0.0198 *
Residuals	210	6.51	0.031		

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```


Error: Subject:PrimeType:ChosenPrime

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeType:ChosenPrime	9	3.150	0.3500	8.704	2.23e-12 ***
AgeGroup:PrimeType:ChosenPrime	9	0.461	0.0512	1.273	0.248
Residuals	630	25.331	0.0402		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> library(ez)
> ezANOVA(data = exp1_mcq, wid = .(Subject),
+         dv = .(Proportion), within = .(PrimeType, ChosenPrime),
+         between = .(AgeGroup))
```

\$ANOVA

	Effect	DFn	DFd	F	p	p<.05
2	AgeGroup	1	70	17.1491768	9.509592e-05	*
3	PrimeType	3	210	0.6695768	5.715913e-01	
5	ChosenPrime	3	210	543.3132776	1.167318e-98	*
4	AgeGroup:PrimeType	3	210	1.5427766	2.044944e-01	
6	AgeGroup:ChosenPrime	3	210	3.3585095	1.975303e-02	*
7	PrimeType:ChosenPrime	9	630	8.7041874	2.229792e-12	*
8	AgeGroup:PrimeType:ChosenPrime	9	630	1.2728800	2.482746e-01	

ges

2	0.0048887153
3	0.0002946951
5	0.6009020113
4	0.0006787483
6	0.0092214030
7	0.0858385191
8	0.0135455171

\$`Mauchly's Test for Sphericity`

	Effect	W	p	p<.05
3	PrimeType	0.7844411130	5.144689e-03	*
4	AgeGroup:PrimeType	0.7844411130	5.144689e-03	*
5	ChosenPrime	0.1272470802	8.462192e-29	*
6	AgeGroup:ChosenPrime	0.1272470802	8.462192e-29	*
7	PrimeType:ChosenPrime	0.0001935578	2.205411e-92	*
8	AgeGroup:PrimeType:ChosenPrime	0.0001935578	2.205411e-92	*

\$`Sphericity Corrections`

	Effect	GGe	p[GG]	p[GG]<.05	HFe
3	PrimeType	0.8717229	5.518176e-01		0.9085971
4	AgeGroup:PrimeType	0.8717229	2.098834e-01		0.9085971
5	ChosenPrime	0.5075355	1.927235e-51	*	0.5164968
6	AgeGroup:ChosenPrime	0.5075355	5.120574e-02		0.5164968
7	PrimeType:ChosenPrime	0.4153369	2.487140e-06	*	0.4416817

```

8 AgeGroup:PrimeType:ChosenPrime 0.4153369 2.820893e-01 0.4416817
  p[HF] p[HF] < .05
3 5.577695e-01
4 2.083657e-01
5 2.660418e-52 *
6 5.032787e-02
7 1.318123e-06 *
8 2.809457e-01

```

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> exp1_errors_lsm = lsmeans::lsmeans(exp1_mcq_aov, c("AgeGroup", "PrimeType", "ChosenPrimeType"))
> prime_effect = cld(exp1_errors_lsm, alpha = 0.05,
+   adjust = "tukey", details = TRUE, by = c("AgeGroup", "PrimeType"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))

```

	contrast	AgeGroup	PrimeType	estimate	SE	df	t.ratio	p.value
2	r - u	Old	b	0.3946738	0.0458883	830.7801	8.600747	0.000000
3	r - p	Old	b	0.3744043	0.0458883	830.7801	8.159034	0.000000
4	b - u	Old	b	0.4507613	0.0458883	830.7801	9.823009	0.000000
5	b - p	Old	b	0.4304918	0.0458883	830.7801	9.381296	0.000000
8	r - u	Old	p	0.3505613	0.0458883	830.7801	7.639446	0.000000
9	r - p	Old	p	0.2691239	0.0458883	830.7801	5.864759	0.000000
10	b - u	Old	p	0.4614647	0.0458883	830.7801	10.056257	0.000000
11	b - p	Old	p	0.3800273	0.0458883	830.7801	8.281570	0.000000
14	r - u	Old	r	0.2237887	0.0458883	830.7801	4.876812	0.000007
15	r - p	Old	r	0.2191375	0.0458883	830.7801	4.775453	0.000012
16	b - u	Old	r	0.6648968	0.0458883	830.7801	14.489458	0.000000
17	b - p	Old	r	0.6602456	0.0458883	830.7801	14.388098	0.000000
18	b - r	Old	r	0.4411081	0.0458883	830.7801	9.612646	0.000000
20	r - u	Old	u	0.2966159	0.0458883	830.7801	6.463865	0.000000
21	r - p	Old	u	0.2461617	0.0458883	830.7801	5.364367	0.000000
22	b - u	Old	u	0.4209759	0.0458883	830.7801	9.173923	0.000000
23	b - p	Old	u	0.3705217	0.0458883	830.7801	8.074424	0.000000
24	b - r	Old	u	0.1243600	0.0458883	830.7801	2.710058	0.034592
26	r - u	Young	b	0.4198547	0.0458883	830.7801	9.149489	0.000000
27	r - p	Young	b	0.3813881	0.0458883	830.7801	8.311224	0.000000
28	b - u	Young	b	0.5026215	0.0458883	830.7801	10.953148	0.000000
29	b - p	Young	b	0.4641549	0.0458883	830.7801	10.114882	0.000000
32	r - u	Young	p	0.3784371	0.0458883	830.7801	8.246916	0.000000
33	r - p	Young	p	0.3354865	0.0458883	830.7801	7.310935	0.000000
34	b - u	Young	p	0.4690886	0.0458883	830.7801	10.222396	0.000000
35	b - p	Young	p	0.4261380	0.0458883	830.7801	9.286416	0.000000
38	r - u	Young	r	0.3487971	0.0458883	830.7801	7.601000	0.000000
39	r - p	Young	r	0.3464828	0.0458883	830.7801	7.550567	0.000000
40	b - u	Young	r	0.5527607	0.0458883	830.7801	12.045784	0.000000
41	b - p	Young	r	0.5504465	0.0458883	830.7801	11.995351	0.000000

42	b - r	Young	r		0.2039636	0.0458883	830.7801	4.444784	0.000059
44	r - u	Young	u		0.3687962	0.0458883	830.7801	8.036821	0.000000
45	r - p	Young	u		0.3671763	0.0458883	830.7801	8.001522	0.000000
46	b - u	Young	u		0.4966483	0.0458883	830.7801	10.822980	0.000000
47	b - p	Young	u		0.4950285	0.0458883	830.7801	10.787681	0.000000
48	b - r	Young	u		0.1278522	0.0458883	830.7801	2.786159	0.027895

```
> ## SPECIFIC OLD COMPARISION T TEST
>
> e1mcq_old_r = exp1_mcq %>% filter(AgeGroup == "Old" & PrimeType == "r")
> e1mcq_old_r_r = e1mcq_old_r %>% filter(ChosenPrime == "r")
> e1mcq_old_r_p = e1mcq_old_r %>% filter(ChosenPrime == "p")
> e1mcq_old_r_b = e1mcq_old_r %>% filter(ChosenPrime == "b")
> e1mcq_old_r_u = e1mcq_old_r %>% filter(ChosenPrime == "u")
> t.test(e1mcq_old_r_r$Proportion, e1mcq_old_r_p$Proportion, paired = TRUE)
```

Paired t-test

```
data: e1mcq_old_r_r$Proportion and e1mcq_old_r_p$Proportion
t = 18.837, df = 35, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.5890888 0.7314023
sample estimates:
mean of the differences
 0.6602456
```

```
> t.test(e1mcq_old_r_r$Proportion, e1mcq_old_r_b$Proportion, paired = TRUE)
```

Paired t-test

```
data: e1mcq_old_r_r$Proportion and e1mcq_old_r_b$Proportion
t = 7.7643, df = 35, p-value = 4.055e-09
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.3257728 0.5564434
sample estimates:
mean of the differences
 0.4411081
```

```
> t.test(e1mcq_old_r_r$Proportion, e1mcq_old_r_u$Proportion, paired = TRUE)
```

Paired t-test

```
data: e1mcq_old_r_r$Proportion and e1mcq_old_r_u$Proportion
t = 19.02, df = 35, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
```

```

95 percent confidence interval:
 0.5939301 0.7358636
sample estimates:
mean of the differences
 0.6648968

```

```

> e1mcq_young_r = exp1_mcq %>% filter(AgeGroup == "Young" & PrimeType == "r")
> e1mcq_young_r_r = e1mcq_young_r %>% filter(ChosenPrime == "r")
> ## comparing young and old
> t.test(e1mcq_young_r_r$Proportion, e1mcq_old_r_r$Proportion)

```

Welch Two Sample t-test

```

data: e1mcq_young_r_r$Proportion and e1mcq_old_r_r$Proportion
t = -1.6868, df = 68.083, p-value = 0.09621
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.20057038 0.01680856
sample estimates:
mean of x mean of y
0.5744779 0.6663588

```

```
>
```

Exp 1: State Data

```

> ## just state
> exp1_state_aov = aov(data = exp1_state, Trials ~ AgeGroup*State +
+                               Error(Subject/State))
> summary(exp1_state_aov)

```

```

Error: Subject
      Df    Sum Sq   Mean Sq F value Pr(>F)
AgeGroup  1 4.450e-27 4.454e-27   6.776 0.0113 *
Residuals 70 4.602e-26 6.570e-28
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:State
      Df Sum Sq Mean Sq F value    Pr(>F)
State    3  50878   16959   62.62 < 2e-16 ***
AgeGroup:State  3   8697    2899   10.70 1.41e-06 ***
Residuals    210  56874     271
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> exp1_state_lsm = lsmeans::lsmeans(exp1_state_aov, c("AgeGroup", "State"))
> prime_effect = cld(exp1_state_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = c("State"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```

	contrast	State	estimate	SE	df	t.ratio	p.value
1	Old - Young	dontknow	13.33333	3.359241	210	3.969150	9.9e-05
3	Old - Young	other	16.52778	3.359241	210	4.920092	1.7e-06

```
> ##state by prime
> exp1_stateprime_aov = aov(data = exp1_state_prime, Trials ~ AgeGroup*PrimeCondition*St
+                               Error(Subject/(PrimeCondition*State)))
> summary(exp1_stateprime_aov)
```

```
Error: Subject
          Df      Sum Sq   Mean Sq F value    Pr(>F)
AgeGroup   1 1.543e-26 1.543e-26   16.12 0.000148 ***
Residuals 70 6.701e-26 9.570e-28
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition
          Df      Sum Sq   Mean Sq F value    Pr(>F)
PrimeCondition   3 3.560e-27 1.186e-27   2.140 0.0963 .
AgeGroup:PrimeCondition   3 3.520e-27 1.174e-27   2.117 0.0991 .
Residuals          210 1.164e-25 5.544e-28
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:State
          Df Sum Sq Mean Sq F value    Pr(>F)
State           3  12719    4240   62.62 < 2e-16 ***
AgeGroup:State   3   2174     725   10.70 1.41e-06 ***
Residuals       210  14218      68
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition:State
          Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition:State   9    613    68.15   9.316 2.38e-13 ***
AgeGroup:PrimeCondition:State   9     36     4.04   0.552 0.836
Residuals              630   4608     7.31
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> library(ez)
> ezANOVA(data = exp1_state_prime, wid = .(Subject),
+         dv = .(Trials), within =.(PrimeCondition, State),
+         between = .(AgeGroup))
```

```
$ANOVA
```

	Effect	DFn	DFd		F	p	p<.05
2	AgeGroup	1	70	-4.436747e-13	1.000000e+00		
3	PrimeCondition	3	210	-3.183567e-14	1.000000e+00		
5	State	3	210	6.261979e+01	5.825585e-29	*	
4	AgeGroup:PrimeCondition	3	210	-2.784393e-14	1.000000e+00		
6	AgeGroup:State	3	210	1.070363e+01	1.413406e-06	*	
7	PrimeCondition:State	9	630	9.316406e+00	2.383097e-13	*	
8	AgeGroup:PrimeCondition:State	9	630	5.524295e-01	8.360652e-01		

```
ges
```

2	1.599722e-32
3	1.112472e-32
5	4.031993e-01
4	9.729835e-33
6	1.035256e-01
7	3.154954e-02
8	1.927995e-03

```
$`Mauchly's Test for Sphericity`
```

	Effect		W	p	p<.05
5	State	0.35869842	8.382815e-14	*	
6	AgeGroup:State	0.35869842	8.382815e-14	*	
7	PrimeCondition:State	0.06370456	5.965946e-19	*	
8	AgeGroup:PrimeCondition:State	0.06370456	5.965946e-19	*	

```
$`Sphericity Corrections`
```

	Effect	GGe	p[GG]	p[GG]<.05	HFe
3	PrimeCondition	0.3333333	1.000000e+00		0.3333333
4	AgeGroup:PrimeCondition	0.3333333	1.000000e+00		0.3333333
5	State	0.5939181	3.167775e-18	*	0.6083639
6	AgeGroup:State	0.5939181	1.022365e-04	*	0.6083639
7	PrimeCondition:State	0.6229953	4.149664e-09	*	0.6834659
8	AgeGroup:PrimeCondition:State	0.6229953	7.563884e-01		0.6834659

```
p[HF] p[HF]<.05
```

3	1.000000e+00	
4	1.000000e+00	
5	1.312388e-18	*
6	8.768867e-05	*
7	8.609985e-10	*
8	7.723887e-01	

```
> options(contrasts = c('contr.sum', 'contr.poly'))
```

```

> exp1_state_lsm = lsmeans::lsmeans(exp1_stateprime_aov, c("AgeGroup", "PrimeCondition",
> prime_effect = cld(exp1_state_lsm, alpha = 0.05,
+ adjust = "tukey", details = TRUE, by = c("PrimeCondition", "AgeGroup"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))

```

	contrast	PrimeCondition	AgeGroup	estimate	SE	df	
t.ratio	p.value						
2	dontknow - other	b	Old	8.833333	1.115866	355.7286	
7.916123	0.0000000						
3	dontknow - TOT	b	Old	7.722222	1.115866	355.7286	
6.920384	0.0000000						
4	know - other	b	Old	10.833333	1.115866	355.7286	
9.708452	0.0000000						
5	know - TOT	b	Old	9.722222	1.115866	355.7286	
8.712714	0.0000000						
8	dontknow - other	b	Young	3.916667	1.115866	355.7286	
3.509979	0.0028337						
10	know - other	b	Young	7.527778	1.115866	355.7286	
6.746130	0.0000000						
11	know - TOT	b	Young	5.194444	1.115866	355.7286	
4.655078	0.0000271						
12	know - dontknow	b	Young	3.611111	1.115866	355.7286	
3.236151	0.0072208						
14	dontknow - other	p	Old	8.277778	1.115866	355.7286	
7.418253	0.0000000						
15	dontknow - TOT	p	Old	7.861111	1.115866	355.7286	
7.044851	0.0000000						
16	know - other	p	Old	11.194444	1.115866	355.7286	10.0320
17	know - TOT	p	Old	10.777778	1.115866	355.7286	
9.658665	0.0000000						
18	know - dontknow	p	Old	2.916667	1.115866	355.7286	
2.613814	0.0458882						
20	dontknow - other	p	Young	3.138889	1.115866	355.7286	
2.812962	0.0265119						
22	know - other	p	Young	8.250000	1.115866	355.7286	
7.393360	0.0000000						
23	know - TOT	p	Young	6.194444	1.115866	355.7286	
5.551243	0.0000003						
24	know - dontknow	p	Young	5.111111	1.115866	355.7286	
4.580398	0.0000380						
26	dontknow - other	r	Old	7.555556	1.115866	355.7286	
6.771023	0.0000000						
27	dontknow - TOT	r	Old	6.805556	1.115866	355.7286	
6.098900	0.0000000						
28	know - other	r	Old	9.694444	1.115866	355.7286	
8.687820	0.0000000						
29	know - TOT	r	Old	8.944444	1.115866	355.7286	

```

8.015697| 0.0000000|
|31 |TOT - other      |r              |Young      | 3.166667| 1.115866| 355.7286|
2.837855| 0.0246829|
|32 |dontknow - other |r              |Young      | 3.444444| 1.115866| 355.7286|
3.086790| 0.0116638|
|34 |know - other      |r              |Young      | 6.944444| 1.115866| 355.7286|
6.223367| 0.0000000|
|35 |know - TOT        |r              |Young      | 3.777778| 1.115866| 355.7286|
3.385512| 0.0043739|
|36 |know - dontknow   |r              |Young      | 3.500000| 1.115866| 355.7286|
3.136577| 0.0099652|
|38 |dontknow - other |u              |Old        | 7.777778| 1.115866| 355.7286|
6.970171| 0.0000000|
|39 |dontknow - TOT    |u              |Old        | 5.861111| 1.115866| 355.7286|
5.252522| 0.0000015|
|40 |know - other      |u              |Old        | 10.194444| 1.115866| 355.7286|
9.135903| 0.0000000|
|41 |know - TOT        |u              |Old        | 8.277778| 1.115866| 355.7286|
7.418253| 0.0000000|
|44 |dontknow - other |u              |Young      | 4.750000| 1.115866| 355.7286|
4.256783| 0.0001555|
|46 |know - other      |u              |Young      | 5.222222| 1.115866| 355.7286|
4.679972| 0.0000242|

```

```

> ### INDIVIDUAL T-TESTS FOR AGExSTATE interaction
>
> e1_young_dk = exp1_state %>% filter(AgeGroup == "Young" & State == "dontknow")
> e1_old_dk = exp1_state %>% filter(AgeGroup == "Old" & State == "dontknow")
> t.test(e1_old_dk$Trials, e1_young_dk$Trials, var.equal = TRUE)

```

Two Sample t-test

```

data: e1_old_dk$Trials and e1_young_dk$Trials
t = 3.0688, df = 70, p-value = 0.003057
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 4.667831 21.998836
sample estimates:
mean of x mean of y
 40.22222  26.88889

```

```

> e1_young_other = exp1_state %>% filter(AgeGroup == "Young" & State == "other")
> e1_old_other = exp1_state %>% filter(AgeGroup == "Old" & State == "other")
> t.test(e1_young_other$Trials, e1_old_other$Trials)

```

Welch Two Sample t-test

```

data: e1_young_other$Trials and e1_old_other$Trials

```



```
t = 7.2009, df = 57.112, p-value = 1.457e-09
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 11.93183 21.12373
sample estimates:
mean of x mean of y
21.88889  5.36111
```

12 Experiment 2: YA-OA Not The Prime

```
> exp2_target = subset(final_j, final_j$StudyNo == '5' | final_j$StudyNo == '6')
> exp2_mcq = subset(final_mcq, final_mcq$StudyNo == '5' | final_mcq$StudyNo == '6')
> exp2_state = subset(final_statedata, final_statedata$StudyNo == '5' |
+                     final_statedata$StudyNo == '6')
> exp2_state_prime = subset(statedata_primetype_long, statedata_primetype_long$StudyNo == '5' |
+                          statedata_primetype_long$StudyNo == '6')
> exp2_state_prime$PrimeCondition = as.factor(as.character(exp2_state_prime$PrimeCondition))
> exp2_state_prime$State = as.factor(as.character(exp2_state_prime$State))
> exp2_state_prime$Subject = as.factor(as.character(exp2_state_prime$Subject))
>
```

Exp 2: Target Accuracy

```
> ### TARGET RETRIEVAL ACCURACY
>
> exp2_target_aov = aov(data = exp2_target, Accuracy ~ AgeGroup*PrimeCondition +
+                      Error (Subject/PrimeCondition))
> summary(exp2_target_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  0.146  0.14631    2.324  0.132
Residuals 62  3.903  0.06295

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition      3  0.3279  0.10929    8.844 1.65e-05 ***
AgeGroup:PrimeCondition  3  0.0580  0.01932    1.564    0.2
Residuals      186  2.2986  0.01236

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> library(ez)
> ezANOVA(data = exp2_target, wid = .(Subject),
+         dv = .(Accuracy), within = .(PrimeCondition),
```

```
+ between = .(AgeGroup))
```

```
$ANOVA
```

	Effect	DFn	DFd	F	p	p<.05	ges
2	AgeGroup	1	62	2.324293	0.1324531919		0.02304922
3	PrimeCondition	3	186	8.843728	0.0000164582	*	0.05021639
4	AgeGroup:PrimeCondition	3	186	1.563613	0.1997101194		0.00926134

```
$`Mauchly's Test for Sphericity`
```

	Effect	W	p	p<.05
3	PrimeCondition	0.7672992	0.006615755	*
4	AgeGroup:PrimeCondition	0.7672992	0.006615755	*

```
$`Sphericity Corrections`
```

	Effect	GGe	p[GG]	p[GG]<.05	HFe
3	PrimeCondition	0.8772502	4.507799e-05	*	0.9196852
4	AgeGroup:PrimeCondition	0.8772502	2.051504e-01		0.9196852

p[HF] p[HF]<.05

3	0.0000318037	*
4	0.2032956187	

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmmeans)
> library(multcomp)
> exp2_target_lsm = lsmmeans::lsmmeans(exp2_target_aov, c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(exp2_target_lsm, alpha = 0.05,
+ adjust = "tukey", details = TRUE, by = c("AgeGroup"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 1 ))
```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
1	Unrelated - Related	Old	0.00750	0.0277915	186	0.2698669	0.9931058
2	Both - Related	Old	0.02875	0.0277915	186	1.0344897	0.7294303
3	Both - Unrelated	Old	0.02125	0.0277915	186	0.7646228	0.8702573
4	Phonological - Related	Old	0.06375	0.0277915	186	2.2938684	0.1031449
5	Phonological - Unrelated	Old	0.05625	0.0277915	186	2.0240015	0.1828861
6	Phonological - Both	Old	0.03500	0.0277915	186	1.2593787	0.5899313
7	Unrelated - Related	Young	0.05000	0.0277915	186	1.7991125	0.2770324
8	Both - Related	Young	0.05000	0.0277915	186	1.7991125	0.2770324
10	Phonological - Related	Young	0.13625	0.0277915	186	4.9025814	0.0000122
11	Phonological - Unrelated	Young	0.08625	0.0277915	186	3.1034690	0.0117671
12	Phonological - Both	Young	0.08625	0.0277915	186	3.1034690	0.0117671

```
> ## specific t-tests
> target_p = exp2_target %>% filter(PrimeCondition == "Phonological")
> target_r = exp2_target %>% filter(PrimeCondition == "Related")
> target_b = exp2_target %>% filter(PrimeCondition == "Both")
```

```
> target_u = exp2_target %>% filter(PrimeCondition == "Unrelated")
> t.test(target_p$Accuracy, target_r$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_r$Accuracy
t = 3.8169, df = 63, p-value = 0.0003109
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.03573319 0.11426681
sample estimates:
mean of the differences
      0.075
```

```
> t.test(target_p$Accuracy, target_b$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_b$Accuracy
t = 2.6942, df = 63, p-value = 0.009033
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01565821 0.10559179
sample estimates:
mean of the differences
      0.060625
```

```
> t.test(target_p$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_u$Accuracy
t = 4.6462, df = 63, p-value = 1.772e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.0548529 0.1376471
sample estimates:
mean of the differences
      0.09625
```

```
> t.test(target_b$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_b$Accuracy and target_u$Accuracy
t = 2.2178, df = 63, p-value = 0.03018
alternative hypothesis: true difference in means is not equal to 0
```

```
95 percent confidence interval:
 0.003525062 0.067724938
sample estimates:
mean of the differences
      0.035625
```

```
> t.test(target_b$Accuracy, target_r$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_b$Accuracy and target_r$Accuracy
t = 0.8146, df = 63, p-value = 0.4184
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.02088892 0.04963892
sample estimates:
mean of the differences
      0.014375
```

```
> t.test(target_r$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_r$Accuracy and target_u$Accuracy
t = 1.0053, df = 63, p-value = 0.3186
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.02099024 0.06349024
sample estimates:
mean of the differences
      0.02125
```

```
>
>
```

Exp 2: Multiple Choice

```
> ## MULTIPLE CHOICE ACCURACY
> library(dplyr)
> exp2_mcq_acc = group_by(exp2_mcq, Subject, PrimeType, AgeGroup) %>%
+   summarise_at(vars(MCQAcc), mean)
> exp2_mcq_acc_aov = aov(data = exp2_mcq_acc, MCQAcc ~ AgeGroup*PrimeType +
+   Error(Subject/PrimeType))
> summary(exp2_mcq_acc_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  0.339  0.3393    4.89 0.0307 *
Residuals 62  4.302  0.0694
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeType
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType  3 0.1183  0.03944    5.414 0.00136 **
AgeGroup:PrimeType  3 0.0726  0.02421    3.323 0.02095 *
Residuals 186 1.3551  0.00729
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## SPECIFIC T TESTS
>
> e2_mcq_p = exp2_mcq_acc %>% filter(PrimeType == "p")
> e2_mcq_r = exp2_mcq_acc %>% filter(PrimeType == "r")
> e2_mcq_b = exp2_mcq_acc %>% filter(PrimeType == "b")
> e2_mcq_u = exp2_mcq_acc %>% filter(PrimeType == "u")
> e2mcq_y_p = e2_mcq_p %>% filter(AgeGroup == "Young")
> e2mcq_o_p = e2_mcq_p %>% filter(AgeGroup == "Old")
> t.test(e2mcq_y_p$MCQAcc, e2mcq_o_p$MCQAcc)
```

Welch Two Sample t-test

```
data: e2mcq_y_p$MCQAcc and e2mcq_o_p$MCQAcc
t = 2.7587, df = 57.666, p-value = 0.007763
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.02880191 0.18119809
sample estimates:
mean of x mean of y
 0.77875  0.67375
```

```
> e2mcq_y_b = e2_mcq_b %>% filter(AgeGroup == "Young")
> e2mcq_o_b = e2_mcq_b %>% filter(AgeGroup == "Old")
> t.test(e2mcq_y_b$MCQAcc, e2mcq_o_b$MCQAcc)
```

Welch Two Sample t-test

```
data: e2mcq_y_b$MCQAcc and e2mcq_o_b$MCQAcc
t = 2.6633, df = 52.43, p-value = 0.01025
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.02312916 0.16437084
```

```
sample estimates:
mean of x mean of y
  0.78000   0.68625
```

```
> e2mcq_y_r = e2_mcq_r %>% filter(AgeGroup == "Young")
> e2mcq_o_r = e2_mcq_r %>% filter(AgeGroup == "Old")
> t.test(e2mcq_y_r$MCQAcc, e2mcq_o_r$MCQAcc)
```

Welch Two Sample t-test

```
data: e2mcq_y_r$MCQAcc and e2mcq_o_r$MCQAcc
t = 1.9968, df = 59.366, p-value = 0.05044
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.0001485681  0.1501485681
sample estimates:
mean of x mean of y
  0.74625   0.67125
```

```
> ezANOVA(data = exp2_mcq_acc, wid = .(Subject),
+         dv = .(MCQAcc), within = .(PrimeType),
+         between = .(AgeGroup))
```

\$ANOVA

	Effect	DFn	DFd	F	p	p<.05	ges
2	AgeGroup	1	62	4.889835	0.030711429	*	0.05658352
3	PrimeType	3	186	5.413597	0.001361785	*	0.02048608
4	AgeGroup:PrimeType	3	186	3.322623	0.020949949	*	0.01267372

\$`Mauchly's Test for Sphericity`

	Effect	W	p	p<.05
3	PrimeType	0.9349764	0.5376336	
4	AgeGroup:PrimeType	0.9349764	0.5376336	

\$`Sphericity Corrections`

	Effect	GGe	p[GG]	p[GG]<.05	HFe	p[HF]
3	PrimeType	0.9562612	0.001636068	*	1.007552	0.001361785
4	AgeGroup:PrimeType	0.9562612	0.022740176	*	1.007552	0.020949949

p[HF]<.05

3	*
4	*

```
> exp2_mcqacc_lsm = lsmeans::lsmeans(exp2_mcq_acc_aov, c("AgeGroup", "PrimeType"))
> prime_effect = cld(exp2_mcqacc_lsm, alpha = 0.05,
+                   adjust = "tukey", details = TRUE, by = c("PrimeType"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.1))
```

contrast	PrimeType	estimate	SE	df	t.ratio	p.value
Young - Old	b	0.09375	0.0377587	103.7754	2.482873	0.0146353
Young - Old	p	0.10500	0.0377587	103.7754	2.780817	0.0064404
Young - Old	r	0.07500	0.0377587	103.7754	1.986298	0.0496371

```
> ## MULTIPLE CHOICE ERRORS
>
> ## before we do ANOVA, we need to replace NAs with 0.
>
> for (i in 1: nrow(exp2_mcq)){
+   if(is.na(exp2_mcq[i,7])){
+     exp2_mcq[i,7] = 0
+   }
+ }
> exp2_mcq_aov = aov(data = exp2_mcq, Proportion ~ AgeGroup*PrimeType*ChosenPrime +
+   Error(Subject/(PrimeType*ChosenPrime)))
> summary(exp2_mcq_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup  1  0.1068  0.10684    27.79 1.82e-06 ***
Residuals 62  0.2384  0.00384
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeType
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeType  3  0.0107  0.003571    1.134  0.337
AgeGroup:PrimeType  3  0.0100  0.003322    1.055  0.369
Residuals 186  0.5856  0.003148

Error: Subject:ChosenPrime
      Df Sum Sq Mean Sq F value    Pr(>F)
ChosenPrime  3  48.69  16.231 549.806 <2e-16 ***
AgeGroup:ChosenPrime  3   0.20   0.065   2.204 0.0891 .
Residuals 186   5.49   0.030

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeType:ChosenPrime
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeType:ChosenPrime  9   0.317  0.03524    0.752  0.661
AgeGroup:PrimeType:ChosenPrime  9   2.514  0.27931    5.961 5.51e-08 ***
Residuals 558 26.145  0.04686
---
```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> library(ez)
> ezANOVA(data = exp2_mcg, wid = .(Subject),
+         dv = .(Proportion), within = .(PrimeType, ChosenPrime),
+         between = .(AgeGroup))
```

```
$ANOVA
```

	Effect	DFn	DFd	F	p	p<.05
2	AgeGroup	1	62	27.7864310	1.816581e-06	*
3	PrimeType	3	186	1.1342790	3.365116e-01	
5	ChosenPrime	3	186	549.8057211	3.571643e-92	*
4	AgeGroup:PrimeType	3	186	1.0551058	3.694977e-01	
6	AgeGroup:ChosenPrime	3	186	2.2043095	8.905461e-02	
7	PrimeType:ChosenPrime	9	558	0.7520851	6.610380e-01	
8	AgeGroup:PrimeType:ChosenPrime	9	558	5.9611226	5.513598e-08	*

```
ges
2 0.0032805450
3 0.0003299406
5 0.6000111427
4 0.0003069176
6 0.0059782028
7 0.0096759793
8 0.0718761191

$`Mauchly's Test for Sphericity`
```

	Effect	W	p	p<.05
3	PrimeType	8.268634e-01	4.162816e-02	*
4	AgeGroup:PrimeType	8.268634e-01	4.162816e-02	*
5	ChosenPrime	1.970579e-01	1.077213e-19	*
6	AgeGroup:ChosenPrime	1.970579e-01	1.077213e-19	*
7	PrimeType:ChosenPrime	4.647268e-05	2.071610e-95	*
8	AgeGroup:PrimeType:ChosenPrime	4.647268e-05	2.071610e-95	*

```
$`Sphericity Corrections`
```

	Effect	GGe	p[GG]	p[GG]<.05	HFe
3	PrimeType	0.8973724	3.340476e-01		0.9419961
4	AgeGroup:PrimeType	0.8973724	3.651262e-01		0.9419961
5	ChosenPrime	0.5929531	8.989760e-56	*	0.6092441
6	AgeGroup:ChosenPrime	0.5929531	1.209834e-01		0.6092441
7	PrimeType:ChosenPrime	0.3599941	5.317429e-01		0.3821883
8	AgeGroup:PrimeType:ChosenPrime	0.3599941	4.542166e-04	*	0.3821883

```
p[HF] p[HF]<.05
3 3.352077e-01
4 3.671298e-01
5 3.132778e-57 *
6 1.195651e-01
7 5.390105e-01
```


8 3.299722e-04

*

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> exp2_errors_lsm = lsmeans::lsmeans(exp2_mcq_aov, c("AgeGroup", "PrimeType", "ChosenPrime"))
> prime_effect = cld(exp2_errors_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = c("PrimeType", "ChosenPrime"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```

	contrast	PrimeType	ChosenPrime	estimate	SE	df	t.ratio	p.value
1	Old - Young	b	b	0.1390686	0.0452231	758.0838	3.075170	0.00231
3	Old - Young	b	r	0.1790249	0.0452231	758.0838	3.958708	0.00008
9	Old - Young	r	b	0.2871214	0.0452231	758.0838	6.349003	0.00000
11	Old - Young	r	r	0.1757002	0.0452231	758.0838	3.885190	0.00010

```
> ## SPECIFIC OLD COMPARISION T TEST
>
> e2mcq_old_r = exp2_mcq %>% filter(AgeGroup == "Old" & PrimeType == "r")
> e2mcq_young_r = exp2_mcq %>% filter(AgeGroup == "Young" & PrimeType == "r")
> e2mcq_old_r_r = e2mcq_old_r %>% filter(ChosenPrime == "r")
> e2mcq_young_r_r = e2mcq_young_r %>% filter(ChosenPrime == "r")
> ## comparing young and old
> t.test(e2mcq_young_r_r$Proportion, e2mcq_old_r_r$Proportion)
```

Welch Two Sample t-test

```
data: e2mcq_young_r_r$Proportion and e2mcq_old_r_r$Proportion
t = -2.7008, df = 51.599, p-value = 0.009336
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.30626887 -0.04513155
sample estimates:
mean of x mean of y
0.4227143 0.5984145
```

```
> e2mcq_old_b = exp2_mcq %>% filter(AgeGroup == "Old" & PrimeType == "b")
> e2mcq_young_b = exp2_mcq %>% filter(AgeGroup == "Young" & PrimeType == "b")
> e2mcq_old_b_b = e2mcq_old_b %>% filter(ChosenPrime == "b")
> e2mcq_young_b_b = e2mcq_young_b %>% filter(ChosenPrime == "b")
> ## comparing young and old
> t.test(e2mcq_young_b_b$Proportion, e2mcq_old_b_b$Proportion)
```

Welch Two Sample t-test

```
data: e2mcq_young_b_b$Proportion and e2mcq_old_b_b$Proportion
t = -2.3168, df = 61.942, p-value = 0.02384
```

```

alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.25906269 -0.01907454
sample estimates:
mean of x mean of y
0.2898719 0.4289405

```

Exp 2: State Data

```

> ## just state
> exp2_state_aov = aov(data = exp2_state, Trials ~ AgeGroup*State +
+                               Error(Subject/State))
> summary(exp2_state_aov)

```

```

Error: Subject
      Df      Sum Sq   Mean Sq F value    Pr(>F)
AgeGroup  1 3.832e-27 3.832e-27   12.44 0.000797 ***
Residuals 62 1.910e-26 3.080e-28
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:State
      Df Sum Sq Mean Sq F value    Pr(>F)
State      3  52217   17406   69.44 < 2e-16 ***
AgeGroup:State  3  10400    3467   13.83 3.52e-08 ***
Residuals    186  46621     251
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> exp2_state_lsm = lsmeans::lsmeans(exp2_state_aov, c("AgeGroup", "State"))
> prime_effect = cld(exp2_state_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = c("State"))
> knitr::kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))

```

	contrast	State	estimate	SE	df	t.ratio	p.value
1	Old - Young	dontknow	18.34375	3.427726	186	5.351580	0.0000003
3	Old - Young	other	13.96875	3.427726	186	4.075223	0.0000680
4	Old - Young	TOT	9.56250	3.427726	186	2.789750	0.0058243

```

> ## Specific t test for Old-Young TOT Differece
>
> y_TOT = exp2_state %>% filter(AgeGroup == "Young" & State == "TOT")
> o_TOT = exp2_state %>% filter(AgeGroup == "Old" & State == "TOT")
> t.test(y_TOT$Trials, o_TOT$Trials)

```

```
Welch Two Sample t-test

data: y_TOT$Trials and o_TOT$Trials
t = 4.4834, df = 59.013, p-value = 3.443e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 5.294682 13.830318
sample estimates:
mean of x mean of y
15.46875   5.90625
```

```
> ##state by prime
> exp2_stateprime_aov = aov(data = exp2_state_prime, Trials ~ AgeGroup*PrimeCondition*St
+                               Error(Subject/(PrimeCondition*State)))
> summary(exp2_stateprime_aov)
```

```
Error: Subject
      Df      Sum Sq    Mean Sq F value Pr(>F)
AgeGroup  1 1.000e-27 1.002e-27   1.025  0.315
Residuals 62 6.065e-26 9.783e-28

Error: Subject:PrimeCondition
      Df      Sum Sq    Mean Sq F value Pr(>F)
PrimeCondition  3 5.670e-27 1.889e-27   1.522  0.210
AgeGroup:PrimeCondition  3 5.850e-27 1.949e-27   1.570  0.198
Residuals      186 2.309e-25 1.241e-27

Error: Subject:State
      Df Sum Sq Mean Sq F value    Pr(>F)
State  3 13054   4351   69.44 < 2e-16 ***
AgeGroup:State  3  2600    867   13.83 3.52e-08 ***
Residuals     186 11655    63

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition:State
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition:State  9   425   47.27  8.284 1.27e-11 ***
AgeGroup:PrimeCondition:State  9    73    8.11  1.422  0.175
Residuals      558  3184    5.71

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> library(ez)
> ezANOVA(data = exp2_state_prime, wid = .(Subject),
+         dv = .(Trials), within = .(PrimeCondition, State),
+         between = .(AgeGroup))
```

```

$ANOVA
      Effect DFn DFd          F          p p<.05
2      AgeGroup      1   62 -1.343107e-14 1.000000e+00
3    PrimeCondition      3  186  0.000000e+00 1.000000e+00
5              State      3  186  6.944189e+01 3.565485e-30      *
4    AgeGroup:PrimeCondition      3  186 -1.042726e-13 1.000000e+00
6    AgeGroup:State      3  186  1.382998e+01 3.516920e-08      *
7    PrimeCondition:State      9  558  8.283654e+00 1.265576e-11      *
8 AgeGroup:PrimeCondition:State      9  558  1.421911e+00 1.751251e-01
ges
2 2.126399e-34
3 0.000000e+00
5 4.680022e-01
4 8.505597e-34
6 1.490822e-01
7 2.786898e-02
8 4.896827e-03

$`Mauchly's Test for Sphericity`
      Effect          W          p p<.05
5              State 0.26102640 4.091098e-16      *
6    AgeGroup:State 0.26102640 4.091098e-16      *
7    PrimeCondition:State 0.09668573 1.761204e-11      *
8 AgeGroup:PrimeCondition:State 0.09668573 1.761204e-11      *

$`Sphericity Corrections`
      Effect          GGe          p[GG] p[GG]<.05          HFe
3    PrimeCondition 0.4345989 1.000000e+00          0.4401109
4    AgeGroup:PrimeCondition 0.4345989 1.000000e+00          0.4401109
5              State 0.5508579 9.451480e-18      * 0.5640238
6    AgeGroup:State 0.5508579 1.947455e-05      * 0.5640238
7    PrimeCondition:State 0.6609432 2.225474e-08      * 0.7389132
8 AgeGroup:PrimeCondition:State 0.6609432 2.055404e-01          0.7389132
p[HF] p[HF]<.05
3 1.000000e+00
4 1.000000e+00
5 4.078064e-18      *
6 1.615936e-05      *
7 3.967172e-09      *
8 1.980792e-01

```

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> exp2_state_lsm = lsmeans::lsmeans(exp2_stateprime_aov, c("AgeGroup", "PrimeCondition",
> prime_effect = cld(exp2_state_lsm, alpha = 0.05,
+       adjust = "tukey", details = TRUE, by = c("PrimeCondition", "State")))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))

```

	contrast	PrimeCondition	State	estimate	SE	df	t.ratio	p.value
	:	:	:	:	:	:	:	:
1	Old - Young	b	dontknow	4.84375	0.9669232	294.1878	5.009447	0.0001904
3	Old - Young	b	other	3.71875	0.9669232	294.1878	3.845962	0.0001904
4	Old - Young	b	TOT	2.09375	0.9669232	294.1878	2.165374	0.033211
5	Old - Young	p	dontknow	5.50000	0.9669232	294.1878	5.688146	0.000000
7	Old - Young	p	other	3.40625	0.9669232	294.1878	3.522772	0.000441
8	Old - Young	p	TOT	2.40625	0.9669232	294.1878	2.488564	0.012500
9	Old - Young	r	dontknow	4.15625	0.9669232	294.1878	4.298428	0.000000
11	Old - Young	r	other	3.12500	0.9669232	294.1878	3.231901	0.000750
12	Old - Young	r	TOT	2.43750	0.9669232	294.1878	2.520883	0.012500
13	Old - Young	u	dontknow	3.84375	0.9669232	294.1878	3.975238	0.000000
14	Old - Young	u	know	2.50000	0.9669232	294.1878	2.585521	0.011719
15	Old - Young	u	other	3.71875	0.9669232	294.1878	3.845962	0.0001904
16	Old - Young	u	TOT	2.62500	0.9669232	294.1878	2.714797	0.008594

```
> ### INDIVIDUAL T-TESTS FOR AGExSTATE interaction
>
> e2_young_dk = exp2_state %>% filter(AgeGroup == "Young" & State == "dontknow")
> e2_old_dk = exp2_state %>% filter(AgeGroup == "Old" & State == "dontknow")
> t.test(e2_old_dk$Trials, e2_young_dk$Trials)
```

Welch Two Sample t-test

```
data: e2_old_dk$Trials and e2_young_dk$Trials
t = 4.0675, df = 44.691, p-value = 0.0001904
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 9.258782 27.428718
sample estimates:
mean of x mean of y
47.75000 29.40625
```

```
> e2_young_other = exp2_state %>% filter(AgeGroup == "Young" & State == "other")
> e2_old_other = exp2_state %>% filter(AgeGroup == "Old" & State == "other")
> t.test(e2_young_other$Trials, e2_old_other$Trials)
```

Welch Two Sample t-test

```
data: e2_young_other$Trials and e2_old_other$Trials
t = 7.7708, df = 44.146, p-value = 8.441e-10
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
10.34629 17.59121
sample estimates:
mean of x mean of y
```

```
17.75000    3.78125
```

```
> e2_young_TOT = exp2_state %>% filter(AgeGroup == "Young" & State == "TOT")
> e2_old_TOT = exp2_state %>% filter(AgeGroup == "Old" & State == "TOT")
> t.test(e2_young_TOT$Trials, e2_old_TOT$Trials)
```

Welch Two Sample t-test

```
data: e2_young_TOT$Trials and e2_old_TOT$Trials
t = 4.4834, df = 59.013, p-value = 3.443e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 5.294682 13.830318
sample estimates:
mean of x mean of y
15.46875    5.90625
```

13 Experiment 3: 48ms

```
> exp3_target = subset(final_j, final_j$StudyNo == '1')
> exp3_mcq = subset(final_mcq, final_mcq$StudyNo == '1')
> exp3_state = subset(final_statedata, final_statedata$StudyNo == '1')
> exp3_state_prime = subset(statedata_primetype_long, statedata_primetype_long$StudyNo == '1')
> exp3_state_prime$PrimeCondition = as.factor(as.character(exp3_state_prime$PrimeCondition))
> exp3_state_prime$State = as.factor(as.character(exp3_state_prime$State))
> exp3_state_prime$Subject = as.factor(as.character(exp3_state_prime$Subject))
```

Exp 3: Target Accuracy

```
> ### TARGET RETRIEVAL ACCURACY
>
> exp3_target_aov = aov(data = exp3_target, Accuracy ~ PrimeCondition +
+                      Error (Subject/PrimeCondition))
> summary(exp3_target_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 35   1.994  0.05699

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition   3  0.1051  0.03503   2.929  0.0371 *
Residuals     105  1.2561  0.01196

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## specific t-tests
> target_p = exp3_target %>% filter(PrimeCondition == "Phonological")
> target_r = exp3_target %>% filter(PrimeCondition == "Related")
> target_b = exp3_target %>% filter(PrimeCondition == "Both")
> target_u = exp3_target %>% filter(PrimeCondition == "Unrelated")
> t.test(target_p$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_u$Accuracy
t = 3.0567, df = 35, p-value = 0.004268
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.0242553 0.1201891
sample estimates:
mean of the differences
      0.0722222
```

```
> library(ez)
> ezANOVA(data = exp3_target, wid = .(Subject),
+         dv = .(Accuracy), within = .(PrimeCondition))
```

\$ANOVA

	Effect	DFn	DFd	F	p	p<.05	ges
2	PrimeCondition	3	105	2.928509	0.03711626	*	0.03131964

\$`Mauchly's Test for Sphericity`

	Effect	W	p	p<.05
2	PrimeCondition	0.6688458	0.01868573	*

\$`Sphericity Corrections`

	Effect	GGe	p[GG]	p[GG]<.05	HFe	p[HF]	p[HF]<.05
2	PrimeCondition	0.8179437	0.04824832	*	0.8842609	0.04384152	*

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> exp3_target_lsm = lsmeans::lsmeans(exp3_target_aov, c("PrimeCondition"))
> prime_effect = cld(exp3_target_lsm, alpha = 0.05,
+                   adjust = "tukey", details = TRUE)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 1 ))
```

contrast	estimate	SE	df	t.ratio	p.value
Related - Unrelated	0.0155556	0.0257799	105	0.6033986	0.9307815
Both - Unrelated	0.0355556	0.0257799	105	1.3791969	0.5151504
Both - Related	0.0200000	0.0257799	105	0.7757982	0.8652374

Phonological - Unrelated	0.0722222	0.0257799	105	2.8014936	0.0303803
Phonological - Related	0.0566667	0.0257799	105	2.1980950	0.1304819
Phonological - Both	0.0366667	0.0257799	105	1.4222968	0.4883835

```
> ## specific t-tests
> target_p = exp3_target %>% filter(PrimeCondition == "Phonological")
> target_r = exp3_target %>% filter(PrimeCondition == "Related")
> target_b = exp3_target %>% filter(PrimeCondition == "Both")
> target_u = exp3_target %>% filter(PrimeCondition == "Unrelated")
> t.test(target_p$Accuracy, target_r$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_r$Accuracy
t = 2.2164, df = 35, p-value = 0.03326
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.004763719 0.108569614
sample estimates:
mean of the differences
      0.0566667
```

```
> t.test(target_p$Accuracy, target_b$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_b$Accuracy
t = 1.3342, df = 35, p-value = 0.1907
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-0.01912394 0.09245727
sample estimates:
mean of the differences
      0.0366667
```

```
> t.test(target_p$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_u$Accuracy
t = 3.0567, df = 35, p-value = 0.004268
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.0242553 0.1201891
sample estimates:
mean of the differences
      0.0722222
```



```
> t.test(target_b$Accuracy, target_r$Accuracy, paired = TRUE)
```

```
Paired t-test
```

```
data: target_b$Accuracy and target_r$Accuracy
t = 0.99061, df = 35, p-value = 0.3287
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.02098702  0.06098702
sample estimates:
mean of the differences
              0.02
```

```
> t.test(target_b$Accuracy, target_u$Accuracy, paired = TRUE)
```

```
Paired t-test
```

```
data: target_b$Accuracy and target_u$Accuracy
t = 1.3711, df = 35, p-value = 0.1791
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.01708853  0.08819964
sample estimates:
mean of the differences
              0.03555556
```

```
> t.test(target_r$Accuracy, target_u$Accuracy, paired = TRUE)
```

```
Paired t-test
```

```
data: target_r$Accuracy and target_u$Accuracy
t = 0.50726, df = 35, p-value = 0.6152
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.04669922  0.07781033
sample estimates:
mean of the differences
              0.01555556
```

```
>
```

Exp 3: Multiple Choice

```
> ## MULTIPLE CHOICE ACCURACY
> library(dplyr)
```

```
> exp3_mcq_acc = group_by(exp3_mcq, Subject, PrimeType) %>%
+   summarise_at(vars(MCQAcc), mean)
> exp3_mcq_acc_aov = aov(data = exp3_mcq_acc, MCQAcc ~ PrimeType +
+   Error(Subject/PrimeType))
> summary(exp3_mcq_acc_aov)
```

Error: Subject

```
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 35  1.638  0.0468
```

Error: Subject:PrimeType

```
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType  3 0.0790 0.026326  3.293 0.0235 *
Residuals 105 0.8394 0.007994
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## SPECIFIC T TESTS
>
> e3_mcq_p = exp3_mcq_acc %>% filter(PrimeType == "p")
> e3_mcq_r = exp3_mcq_acc %>% filter(PrimeType == "r")
> e3_mcq_b = exp3_mcq_acc %>% filter(PrimeType == "b")
> e3_mcq_u = exp3_mcq_acc %>% filter(PrimeType == "u")
> t.test(e3_mcq_r$MCQAcc, e3_mcq_u$MCQAcc, paired = TRUE) ## sig
```

Paired t-test

```
data: e3_mcq_r$MCQAcc and e3_mcq_u$MCQAcc
t = -2.8619, df = 35, p-value = 0.007063
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.09686401 -0.01646933
sample estimates:
mean of the differences
 -0.05666667
```

```
> t.test(e3_mcq_r$MCQAcc, e3_mcq_p$MCQAcc, paired = TRUE) ## not sig
```

Paired t-test

```
data: e3_mcq_r$MCQAcc and e3_mcq_p$MCQAcc
t = -2.3095, df = 35, p-value = 0.02694
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.10856663 -0.00698893
sample estimates:
mean of the differences
 -0.05777778
```

```
> ezANOVA(data = exp3_mcq_acc, wid = .(Subject),
+         dv = .(MCQAcc), within = .(PrimeType))
```

```
$ANOVA
```

```
      Effect DFn DFd      F      p p<.05      ges
2 PrimeType   3 105 3.293006 0.02348118 * 0.03089414
```

```
$`Mauchly's Test for Sphericity`
```

```
      Effect      W      p p<.05
2 PrimeType 0.7945667 0.1704433
```

```
$`Sphericity Corrections`
```

```
      Effect      GGe      p[GG] p[GG]<.05      HFe      p[HF] p[HF]<.05
2 PrimeType 0.8816733 0.02906934 * 0.9603953 0.02521661 *
```

```
> exp3_mcqacc_lsm = lsmeans::lsmeans(exp3_mcq_acc_aov, c("PrimeType"))
> prime_effect = cld(exp3_mcqacc_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.1))
```

	contrast	estimate	SE	df	t.ratio	p.value
2	u - r	0.0566667	0.0210746	105	2.688861	0.0409296
4	p - r	0.0577778	0.0210746	105	2.741584	0.0356502

```
> ## MULTIPLE CHOICE ERRORS
>
> ## before we do ANOVA, we need to replace NAs with 0.
>
> for (i in 1: nrow(exp3_mcq)){
+   if(is.na(exp3_mcq[i,7])){
+     exp3_mcq[i,7] = 0
+   }
+ }
> exp3_mcq_aov = aov(data = exp3_mcq, Proportion ~ PrimeType*ChosenPrime +
+                   Error(Subject/(PrimeType*ChosenPrime)))
> summary(exp3_mcq_aov)
```

```
Error: Subject
```

```
      Df      Sum Sq      Mean Sq F value Pr(>F)
Residuals 35 0.0005022 1.435e-05
```

```
Error: Subject:PrimeType
```

```
      Df      Sum Sq      Mean Sq F value Pr(>F)
PrimeType 3 0.000043 1.435e-05      1 0.396
Residuals 105 0.001507 1.435e-05
```

```
Error: Subject:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
ChosenPrime    3 30.305   10.102    448 <2e-16 ***
Residuals   105   2.368    0.023
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeType:ChosenPrime
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeType:ChosenPrime    9   2.021   0.22458     6.86 5.07e-09 ***
Residuals              315  10.313   0.03274
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> library(ez)
> ezANOVA(data = exp3_mcq, wid = .(Subject),
+         dv = .(Proportion), within = .(PrimeType, ChosenPrime))
```

```
$ANOVA
      Effect DFn DFd      F      p p<.05      ges
2      PrimeType    3 105   1.00000 3.959719e-01    3.393963e-06
3      ChosenPrime    3 105 447.96747 1.141515e-59    * 7.049728e-01
4 PrimeType:ChosenPrime    9 315   6.85965 5.070984e-09    * 1.374616e-01

$`Mauchly's Test for Sphericity`
      Effect      W      p p<.05
3      ChosenPrime 1.284794e-01 1.621597e-13    *
4 PrimeType:ChosenPrime 4.310013e-05 6.057443e-43    *

$`Sphericity Corrections`
      Effect      GGe      p[GG] p[GG]<.05      HFe      p[HF]
2      PrimeType 0.3333333 3.241743e-01    0.3333333 3.241743e-01
3      ChosenPrime 0.4657570 4.011334e-29    * 0.4791452 6.862229e-30
4 PrimeType:ChosenPrime 0.3828900 1.260879e-04    * 0.4297972 5.764997e-05
p[HF]<.05
2
3      *
4      *
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> exp3_errors_lsm = lsmeans::lsmeans(exp3_mcq_aov, c("PrimeType", "ChosenPrime"))
> prime_effect = cld(exp3_errors_lsm, alpha = 0.05,
+                   adjust = "tukey", details = TRUE, by = c("PrimeType"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.8))
```

	contrast	PrimeType	estimate	SE	df	t.ratio	p.value
:	--	:	-----	:	-----	:	-----

2	r - u	b		0.4447289	0.040955	411.2189	10.8589586	0.0000000
3	r - p	b		0.4414320	0.040955	411.2189	10.7784591	0.0000000
4	b - u	b		0.5154182	0.040955	411.2189	12.5849824	0.0000000
5	b - p	b		0.5121214	0.040955	411.2189	12.5044829	0.0000000
6	b - r	b		0.0706893	0.040955	411.2189	1.7260238	0.3114464
7	p - u	p		0.0395395	0.040955	411.2189	0.9654384	0.7691747
8	r - u	p		0.3744838	0.040955	411.2189	9.1437816	0.0000000
9	r - p	p		0.3349442	0.040955	411.2189	8.1783432	0.0000000
10	b - u	p		0.5006592	0.040955	411.2189	12.2246113	0.0000000
11	b - p	p		0.4611197	0.040955	411.2189	11.2591729	0.0000000
12	b - r	p		0.1261754	0.040955	411.2189	3.0808297	0.0117776
14	r - u	r		0.2877852	0.040955	411.2189	7.0268602	0.0000000
15	r - p	r		0.2702963	0.040955	411.2189	6.5998337	0.0000000
16	b - u	r		0.6669481	0.040955	411.2189	16.2848929	0.0000000
17	b - p	r		0.6494593	0.040955	411.2189	15.8578664	0.0000000
18	b - r	r		0.3791629	0.040955	411.2189	9.2580327	0.0000000
20	r - u	u		0.4185845	0.040955	411.2189	10.2205905	0.0000000
21	r - p	u		0.3879087	0.040955	411.2189	9.4715784	0.0000000
22	b - u	u		0.4818965	0.040955	411.2189	11.7664814	0.0000000
23	b - p	u		0.4512207	0.040955	411.2189	11.0174693	0.0000000
24	b - r	u		0.0633120	0.040955	411.2189	1.5458909	0.4110525

Exp 3: State Data

```
> ## just state
> exp3_state_aov = aov(data = exp3_state, Trials ~ State +
+                               Error(Subject/State))
> summary(exp3_state_aov)
```

```
Error: Subject
      Df    Sum Sq   Mean Sq F value Pr(>F)
Residuals 35 1.598e-26 4.564e-28
```

```
Error: Subject:State
      Df Sum Sq Mean Sq F value    Pr(>F)
State    3  13661    4554    21.8 4.71e-11 ***
Residuals 105  21929      209
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> exp3_state_lsm = lsmeans::lsmeans(exp3_state_aov, c("State"))
> prime_effect = cld(exp3_state_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```

	contrast	estimate	SE	df	t.ratio	p.value
2	dontknow - TOT	13.88889	3.406253	105	4.077469	0.0005089
3	dontknow - other	11.91667	3.406253	105	3.498468	0.0037854
4	know - TOT	24.13889	3.406253	105	7.086641	0.0000000
5	know - other	22.16667	3.406253	105	6.507640	0.0000000
6	know - dontknow	10.25000	3.406253	105	3.009172	0.0170295

```
> ##state by prime
> exp3_stateprime_aov = aov(data = exp3_state_prime, Trials ~ PrimeCondition*State +
+                               Error(Subject/(PrimeCondition*State)))
> summary(exp3_stateprime_aov)
```

```
Error: Subject
      Df      Sum Sq    Mean Sq F value Pr(>F)
Residuals 35 1.729e-27  4.939e-29

Error: Subject:PrimeCondition
      Df      Sum Sq    Mean Sq F value Pr(>F)
PrimeCondition  3 2.480e-28  8.258e-29    1.256  0.293
Residuals     105 6.905e-27  6.576e-29

Error: Subject:State
      Df Sum Sq Mean Sq F value    Pr(>F)
State   3   3415   1138.4    21.8 4.71e-11 ***
Residuals 105   5482    52.2

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition:State
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition:State  9   249.6    27.73    4.202 3.79e-05 ***
Residuals           315 2078.9     6.60

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> library(ez)
> ezANOVA(data = exp3_state_prime, wid = .(Subject),
+          dv = .(Trials), within = .(PrimeCondition, State))
```

```
$ANOVA
      Effect DFn DFd      F      p p<.05      ges
2    PrimeCondition  3 105 2.351171e-13 1.000000e+00 4.201945e-32
3          State    3 105 2.180421e+01 4.706186e-11 * 3.111485e-01
4 PrimeCondition:State  9 315 4.202170e+00 3.793581e-05 * 3.195577e-02

$`Mauchly's Test for Sphericity`
```

```

          Effect          W          p p<.05
3          State 0.50971868 3.843874e-04 *
4 PrimeCondition:State 0.05408547 2.925935e-05 *

`Sphericity Corrections`
          Effect          GGe          p[GG] p[GG]<.05          HFe          p[HF]
2          PrimeCondition 0.5676417 1.000000e+00          0.5936989 1.000000e+00
3          State 0.6848415 2.997089e-08          * 0.7281008 1.231925e-08
4 PrimeCondition:State 0.6056857 8.345856e-04          * 0.7304000 3.113067e-04
p[HF]<.05
2
3          *
4          *

```

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> exp3_state_lsm = lsmeans::lsmeans(exp3_stateprime_aov, c("PrimeCondition", "State"))
> prime_effect = cld(exp3_state_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = c("PrimeCondition"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.1))

```

	contrast	PrimeCondition	estimate	SE	df	t.ratio	p.value
2	dontknow - TOT	b	3.333333	1.000074	190.5969	3.333088	0.005648
3	dontknow - other	b	2.750000	1.000074	190.5969	2.749798	0.032842
4	know - TOT	b	6.861111	1.000074	190.5969	6.860607	0.000000
5	know - other	b	6.277778	1.000074	190.5969	6.277317	0.000000
6	know - dontknow	b	3.527778	1.000074	190.5969	3.527519	0.002932
8	dontknow - TOT	p	3.250000	1.000074	190.5969	3.249761	0.007404
9	dontknow - other	p	2.805556	1.000074	190.5969	2.805349	0.028162
10	know - TOT	p	5.861111	1.000074	190.5969	5.860680	0.000000
11	know - other	p	5.416667	1.000074	190.5969	5.416269	0.000001
12	know - dontknow	p	2.611111	1.000074	190.5969	2.610919	0.047588
14	dontknow - TOT	r	3.388889	1.000074	190.5969	3.388640	0.004699
15	dontknow - other	r	2.472222	1.000074	190.5969	2.472041	0.067621
16	know - TOT	r	7.694444	1.000074	190.5969	7.693879	0.000000
17	know - other	r	6.777778	1.000074	190.5969	6.777280	0.000000
18	know - dontknow	r	4.305556	1.000074	190.5969	4.305239	0.000155
20	dontknow - TOT	u	3.722222	1.000074	190.5969	3.721949	0.001473
21	dontknow - other	u	3.694444	1.000074	190.5969	3.694173	0.001628
22	know - TOT	u	3.916667	1.000074	190.5969	3.916379	0.000717
23	know - other	u	3.888889	1.000074	190.5969	3.888603	0.000796

```

> ## specific t
>
> ## for related primes
> e3mcq_r = exp3_mcq %>% filter(PrimeType == "r")
> e3mcq_r_r = e3mcq_r %>% filter(ChosenPrime == "r")

```

```
> e3mcq_r_b = e3mcq_r %>% filter(ChosenPrime == "b")
> t.test(e3mcq_r_r$Proportion, e3mcq_r_b$Proportion, paired = TRUE)
```

Paired t-test

```
data: e3mcq_r_r$Proportion and e3mcq_r_b$Proportion
t = 6.3399, df = 35, p-value = 2.761e-07
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.2577504 0.5005755
sample estimates:
mean of the differences
      0.3791629
```

```
> ## for both primes
> e3mcq_b = exp3_mcq %>% filter(PrimeType == "b")
> e3mcq_b_r = e3mcq_b %>% filter(ChosenPrime == "r")
> e3mcq_b_b = e3mcq_b %>% filter(ChosenPrime == "b")
> t.test(e3mcq_b_r$Proportion, e3mcq_b_b$Proportion, paired = TRUE)
```

Paired t-test

```
data: e3mcq_b_r$Proportion and e3mcq_b_b$Proportion
t = -0.94029, df = 35, p-value = 0.3535
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.2233092 0.0819305
sample estimates:
mean of the differences
     -0.07068934
```

```
>
```

14 Comparing YA 48 ms with OA NotthePrime

```
> for (i in 1: nrow(final_mcq)){
+   if(is.na(final_mcq[i,7])){
+     final_mcq[i,7] = 0
+   }
+ }
> exp3_compare_1 = subset(final_mcq, final_mcq$StudyNo == '6' |
+   final_mcq$StudyNo == '1')
> compare_aov_1 = aov(data = exp3_compare_1, Proportion ~ StudyNo*PrimeType*ChosenPrime)
> summary(compare_aov_1)
```



```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
StudyNo 1 0.1373 0.13726    50.99 9.1e-10 ***
Residuals 66 0.1777 0.00269
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeType
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType 3 0.0063 0.002101    1.031 0.380
StudyNo:PrimeType 3 0.0068 0.002255    1.107 0.347
Residuals 198 0.4034 0.002038

Error: Subject:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
ChosenPrime 3 52.52 17.508 698.864 <2e-16 ***
StudyNo:ChosenPrime 3 0.21 0.070 2.777 0.0424 *
Residuals 198 4.96 0.025
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeType:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType:ChosenPrime 9 2.722 0.30247 8.898 1.22e-12 ***
StudyNo:PrimeType:ChosenPrime 9 0.187 0.02075 0.610 0.789
Residuals 594 20.192 0.03399
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## specific t
>
> e3_old_b = exp3_compare_1 %>% filter(AgeGroup == "Old" & ChosenPrime == "b")
> mean_old = group_by(e3_old_b, Subject) %>%
+   summarise_at(vars(Proportion), mean)
> e3_young_b = exp3_compare_1 %>% filter(AgeGroup == "Young" & ChosenPrime == "b")
> mean_young = group_by(e3_young_b, Subject) %>%
+   summarise_at(vars(Proportion), mean)
> t.test(mean_young$Proportion, mean_old$Proportion)
```

Welch Two Sample t-test

```
data: mean_young$Proportion and mean_old$Proportion
t = 3.1003, df = 65.235, p-value = 0.002854
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.02334383 0.10784904
sample estimates:
```

```
mean of x mean of y
0.4125660 0.3469696
```

```
> ### e2 young and e3 young
> exp3_compare_2 = subset(final_mcq, final_mcq$StudyNo == '1' |
+                           final_mcq$StudyNo == '5')
> compare_aov_2 = aov(data = exp3_compare_2, Proportion ~ StudyNo*PrimeType*ChosenPrime)
> summary(compare_aov_2)
```

```
Error: Subject
```

```
      Df Sum Sq Mean Sq F value Pr(>F)
StudyNo  1 0.00117 0.0011663    1.247  0.268
Residuals 66 0.06173 0.0009353
```

```
Error: Subject:PrimeType
```

```
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType  3 0.00343 0.0011444    1.224  0.302
StudyNo:PrimeType  3 0.00426 0.0014210    1.519  0.211
Residuals 198 0.18518 0.0009353
```

```
Error: Subject:ChosenPrime
```

```
      Df Sum Sq Mean Sq F value Pr(>F)
ChosenPrime  3 56.76 18.921 711.408 <2e-16 ***
StudyNo:ChosenPrime  3  0.00  0.001  0.028  0.994
Residuals 198  5.27  0.027
```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Error: Subject:PrimeType:ChosenPrime
```

```
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType:ChosenPrime  9  0.123  0.0136  0.305  0.973
StudyNo:PrimeType:ChosenPrime  9  3.842  0.4269  9.540 1.2e-13 ***
Residuals 594 26.579  0.0447
```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> compare_lsm = lsmeans::lsmeans(compare_aov_1, c("StudyNo", "ChosenPrime"))
> prime_effect = multcomp::cld(compare_lsm, alpha = 0.05,
+                               adjust = "tukey", details = TRUE, by = c("ChosenPrime"))
> knitr::kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```

contrast	ChosenPrime	estimate	SE	df	t.ratio	p.value
1 - 6	b	0.0655964	0.016947	211.6232	3.870686	0.0001446

```
>
```

15 Comparing Prime Type Across Experiments

```
> ## final_j contains all experiments
>
> for(i in 1:nrow(final_j)){
+   if(final_j[i,3] == "2" | final_j[i,3] == "4"){
+     final_j[i,"Experiment"] = "Experiment1"
+   }
+   else if(final_j[i,3] == "5" | final_j[i,3] == "6"){
+     final_j[i,"Experiment"] = "Experiment2"
+   }
+   else
+     final_j[i,"Experiment"] = "Experiment3"
+ }
> final_j$Experiment = as.factor(as.character(final_j$Experiment))
> combined_targetacc = aov(data = final_j, Accuracy ~ PrimeCondition +
+                           Error(Subject/PrimeCondition))
> summary(combined_targetacc)
```

Error: Subject

```
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 171   11.01   0.0644
```

Error: Subject:PrimeCondition

```
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition   3   1.117   0.3725   30.76 <2e-16 ***
Residuals      513   6.212   0.0121
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> combined_targetacc_lsm = lsmeans::lsmeans(combined_targetacc,
+                                             c("PrimeCondition"))
> prime_effect = cld(combined_targetacc_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 1 ))
```

contrast	estimate	SE	df	t.ratio	p.value
Related - Unrelated	0.0076744	0.0118663	513	0.6467402	0.9166711
Both - Unrelated	0.0360465	0.0118663	513	3.0377192	0.0133283
Both - Related	0.0283721	0.0118663	513	2.3909790	0.0800700
Phonological - Unrelated	0.1023256	0.0118663	513	8.6232028	0.0000000
Phonological - Related	0.0946512	0.0118663	513	7.9764626	0.0000000
Phonological - Both	0.0662791	0.0118663	513	5.5854837	0.0000002

```
> ## PAIRWISE COMPARISONS
>
> compare_p = final_j %>% filter(PrimeCondition == "Phonological")
> compare_r = final_j %>% filter(PrimeCondition == "Related")
> compare_b = final_j %>% filter(PrimeCondition == "Both")
> compare_u = final_j %>% filter(PrimeCondition == "Unrelated")
> t.test(compare_p$Accuracy, compare_r$Accuracy, paired = TRUE)
```

Paired t-test

```
data: compare_p$Accuracy and compare_r$Accuracy
t = 7.81, df = 171, p-value = 5.503e-13
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.07072869 0.11857364
sample estimates:
mean of the differences
 0.09465116
```

16 Multiple Choice: Only R and B

```
> ### EXPERIMENT 1 ###
> ## MULTIPLE CHOICE ACCURACY
> exp1_mcqacc_subset = subset(exp1_mcq_acc, exp1_mcq_acc$PrimeType == 'r' |
+                               exp1_mcq_acc$PrimeType == 'b')
> exp1_mcqacc_subset_aov = aov(data = exp1_mcqacc_subset, MCQAcc ~ PrimeType +
+                               Error(Subject/PrimeType))
> summary(exp1_mcqacc_subset_aov)
```

Error: Subject

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Residuals	71	3.301	0.04649		

Error: Subject:PrimeType

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeType	1	0.0324	0.03240	7.126	0.00941 **
Residuals	71	0.3228	0.00455		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> ezANOVA(data = exp1_mcqacc_subset, wid = .(Subject),
+          dv = .(MCQAcc), within = .(PrimeType),
+          between = .(AgeGroup))
```

```
$ANOVA
```

	Effect	DFn	DFd	F	p	p<.05	ges
2	AgeGroup	1	70	0.2421229	0.624216686		0.003146432
3	PrimeType	1	70	7.1934029	0.009121006	*	0.008908169
4	AgeGroup:PrimeType	1	70	1.6676064	0.200827502		0.002079360

```
> ## MULTIPLE CHOICE ERRORS: only when they chose b or r
>
> exp1_mcq_subset = subset(exp1_mcq, exp1_mcq$ChosenPrime == "r" |
+                           exp1_mcq$ChosenPrime == 'b')
> ## before we do ANOVA, we need to replace NAs with 0.
>
> for (i in 1:nrow(exp1_mcq_subset)){
+   if(is.na(exp1_mcq_subset[i,7])){
+     exp1_mcq_subset[i,7] = 0
+   }
+ }
> exp1_mcq_subset_aov = aov(data = exp1_mcq_subset, Proportion ~ AgeGroup*PrimeType*ChosenPrime,
+                           Error(Subject/(PrimeType*ChosenPrime)))
> summary(exp1_mcq_subset_aov)
```

```
Error: Subject
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup	1	0.2977	0.29768	14.65	0.000278 ***
Residuals	70	1.4220	0.02031		

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Error: Subject:PrimeType
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeType	3	0.1540	0.05133	3.567	0.015 *
AgeGroup:PrimeType	3	0.0729	0.02430	1.689	0.171
Residuals	210	3.0219	0.01439		

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Error: Subject:ChosenPrime
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
ChosenPrime	1	2.074	2.0735	28.457	1.12e-06 ***
AgeGroup:ChosenPrime	1	0.177	0.1771	2.431	0.123
Residuals	70	5.101	0.0729		

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Error: Subject:PrimeType:ChosenPrime
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
--	----	--------	---------	---------	--------

```

PrimeType:ChosenPrime      3  2.783  0.9277  9.172 9.93e-06 ***
AgeGroup:PrimeType:ChosenPrime 3  0.339  0.1131  1.118  0.343
Residuals                  210 21.241  0.1011
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> library(ez)
> ezANOVA(data = expl_mcq_subset, wid = .(Subject),
+         dv = .(Proportion), within = .(PrimeType, ChosenPrime),
+         between = .(AgeGroup))

```

```

$ANOVA
      Effect DFn DFd      F      p p<.05
2      AgeGroup      1  70 14.653612 2.781791e-04 *
3      PrimeType      3 210  3.567095 1.501920e-02 *
5      ChosenPrime      1  70 28.457363 1.117738e-06 *
4      AgeGroup:PrimeType      3 210  1.688635 1.705102e-01
6      AgeGroup:ChosenPrime      1  70  2.431032 1.234646e-01
7      PrimeType:ChosenPrime      3 210  9.171915 9.925084e-06 *
8 AgeGroup:PrimeType:ChosenPrime      3 210  1.117860 3.428037e-01
ges
2 0.009576765
3 0.004977229
5 0.063103733
4 0.002362373
6 0.005720946
7 0.082909518
8 0.010898344

$`Mauchly's Test for Sphericity`
      Effect      W      p p<.05
3      PrimeType 0.8478173 0.04496936 *
4      AgeGroup:PrimeType 0.8478173 0.04496936 *
7      PrimeType:ChosenPrime 0.9078093 0.24830092
8 AgeGroup:PrimeType:ChosenPrime 0.9078093 0.24830092

$`Sphericity Corrections`
      Effect      GGe      p[GG] p[GG]<.05      HFe
3      PrimeType 0.9074415 1.831832e-02 * 0.9477390
4      AgeGroup:PrimeType 0.9074415 1.756166e-01 0.9477390
7      PrimeType:ChosenPrime 0.9427793 1.629124e-05 * 0.9865866
8 AgeGroup:PrimeType:ChosenPrime 0.9427793 3.413039e-01 0.9865866
p[HF] p[HF]<.05
3 1.679918e-02 *
4 1.733916e-01
7 1.114679e-05 *
8 3.424713e-01

```

```

> ##### EXPERIMENT 2 #####
>
> exp2_mcq_subset = subset(exp2_mcq, exp2_mcq$ChosenPrime == "r" |
+                           exp2_mcq$ChosenPrime == 'b')
> ## before we do ANOVA, we need to replace NAs with 0.
>
> for (i in 1: nrow(exp2_mcq_subset)){
+   if(is.na(exp2_mcq_subset[i,7])){
+     exp2_mcq_subset[i,7] = 0
+   }
+ }
> exp2_mcq_subset_aov = aov(data = exp2_mcq_subset, Proportion ~ AgeGroup*PrimeType*ChosenPrime
+                           Error(Subject/(PrimeType*ChosenPrime)))
> summary(exp2_mcq_subset_aov)

```

```

Error: Subject
      Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup    1  0.2348  0.23477    13.82 0.000435 ***
Residuals  62  1.0534  0.01699
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Error: Subject:PrimeType
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeType    3  0.0373  0.012425    1.252  0.292
AgeGroup:PrimeType    3  0.0265  0.008833    0.890  0.447
Residuals    186  1.8458  0.009924
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Error: Subject:ChosenPrime
      Df Sum Sq Mean Sq F value    Pr(>F)
ChosenPrime    1  2.529  2.5291  41.372 2.05e-08 ***
AgeGroup:ChosenPrime    1  0.062  0.0621  1.016  0.317
Residuals    62  3.790  0.0611
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> library(ez)
> ezANOVA(data = exp2_mcq_subset, wid = .(Subject),

```

```
+ dv = .(Proportion), within =.(PrimeType, ChosenPrime),
+ between = .(AgeGroup))
```

```
$ANOVA
```

	Effect	DFn	DFd	F	p	p<.05
2	AgeGroup	1	62	13.8183258	4.348315e-04	*
3	PrimeType	3	186	1.2521078	2.922707e-01	
5	ChosenPrime	1	62	41.3724846	2.045404e-08	*
4	AgeGroup:PrimeType	3	186	0.8901218	4.473292e-01	
6	AgeGroup:ChosenPrime	1	62	1.0161089	3.173617e-01	
7	PrimeType:ChosenPrime	3	186	0.7097012	5.473375e-01	
8	AgeGroup:PrimeType:ChosenPrime	3	186	6.6690679	2.665704e-04	*

ges

2	0.0078637410
3	0.0012568730
5	0.0786672859
4	0.0008938342
6	0.0020926497
7	0.0087839000
8	0.0768723864

```
$`Mauchly's Test for Sphericity`
```

	Effect	W	p	p<.05
3	PrimeType	0.7286939	0.001752956	*
4	AgeGroup:PrimeType	0.7286939	0.001752956	*
7	PrimeType:ChosenPrime	0.7945319	0.015840376	*
8	AgeGroup:PrimeType:ChosenPrime	0.7945319	0.015840376	*

```
$`Sphericity Corrections`
```

	Effect	GGe	p[GG]	p[GG]<.05	HFe
3	PrimeType	0.8362614	0.2919991603		0.8743778
4	AgeGroup:PrimeType	0.8362614	0.4325511356		0.8743778
7	PrimeType:ChosenPrime	0.8589321	0.5271655813		0.8994139
8	AgeGroup:PrimeType:ChosenPrime	0.8589321	0.0005924395	*	0.8994139

p[HF] p[HF]<.05

3	0.2922084949	
4	0.4362712483	
7	0.5332607959	
8	0.0004708642	*

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> exp2_errors_subset_lsm = lsmeans::lsmeans(exp2_mcq_subset_aov,
+                                           c("AgeGroup", "PrimeType", "ChosenPrime"))
> prime_effect = cld(exp2_errors_subset_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE, by = c("PrimeType", "ChosenPrime"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```


	contrast	PrimeType	ChosenPrime	estimate	SE	df	t.ratio	p.value
	:	:	:	:	:	:	:	:
1	Old - Young	b	b	0.1390686	0.0610935	283.4801	2.276325	0.0241
2	Old - Young	b	r	0.1790249	0.0610935	283.4801	2.930344	0.0034
5	Old - Young	r	b	0.2871214	0.0610935	283.4801	4.699706	0.0001
6	Old - Young	r	r	0.1757002	0.0610935	283.4801	2.875924	0.0044

17 Recoding RPBUs to Sound Meaning

```
> for(i in 1: nrow(final_j)) {
+
+   if(final_j[i,5] == "Related"){
+     final_j[i,7] = "No"
+     final_j[i,8] = "Yes"
+   }
+   else if(final_j[i,5] == "Both"){
+     final_j[i,7] = "Yes"
+     final_j[i,8] = "Yes"
+   }
+   else if(final_j[i,5] == "Phonological"){
+     final_j[i,7] = "Yes"
+     final_j[i,8] = "No"
+   }
+   else {
+     final_j[i,7] = "No"
+     final_j[i,8] = "No"
+   }
+ }
> colnames(final_j) = c("AgeGroup", "Subject", "StudyNo", "PrimeInstruction", "PrimeCondition",
+   "Accuracy", "Sound", "Meaning")
>
```

18 Collapsing the 4 experiments

```
> final_mcq_main4 = subset(final_mcq, final_mcq$StudyNo != '1')
> for (i in 1: nrow(final_mcq_main4)){
+   if(is.na(final_mcq_main4[i,7])){
+     final_mcq_main4[i,7] = 0
+   }
+ }
> fourway_aov = aov(data = final_mcq_main4, Proportion ~ AgeGroup*PrimeInstruction*PrimeCondition)
```

```
> summary(fourway_aov)
```

```
Error: Subject
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup	1	0.2700	0.26998	39.117	5.15e-09 ***
PrimeInstruction	1	0.0201	0.02009	2.911	0.0903 .
AgeGroup:PrimeInstruction	1	0.0017	0.00165	0.239	0.6254
Residuals	132	0.9111	0.00690		

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Error: Subject:PrimeType
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeType	3	0.0103	0.003432	0.839	0.473
AgeGroup:PrimeType	3	0.0227	0.007556	1.848	0.138
PrimeInstruction:PrimeType	3	0.0103	0.003435	0.840	0.473
AgeGroup:PrimeInstruction:PrimeType	3	0.0101	0.003361	0.822	0.482
Residuals	396	1.6194	0.004089		

```
Error: Subject:ChosenPrime
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
ChosenPrime	3	99.14	33.05	1090.679	<2e-16 ***
AgeGroup:ChosenPrime	3	0.49	0.16	5.392	0.0012 **
PrimeInstruction:ChosenPrime	3	0.06	0.02	0.680	0.5649
AgeGroup:PrimeInstruction:ChosenPrime	3	0.02	0.01	0.190	0.9030
Residuals	396	12.00	0.03		

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Error: Subject:PrimeType:ChosenPrime
```

	Df	Sum Sq	Mean Sq	F value
PrimeType:ChosenPrime	9	1.39	0.15483	3.573
AgeGroup:PrimeType:ChosenPrime	9	2.23	0.24759	5.714
PrimeInstruction:PrimeType:ChosenPrime	9	2.07	0.23039	5.317
AgeGroup:PrimeInstruction:PrimeType:ChosenPrime	9	0.75	0.08290	1.913
Residuals	1188	51.48	0.04333	

```
Pr(>F)
```

PrimeType:ChosenPrime	0.000213	***
AgeGroup:PrimeType:ChosenPrime	8.70e-08	***
PrimeInstruction:PrimeType:ChosenPrime	3.84e-07	***
AgeGroup:PrimeInstruction:PrimeType:ChosenPrime	0.046489	*
Residuals		

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> library(ez)
```

```
> ezANOVA(data = final_mcq_main4, wid = .(Subject),
+         dv = .(Proportion), within = .(PrimeType, ChosenPrime),
```

```
+      between = .(AgeGroup, PrimeInstruction)) ## IMPORTANT SPHERICITY
```

```
$ANOVA
```

	Effect	DFn	DFd	F
2	AgeGroup	1	132	39.1165536
3	PrimeInstruction	1	132	2.9111506
5	PrimeType	3	396	0.8392819
9	ChosenPrime	3	396	1090.6792585
4	AgeGroup:PrimeInstruction	1	132	0.2394847
6	AgeGroup:PrimeType	3	396	1.8477392
7	PrimeInstruction:PrimeType	3	396	0.8400384
10	AgeGroup:ChosenPrime	3	396	5.3923100
11	PrimeInstruction:ChosenPrime	3	396	0.6797443
13	PrimeType:ChosenPrime	9	1188	3.5732802
8	AgeGroup:PrimeInstruction:PrimeType	3	396	0.8217854
12	AgeGroup:PrimeInstruction:ChosenPrime	3	396	0.1902778
14	AgeGroup:PrimeType:ChosenPrime	9	1188	5.7140307
15	PrimeInstruction:PrimeType:ChosenPrime	9	1188	5.3170331
16	AgeGroup:PrimeInstruction:PrimeType:ChosenPrime	9	1188	1.9131806

	p	p<.05	ges
2	5.147210e-09	*	0.0040736584
3	9.032079e-02		0.0003043192
5	4.729393e-01		0.0001559683
9	5.790120e-191	*	0.6003149154
4	6.253907e-01		0.0000250417
6	1.379397e-01		0.0003433111
7	4.725329e-01		0.0001561089
10	1.202800e-03	*	0.0073709922
11	5.648524e-01		0.0009351975
13	2.134711e-04	*	0.0206752695
8	4.824149e-01		0.0001527173
12	9.030058e-01		0.0002619621
14	8.704312e-08	*	0.0326573025
15	3.837601e-07	*	0.0304574555
16	4.648859e-02	*	0.0111771697

```
$`Mauchly's Test for Sphericity`
```

	Effect	W	p
5	PrimeType	0.8330584683	2.294453e-04
6	AgeGroup:PrimeType	0.8330584683	2.294453e-04
7	PrimeInstruction:PrimeType	0.8330584683	2.294453e-04
8	AgeGroup:PrimeInstruction:PrimeType	0.8330584683	2.294453e-04
9	ChosenPrime	0.1930351611	1.812791e-44
10	AgeGroup:ChosenPrime	0.1930351611	1.812791e-44
11	PrimeInstruction:ChosenPrime	0.1930351611	1.812791e-44
12	AgeGroup:PrimeInstruction:ChosenPrime	0.1930351611	1.812791e-44
13	PrimeType:ChosenPrime	0.0002090455	1.871522e-199
14	AgeGroup:PrimeType:ChosenPrime	0.0002090455	1.871522e-199

```

15 PrimeInstruction:PrimeType:ChosenPrime 0.0002090455 1.871522e-199
16 AgeGroup:PrimeInstruction:PrimeType:ChosenPrime 0.0002090455 1.871522e-199
p<.05
5 *
6 *
7 *
8 *
9 *
10 *
11 *
12 *
13 *
14 *
15 *
16 *

```

\$`Sphericity Corrections`

	Effect	GGe	p[GG]
5	PrimeType	0.8932904	4.619944e-01
6	AgeGroup:PrimeType	0.8932904	1.447530e-01
7	PrimeInstruction:PrimeType	0.8932904	4.616120e-01
8	AgeGroup:PrimeInstruction:PrimeType	0.8932904	4.709164e-01
9	ChosenPrime	0.5505086	1.669246e-106
10	AgeGroup:ChosenPrime	0.5505086	8.410390e-03
11	PrimeInstruction:ChosenPrime	0.5505086	4.808760e-01
12	AgeGroup:PrimeInstruction:ChosenPrime	0.5505086	7.847361e-01
13	PrimeType:ChosenPrime	0.3974174	9.334130e-03
14	AgeGroup:PrimeType:ChosenPrime	0.3974174	3.175525e-04
15	PrimeInstruction:PrimeType:ChosenPrime	0.3974174	5.975158e-04
16	AgeGroup:PrimeInstruction:PrimeType:ChosenPrime	0.3974174	1.149206e-01

p[GG]<.05

	HFe	p[HF]	p[HF]<.05
5	0.9135543	4.641762e-01	
6	0.9135543	1.434431e-01	
7	0.9135543	4.637891e-01	
8	0.9135543	4.732063e-01	
9	* 0.5565925	1.199258e-107	*
10	* 0.5565925	8.189554e-03	*
11	0.5565925	4.824091e-01	
12	0.5565925	7.872370e-01	
13	* 0.4098502	8.613284e-03	*
14	* 0.4098502	2.672129e-04	*
15	* 0.4098502	5.117689e-04	*
16	0.4098502	1.127332e-01	

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> fourway_lsm = lsmeans::lsmeans(fourway_aov,
+                               c("AgeGroup", "PrimeInstruction", "PrimeType", "ChosenPrime"),
> prime_effect = cld(fourway_lsm, alpha = 0.05,

```

```
+ adjust = "tukey", details = TRUE, by = c("AgeGroup", "PrimeType", "ChosenPrime")
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```

	contrast	AgeGroup	PrimeType	ChosenPrime	estimate	SE	df	t.ratio	p.value
17	NoInstruction - NotThePrime	Young	b	b	0.2183051	0.042950			
19	NoInstruction - NotThePrime	Young	b	r	0.2253158	0.042950			
25	NoInstruction - NotThePrime	Young	r	b	0.1720295	0.042950			
27	NoInstruction - NotThePrime	Young	r	r	0.1517636	0.042950			

```
> ## SPECIFIC T-TEST
>
> ## Effect of Instruction on Young
>
> ## Semantic
>
> y_r = final_mcq_main4 %>% filter(AgeGroup == "Young" & PrimeType == "r")
> y_r_r_no = y_r %>% filter(PrimeInstruction == "NoInstruction" & ChosenPrime == "r")
> y_r_r_yes = y_r %>% filter(PrimeInstruction != "NoInstruction" & ChosenPrime == "r")
> t.test(y_r_r_no$Proportion, y_r_r_yes$Proportion)
```

Welch Two Sample t-test

```
data: y_r_r_no$Proportion and y_r_r_yes$Proportion
t = 2.1908, df = 59.19, p-value = 0.03241
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01316032 0.29036684
sample estimates:
mean of x mean of y
0.5744779 0.4227143
```

```
> ## Both
>
> y_b = final_mcq_main4 %>% filter(AgeGroup == "Young" & PrimeType == "b")
> y_b_b_no = y_b %>% filter(PrimeInstruction == "NoInstruction" & ChosenPrime == "b")
> y_b_b_yes = y_b %>% filter(PrimeInstruction != "NoInstruction" & ChosenPrime == "b")
> t.test(y_b_b_no$Proportion, y_b_b_yes$Proportion)
```

Welch Two Sample t-test

```
data: y_b_b_no$Proportion and y_b_b_yes$Proportion
t = 3.8063, df = 63.702, p-value = 0.0003192
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.1037192 0.3328911
```

```
sample estimates:
mean of x mean of y
0.5081771 0.2898719
```

```
> ## Effect of Instruction on Old
>
> ## Semantic
> o_r = final_mcq_main4 %>% filter(AgeGroup == "Old" & PrimeType == "r")
> o_r_r_no = o_r %>% filter(PrimeInstruction == "NoInstruction" & ChosenPrime == "r")
> o_r_r_yes = o_r %>% filter(PrimeInstruction != "NoInstruction" & ChosenPrime == "r")
> t.test(o_r_r_no$Proportion, o_r_r_yes$Proportion)
```

Welch Two Sample t-test

```
data: o_r_r_no$Proportion and o_r_r_yes$Proportion
t = 1.3867, df = 65.932, p-value = 0.1702
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.02988145  0.16577001
sample estimates:
mean of x mean of y
0.6663588 0.5984145
```

```
> ## Both
>
> o_b = final_mcq_main4 %>% filter(AgeGroup == "Old" & PrimeType == "b")
> o_b_b_no = o_b %>% filter(PrimeInstruction == "NoInstruction" & ChosenPrime == "b")
> o_b_b_yes = o_b %>% filter(PrimeInstruction != "NoInstruction" & ChosenPrime == "b")
> t.test(o_b_b_no$Proportion, o_b_b_yes$Proportion)
```

Welch Two Sample t-test

```
data: o_b_b_no$Proportion and o_b_b_yes$Proportion
t = 0.38076, df = 64.995, p-value = 0.7046
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.09263325  0.13627479
sample estimates:
mean of x mean of y
0.4507613 0.4289405
```

```
>
>
>
```

19 Tables and Figures

Experiment 1

State data

```
> exp1_fig_state = Rmisc::summarySE(exp1_state_prime,
+                                 measurevar = "Trials",
+                                 groupvars = c("AgeGroup", "PrimeCondition", "State"))
> library(ggplot2)
> library(ggthemes)
> state_1 = exp1_fig_state %>% mutate(PrimeType = factor(PrimeCondition,
+                                                         levels = unique(PrimeCondition)),
+                                     labels = c("Both", "Phonological",
+                                                 "Semantic", "Unrelated")),
+                                     RetrievalState = factor(State, levels = unique(State),
+                                                             labels = c("Dont Know", "Know", "Other", "TOT")))%>%
+ ggplot(aes(x = PrimeType, y = Trials,
+            fill = RetrievalState, group = RetrievalState))+
+ geom_bar(stat = "identity", position = "dodge", width = 0.7, color = "black")+
+ geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+ facet_wrap(~AgeGroup)+
+ theme_few()+
+ scale_fill_colorblind()+
+ xlab("") + ylab("") +
+ ggtitle("E1: Young and Old Adults (Without Instructions)") +
+ ggtitle("E1: Young vs. Old (Without Instructions)") +
+ theme(axis.text = element_text(size = rel(1)),
+       axis.title = element_text(face = "bold", size = rel(1)),
+       legend.title = element_text(face = "bold", size = rel(1)),
+       plot.title = element_text(hjust = .5),
+       strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

State by Prime

```
> exp1_fig_state_prime = Rmisc::summarySE(exp1_state_prime,
+                                          measurevar = "Trials",
+                                          groupvars = c("PrimeCondition", "State"))
> library(ggplot2)
> library(ggthemes)
> state_1_prime = exp1_fig_state_prime %>% mutate(PrimeType =
+                                                  factor(PrimeCondition,
+                                                      levels = unique(PrimeCondition)),
+                                                  labels = c("Both", "Phonological",
+                                                            "Semantic", "Unrelated")),
+                                                  RetrievalState = factor(State,
```

```

+           levels = unique(State),
+           labels = c("Dont Know", "Know", "Other", "TOT")))%>%
+ ggplot(aes(x = PrimeType, y = Trials,
+           group = RetrievalState, fill = RetrievalState))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_colorblind()+
+   xlab("") + ylab("") +
+   ggtitle("E1: Young and Old Adults (Without Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
+
+

```

State ONLY

```

> exp1_fig_state_only = Rmisc::summarySE(exp1_state,
+   measurevar = "Trials",
+   groupvars = c("AgeGroup", "State"))
> exp1_fig_state_only = arrange(exp1_fig_state_only,
+   desc(AgeGroup))
> library(ggplot2)
> library(ggthemes)
> state_1_only = exp1_fig_state_only %>% mutate(RetrievalState = factor(State,
+   levels = unique(State),
+   labels = c("Dont Know", "Know", "Other", "TOT")),
+   Age = factor(AgeGroup, levels = unique(AgeGroup),
+   labels = c("Young", "Old")))%>%
+ ggplot(aes(x = RetrievalState, y = Trials,
+   group = Age, fill = Age))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("") +
+   ggtitle("E1: Young and Old Adults (Without Instructions)") +
+   ggtitle("E1: Young vs. Old (Without Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),

```



```
+       plot.title = element_text(hjust = .5),
+       strip.text.x = element_text(face = "bold", size = rel(1.4)))
>
```

State ONLY

```
> exp1_fig_state_only = Rmisc::summarySE(exp1_state,
+       measurevar = "Trials",
+       groupvars = c("AgeGroup", "State"))
> exp1_fig_state_only = arrange(exp1_fig_state_only,
+       desc(AgeGroup))
> library(ggplot2)
> library(ggthemes)
> state_1_only = exp1_fig_state_only %>% mutate(RetrievalState = factor(State,
+       levels = unique(State),
+       labels = c("Dont Know", "Know", "Other", "TOT")),
+       Age = factor(AgeGroup, levels = unique(AgeGroup),
+       labels = c("Young", "Old")))%>%
+ ggplot(aes(x = RetrievalState, y = Trials,
+       group = Age, fill = Age))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+       width=.2, color = "gray26",
+       position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("") +
+   ggtitle("E1: Young vs. Old (Without Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+       axis.title = element_text(face = "bold", size = rel(1)),
+       legend.title = element_text(face = "bold", size = rel(1)),
+       plot.title = element_text(hjust = .5),
+       strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

Target Accuracy

```
> exp1_fig_target = Rmisc::summarySE(exp1_target,
+       measurevar = "Accuracy",
+       groupvars = c("AgeGroup", "PrimeCondition"))
> exp1_fig_target = arrange(exp1_fig_target, desc(AgeGroup))
> library(ggplot2)
> library(ggthemes)
> targetacc_1 = exp1_fig_target %>% mutate(PrimeType = factor(PrimeCondition,
+       levels = unique(PrimeCondition)),
+       labels = c("Both", "Phonological",
+       "Semantic", "Unrelated")),
```

```

+           Age = factor(AgeGroup, levels = unique(AgeGroup),
+           labels = c("Young", "Old")))%>%
+ ggplot(aes(x = PrimeType, y = Accuracy,
+           fill = Age, group = Age))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color = "black")+
+   geom_errorbar(aes(ymin=Accuracy - se, ymax=Accuracy + se),
+           width=.2, color = "gray26",
+           position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("") +
+   ggtitle("Young and Old Adults (No Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1.2)),
+         plot.title = element_text(hjust = .5),
+         legend.text = element_text(face = "bold", size = rel(1.1)),
+         axis.text.x = element_text(face = "bold", size = rel(1.2)),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))

```

MCQ Table

```

> ## CODE BELOW IS IF WE WANT MCQ NUMBERS FOR SAME/DIFFERENT PRIME CHOICE
> # for(i in 1:nrow(exp1_mcq)){
> #   if(exp1_mcq[i,"PrimeType"] == exp1_mcq[i,"ChosenPrime"]){
> #     exp1_mcq[i,"MCQChoice"] = "Same"
> #   }
> #   else {
> #     exp1_mcq[i,"MCQChoice"] = "Different"
> #   }
> # }
> #
> # e1_mcq_yn = group_by(exp1_mcq, Subject, AgeGroup, StudyNo,
> #                       PrimeType, MCQChoice ) %>%
> #               summarise_at(vars(Proportion), sum)
> # library(Rmisc)
> # e1_mcq_agg_yn = summarySE(e1_mcq_agg,
> #                           measurevar = "Proportion",
> #                           groupvars = c("AgeGroup", "PrimeType", "MCQChoice"))
>
> ## CODE BELOW ONLY FOR R AND B CHOICES in MCQ
>
> e1_mcq_agg = Rmisc::summarySE(exp1_mcq,
+                               measurevar = "Proportion",
+                               groupvars = c("AgeGroup", "PrimeType", "ChosenPrime"))
> e1_mcq_main = e1_mcq_agg %>% filter(PrimeType %in% c("b", "r") &

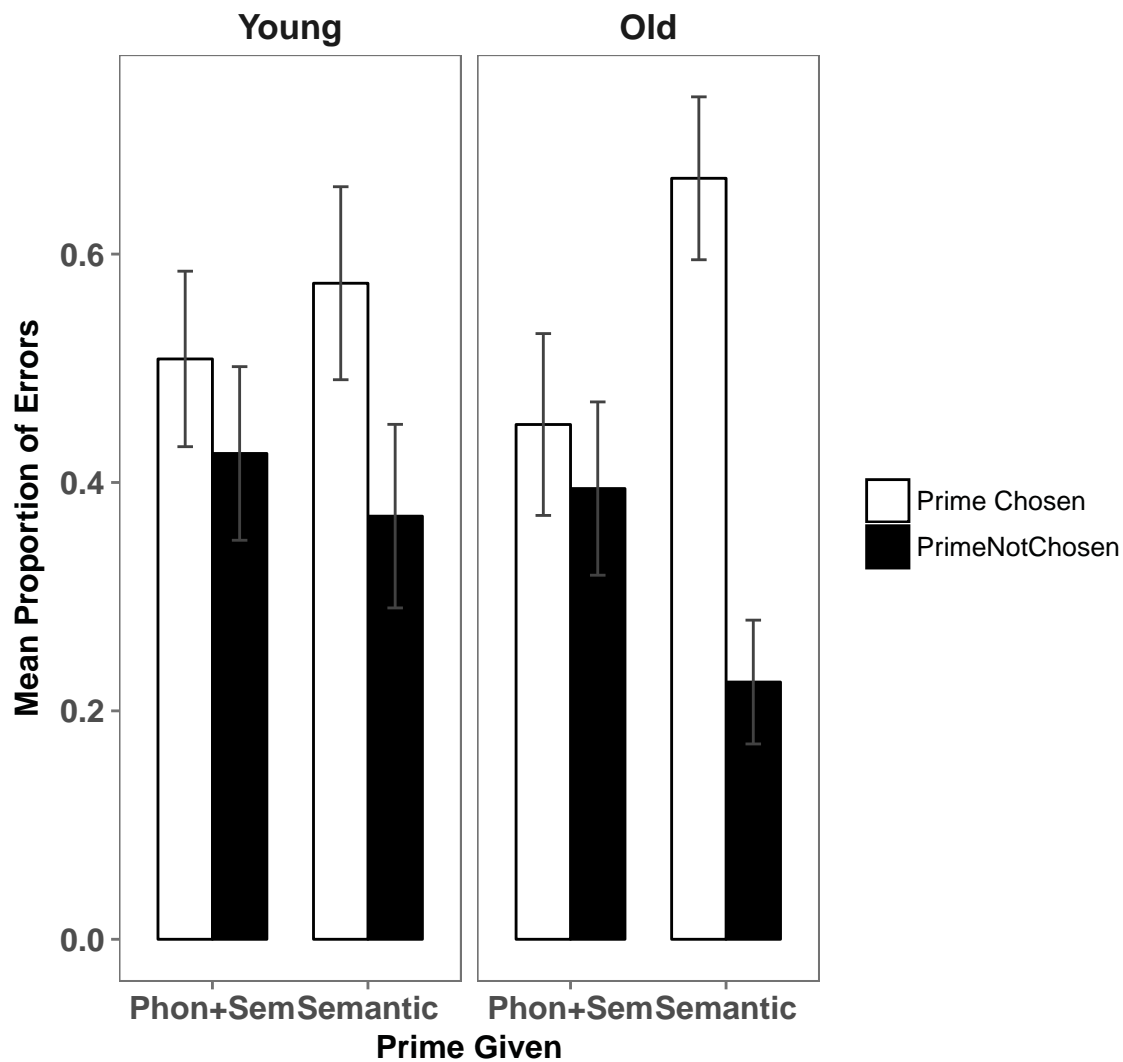
```

```

+                               ChosenPrime %in% c("b", "r"))
> e1_mcq_main$ChoseThePrime = c("1_Yes", "2_No", "2_No", "1_Yes",
+                               "1_Yes", "2_No", "2_No", "1_Yes")
> e1_mcq_main = dplyr::arrange(e1_mcq_main, desc(AgeGroup))
> library(ggplot2)
> library(ggthemes)
> e1_mcq_main %>% mutate(PrimeCondition = factor(PrimeType,
+                               levels = unique(PrimeType),
+                               labels = c("Phon+Sem","Semantic")),
+                               Choice = factor(ChoseThePrime,
+                               levels = unique(ChoseThePrime),
+                               labels = c("Prime Chosen","PrimeNotChosen")),
+                               Age = factor(AgeGroup, levels = unique(AgeGroup),
+                               labels = c("Young", "Old")))%>%
+ ggplot(aes(x = PrimeCondition, y = Proportion,
+                               fill = Choice, group = Choice))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color = "black")+
+   geom_errorbar(aes(ymin=Proportion - ci, ymax=Proportion + ci),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   facet_wrap(~Age)+
+   theme_few()+
+   scale_fill_manual(values = c("white", "black"))+
+   xlab("Prime Given") + ylab("Mean Proportion of Errors") +
+   ggtitle("Experiment 1: Multiple-Choice Errors") +
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_blank(),
+         plot.title = element_text(face = "bold", size = rel(1.5), hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
>
> ## Stored and formatted in excel file: JuliePaperTables.xlsx

```

Experiment 1: Multiple-Choice Errors



Experiment 2

State data

```
> exp1_fig_state = Rmisc::summarySE(exp2_state_prime,
+                                   measurevar = "Trials",
+                                   groupvars = c("AgeGroup", "PrimeCondition", "State"))
> library(ggplot2)
> library(ggthemes)
> state_2 = exp1_fig_state %>% mutate(PrimeType = factor(PrimeCondition,
+                                                         levels = unique(PrimeCondition)),
```

```

+           labels = c("Both", "Phonological",
+                       "Semantic", "Unrelated")),
+           RetrievalState = factor(State, levels = unique(State),
+                                   labels = c("Dont Know", "Know", "Other", "TOT")))%>%
+ ggplot(aes(x = PrimeType, y = Trials,
+            fill = RetrievalState, group = RetrievalState))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   facet_wrap(~AgeGroup)+
+   theme_few()+
+   scale_fill_colorblind()+
+   xlab("") + ylab("Mean Number of Trials") +
+   ggtitle("E2: Young and Old Adults (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))

```

State by Prime

```

> exp2_fig_state_prime = Rmisc::summarySE(exp2_state_prime,
+    measurevar = "Trials",
+    groupvars = c("PrimeCondition", "State"))
> library(ggplot2)
> library(ggthemes)
> state_2_prime = exp2_fig_state_prime %>% mutate(PrimeType =
+    factor(PrimeCondition,
+          levels = unique(PrimeCondition)),
+    labels = c("Both", "Phonological",
+              "Semantic", "Unrelated")),
+    RetrievalState = factor(State,
+                            levels = unique(State),
+                            labels = c("Dont Know", "Know", "Other", "TOT")))%>%
+ ggplot(aes(x = PrimeType, y = Trials,
+            group = RetrievalState, fill = RetrievalState))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_colorblind()+
+   xlab("") + ylab("Mean Number of Trials") +
+   ggtitle("E2: Young and Old Adults (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),

```

```
+      axis.title = element_text(face = "bold", size = rel(1)),
+      legend.title = element_text(face = "bold", size = rel(1)),
+      plot.title = element_text(hjust = .5),
+      strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

State ONLY

```
> exp2_fig_state_only = Rmisc::summarySE(exp2_state,
+      measurevar = "Trials",
+      groupvars = c("AgeGroup", "State"))
> exp2_fig_state_only = arrange(exp2_fig_state_only,
+      desc(AgeGroup))
> library(ggplot2)
> library(ggthemes)
> state_2_only = exp2_fig_state_only %>% mutate(RetrievalState = factor(State,
+      levels = unique(State),
+      labels = c("Dont Know", "Know", "Other", "TOT")),
+      Age = factor(AgeGroup, levels = unique(AgeGroup),
+      labels = c("Young", "Old")))%>%
+ ggplot(aes(x = RetrievalState, y = Trials,
+      group = Age, fill = Age))+
+ geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+ geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+      width=.2, color = "gray26",
+      position = position_dodge(0.7))+
+ theme_few()+
+ scale_fill_manual(values = c("royalblue4", "slategray1"))+
+ xlab("") + ylab("Mean Number of Trials") +
+ ggtitle("E2: Young and Old Adults (With Instructions)") +
+ ggtitle("E2: Young vs. Old (With Instructions)") +
+ theme(axis.text = element_text(size = rel(1)),
+      axis.title = element_text(face = "bold", size = rel(1)),
+      legend.title = element_text(face = "bold", size = rel(1)),
+      plot.title = element_text(hjust = .5),
+      strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

State by Prime

```
> exp2_fig_state_prime = Rmisc::summarySE(exp2_state_prime,
+      measurevar = "Trials",
+      groupvars = c("PrimeCondition", "State"))
> library(ggplot2)
> library(ggthemes)
> state_2_prime = exp2_fig_state_prime %>% mutate(PrimeType =
+      factor(PrimeCondition,
+      levels = unique(PrimeCondition)),
```

```

+           labels = c("Both", "Phonological",
+                     "Semantic", "Unrelated")),
+           RetrievalState = factor(State,
+                                   levels = unique(State),
+                                   labels = c("Dont Know", "Know", "Other", "TOT")))%>%
+ ggplot(aes(x = PrimeType, y = Trials,
+           group = RetrievalState, fill = RetrievalState))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_colorblind()+
+   xlab("") + ylab("Mean Number of Trials") +
+   ggtitle("E2: Young vs. Old (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))

```

State ONLY

```

> exp2_fig_state_only = Rmisc::summarySE(exp2_state,
+   measurevar = "Trials",
+   groupvars = c("AgeGroup", "State"))
> exp2_fig_state_only = arrange(exp2_fig_state_only,
+   desc(AgeGroup))
> library(ggplot2)
> library(ggthemes)
> state_2_only = exp2_fig_state_only %>% mutate(RetrievalState = factor(State,
+   levels = unique(State),
+   labels = c("Dont Know", "Know", "Other", "TOT")),
+   Age = factor(AgeGroup, levels = unique(AgeGroup),
+   labels = c("Young", "Old")))%>%
+ ggplot(aes(x = RetrievalState, y = Trials,
+   group = Age, fill = Age))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("") +
+   ggtitle("E2: Young vs. Old (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),

```

```
+       legend.title = element_text(face = "bold", size = rel(1)),
+       plot.title = element_text(hjust = .5),
+       strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

Target Accuracy

```
> exp2_fig_target = Rmisc::summarySE(exp2_target,
+       measurevar = "Accuracy",
+       groupvars = c("AgeGroup", "PrimeCondition"))
> exp2_fig_target = arrange(exp2_fig_target, desc(AgeGroup))
> library(ggplot2)
> library(ggthemes)
> targetacc_2 = exp2_fig_target %>% mutate(PrimeType = factor(PrimeCondition,
+       levels = unique(PrimeCondition)),
+       labels = c("Both", "Phonological",
+       "Semantic", "Unrelated")),
+       Age = factor(AgeGroup, levels = unique(AgeGroup),
+       labels = c("Young", "Old")))%>%
+ ggplot(aes(x = PrimeType, y = Accuracy,
+       fill = Age, group = Age))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color = "black")+
+   geom_errorbar(aes(ymin=Accuracy - se, ymax=Accuracy + se),
+       width=.2, color = "gray26",
+       position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   xlab("") + ylab("Mean Target Accuracy") +
+   ggtitle("Young and Old Adults (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+       axis.title = element_text(face = "bold", size = rel(1)),
+       legend.title = element_text(face = "bold", size = rel(1.2)),
+       plot.title = element_text(hjust = .5),
+       legend.text = element_text(face = "bold", size = rel(1.1)),
+       axis.text.x = element_text(face = "bold", size = rel(1.2)),
+       strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

MCQ Table

```
> e2_mcq_agg = Rmisc::summarySE(exp2_mcq,
+       measurevar = "Proportion",
+       groupvars = c("AgeGroup", "PrimeType", "ChosenPrime"))
> e2_mcq_main = e2_mcq_agg %>% filter(PrimeType %in% c("b", "r") &
+       ChosenPrime %in% c("b", "r"))
> e2_mcq_main$ChoseThePrime = c("1_Yes", "2_No", "2_No", "1_Yes",
+       "1_Yes", "2_No", "2_No", "1_Yes")
> e2_mcq_main = arrange(e2_mcq_main, desc(AgeGroup))
```

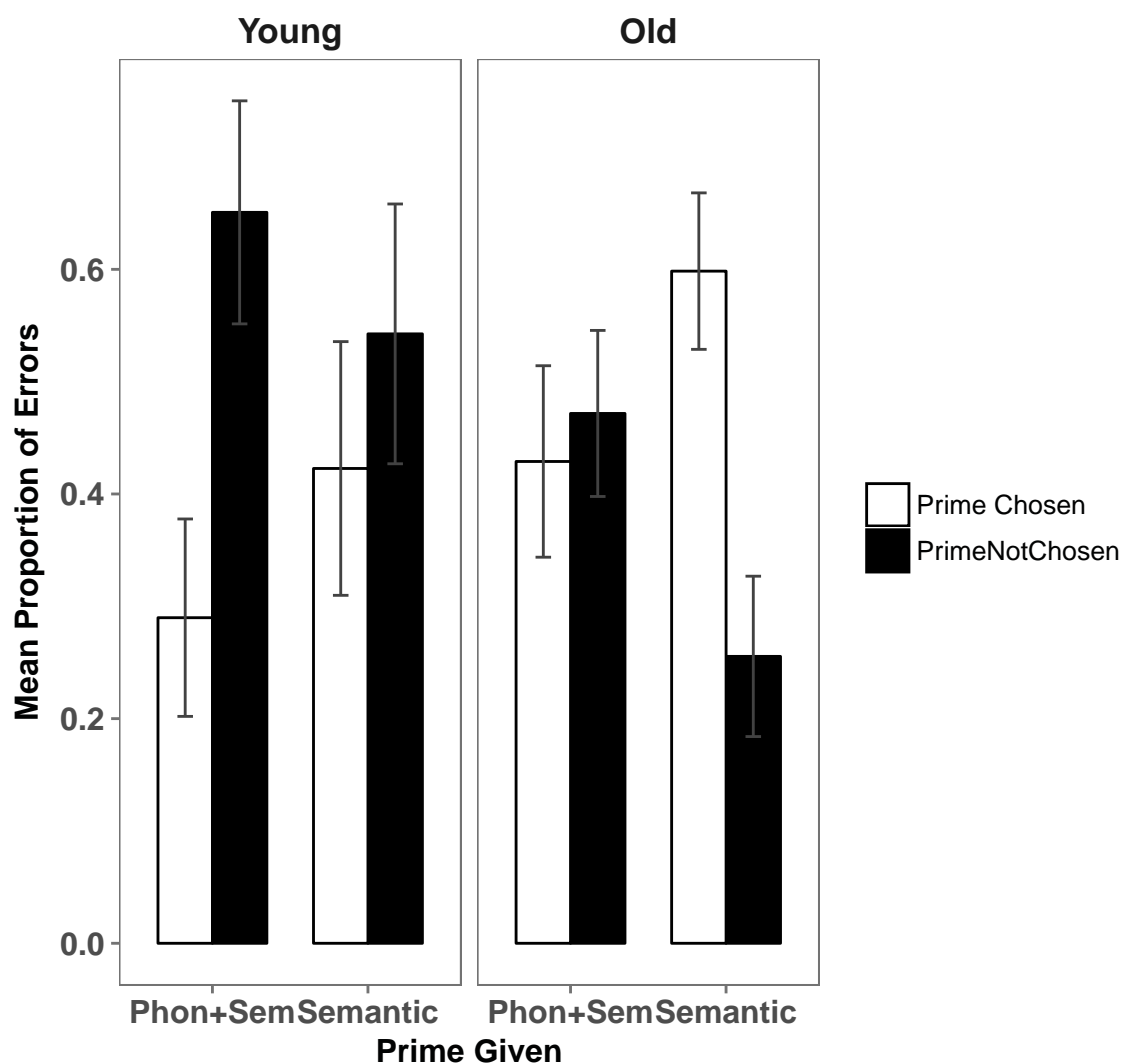


```

> library(ggplot2)
> library(ggthemes)
> e2_mcq_main %>% mutate(PrimeCondition = factor(PrimeType,
+                                           levels = unique(PrimeType),
+                                           labels = c("Phon+Sem","Semantic")),
+                       Choice = factor(ChoseThePrime,
+                                       levels = unique(ChoseThePrime),
+                                       labels = c("Prime Chosen","PrimeNotChosen")),
+                       Age = factor(AgeGroup, levels = unique(AgeGroup),
+                                   labels = c("Young", "Old")))%>%
+ ggplot(aes(x = PrimeCondition, y = Proportion,
+           fill = Choice, group = Choice))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color = "black")+
+   geom_errorbar(aes(ymin=Proportion - ci, ymax=Proportion + ci),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   facet_wrap(~Age)+
+   theme_few()+
+   scale_fill_manual(values = c("white", "black"))+
+   xlab("Prime Given") + ylab("Mean Proportion of Errors") +
+   ggtitle("Experiment 2: Multiple-Choice Errors") +
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_blank(),
+         plot.title = element_text(face = "bold", size = rel(1.5), hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
>
> ## Stored and formatted in excel file: JuliePaperTables.xlsx
>

```

Experiment 2: Multiple-Choice Errors



Experiment 3

State data

```
> exp3_fig_state = Rmisc::summarySE(exp3_state_prime,
+                                   measurevar = "Trials",
+                                   groupvars = c("AgeGroup", "PrimeCondition", "State"))
> library(ggplot2)
> library(ggthemes)
> state_3 = exp3_fig_state %>% mutate(PrimeType = factor(PrimeCondition,
+                                                         levels = unique(PrimeCondition)),
```

```

+         labels = c("Both", "Phonological",
+                     "Semantic", "Unrelated")),
+         RetrievalState = factor(State, levels = unique(State),
+                                 labels = c("Dont Know", "Know", "Other", "TOT")))%>%
+ ggplot(aes(x = PrimeType, y = Trials,
+             fill = RetrievalState, group = RetrievalState))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_colorblind()+
+   xlab("Prime Condition") + ylab("") +
+   ggtitle("E3: Young Adults (Threshold Priming: 48 ms)") +
+   ggtitle("E3: Young (Threshold Priming: 48 ms)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))

```

State by Prime

```

> exp3_fig_state_prime = Rmisc::summarySE(exp3_state_prime,
+     measurevar = "Trials",
+     groupvars = c("PrimeCondition","State"))
> library(ggplot2)
> library(ggthemes)
> state_3_prime = exp3_fig_state_prime %>% mutate(PrimeType =
+     factor(PrimeCondition,
+           levels = unique(PrimeCondition)),
+     labels = c("Both", "Phonological",
+               "Semantic", "Unrelated")),
+     RetrievalState = factor(State,
+                             levels = unique(State),
+                             labels = c("Dont Know", "Know", "Other", "TOT")))%>%
+ ggplot(aes(x = PrimeType, y = Trials,
+             group = RetrievalState, fill = RetrievalState))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_colorblind()+
+   xlab("Prime Condition") + ylab("") +
+   ggtitle("E3: Young Adults (Threshold Priming: 48 ms)") +
+   theme(axis.text = element_text(size = rel(1)),

```

```
+      axis.title = element_text(face = "bold", size = rel(1)),
+      legend.title = element_text(face = "bold", size = rel(1)),
+      plot.title = element_text(hjust = .5),
+      strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

State ONLY

```
> exp3_fig_state_only = Rmisc::summarySE(exp3_state,
+      measurevar = "Trials",
+      groupvars = c("AgeGroup", "State"))
> library(ggplot2)
> library(ggthemes)
> state_3_only = exp3_fig_state_only %>% mutate(RetrievalState = factor(State,
+      levels = unique(State),
+      labels = c("Dont Know", "Know", "Other", "TOT")),
+      Age = factor(AgeGroup, levels = unique(AgeGroup),
+      labels = c("Young")))%>%
+ ggplot(aes(x = RetrievalState, y = Trials,
+      fill = Age, group = Age))+
+ geom_bar(stat = "identity", position = "dodge", width = 0.7,
+      color= "black")+
+ geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+      width=.2, color = "gray26",
+      position = position_dodge(0.7))+
+ scale_fill_manual(values = c("royalblue4"))+
+ theme_few()+
+ xlab("") + ylab("") +
+ ggtitle("E3: Young Adults (Threshold Priming: 48 ms)") +
+ ggtitle("E3: Young (Threshold Priming: 48 ms)") +
+ theme(axis.text = element_text(size = rel(1)),
+      axis.title = element_text(face = "bold", size = rel(1)),
+      legend.title = element_text(face = "bold", size = rel(1)),
+      plot.title = element_text(hjust = .5),
+      strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

State ONLY

```
> exp3_fig_state_only = Rmisc::summarySE(exp3_state,
+      measurevar = "Trials",
+      groupvars = c("AgeGroup", "State"))
> library(ggplot2)
> library(ggthemes)
> state_3_only = exp3_fig_state_only %>% mutate(RetrievalState = factor(State,
+      levels = unique(State),
+      labels = c("Dont Know", "Know", "Other", "TOT")),
+      Age = factor(AgeGroup, levels = unique(AgeGroup),
```

```

+           labels = c("Young")))%>%
+ ggplot(aes(x = RetrievalState, y = Trials,
+           fill = Age, group = Age))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7,
+           color = "black")+
+   geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   scale_fill_manual(values = c("royalblue4"))+
+   theme_few()+
+   xlab("") + ylab("") +
+   ggtitle("E3: Young (48 ms)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))

```

Target Accuracy

```

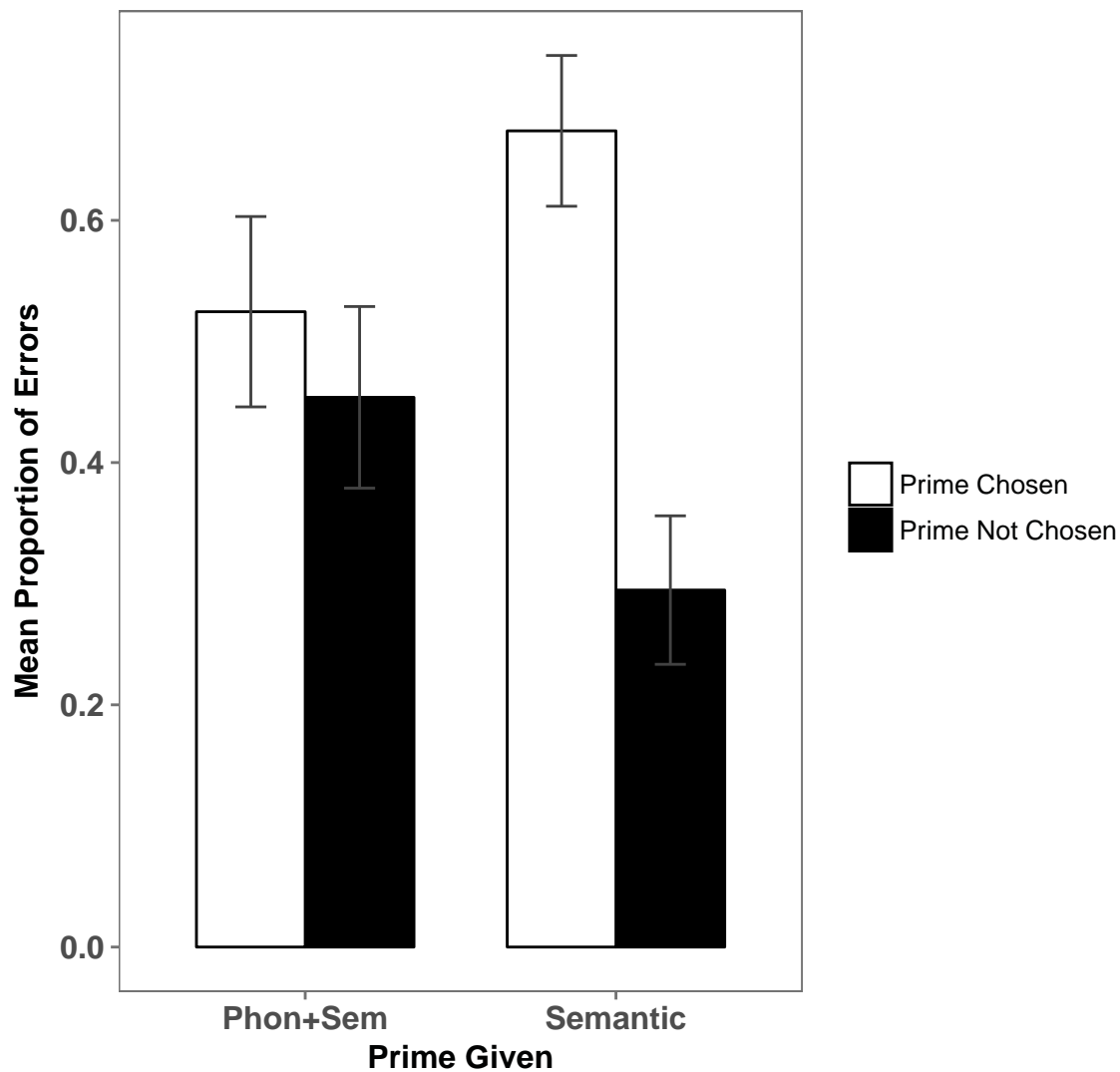
> exp3_fig_target = Rmisc::summarySE(exp3_target,
+                                   measurevar = "Accuracy",
+                                   groupvars = c("AgeGroup", "PrimeCondition"))
> library(ggplot2)
> library(ggthemes)
> targetacc_3 = exp3_fig_target %>% mutate(PrimeType = factor(PrimeCondition,
+                                   levels = unique(PrimeCondition)),
+                                   labels = c("Both", "Phonological",
+                                   "Semantic", "Unrelated")),
+                                   Age = factor(AgeGroup, levels = unique(AgeGroup),
+                                   labels = c("Young")))%>%
+ ggplot(aes(x = PrimeType, y = Accuracy, fill = Age, group = Age))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color = "black")+
+   geom_errorbar(aes(ymin=Accuracy - se, ymax=Accuracy + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_manual(values = c("royalblue4", "slategray1"))+
+   #scale_fill_manual(values = c("darkred", "forestgreen"))+
+   xlab("Prime Condition") + ylab("") +
+   ggtitle("E3: Young Adults (Threshold Priming: 48 ms)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))

```

MCQ Table

```
> e3_mcqoverall_agg = group_by(exp3_mcq, AgeGroup, PrimeType, ChosenPrime)%>%
+   summarise_at(vars(MCQAcc), mean)
> ## Plotting
>
> e3_mcq_agg = Rmisc::summarySE(exp3_mcq,
+                               measurevar = "Proportion",
+                               groupvars = c("PrimeType", "ChosenPrime"))
> e3_mcq_main = e3_mcq_agg %>% filter(PrimeType %in% c("b", "r") &
+                               ChosenPrime %in% c("b", "r"))
> e3_mcq_main$ChoseThePrime = c("1_Yes", "2_No", "2_No", "1_Yes")
> library(ggplot2)
> library(ggthemes)
> e3_mcq_main %>% mutate(PrimeCondition = factor(PrimeType,
+                                               levels = unique(PrimeType),
+                                               labels = c("Phon+Sem","Semantic")),
+                       Choice = factor(ChoseThePrime,
+                                       levels = unique(ChoseThePrime),
+                                       labels = c("Prime Chosen","Prime Not Chosen")))%>%
+ ggplot(aes(x = PrimeCondition, y = Proportion,
+           fill = Choice, group = Choice))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color = "black")+
+   geom_errorbar(aes(ymin=Proportion - ci, ymax=Proportion + ci),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   scale_fill_manual(values = c("white", "black"))+
+   xlab("Prime Given") + ylab("Mean Proportion of Errors") +
+   ggtitle("Experiment 3: Multiple-Choice Errors") +
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_blank(),
+         plot.title = element_text(face = "bold", size = rel(1.5), hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
>
> ## Stored and formatted in excel file: JuliePaperTables.xlsx
>
```

Experiment 3: Multiple-Choice Errors



E3 and E2 compare

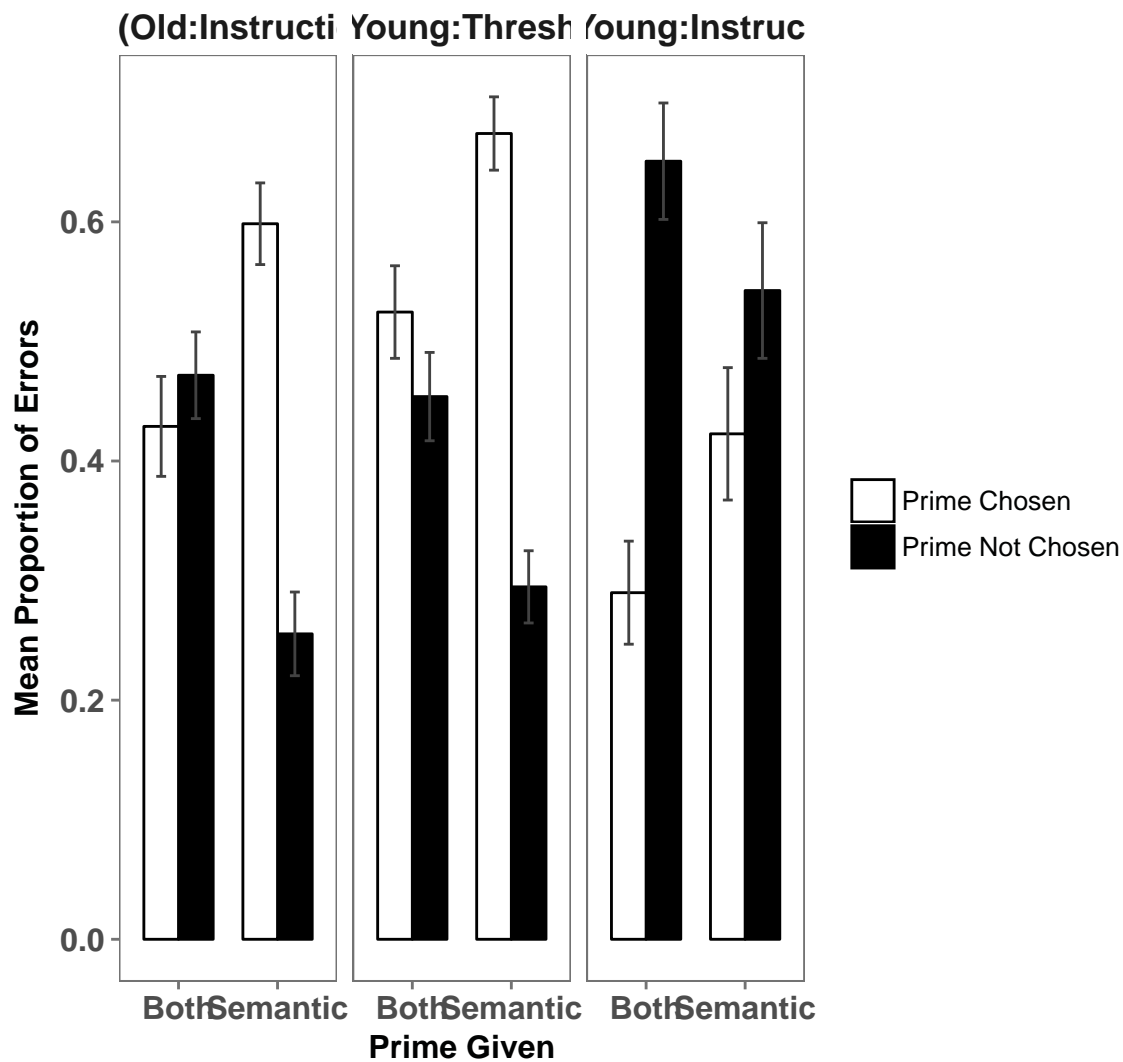
```
> exp3_fig_compare_1 = Rmisc::summarySE(exp3_compare_1,
+                                     measurevar = "Proportion",
+                                     groupvars = c("StudyNo", "PrimeType", "ChosenPrime"))
> exp3_fig_compare_2 = Rmisc::summarySE(exp3_compare_2,
+                                     measurevar = "Proportion",
+                                     groupvars = c("StudyNo", "PrimeType", "ChosenPrime"))
> e3_main_1 = exp3_fig_compare_1 %>% filter(PrimeType %in% c("b", "r") &
+                                     ChosenPrime %in% c("b", "r"))
```

```

> e3_main_2 = exp3_fig_compare_2 %>% filter(PrimeType %in% c("b", "r") &
+                                           ChosenPrime %in% c("b", "r"))
> exp3_mainfig = full_join(e3_main_1, e3_main_2)
> exp3_mainfig$ChoseThePrime = c("1_Yes", "2_No", "2_No", "1_Yes",
+                                "1_Yes", "2_No", "2_No", "1_Yes",
+                                "1_Yes", "2_No", "2_No", "1_Yes")
> exp3_mainfig5 = exp3_mainfig %>% filter(StudyNo== "5")
> exp3_mainfig1 = exp3_mainfig %>% filter(StudyNo == "1")
> exp3_mainfig6 = exp3_mainfig %>% filter(StudyNo == "6")
> final_mainfig = rbind(exp3_mainfig6, exp3_mainfig1, exp3_mainfig5)
> library(ggplot2)
> library(ggthemes)
> final_mainfig %>% mutate(PrimeCondition = factor(PrimeType,
+                                                  levels = unique(PrimeType),
+                                                  labels = c("Both","Semantic")),
+                          ChosenPrime = factor(ChosenPrime,
+                                                  levels = unique(PrimeType),
+                                                  labels = c("Both","Semantic")),
+                          Experiment = factor(StudyNo,
+                                                  levels = unique(StudyNo),
+                                                  labels = c("E2 (Old:Instruction)","E3 (Young:Threshold)",
+                                                            "E2 (Young:Instruction)")),
+                          Choice = factor(ChoseThePrime,
+                                                  levels = unique(ChoseThePrime),
+                                                  labels = c("Prime Chosen","Prime Not Chosen")))%>%
+ ggplot(aes(x = PrimeCondition, y = Proportion,
+            fill = Choice, group = Choice))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7, color = "black")+
+   geom_errorbar(aes(ymin=Proportion - se, ymax=Proportion + se),
+                 width=.2, color = "gray26",
+                 position = position_dodge(0.7))+
+   facet_wrap(~Experiment)+
+   theme_few()+
+   scale_fill_manual(values = c("white", "black"))+
+   xlab("Prime Given") + ylab("Mean Proportion of Errors") +
+   ggtitle("Experiment 2 vs 3: Multiple-Choice Errors") +
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_blank(),
+         plot.title = element_text(face = "bold", size = rel(1.5), hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))

```


Experiment 2 vs 3: Multiple-Choice Errors

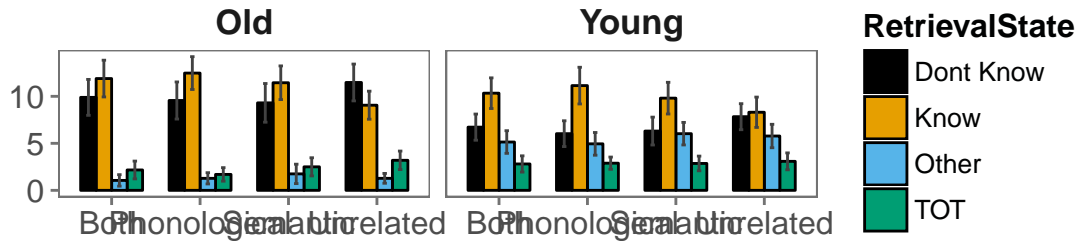


Combined State Data

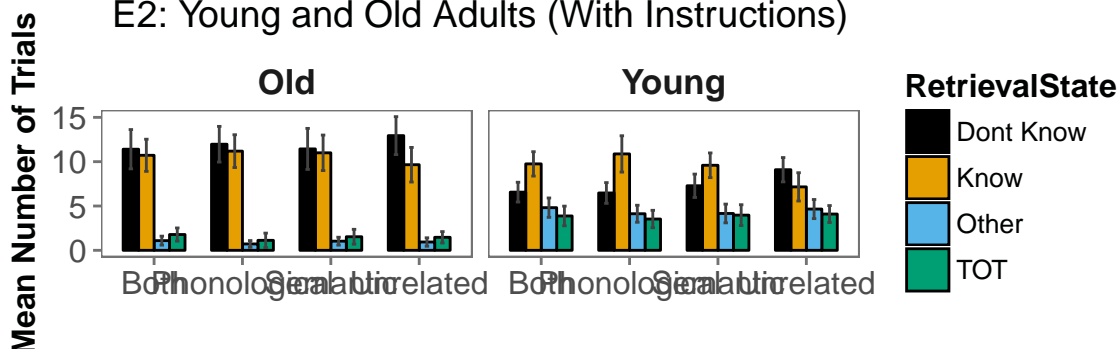
```
> library(grid)
> gridExtra::grid.arrange(state_1, state_2, state_3, nrow = 3, ncol = 1,
+                           top=textGrob("Retrieval States Across Experiments 1, E2, E3",
+                                         gp=gpar(fontsize=20)))
```

Retrieval States Across Experiments 1, E2, E3

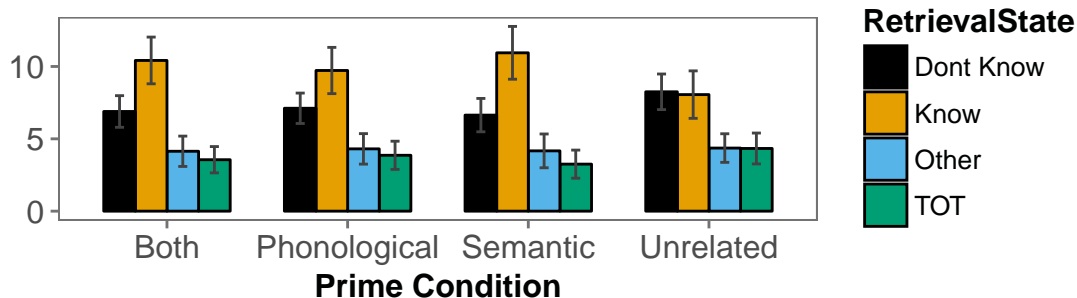
E1: Young vs. Old (Without Instructions)



E2: Young and Old Adults (With Instructions)



E3: Young (Threshold Priming: 48 ms)

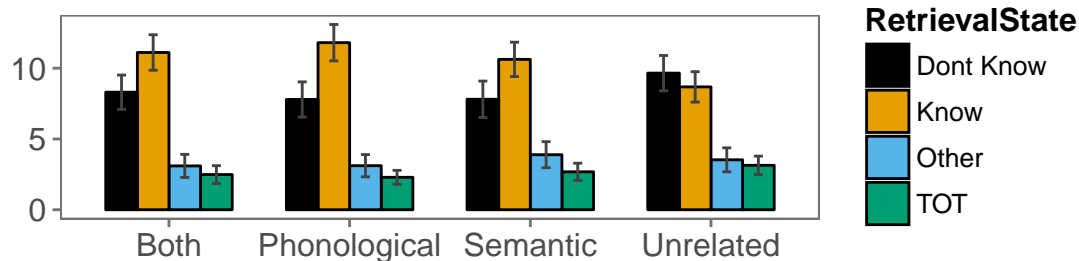


Combined State Prime Data

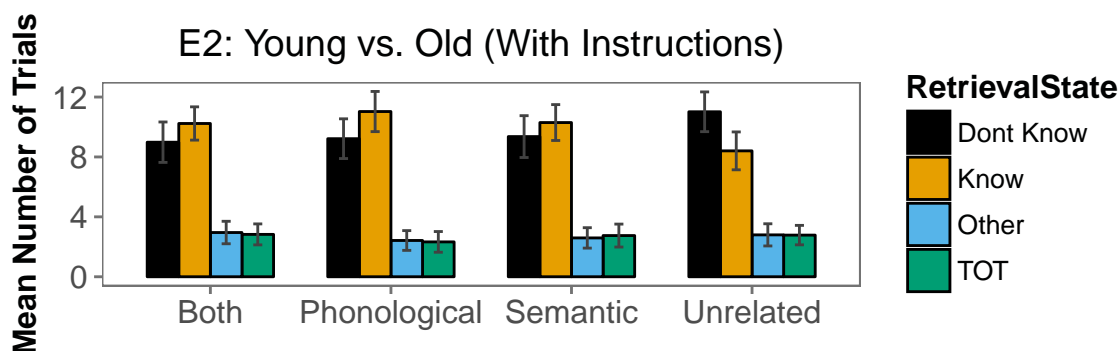
```
> library(grid)
> gridExtra::grid.arrange(state_1_prime, state_2_prime, state_3_prime,
+                           nrow = 3, ncol = 1,
+                           top=textGrob("Retrieval States Across Experiments E1, E2, E3",
+                                       gp=gpar(fontsize=20)))
```

Retrieval States Across Experiments E1, E2, E3

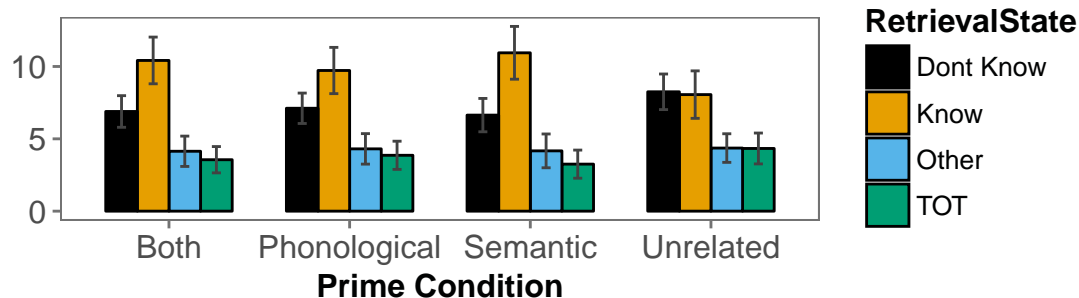
E1: Young and Old Adults (Without Instructions)



E2: Young vs. Old (With Instructions)



E3: Young Adults (Threshold Priming: 48 ms)

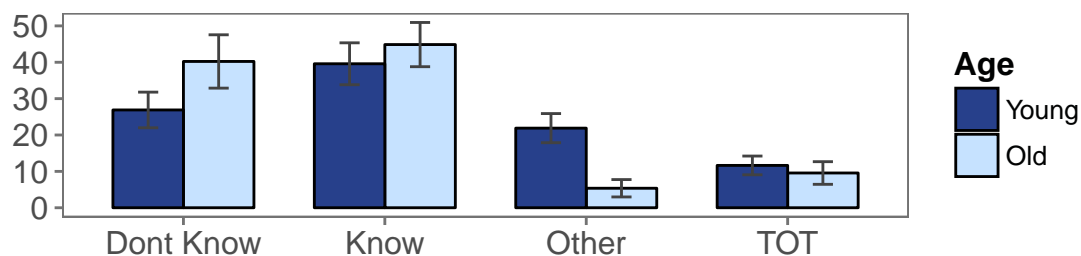


Combined State ONLY Data

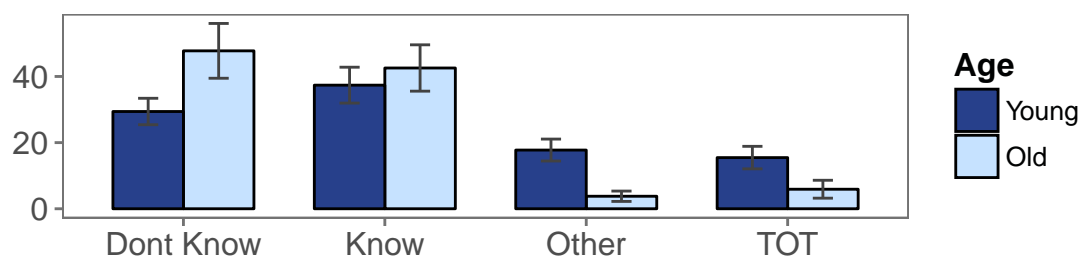
```
> library(grid)
> gridExtra::grid.arrange(state_1_only, state_2_only, state_3_only,
+                           nrow = 3, ncol = 1,
+                           top=textGrob("Retrieval States Across Experiments E1, E2, E3",
+                                       gp=gpar(fontsize=20)))
```

Retrieval States Across Experiments E1, E2, E3

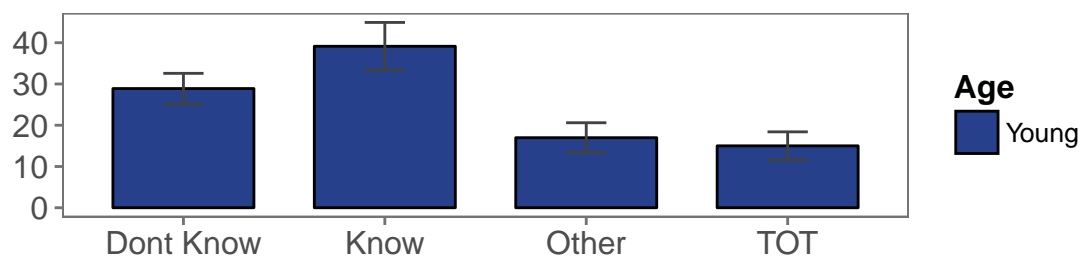
E1: Young vs. Old (Without Instructions)



E2: Young vs. Old (With Instructions)



E3: Young (48 ms)

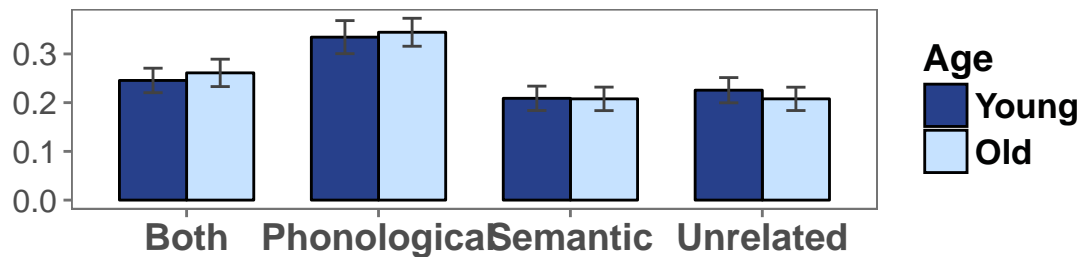


Combined Target Accuracy Data

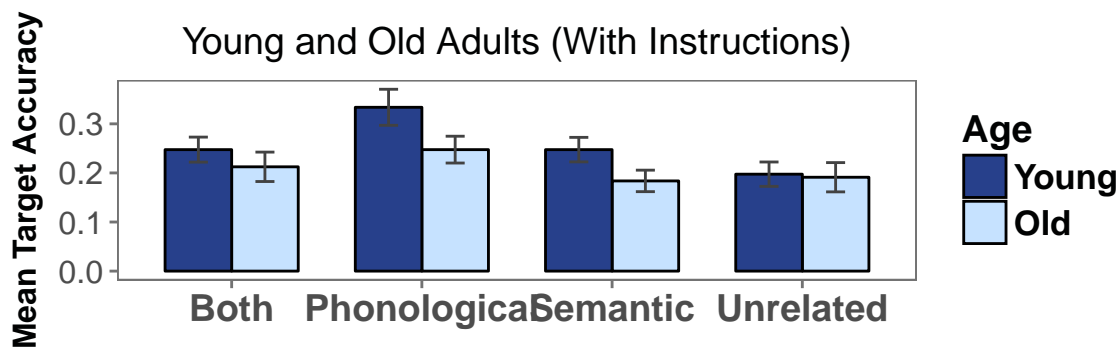
```
> library(grid)
> gridExtra::grid.arrange(targetacc_1, targetacc_2, targetacc_3, nrow = 3, ncol = 1,
+                           top=textGrob("Target Accuracy Across Experiments 1, E2, E3",
+                                         gp=gpar(fontsize=20)))
```

Target Accuracy Across Experiments 1, E2, E3

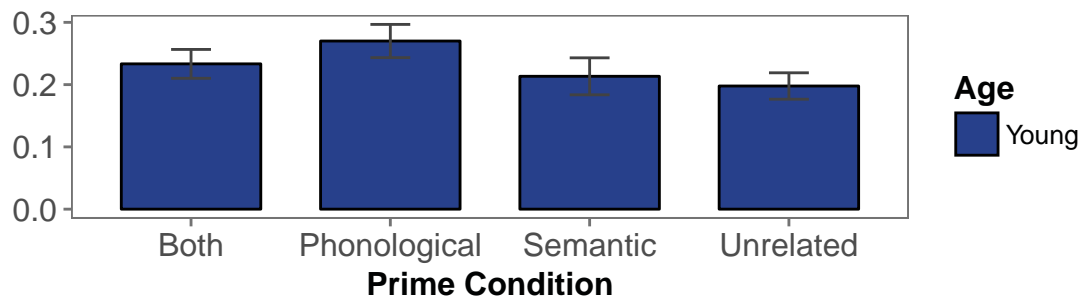
Young and Old Adults (No Instructions)



Young and Old Adults (With Instructions)



E3: Young Adults (Threshold Priming: 48 ms)

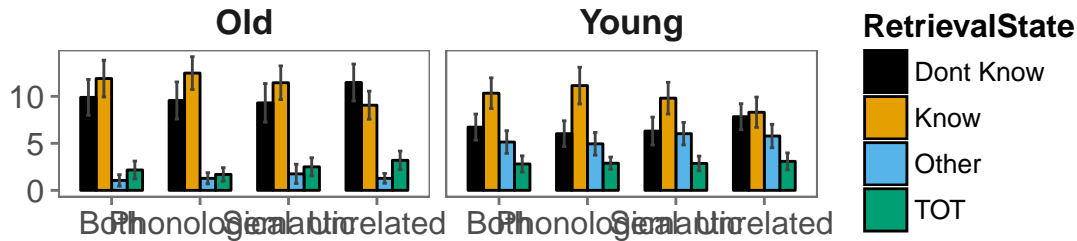


20 HLM Approaches

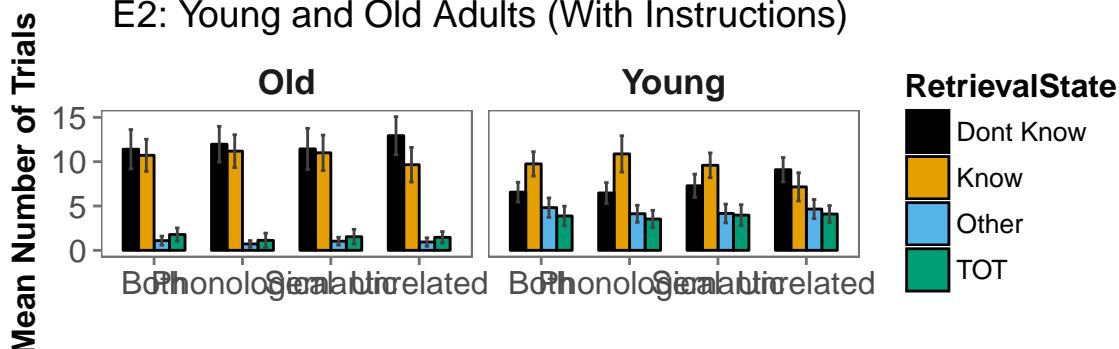
```
> library(grid)
> gridExtra::grid.arrange(state_1, state_2, state_3, nrow = 3, ncol = 1,
+                           top=textGrob("Retrieval States Across Experiments 1, 2 and 3",
+                                       gp=gpar(fontsize=20)))
```

Retrieval States Across Experiments 1, 2 and 3

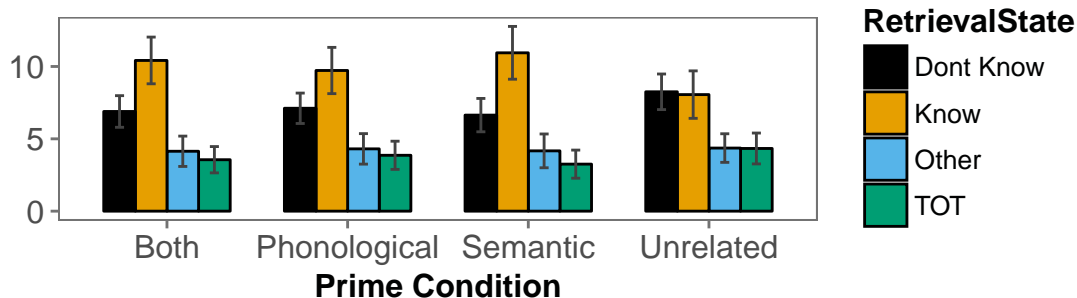
E1: Young vs. Old (Without Instructions)



E2: Young and Old Adults (With Instructions)



E3: Young (Threshold Priming: 48 ms)

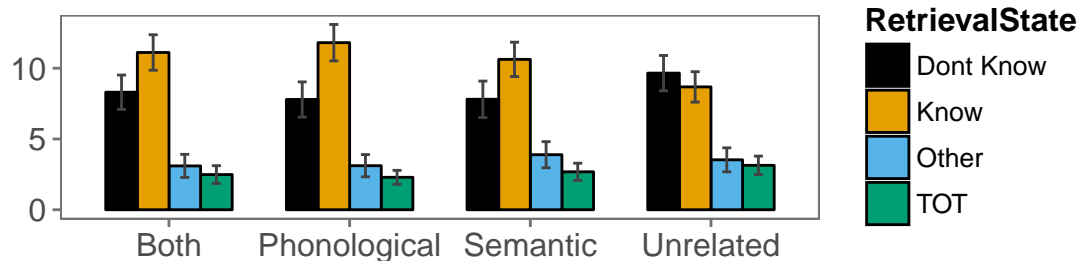


Combined State Prime Data

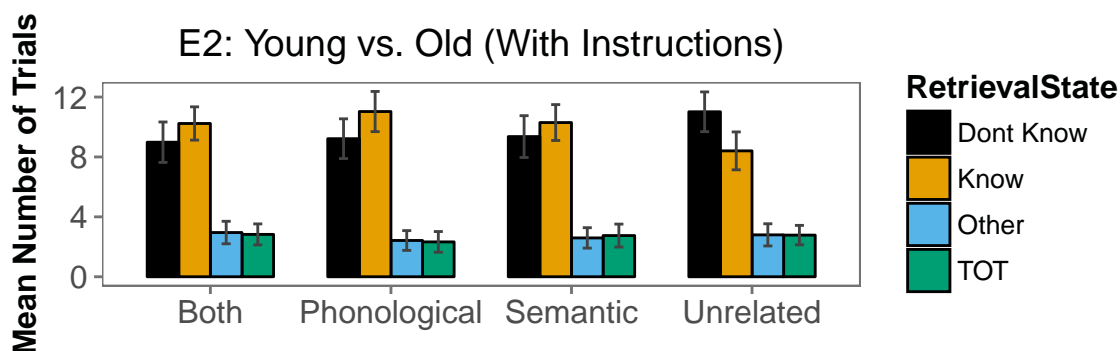
```
> library(grid)
> gridExtra::grid.arrange(state_1_prime, state_2_prime, state_3_prime,
+                           nrow = 3, ncol = 1,
+                           top=textGrob("Retrieval States Across Experiments E1, E2, E3",
+                                       gp=gpar(fontsize=20)))
```

Retrieval States Across Experiments E1, E2, E3

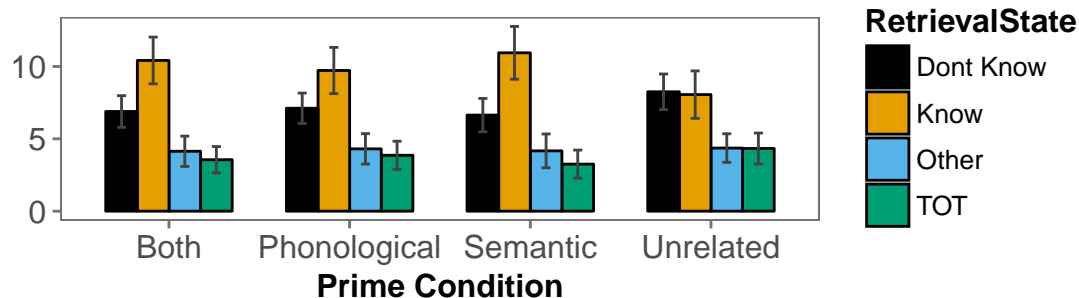
E1: Young and Old Adults (Without Instructions)



E2: Young vs. Old (With Instructions)



E3: Young Adults (Threshold Priming: 48 ms)

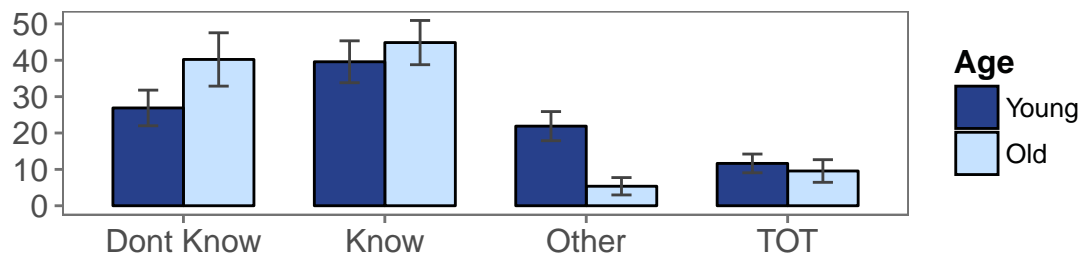


Combined State ONLY Data

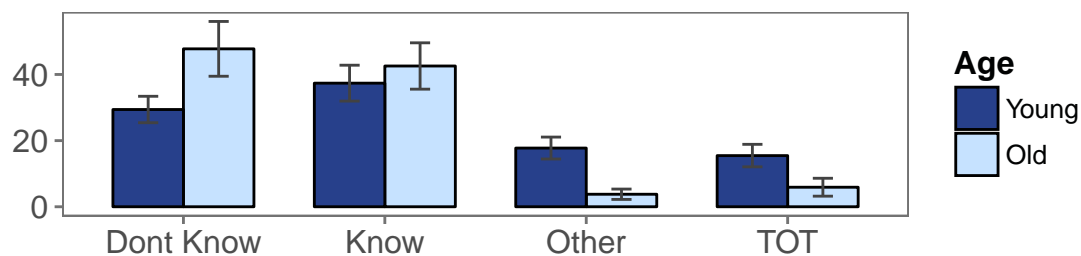
```
> library(grid)
> gridExtra::grid.arrange(state_1_only, state_2_only, state_3_only,
+                           nrow = 3, ncol = 1,
+                           top=textGrob("Retrieval States Across Experiments E1, E2, E3",
+                                       gp=gpar(fontsize=20)))
```

Retrieval States Across Experiments E1, E2, E3

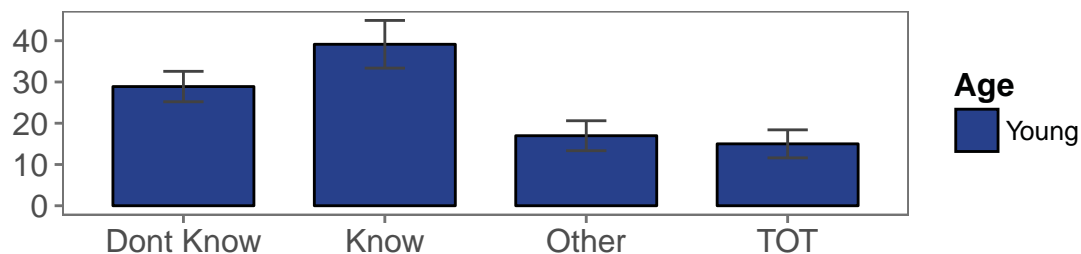
E1: Young vs. Old (Without Instructions)



E2: Young vs. Old (With Instructions)



E3: Young (48 ms)

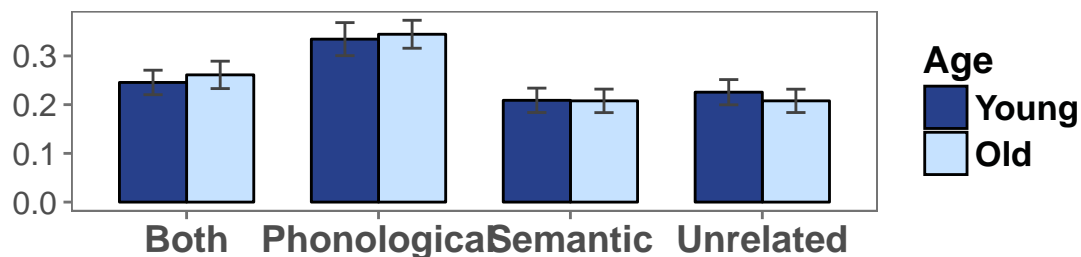


Combined Target Accuracy Data

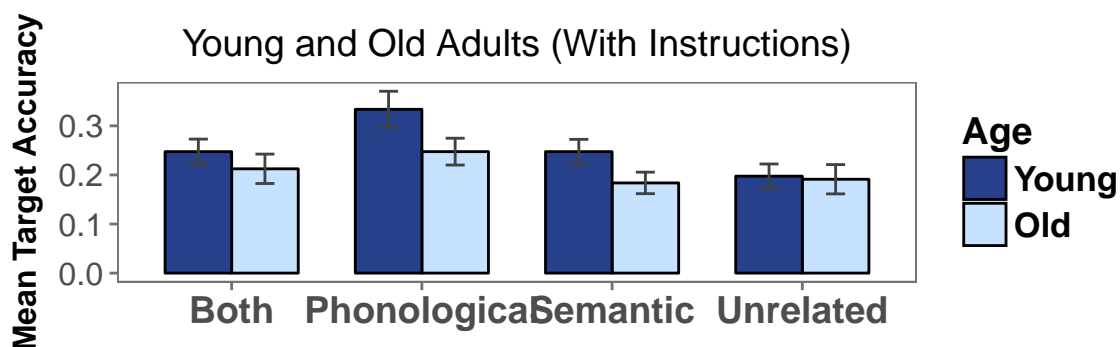
```
> library(grid)
> gridExtra::grid.arrange(targetacc_1, targetacc_2, targetacc_3, nrow = 3, ncol = 1,
+                           top=textGrob("Target Accuracy Across Experiments E1, E2, E3",
+                                         gp=gpar(fontsize=20)))
```


Target Accuracy Across Experiments E1, E2, E3

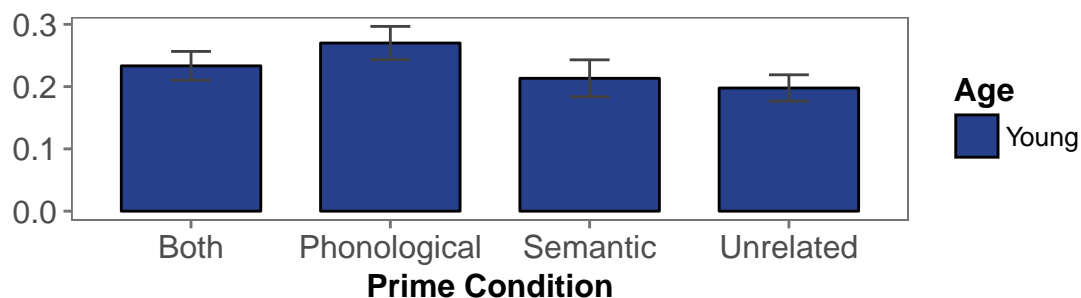
Young and Old Adults (No Instructions)



Young and Old Adults (With Instructions)



E3: Young Adults (Threshold Priming: 48 ms)



21 M Turk Rating Data

Calculating item level accuracies

```
> itemratings= read.csv("item_ratings_wide.csv",
+                       header = TRUE, sep = ",")
> main = read.csv("Julie_Main5Studies.csv", header = TRUE, sep = ",")
> main_item = merge(main, itemratings, by = "Target")
> main_item = dplyr::arrange(main_item, StudyNo, Subject, TargetNo, PrimeType)
> ## but we also need item-level accuracy data
```

```

> library(dplyr)
> item_acc = group_by(main_item, TargetNo) %>%
+   summarise_at(vars(Accuracy), mean)
> colnames(item_acc) = c("TargetNo", "ItemAcc")
> main_item = merge(main_item, item_acc, by = c("TargetNo"))
> main_item = dplyr::arrange(main_item, StudyNo, Subject, TargetNo, PrimeType)
> ## Now we run an HLM for each prime condition separately

```

Predicting Accuracy Using Rating

```

> main_item$StudyNo = as.factor(as.character(main_item$StudyNo))
> Phon = main_item %>% filter(PrimeCondition == "P" &
+   PrimeType == "Phonological")
> Sem = main_item %>% filter(PrimeCondition == "R" &
+   PrimeType == "Semantic")
> Both_Phon = main_item %>% filter(PrimeCondition == "B" &
+   PrimeType == "Both")
> Both_Sem = main_item %>% filter(PrimeCondition == "B" &
+   PrimeType == "Both")

```

Models

Models with Only Rating

```

> library(lme4)
> phon_model = glmer(data = Phon, Accuracy ~ SoundRating +
+   (1|Subject), family = "binomial")
> summary(phon_model)

```

```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial (logit)
Formula: Accuracy ~ SoundRating + (1 | Subject)
Data: Phon

      AIC      BIC   logLik deviance df.resid
5066.7   5085.8  -2530.3   5060.7     4347

Scaled residuals:
    Min       1Q   Median       3Q      Max
-1.6039 -0.6436 -0.4663  0.9428  3.0318

Random effects:
Groups   Name              Variance Std.Dev.
Subject (Intercept) 0.7018      0.8377
Number of obs: 4350, groups: Subject, 174

```

```

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.48902     0.24847  -5.993 2.06e-09 ***
SoundRating  0.11782     0.05108   2.307  0.0211 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
              (Intr)
SoundRating -0.955
convergence code: 0
Model failed to converge with max|grad| = 0.00167732 (tol = 0.001, component 1)

```

```

> phon_model_2 = glmer(data = Phon, Accuracy ~ SoundRating +
+                      (1|Subject), family = "binomial")
> summary(phon_model_2)

```

```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial ( logit )
Formula: Accuracy ~ SoundRating + (1 | Subject)
Data: Phon

             AIC      BIC   logLik deviance df.resid
5066.7    5085.8  -2530.3   5060.7     4347

Scaled residuals:
      Min       1Q   Median       3Q      Max
-1.6039 -0.6436 -0.4663  0.9428  3.0318

Random effects:
 Groups Name      Variance Std.Dev.
 Subject (Intercept) 0.7018   0.8377
Number of obs: 4350, groups: Subject, 174

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.48902     0.24847  -5.993 2.06e-09 ***
SoundRating  0.11782     0.05108   2.307  0.0211 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
              (Intr)
SoundRating -0.955
convergence code: 0
Model failed to converge with max|grad| = 0.00167732 (tol = 0.001, component 1)

```

```
> sem_model = glmer(data = Sem, Accuracy ~ MeaningRating*ItemAcc +
+                   (1|Subject), family = "binomial")
> summary(sem_model)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial (logit)
Formula: Accuracy ~ MeaningRating * ItemAcc + (1 | Subject)
Data: Sem

      AIC      BIC   logLik deviance df.resid
 3325.5   3357.4  -1657.8   3315.5     4345

Scaled residuals:
    Min       1Q   Median       3Q      Max
-9.1149 -0.3960 -0.2297 -0.1154  7.5220

Random effects:
 Groups Name      Variance Std.Dev.
 Subject (Intercept) 1.052    1.026
Number of obs: 4350, groups: Subject, 174

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)      -2.3150     0.4821  -4.802 1.57e-06 ***
MeaningRating     -0.2790     0.1019  -2.738 0.006188 **
ItemAcc           4.6065     1.3615   3.383 0.000716 ***
MeaningRating:ItemAcc 0.5063     0.2842   1.781 0.074857 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
          (Intr) MnngRt ItmAcc
MeaningRtng -0.962
ItemAcc      -0.822  0.806
MnngRtng:IA  0.814 -0.837 -0.980
```

```
> both_sem_model = glmer(data = Both_Sem, Accuracy ~ MeaningRating*ItemAcc +
+                   (1|Subject), family = "binomial",
+                   control=glmerControl(optimizer="bobyqa",
+                   optCtrl=list(maxfun=100000)))
> summary(both_sem_model)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial (logit)
Formula: Accuracy ~ MeaningRating * ItemAcc + (1 | Subject)
```

```

Data: Both_Sem
Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+05))

      AIC      BIC    logLik deviance df.resid
3653.2   3685.1  -1821.6   3643.2     4345

Scaled residuals:
    Min       1Q   Median       3Q      Max
-4.5593 -0.4370 -0.2605 -0.1207  6.0380

Random effects:
 Groups Name      Variance Std.Dev.
 Subject (Intercept) 0.89     0.9434
Number of obs: 4350, groups: Subject, 174

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)    -2.86358    0.37537  -7.629 2.37e-14 ***
MeaningRating   -0.08847    0.08385  -1.055  0.291
ItemAcc          7.13656    0.96061   7.429 1.09e-13 ***
MeaningRating:ItemAcc -0.09391    0.20369  -0.461  0.645
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
      (Intr) MnngRt ItmAcc
MeaningRtng -0.947
ItemAcc      -0.788  0.742
MnngRtng:IA  0.791 -0.808 -0.963

```

```

> both_phon_model = glmer(data = Both_Phon, Accuracy ~ SoundRating*ItemAcc +
+                          (1|Subject), family = "binomial",
+                          control=glmerControl(optimizer="bobyqa",
+                          optCtrl=list(maxfun=100000)))
> summary(both_phon_model)

```

```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial (logit)
Formula: Accuracy ~ SoundRating * ItemAcc + (1 | Subject)
Data: Both_Phon
Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+05))

      AIC      BIC    logLik deviance df.resid
3652.7   3684.6  -1821.4   3642.7     4345

Scaled residuals:
    Min       1Q   Median       3Q      Max

```

```

-4.0686 -0.4343 -0.2606 -0.1232  5.5756

Random effects:
  Groups   Name      Variance Std.Dev.
  Subject (Intercept) 0.8916   0.9443
Number of obs: 4350, groups:  Subject, 174

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)    -4.26470    0.44795  -9.520  < 2e-16 ***
SoundRating      0.24028    0.09662   2.487   0.0129 *
ItemAcc          8.52187    1.18812   7.173 7.36e-13 ***
SoundRating:ItemAcc -0.43708    0.26064  -1.677   0.0935 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
              (Intr) SndRtn ItmAcc
SoundRating  -0.964
ItemAcc       -0.846  0.828
SndRtn:ItmAcc  0.822 -0.843 -0.977

```

```

>
>
> ## seems that ratings have an overall effect on accuracy, but not above and beyond the

```

Models with Rating and Average Performance

```

> library(lme4)
> phon_model_2 = glmer(data = Phon, Accuracy ~ SoundRating*ItemAcc +
+                      (1|Subject), family = "binomial")
> summary(phon_model_2)

```

```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial ( logit )
Formula: Accuracy ~ SoundRating * ItemAcc + (1 | Subject)
Data: Phon

      AIC      BIC    logLik deviance df.resid
 4041.0   4072.8  -2015.5   4031.0     4345

Scaled residuals:
    Min       1Q   Median       3Q      Max
-5.8014 -0.4918 -0.2828  0.4313  7.8353

Random effects:

```

```

Groups   Name             Variance Std.Dev.
Subject (Intercept) 1.13      1.063
Number of obs: 4350, groups: Subject, 174

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)    -3.2402    0.4971  -6.518 7.11e-11 ***
SoundRating      0.1017    0.1033   0.985  0.32474
ItemAcc         5.5676    1.4622   3.808  0.00014 ***
SoundRating:ItemAcc 0.2363    0.3094   0.764  0.44507
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
              (Intr) SndRtn ItmAcc
SoundRating  -0.972
ItemAcc       -0.811  0.810
SndRtn:ItmAcc 0.793 -0.817 -0.985

```

```

> sem_model_2 = glmer(data = Sem, Accuracy ~ MeaningRating*ItemAcc +
+                      (1|Subject), family = "binomial")
> summary(sem_model_2)

```

```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial ( logit )
Formula: Accuracy ~ MeaningRating * ItemAcc + (1 | Subject)
Data: Sem

```

AIC	BIC	logLik	deviance	df.resid
3325.5	3357.4	-1657.8	3315.5	4345

```

Scaled residuals:
      Min       1Q   Median       3Q      Max
-9.1149 -0.3960 -0.2297 -0.1154  7.5220

```

```

Random effects:
Groups   Name             Variance Std.Dev.
Subject (Intercept) 1.052      1.026
Number of obs: 4350, groups: Subject, 174

```

```

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)    -2.3150    0.4821  -4.802 1.57e-06 ***
MeaningRating   -0.2790    0.1019  -2.738 0.006188 **
ItemAcc         4.6065    1.3615   3.383 0.000716 ***
MeaningRating:ItemAcc 0.5063    0.2842   1.781 0.074857 .
---

```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Correlation of Fixed Effects:
      (Intr) MnngRt ItmAcc
MeaningRtng -0.962
ItemAcc      -0.822  0.806
MnngRtng:IA  0.814 -0.837 -0.980
```

```
> both_phon_model_2 = glmer(data = Both_Phon, Accuracy ~ SoundRating*ItemAcc +
+                           (1|Subject), family = "binomial")
> summary(both_phon_model_2)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial (logit)
Formula: Accuracy ~ SoundRating * ItemAcc + (1 | Subject)
Data: Both_Phon
```

AIC	BIC	logLik	deviance	df.resid
3652.7	3684.6	-1821.4	3642.7	4345

Scaled residuals:

Min	1Q	Median	3Q	Max
-4.0686	-0.4343	-0.2606	-0.1232	5.5756

Random effects:

Groups	Name	Variance	Std.Dev.
Subject	(Intercept)	0.8916	0.9443

Number of obs: 4350, groups: Subject, 174

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-4.26472	0.44777	-9.524	< 2e-16 ***
SoundRating	0.24028	0.09658	2.488	0.0128 *
ItemAcc	8.52188	1.18756	7.176	7.18e-13 ***
SoundRating:ItemAcc	-0.43708	0.26052	-1.678	0.0934 .

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Correlation of Fixed Effects:
      (Intr) SndRtn ItmAcc
SoundRating -0.964
ItemAcc      -0.846  0.828
SndRtng:ItA  0.822 -0.843 -0.977
```

```
> both_sem_model_2 = glmer(data = Both_Sem, Accuracy ~ MeaningRating*ItemAcc +
+                           (1|Subject), family = "binomial",
+                           control=glmerControl(optimizer="bobyqa"),
```



```
+      optCtrl=list(maxfun=100000)))
> summary(both_sem_model_2)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial (logit)
Formula: Accuracy ~ MeaningRating * ItemAcc + (1 | Subject)
Data: Both_Sem
Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+05))

      AIC      BIC    logLik deviance df.resid
3653.2   3685.1  -1821.6   3643.2     4345

Scaled residuals:
    Min       1Q   Median       3Q      Max
-4.5593 -0.4370 -0.2605 -0.1207  6.0380

Random effects:
 Groups Name      Variance Std.Dev.
Subject (Intercept) 0.89      0.9434
Number of obs: 4350, groups: Subject, 174

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)    -2.86358    0.37537  -7.629 2.37e-14 ***
MeaningRating   -0.08847    0.08385  -1.055  0.291
ItemAcc          7.13656    0.96061   7.429 1.09e-13 ***
MeaningRating:ItemAcc -0.09391    0.20369  -0.461  0.645
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
      (Intr) MnngRt ItmAcc
MeaningRtng -0.947
ItemAcc      -0.788  0.742
MnngRtng:IA  0.791 -0.808 -0.963

>
> ## seems that ratings have an overall effect on accuracy, but not above and beyond the
>
```

Plotting Model Fits: Rating and Mean Accuracy

Phonological

```
> fixed.frame <- Phon %>%
+   dplyr::summarise(mean = mean(ItemAcc, na.rm = T),
```

```

+         sd = sd(ItemAcc, na.rm = T))
> fixed.frame <-
+   data.frame(
+     expand.grid(
+       # here, you add values for your time variable and predictors
+       SoundRating = seq(1,7,1),
+       ItemAcc = c(fixed.frame$mean-fixed.frame$sd,
+                   fixed.frame$mean,
+                   fixed.frame$mean+fixed.frame$sd)))
> fixed.frame$pred = predict(phon_model_2, newdata = fixed.frame, re.form = NA)
> fixed.frame$odds = exp(fixed.frame$pred)
> fixed.frame$prob = fixed.frame$odds/(1+fixed.frame$odds)
> a2 = fixed.frame %>%
+ mutate(ItemAccuracy = factor(ItemAcc, levels = unique(ItemAcc),
+                               labels = c("-1SD", "0SD", "1SD"))) %>%
+   ggplot(aes(x = SoundRating, y = prob, color = ItemAccuracy)) +
+   geom_line(size = 1) +
+   labs(x = "Sound Rating",
+        y = "",
+        title = "Phonological Condition (Sound Rating)") +
+   theme_few()+
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(face = "bold", size = rel(1), hjust = .5))

```

Semantic

```

> fixed.frame <- Phon %>%
+   dplyr::summarise(mean = mean(ItemAcc, na.rm = T),
+                     sd = sd(ItemAcc, na.rm = T))
> fixed.frame <-
+   data.frame(
+     expand.grid(
+       # here, you add values for your time variable and predictors
+       MeaningRating = seq(1,7,1),
+       ItemAcc = c(fixed.frame$mean-fixed.frame$sd,
+                   fixed.frame$mean,
+                   fixed.frame$mean+fixed.frame$sd)))
> fixed.frame$pred = predict(sem_model_2, newdata = fixed.frame, re.form = NA)
> fixed.frame$odds = exp(fixed.frame$pred)
> fixed.frame$prob = fixed.frame$odds/(1+fixed.frame$odds)
> b2 = fixed.frame %>%
+ mutate(ItemAccuracy = factor(ItemAcc, levels = unique(ItemAcc),
+                               labels = c("-1SD", "0SD", "1SD"))) %>%
+   ggplot(aes(x = MeaningRating, y = prob, color = ItemAccuracy)) +
+   geom_line(size = 1) +

```

```
+      labs(x = "Meaning Rating",
+          y = "",
+          title = "Semantic Condition (Meaning Rating)") +
+    theme_few()+
+    theme(axis.text = element_text(face = "bold", size = rel(1)),
+          axis.title = element_text(face = "bold", size = rel(1)),
+          legend.title = element_text(face = "bold", size = rel(1)),
+          plot.title = element_text(face = "bold", size = rel(1), hjust = .5))
```

21.0.1 BothSem

```
> fixed.frame <- Phon %>%
+   dplyr::summarise(mean = mean(ItemAcc, na.rm = T),
+                   sd = sd(ItemAcc, na.rm = T))
> fixed.frame <-
+   data.frame(
+     expand.grid(
+       # here, you add values for your time variable and predictors
+       MeaningRating = seq(1,7,1),
+       ItemAcc = c(fixed.frame$mean-fixed.frame$sd,
+                   fixed.frame$mean,
+                   fixed.frame$mean+fixed.frame$sd)))
> fixed.frame$pred = predict(both_sem_model_2,
+                             newdata = fixed.frame, re.form = NA)
> fixed.frame$odds = exp(fixed.frame$pred)
> fixed.frame$prob = fixed.frame$odds/(1+fixed.frame$odds)
> c2 = fixed.frame %>%
+   mutate(ItemAccuracy = factor(ItemAcc, levels = unique(ItemAcc),
+                                 labels = c("-1SD", "0SD", "1SD"))) %>%
+   ggplot(aes(x = MeaningRating, y = prob, color = ItemAccuracy)) +
+   geom_line(size = 1) +
+     labs(x = "Meaning Rating",
+          y = "",
+          title = "Both Condition (Meaning Rating)") +
+   theme_few()+
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(face = "bold", size = rel(1), hjust = .5))
```

21.0.2 BothPhon

```
> fixed.frame <- Phon %>%
+   dplyr::summarise(mean = mean(ItemAcc, na.rm = T),
+                   sd = sd(ItemAcc, na.rm = T))
> fixed.frame <-
```

```

+   data.frame(
+     expand.grid(
+       # here, you add values for your time variable and predictors
+       SoundRating = seq(1,7,1),
+       ItemAcc = c(fixed.frame$mean-fixed.frame$sd,
+                   fixed.frame$mean,
+                   fixed.frame$mean+fixed.frame$sd)))
> fixed.frame$pred = predict(both_phon_model_2,
+                             newdata = fixed.frame, re.form = NA)
> fixed.frame$odds = exp(fixed.frame$pred)
> fixed.frame$prob = fixed.frame$odds/(1+fixed.frame$odds)
> d2 = fixed.frame %>%
+ mutate(ItemAccuracy = factor(ItemAcc, levels = unique(ItemAcc),
+                               labels = c("-1SD", "OSD", "1SD"))) %>%
+   ggplot(aes(x = SoundRating, y = prob, color = ItemAccuracy)) +
+   geom_line(size = 1) +
+   labs(x = "Sound Rating",
+        y = "",
+        title = "Both Condition (Sound Rating)") +
+   theme_few()+
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(face = "bold", size = rel(1), hjust = .5))

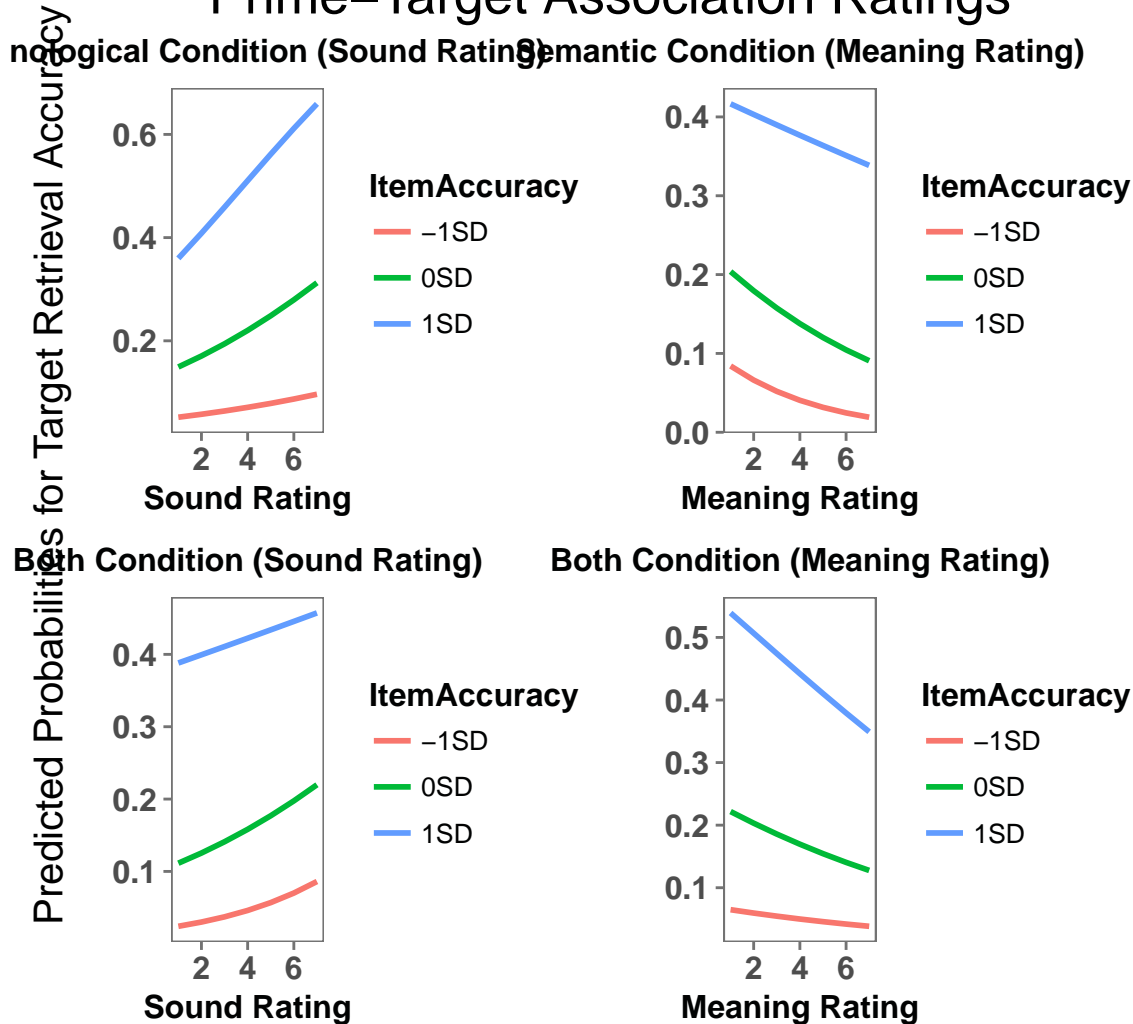
```

```

> library(gridExtra)
> grid.arrange(a2,b2,d2,c2,
+               top=textGrob("Target Retrieval Accuracy as a function of\nPrime-Target Ass
+                               gp=gpar(fontsize=20)),
+               left = textGrob("Predicted Probabilities for Target Retrieval Accuracy",

```

Target Retrieval Accuracy as a function of Prime–Target Association Ratings



21.1 State RT data

21.1.1 z-scoring RTs

```
> state_firsttrim = main # %>% filter(State.RT > 250 )
> ## aggregate per subject all IVs and DVs
> meanRT = group_by(state_firsttrim, Subject) %>%
+   summarise_at(vars(State.RT), mean)
> colnames(meanRT) = c("Subject", "MeanRT")
> sdRT = group_by(state_firsttrim, Subject) %>%
+   summarise_at(vars(State.RT), sd)
```

```

> colnames(sdRT) = c("Subject", "sdRT")
> RT_agg = merge(meanRT, sdRT, by = "Subject")
> ## merge aggregate info with long data
> state_z = merge(state_firsttrim, RT_agg, by = "Subject", all.x = T)
> ## person and grand-mean centered scores using original and aggregate
> library(dplyr)
> ## trimming z rt
> state_z$Upper = state_z$MeanRT + 2.5*state_z$sdRT
> state_z$Lower = state_z$MeanRT - 2.5*state_z$sdRT
> state_z_trimmed = subset(state_z, state_z$State.RT < state_z$Upper &
+                           state_z$State.RT > state_z$Lower)

```

21.1.2 State by Age by RT

```

> state_z_trimmed$Question.RESP = as.factor(state_z_trimmed$Question.RESP)
> e1_stateRT = state_z_trimmed %>% filter(StudyNo == "2" | StudyNo == "4")
> e2_stateRT = state_z_trimmed %>% filter(StudyNo == "5" | StudyNo == "6")
> e3_stateRT = state_z_trimmed %>% filter(StudyNo == "1")
>

```

21.1.3 E1

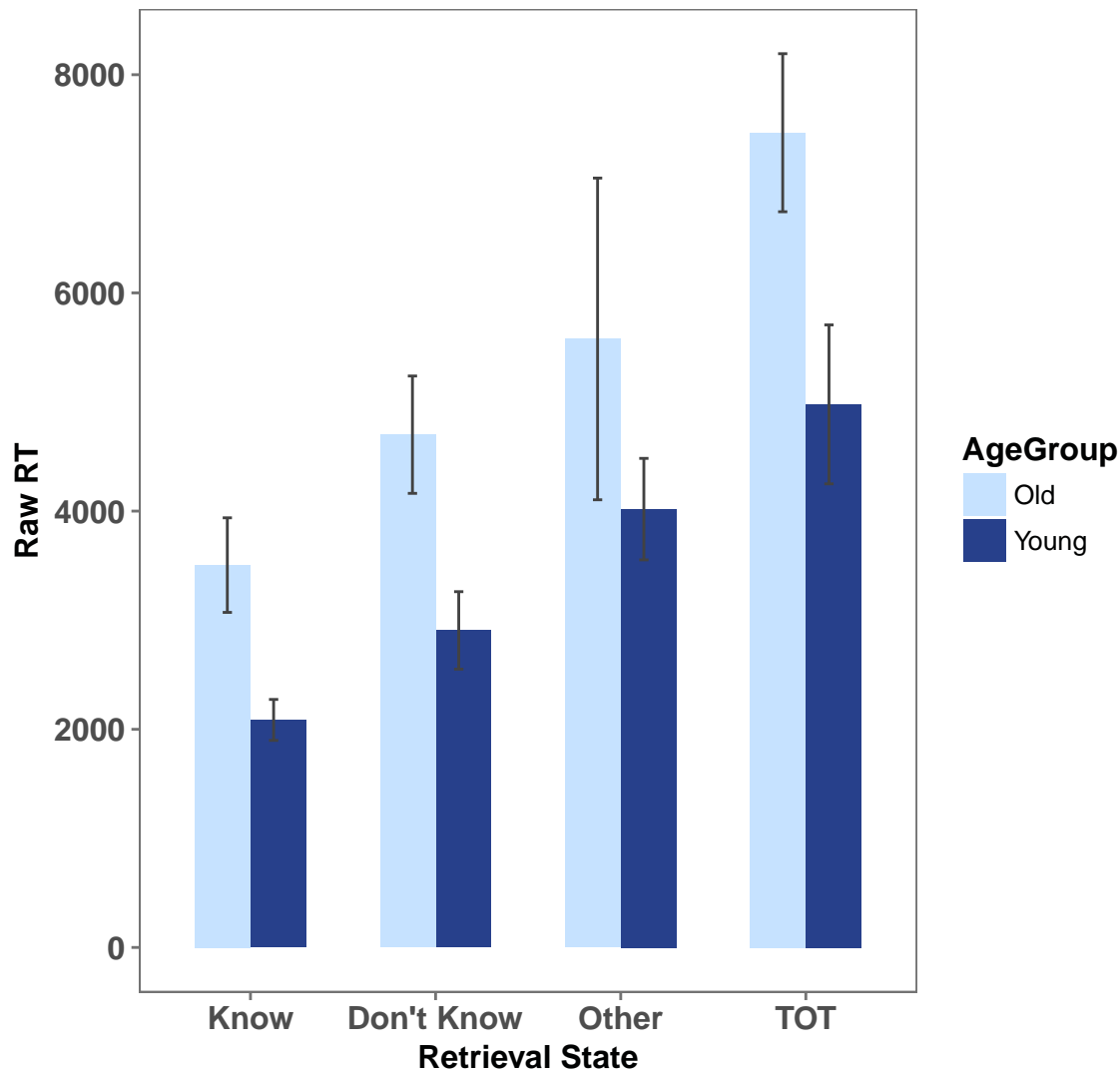
```

> e1_stateRT_agg = group_by(e1_stateRT, AgeGroup, Subject, Question.RESP) %>%
+   summarize_at(vars(State.RT), mean)
> e1_stateRT_rmisc = Rmisc::summarySE(e1_stateRT_agg,
+                                     measurevar = "State.RT",
+                                     groupvars = c("AgeGroup", "Question.RESP"))
> e1_stateRT_rmisc %>% mutate(RetrievalState = factor(Question.RESP,
+                                                     levels = unique(Question.RESP),
+                                                     labels = c("Know", "Don't Know", "Other", "TOT")),
+                             Age = factor(AgeGroup, levels = unique(AgeGroup),
+                             labels = c("Young", "Old")))%>%
+   ggplot(aes(x = RetrievalState, y = State.RT, group = AgeGroup,
+             fill = AgeGroup)) +
+   geom_bar(stat = "identity", position = "dodge", width = 0.6)+
+   geom_errorbar(aes(ymin=State.RT - ci, ymax=State.RT + ci),
+               width=.1, color = "gray26",
+               position = position_dodge(0.5))+
+   theme_few()+
+   scale_fill_manual(values = c("slategray1", "royalblue4"))+
+   xlab("Retrieval State") + ylab("Raw RT") +
+   ggtitle("Experiment 1: Retrieval State RTs by Age Group") +
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(face = "bold", size = rel(1.4), hjust = .5)),

```

```
+ strip.text.x = element_text(face = "bold", size = rel(1.4)))
>
```

Experiment 1: Retrieval State RTs by Age Group



21.1.4 E2

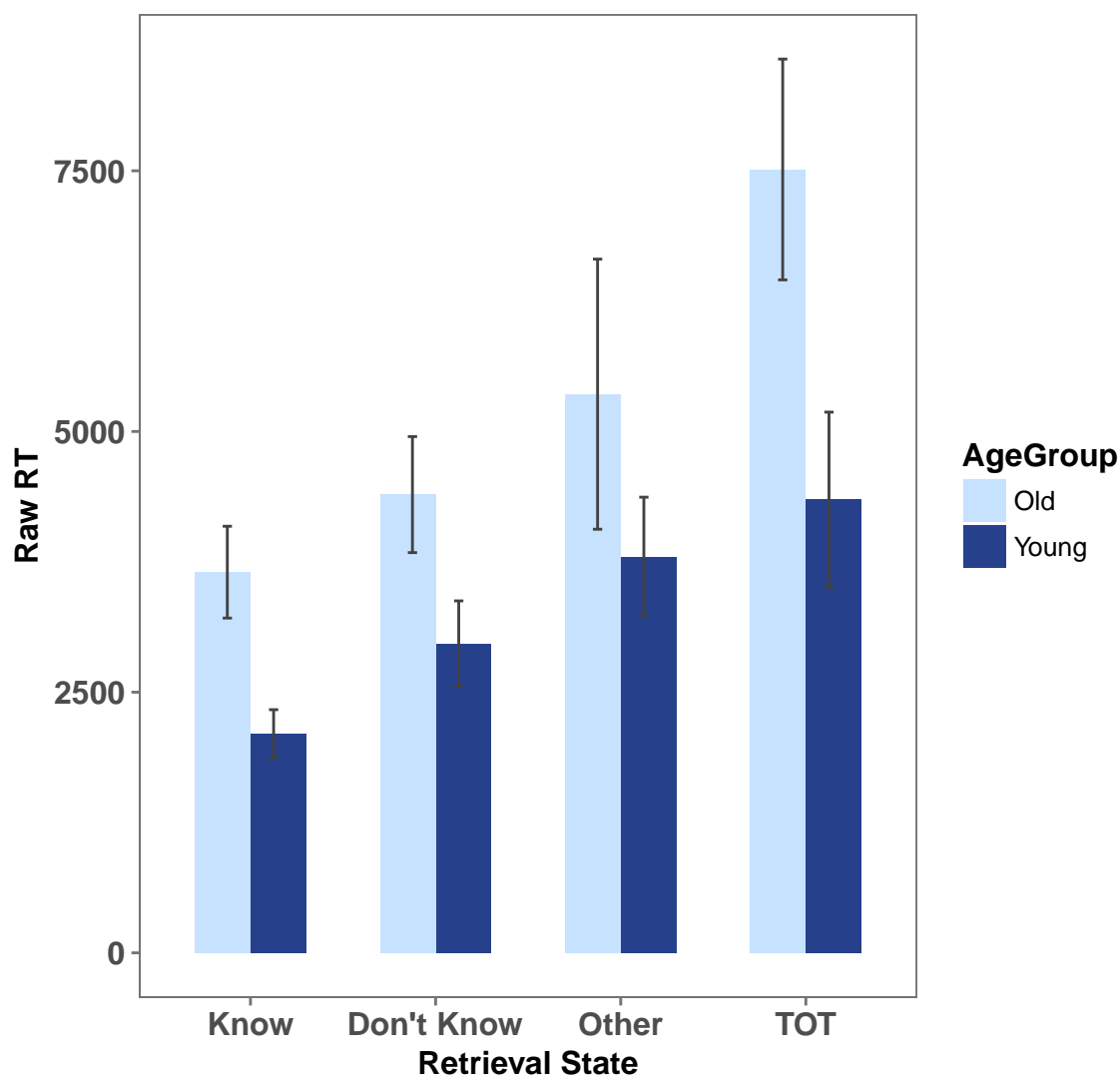
```
> e2_stateRT_agg = group_by(e2_stateRT, AgeGroup, Subject, Question.RESP) %>%
+   summarize_at(vars(State.RT), mean)
> e2_stateRT_rmisc = Rmisc::summarySE(e2_stateRT_agg,
+                                     measurevar = "State.RT",
+                                     groupvars = c("AgeGroup", "Question.RESP"))
> e2_stateRT_rmisc %>% mutate(RetrievalState = factor(Question.RESP,
```

```

+             levels = unique(Question.RESP),
+             labels = c("Know", "Don't Know", "Other", "TOT")),
+             Age = factor(AgeGroup, levels = unique(AgeGroup),
+             labels = c("Young", "Old"))))%>%
+   ggplot(aes(x = RetrievalState, y = State.RT, group = AgeGroup,
+             fill = AgeGroup)) +
+     geom_bar(stat = "identity", position = "dodge", width = 0.6)+
+     geom_errorbar(aes(ymin=State.RT - ci, ymax=State.RT + ci),
+                   width=.1, color = "gray26",
+                   position = position_dodge(0.5))+
+   theme_few()+
+   scale_fill_manual(values = c("slategray1", "royalblue4"))+
+   xlab("Retrieval State") + ylab("Raw RT") +
+   ggtitle("Experiment 2: Retrieval State RTs by Age Group") +
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(face = "bold", size = rel(1.4), hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
+
>

```


Experiment 2: Retrieval State RTs by Age Group



21.1.5 E3

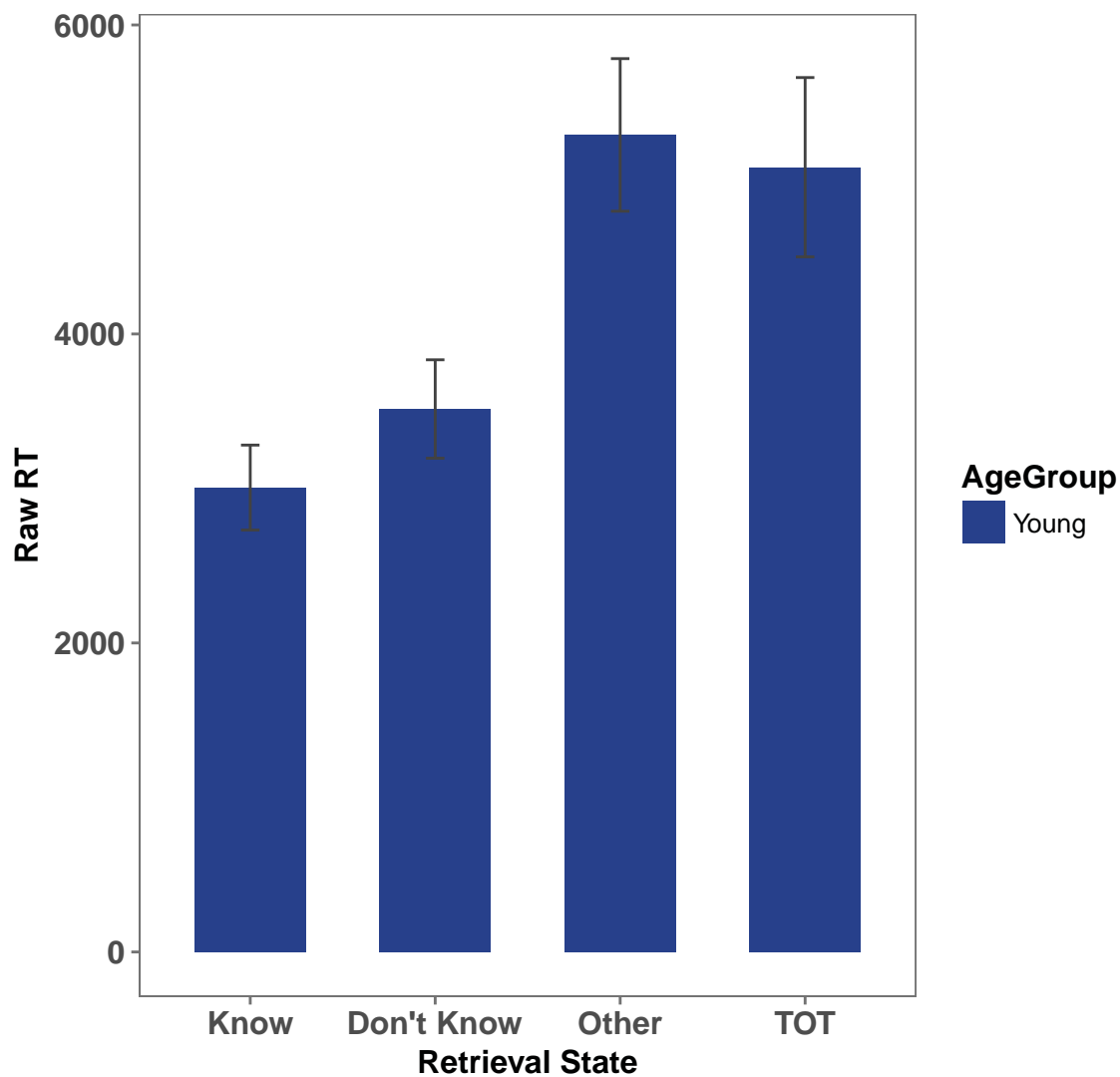
```
> e3_stateRT_agg = group_by(e3_stateRT, AgeGroup, Subject, Question.RESP) %>%  
+   summarize_at(vars(State.RT), mean)  
> e3_stateRT_rmisc = Rmisc::summarySE(e3_stateRT_agg,  
+                                     measurevar = "State.RT",  
+                                     groupvars = c("AgeGroup", "Question.RESP"))  
> e3_stateRT_rmisc %>% mutate(RetrievalState = factor(Question.RESP,  
+                                                     levels = unique(Question.RESP),  
+                                                     labels = c("Know", "Don't Know", "Other", "TOT")),  
+                             Age = factor(AgeGroup, levels = unique(AgeGroup),
```

```

+           labels = c("Young"))))%>%
+   ggplot(aes(x = RetrievalState, y = State.RT, group = AgeGroup,
+             fill = AgeGroup)) +
+     geom_bar(stat = "identity", position = "dodge", width = 0.6)+
+     geom_errorbar(aes(ymin=State.RT - ci, ymax=State.RT + ci),
+                   width=.1, color = "gray26",
+                   position = position_dodge(0.5))+
+   theme_few()+
+   scale_fill_manual(values = c("royalblue4"))+
+   xlab("Retrieval State") + ylab("Raw RT") +
+   ggtitle("Experiment 3: Retrieval State RTs") +
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(face = "bold", size = rel(1.4), hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
+
>

```

Experiment 3: Retrieval State RTs



21.2 HLMs

```
> library(lme4)
> e1_stateRT_hlm = lmer (data = e1_stateRT_agg, State.RT ~ Question.RESP*AgeGroup +
+                        (1|Subject))
> summary(e1_stateRT_hlm)
```

Linear mixed model fit by REML t-tests use Satterthwaite approximations to degrees of freedom [lmerMod]
Formula: State.RT ~ Question.RESP * AgeGroup + (1 | Subject)

```

Data: e1_stateRT_agg

REML criterion at convergence: 4828.9

Scaled residuals:
    Min       1Q   Median       3Q      Max
-3.3841 -0.3753 -0.0328  0.3429  4.6352

Random effects:
 Groups   Name                Variance Std.Dev.
 Subject (Intercept) 1011246    1006
 Residual                2473381    1573
Number of obs: 277, groups: Subject, 73

Fixed effects:
              Estimate Std. Error      df t value Pr(>|t|)
(Intercept)      4377.57      151.73    70.61  28.851 < 2e-16 ***
Question.RESP1    -1582.19      161.59   268.64  -9.791 < 2e-16 ***
Question.RESP2     -574.21      161.59   268.64  -3.554 0.000449 ***
Question.RESP3      365.48      170.83   273.70   2.139 0.033287 *
AgeGroup1          878.79      151.73    70.61   5.792 1.78e-07 ***
Question.RESP1:AgeGroup1 -169.27      161.59   268.64  -1.048 0.295803
Question.RESP2:AgeGroup1  18.25      161.59   268.64   0.113 0.910184
Question.RESP3:AgeGroup1 -160.54      170.83   273.70  -0.940 0.348173
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
      (Intr) Qs.RESP1 Qs.RESP2 Qs.RESP3 AgGrp1 Q.RESP1: Q.RESP2:
Qustn.RESP1 -0.028
Qustn.RESP2 -0.028 -0.298
Qustn.RESP3  0.038 -0.342 -0.342
AgeGroup1    0.040 -0.026 -0.026  0.030
Q.RESP1:AG1 -0.026  0.037  0.020 -0.032 -0.028
Q.RESP2:AG1 -0.026  0.020  0.037 -0.032 -0.028 -0.298
Q.RESP3:AG1  0.030 -0.032 -0.032  0.120  0.038 -0.342 -0.342

> car::Anova(e1_stateRT_hlm)

Analysis of Deviance Table (Type II Wald chisquare tests)

Response: State.RT
              Chisq Df Pr(>Chisq)
Question.RESP 168.1878  3 < 2.2e-16 ***
AgeGroup       33.4383  1 7.356e-09 ***
Question.RESP:AgeGroup  4.0654  3 0.2545
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
> e2_stateRT_hlm = lmer (data = e2_stateRT_agg, State.RT ~ Question.RESP*AgeGroup +
+ (1|Subject))
> summary(e2_stateRT_hlm)
```

Linear mixed model fit by REML t-tests use Satterthwaite approximations to degrees of freedom [lmerMod]

Formula: State.RT ~ Question.RESP * AgeGroup + (1 | Subject)
Data: e2_stateRT_agg

REML criterion at convergence: 4259.7

Scaled residuals:

Min	1Q	Median	3Q	Max
-3.3331	-0.4054	-0.0413	0.3798	3.7284

Random effects:

Groups	Name	Variance	Std.Dev.
Subject	(Intercept)	1367769	1170
Residual		2136278	1462

Number of obs: 246, groups: Subject, 65

Fixed effects:

	Estimate	Std. Error	df	t value	Pr(> t)
(Intercept)	4236.69	173.25	63.59	24.454	< 2e-16 ***
Question.RESP1	-1360.77	159.39	83.95	-8.537	5.09e-13 ***
Question.RESP2	-556.72	159.39	83.95	-3.493	0.000765 ***
Question.RESP3	248.36	167.67	84.99	1.481	0.142233
AgeGroup1	929.51	173.25	63.59	5.365	1.21e-06 ***
Question.RESP1:AgeGroup1	-154.63	159.39	83.95	-0.970	0.334766
Question.RESP2:AgeGroup1	-214.83	159.39	83.95	-1.348	0.181347
Question.RESP3:AgeGroup1	-243.18	167.67	84.99	-1.450	0.150631

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:

	(Intr)	Qs.RESP1	Qs.RESP2	Qs.RESP3	AgGrp1	Q.RESP1: Q.RESP2:
Qustn.RESP1	-0.027					
Qustn.RESP2	-0.027	-0.294				
Qustn.RESP3	0.028	-0.337	-0.337			
AgeGroup1	0.037	-0.024	-0.024	0.030		
Q.RESP1:AG1	-0.024	0.041	0.022	-0.038	-0.027	
Q.RESP2:AG1	-0.024	0.022	0.041	-0.038	-0.027	-0.294
Q.RESP3:AG1	0.030	-0.038	-0.038	0.134	0.028	-0.337

```
> car::Anova(e2_stateRT_hlm)
```

Analysis of Deviance Table (Type II Wald chisquare tests)

```

Response: State.RT
              Chisq Df Pr(>Chisq)
Question.RESP      133.723   3 < 2.2e-16 ***
AgeGroup           27.953   1 1.243e-07 ***
Question.RESP:AgeGroup 13.518   3   0.00364 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> e3_stateRT_hlm = lmer (data = e3_stateRT_agg, State.RT ~ Question.RESP +
+                        (1|Subject))
> summary(e3_stateRT_hlm)

```

```

Linear mixed model fit by REML t-tests use Satterthwaite approximations to
degrees of freedom [lmerMod]
Formula: State.RT ~ Question.RESP + (1 | Subject)
Data: e3_stateRT_agg

```

```
REML criterion at convergence: 2380.6
```

```

Scaled residuals:
      Min       1Q   Median       3Q      Max
-1.78497 -0.58810 -0.06558  0.46680  3.09878

```

```

Random effects:
 Groups   Name      Variance Std.Dev.
Subject  (Intercept) 804737   897.1
Residual                849748   921.8
Number of obs: 144, groups: Subject, 36

```

```

Fixed effects:
              Estimate Std. Error      df t value Pr(>|t|)
(Intercept)    4221.9      168.1    35.0  25.117 < 2e-16 ***
Question.RESP1 -1216.1      133.1   109.9  -9.140 3.55e-15 ***
Question.RESP2  -707.6      133.1   109.9  -5.318 5.57e-07 ***
Question.RESP3  1066.3      133.1   109.9   8.014 1.28e-12 ***
---

```

```

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Correlation of Fixed Effects:
      (Intr) Q.RESP1 Q.RESP2
Qustn.RESP1  0.000
Qustn.RESP2  0.000 -0.333
Qustn.RESP3  0.000 -0.333 -0.333

```

```

> car::Anova(e3_stateRT_hlm)

```

```
Analysis of Deviance Table (Type II Wald chisquare tests)
```

```
Response: State.RT
```

```
          Chisq Df Pr(>Chisq)  
Question.RESP 163.18  3  < 2.2e-16 ***
```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
>
```

22 Word Type Analysis

```
> main = read.csv("Julie_Main5Studies.csv", header = TRUE, sep = ",")  
> main$StudyNo = as.factor(main$StudyNo)  
> main$PrimeCondition = as.factor(main$PrimeCondition)  
> word_type = read.csv("ItemWordTypes.csv", header = TRUE, sep = ",")  
> main_word = merge(main, word_type, by = c("Target"))  
> library(dplyr)  
> word_type_prime = group_by(main_word,  
+                             ExperimentName, AgeGroup, Subject, PrimeCondition, Proper) %>%  
+   summarise_at(vars(Accuracy), mean)  
> word_type_prime$Subject = as.factor(word_type_prime$Subject)  
> word_type_prime_E1 = word_type_prime %>%  
+   filter(ExperimentName == "tot extended prime")  
> word_type_prime_E2 = word_type_prime %>%  
+   filter(ExperimentName == "tot not the prime")  
> word_type_prime_E3 = word_type_prime %>%  
+   filter(ExperimentName == "tot 48 ms")  
> word_type_age = group_by(main_word, ExperimentName,  
+                             AgeGroup, Proper) %>%  
+   summarise_at(vars(Accuracy), mean)  
> word_type_state_sub = group_by(main_word, Subject,  
+                             Proper, Question.RESP) %>%  
+   summarise(Trials = n())  
> word_type_state_experiment = group_by(main_word, ExperimentName,  
+                             Proper, Question.RESP) %>%  
+   summarise(Trials = n())  
> word_type_state_sub_age = group_by(main_word, Subject, AgeGroup,  
+                             Proper, Question.RESP) %>%  
+   summarise(Trials = n())  
>
```

22.1 E1 E2 E3: proper name ANOVA

```
> e1_proper_aov = aov(data = word_type_prime_E1, Accuracy ~ AgeGroup*PrimeCondition*Proper
+ Error(Subject/(PrimeCondition*Proper)))
> summary(e1_proper_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  0.058  0.05821    0.364  0.548
Residuals 71 11.355  0.15993

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3  1.456  0.4852  24.974 6.93e-14 ***
AgeGroup:PrimeCondition  3  0.032  0.0108  0.556  0.645
Residuals      213  4.139  0.0194

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:Proper
      Df Sum Sq Mean Sq F value Pr(>F)
Proper  1  4.728  4.728 107.969 6.71e-16 ***
AgeGroup:Proper  1  0.218  0.218  4.981  0.0288 *
Residuals      71  3.109  0.044

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition:Proper
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition:Proper  3  0.069  0.02313  1.433  0.234
AgeGroup:PrimeCondition:Proper  3  0.065  0.02172  1.346  0.261
Residuals      213  3.437  0.01614
```

```
> e2_proper_aov = aov(data = word_type_prime_E2, Accuracy ~ AgeGroup*PrimeCondition*Proper
+ Error(Subject/(PrimeCondition*Proper)))
> summary(e2_proper_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  0.214  0.2136  1.391  0.243
Residuals 63  9.677  0.1536

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3  0.625  0.2082  10.26 2.74e-06 ***
AgeGroup:PrimeCondition  3  0.083  0.0276  1.36  0.256
Residuals      189  3.836  0.0203

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



```
Error: Subject:Proper
      Df Sum Sq Mean Sq F value    Pr(>F)
Proper      1   3.584    3.584  65.795 2.28e-11 ***
AgeGroup:Proper  1   0.080    0.080   1.468    0.23
Residuals    63   3.432    0.054
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition:Proper
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition:Proper      3   0.009  0.002969   0.170   0.916
AgeGroup:PrimeCondition:Proper  3   0.004  0.001365   0.078   0.972
Residuals                 189   3.293  0.017425
```

```
> e3_proper_aov = aov(data = word_type_prime_E3, Accuracy ~ PrimeCondition*Proper +
+                      Error(Subject/(PrimeCondition*Proper)))
> summary(e3_proper_aov)
```

```
Error: Subject
      Df Sum Sq Mean Sq F value    Pr(>F)
Residuals 35   4.794    0.137

Error: Subject:PrimeCondition
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition      3  0.2251  0.07503   3.914 0.0108 *
Residuals          105  2.0128  0.01917
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:Proper
      Df Sum Sq Mean Sq F value    Pr(>F)
Proper      1   1.095    1.095  35.28 9.28e-07 ***
Residuals 35   1.086    0.031
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Subject:PrimeCondition:Proper
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition:Proper      3  0.0216  0.007189   0.465   0.707
Residuals                 105  1.6237  0.015464
```

Word Type, Experiment and Prime Type

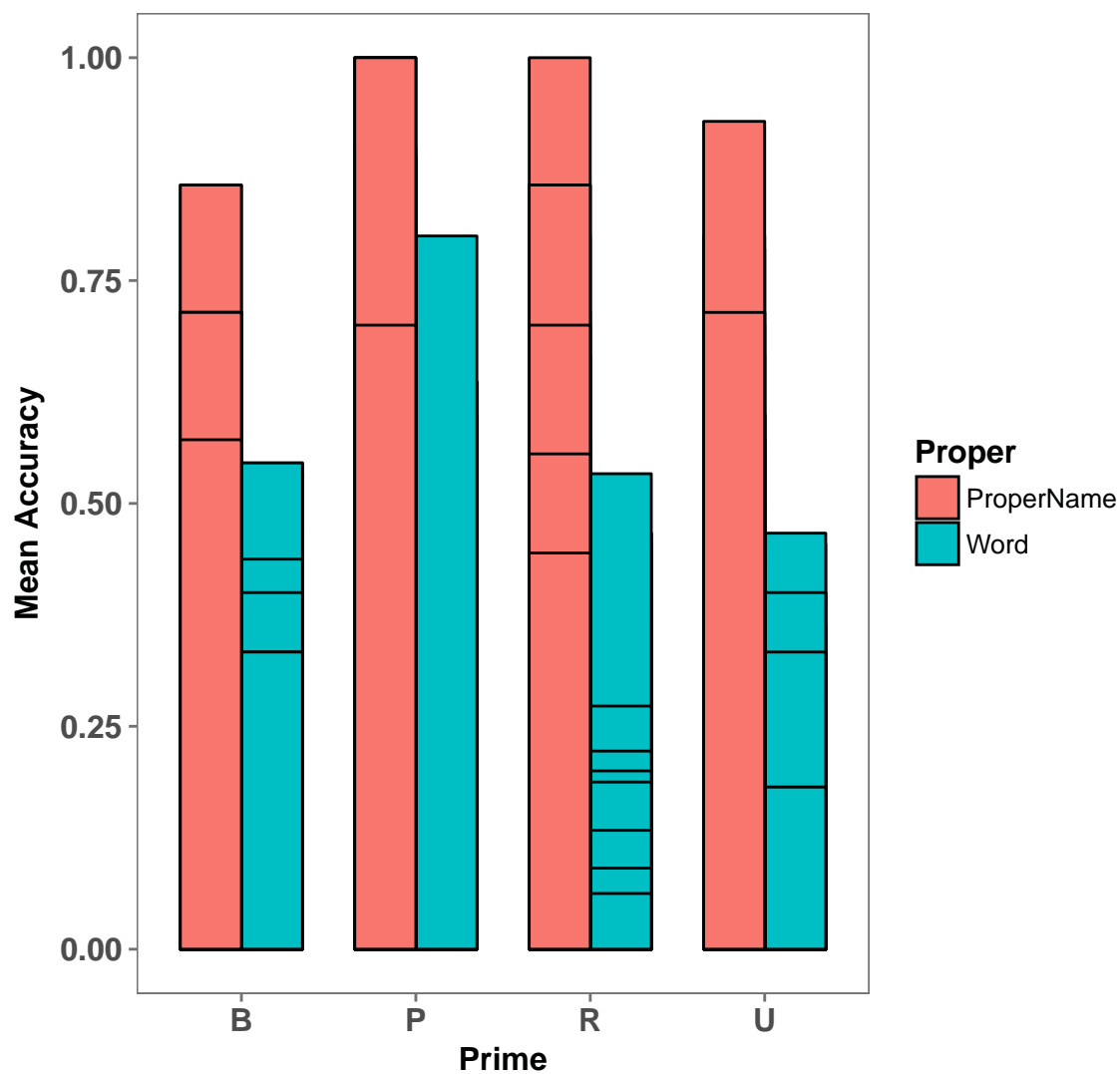
```
> library(ggplot2)
> library(ggthemes)
> word_type_prime_E1 %>%
```

```

+ ggplot(aes(x = PrimeCondition, y = Accuracy,
+           group = Proper, fill = Proper))+
+   geom_bar(stat = "identity", position = "dodge",
+           width = 0.7, color = "black")+
+   theme_few()+
+   # facet_wrap(~ExperimentName)+
+   xlab("Prime") + ylab("Mean Accuracy") +
+   ggtitle("Word Types and Accuracy across Primes") +
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(face = "bold", size = rel(1.5), hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))

```

Word Types and Accuracy across Primes

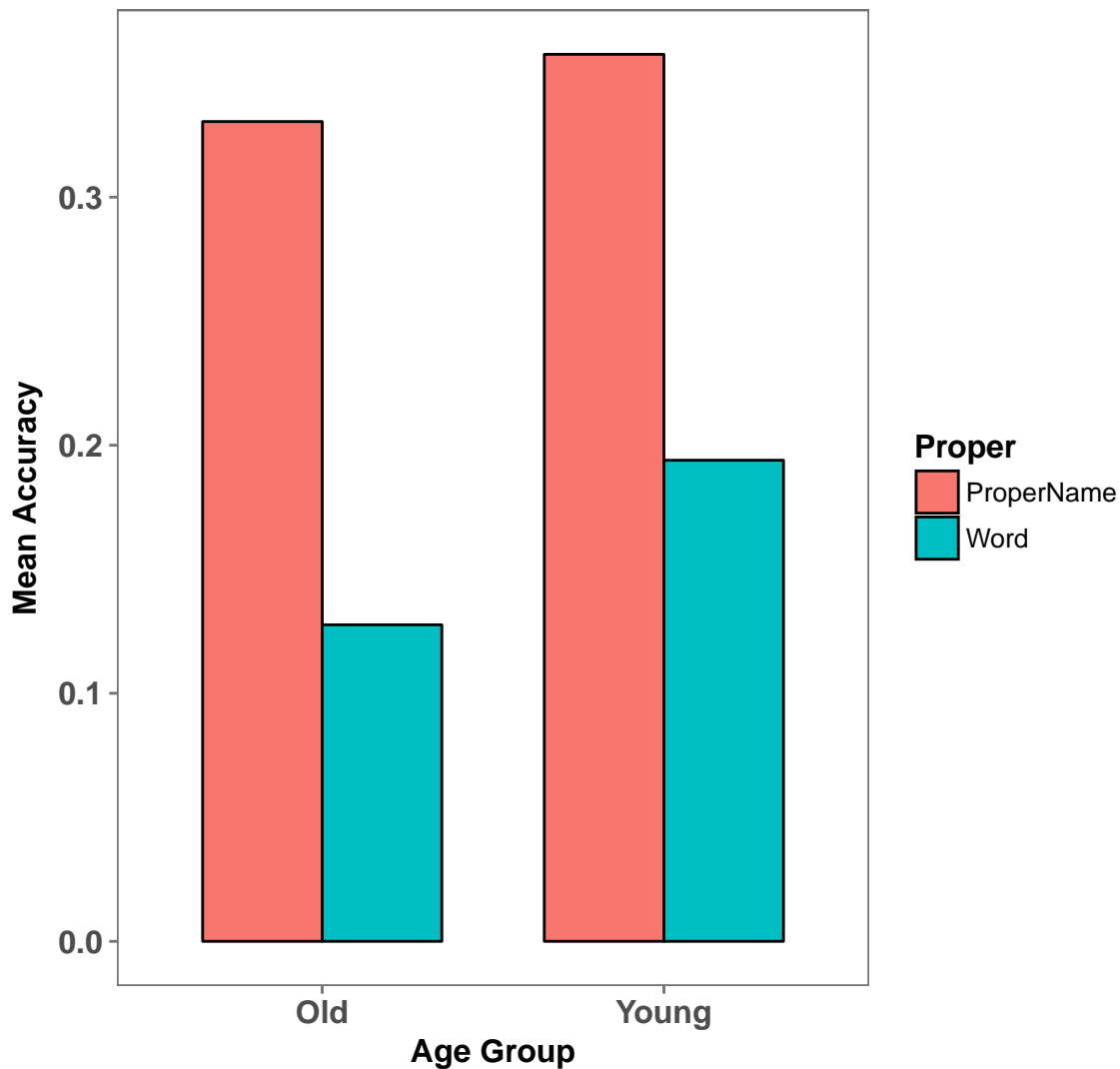


Word Type and AgeGroup

```
> library(ggplot2)
> library(ggthemes)
> word_type_age %>% filter(ExperimentName == "tot not the prime") %>%
+   ggplot(aes(x = AgeGroup, y = Accuracy,
+             group = Proper, fill = Proper))+
+   geom_bar(stat = "identity", position = "dodge",
+           width = 0.7, color = "black")+
+   theme_few()
```

```
+ # facet_wrap(~ExperimentName)+
+ xlab("Age Group") + ylab("Mean Accuracy") +
+ ggtitle("Word Types and Accuracy across Age Groups") +
+ theme(axis.text = element_text(face = "bold", size = rel(1)),
+       axis.title = element_text(face = "bold", size = rel(1)),
+       legend.title = element_text(face = "bold", size = rel(1)),
+       plot.title = element_text(face = "bold", size = rel(1.5), hjust = .5),
+       strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

Word Types and Accuracy across Age Groups



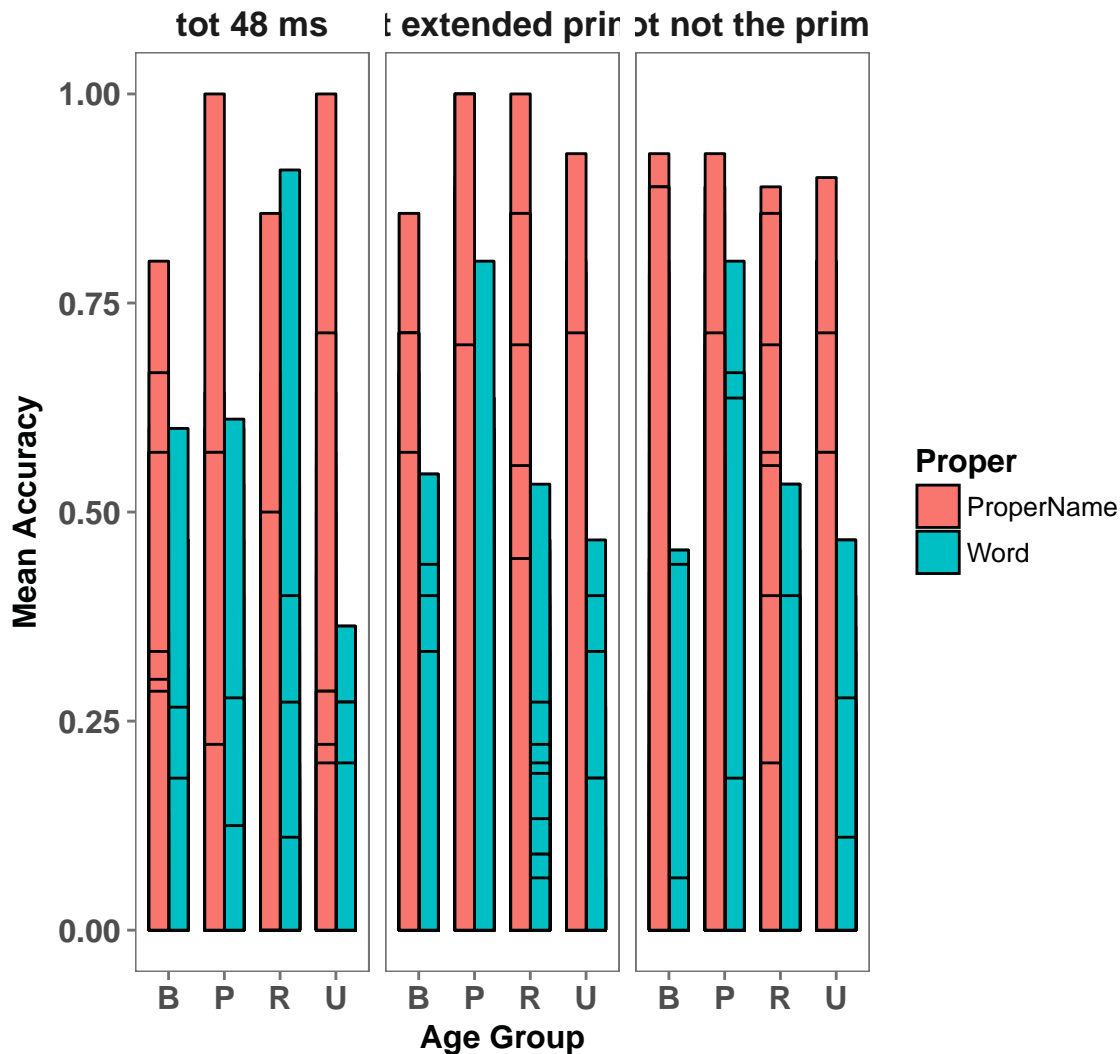
Word Type and Prime

```

> library(ggplot2)
> library(ggthemes)
> word_type_prime %>%
+   ggplot(aes(x = PrimeCondition, y = Accuracy,
+             group = Proper, fill = Proper))+
+   geom_bar(stat = "identity", position = "dodge",
+           width = 0.7, color = "black")+
+   theme_few()+
+   facet_wrap(~ExperimentName)+
+   xlab("Age Group") + ylab("Mean Accuracy") +
+   ggtitle("Word Types and Accuracy across Age Groups") +
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(face = "bold", size = rel(1.5), hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))

```

Word Types and Accuracy across Age Groups



Word Type and State

```
> library(ggplot2)
> library(ggthemes)
> word_type_state_sub$Question.RESP = as.factor(word_type_state_sub$Question.RESP)
> word_type_state = Rmisc::summarySE(word_type_state_sub,
+                                   measurevar = "Trials",
+                                   groupvars = c("Proper", "Question.RESP"))
> word_type_state %>%
+   mutate(RetrievalState = factor(Question.RESP,
```

```

+           levels = unique(Question.RESP),
+           labels = c("Know", "Dont Know", "Other", "TOT")))%>%
+ ggplot(aes(x = RetrievalState, y = Trials,
+           group = Proper, fill = Proper))+
+ geom_bar(stat = "identity", position = "dodge",
+           width = 0.7, color = "black")+
+ theme_few()+
+ xlab("State") + ylab("Mean Trials") +
+ ggtitle("Word Types and Trials across States") +
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(face = "bold", size = rel(1.5), hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))

```

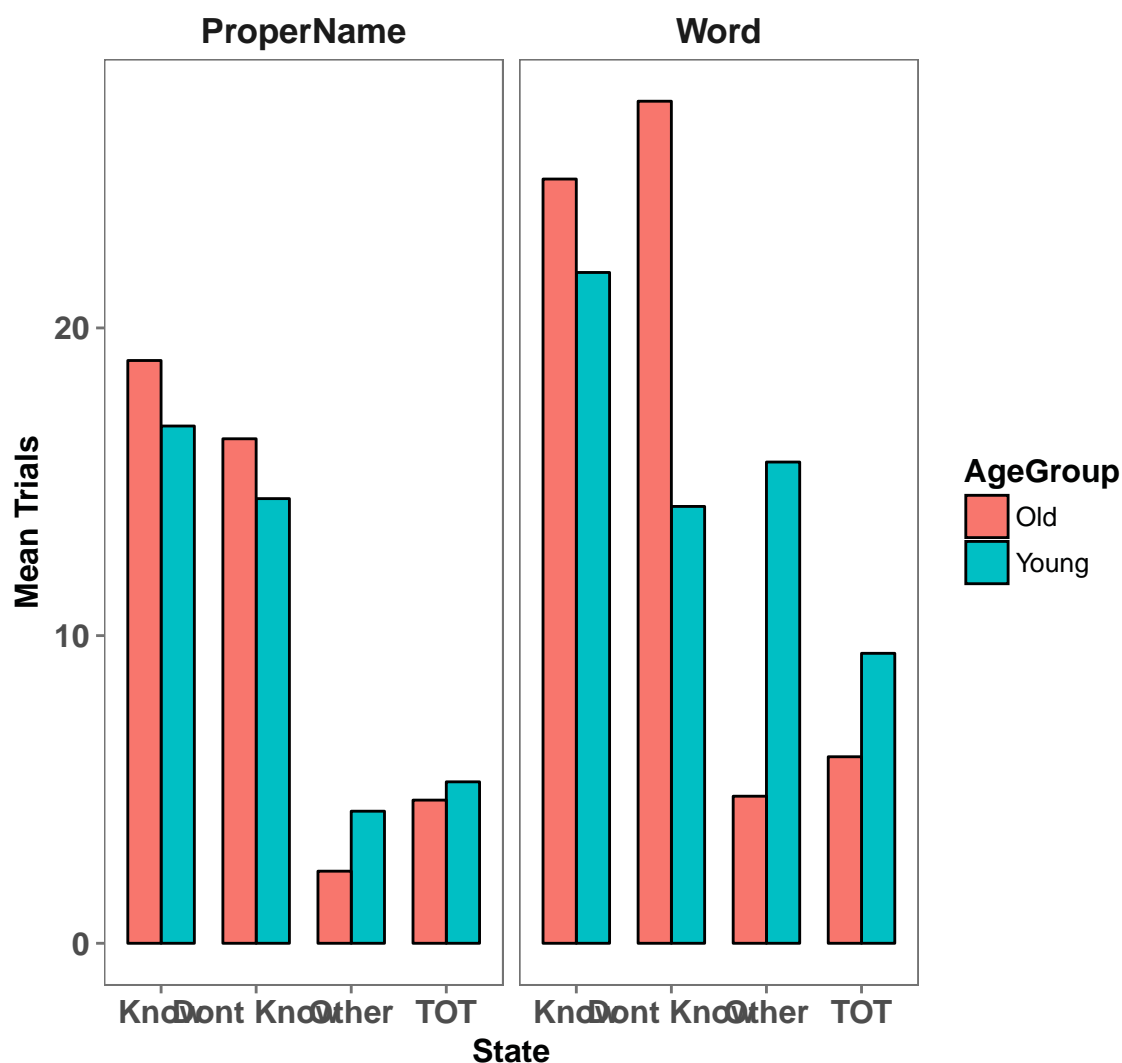
Word Type, Age and State

```

> library(ggplot2)
> library(ggthemes)
> word_type_state_age = Rmisc::summarySE(word_type_state_sub_age,
+           measurevar = "Trials",
+           groupvars = c("AgeGroup", "Proper", "Question.RESP"))
> word_type_state_age %>%
+   mutate(RetrievalState = factor(Question.RESP,
+           levels = unique(Question.RESP),
+           labels = c("Know", "Dont Know", "Other", "TOT")))%>%
+ ggplot(aes(x = RetrievalState, y = Trials,
+           group = AgeGroup, fill = AgeGroup))+
+ geom_bar(stat = "identity", position = "dodge",
+           width = 0.7, color = "black")+
+ theme_few()+
+ facet_wrap(~Proper)+
+ xlab("State") + ylab("Mean Trials") +
+ ggtitle("Word Types and Trials Across State and Age") +
+   theme(axis.text = element_text(face = "bold", size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(face = "bold", size = rel(1.5), hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))

```

Word Types and Trials Across State and Age



22.2 HLMs with WordType

22.2.1 Target Accuracy

```
> contrasts(main_word$PrimeCondition) = contr.treatment(4, base = 2)
> contrasts(main_word$AgeGroup) = contr.treatment(2, base = 1)
> contrasts(main_word$Proper) = contr.treatment(2, base = 1)
> e1_proper = main_word %>% filter(StudyNo == "2" | StudyNo == "4")
> e2_proper = main_word %>% filter(StudyNo == "5" | StudyNo == "6")
> e3_proper = main_word %>% filter(StudyNo == "1")
> exp1_acc_hlm_M1 = glmer(data = e1_proper ,
```



```

+           Accuracy ~ AgeGroup*PrimeCondition*Proper +
+           (1|Subject) + (1|Target), family = "binomial",
+           control=glmerControl(optimizer="bobyqa",
+           optCtrl=list(maxfun=100000)))
> summary(exp1_acc_hlm_M1)

```

```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial (logit )
Formula: Accuracy ~ AgeGroup * PrimeCondition * Proper + (1 | Subject) +
(1 | Target)
Data: e1_proper
Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+05))

```

AIC	BIC	logLik	deviance	df.resid
6177.2	6301.3	-3070.6	6141.2	7282

Scaled residuals:

Min	1Q	Median	3Q	Max
-4.7354	-0.4659	-0.2271	0.2196	10.3404

Random effects:

Groups	Name	Variance	Std.Dev.
Target	(Intercept)	1.802	1.343
Subject	(Intercept)	1.016	1.008

Number of obs: 7300, groups: Target, 100; Subject, 73

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	0.01153	0.29777	0.039	0.969106
AgeGroup2	-0.53684	0.29446	-1.823	0.068283 .
PrimeCondition1	-0.51641	0.17659	-2.924	0.003452 **
PrimeCondition3	-0.95925	0.18078	-5.306	1.12e-07 ***
PrimeCondition4	-1.01087	0.18144	-5.571	2.53e-08 ***
Proper2	-1.74437	0.32705	-5.334	9.63e-08 ***
AgeGroup2:PrimeCondition1	0.29016	0.24979	1.162	0.245392
AgeGroup2:PrimeCondition3	0.45516	0.25408	1.791	0.073231 .
AgeGroup2:PrimeCondition4	0.46021	0.25545	1.802	0.071617 .
AgeGroup2:Proper2	0.84169	0.24448	3.443	0.000576 ***
PrimeCondition1:Proper2	-0.27540	0.25768	-1.069	0.285180
PrimeCondition3:Proper2	-0.25521	0.26784	-0.953	0.340670
PrimeCondition4:Proper2	-0.13164	0.26817	-0.491	0.623513
AgeGroup2:PrimeCondition1:Proper2	-0.58325	0.35979	-1.621	0.105000
AgeGroup2:PrimeCondition3:Proper2	-0.65124	0.37342	-1.744	0.081161 .
AgeGroup2:PrimeCondition4:Proper2	-0.36841	0.36998	-0.996	0.319365

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> car::Anova(exp1_acc_hlm_M1)
```

Analysis of Deviance Table (Type II Wald chisquare tests)

Response: Accuracy

	Chisq	Df	Pr(>Chisq)
AgeGroup	0.0243	1	0.8760249
PrimeCondition	151.0413	3	< 2.2e-16 ***
Proper	33.0953	1	8.775e-09 ***
AgeGroup:PrimeCondition	2.8165	3	0.4207988
AgeGroup:Proper	12.2214	1	0.0004724 ***
PrimeCondition:Proper	13.7021	3	0.0033399 **
AgeGroup:PrimeCondition:Proper	3.9393	3	0.2680885

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> sjPlot::plot_model(exp1_acc_hlm_M1, type = "pred",
+                     terms = c("AgeGroup","Proper"))
> ## OA better than YA for proper names than words
>
> #sjPlot::plot_model(exp1_acc_hlm_M1, type = "pred",
> #                     terms = c("Proper","PrimeCondition"))
>
> ## Words show more phon. facilitation than Proper Names
>
> exp2_acc_hlm_M1 = glmer(data = e2_proper ,
+                          Accuracy ~ AgeGroup*PrimeCondition*Proper +
+                          (1|Subject) + (1|Target), family = "binomial",
+                          control=glmerControl(optimizer="bobyqa",
+                          optCtrl=list(maxfun=100000)))
> summary(exp2_acc_hlm_M1)
```

Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [glmerMod]

Family: binomial (logit)

Formula: Accuracy ~ AgeGroup * PrimeCondition * Proper + (1 | Subject) + (1 | Target)

Data: e2_proper

Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+05))

AIC	BIC	logLik	deviance	df.resid
5308.2	5430.3	-2636.1	5272.2	6482

Scaled residuals:

Min	1Q	Median	3Q	Max
-3.7065	-0.4408	-0.2260	-0.0637	13.4160

```

Random effects:
  Groups   Name      Variance Std.Dev.
Target    (Intercept) 1.834    1.354
Subject    (Intercept) 1.003    1.002
Number of obs: 6500, groups: Target, 100; Subject, 65

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -0.7536    0.3106  -2.426   0.0153 *
AgeGroup2      0.4502    0.3128   1.439   0.1500
PrimeCondition1 -0.2305    0.1979  -1.165   0.2441
PrimeCondition3 -0.3809    0.1992  -1.913   0.0558 .
PrimeCondition4 -0.4404    0.1994  -2.209   0.0272 *
Proper2       -1.6923    0.3442  -4.916 8.83e-07 ***
AgeGroup2:PrimeCondition1 -0.2973    0.2725  -1.091   0.2752
AgeGroup2:PrimeCondition3 -0.1817    0.2729  -0.666   0.5054
AgeGroup2:PrimeCondition4 -0.4869    0.2770  -1.758   0.0788 .
AgeGroup2:Proper2 0.4739    0.2714   1.747   0.0807 .
PrimeCondition1:Proper2 -0.2817    0.2951  -0.955   0.3398
PrimeCondition3:Proper2 -0.3989    0.3019  -1.321   0.1864
PrimeCondition4:Proper2 -0.3825    0.3057  -1.251   0.2109
AgeGroup2:PrimeCondition1:Proper2 0.2116    0.3962   0.534   0.5933
AgeGroup2:PrimeCondition3:Proper2 0.2224    0.4028   0.552   0.5809
AgeGroup2:PrimeCondition4:Proper2 0.2172    0.4125   0.527   0.5985
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
> car::Anova(exp2_acc_hlm_M1)
```

```

Analysis of Deviance Table (Type II Wald chisquare tests)

Response: Accuracy
              Chisq Df Pr(>Chisq)
AgeGroup      3.6418  1  0.05634 .
PrimeCondition 73.2380  3 8.644e-16 ***
Proper        29.6173  1 5.263e-08 ***
AgeGroup:PrimeCondition 3.9008  3 0.27238
AgeGroup:Proper 18.6134  1 1.601e-05 ***
PrimeCondition:Proper 2.4348  3 0.48719
AgeGroup:PrimeCondition:Proper 0.4567  3 0.92829
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> #sjPlot::plot_model(exp2_acc_hlm_M1, type = "pred",
> #                    terms = c("AgeGroup", "Proper"))
>
> # Not clear what this interaction is: DA worse than YA in Words
>

```

```

>
> exp3_acc_hlm_M1 = glmer(data = e3_proper ,
+                           Accuracy ~ PrimeCondition*Proper +
+                           (1|Subject) + (1|Target), family = "binomial",
+                           control=glmerControl(optimizer="bobyqa",
+                           optCtrl=list(maxfun=100000)))
> summary(exp1_acc_hlm_M1)

```

```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial ( logit )
Formula: Accuracy ~ AgeGroup * PrimeCondition * Proper + (1 | Subject) +
(1 | Target)
Data: e1_proper
Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+05))

```

AIC	BIC	logLik	deviance	df.resid
6177.2	6301.3	-3070.6	6141.2	7282

Scaled residuals:

Min	1Q	Median	3Q	Max
-4.7354	-0.4659	-0.2271	0.2196	10.3404

Random effects:

Groups	Name	Variance	Std.Dev.
Target	(Intercept)	1.802	1.343
Subject	(Intercept)	1.016	1.008

Number of obs: 7300, groups: Target, 100; Subject, 73

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	0.01153	0.29777	0.039	0.969106
AgeGroup2	-0.53684	0.29446	-1.823	0.068283 .
PrimeCondition1	-0.51641	0.17659	-2.924	0.003452 **
PrimeCondition3	-0.95925	0.18078	-5.306	1.12e-07 ***
PrimeCondition4	-1.01087	0.18144	-5.571	2.53e-08 ***
Proper2	-1.74437	0.32705	-5.334	9.63e-08 ***
AgeGroup2:PrimeCondition1	0.29016	0.24979	1.162	0.245392
AgeGroup2:PrimeCondition3	0.45516	0.25408	1.791	0.073231 .
AgeGroup2:PrimeCondition4	0.46021	0.25545	1.802	0.071617 .
AgeGroup2:Proper2	0.84169	0.24448	3.443	0.000576 ***
PrimeCondition1:Proper2	-0.27540	0.25768	-1.069	0.285180
PrimeCondition3:Proper2	-0.25521	0.26784	-0.953	0.340670
PrimeCondition4:Proper2	-0.13164	0.26817	-0.491	0.623513
AgeGroup2:PrimeCondition1:Proper2	-0.58325	0.35979	-1.621	0.105000
AgeGroup2:PrimeCondition3:Proper2	-0.65124	0.37342	-1.744	0.081161 .
AgeGroup2:PrimeCondition4:Proper2	-0.36841	0.36998	-0.996	0.319365

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> car::Anova(exp3_acc_hlm_M1)
```

```
Analysis of Deviance Table (Type II Wald chisquare tests)
```

```
Response: Accuracy
```

	Chisq	Df	Pr(>Chisq)
PrimeCondition	19.757	3	0.0001906 ***
Proper	16.383	1	5.175e-05 ***
PrimeCondition:Proper	0.630	3	0.8895350

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> #sjPlot::plot_model(exp3_acc_hlm_M1, type = "int")
>
>
> exp1_plot_data <- effects::effect("PrimeCondition*Proper",
+                                 exp2_acc_hlm_M1,
+ xlevels = list(PrimeCondition = c("B", "P", "R", "U"),
+               AgeGroup = c("Young", "Old"),
+               Proper = c("ProperName", "Word")))
> # plot(exp1_plot_data, main = "Exp1: WordType x PrimeCondition")
>
> exp1_plot_data <- effects::effect("AgeGroup*Proper",
+                                 exp1_acc_hlm_M1,
+ xlevels = list(PrimeCondition = c("B", "P", "R", "U"),
+               AgeGroup = c("Young", "Old"),
+               Proper = c("ProperName", "Word")))
> t1 = plot(exp1_plot_data, main = "Exp1: WordType x Age")
> exp2_plot_data <- effects::effect("AgeGroup*Proper",
+                                 exp2_acc_hlm_M1,
+ xlevels = list(PrimeCondition = c("B", "P", "R", "U"),
+               AgeGroup = c("Young", "Old"),
+               Proper = c("ProperName", "Word")))
> t2 =plot(exp1_plot_data, main = "Exp2: WordType x Age")
>
> #gridExtra::grid.arrange(t1,t2, nrow = 1, ncol = 2)
```

22.2.2 States: E3

```
> ### MULTINOMIAL LOGISTIC REGRESSION ###
>
> library(nnet)
> library(dplyr)
> e3_proper$State = ifelse(e3_proper$Question.RESP == "1", "Know",
+                          (ifelse(e3_proper$Question.RESP == "2", "DontKnow",
```

```
+         ifelse(e3_proper$Question.RESP == "3", "Other", "TOT"))))
> e3_proper_state_multinomial = nnet::multinom(data = e3_proper,
+         State ~ Proper*PrimeCondition +
+         (1|Subject))
```

```
# weights: 40 (27 variable)
initial value 4990.659700
iter 10 value 4653.511178
iter 20 value 4631.521716
iter 30 value 4630.791291
iter 30 value 4630.791288
iter 30 value 4630.791288
final value 4630.791288
converged
```

```
> summary(e3_proper_state_multinomial)
```

Call:

```
nnet::multinom(formula = State ~ Proper * PrimeCondition + (1 |
  Subject), data = e3_proper)
```

Coefficients:

	(Intercept)	Proper2	PrimeCondition1	PrimeCondition3	PrimeCondition4
Know	0.04800516	0.3973448	0.14515455	0.2439883	-0.3124643
Other	-0.66673875	1.2850940	0.09306014	-0.2434206	-0.1440636
TOT	-0.42625599	0.4391525	-0.03124165	-0.1590846	-0.1083642
1 Subject1					
		Proper2:PrimeCondition1	Proper2:PrimeCondition3		
Know	-0.04800516		-0.08453683	-0.09638129	
Other	0.66673875		-0.15267999	0.34942559	
TOT	0.42625599		-0.04287940	0.09222280	
		Proper2:PrimeCondition4			
Know		-0.042301717			
Other		0.007411718			
TOT		0.119629224			

Std. Errors:

	(Intercept)	Proper2	PrimeCondition1	PrimeCondition3	PrimeCondition4
Know	0.06081547	0.1657584	0.1719073	0.1701203	0.1727395
Other	0.09638998	0.2308735	0.2720285	0.2922803	0.2707143
TOT	0.08052036	0.2138057	0.2328056	0.2385910	0.2240295
1 Subject1					
		Proper2:PrimeCondition1	Proper2:PrimeCondition3		
Know	0.06081547		0.2337365	0.2335841	
Other	0.09638998		0.3267782	0.3435708	
TOT	0.08052036		0.3076877	0.3144480	
		Proper2:PrimeCondition4			
Know		0.2348612			
Other		0.3229731			
TOT		0.2946621			

```
Residual Deviance: 9261.583
AIC: 9309.583
```

```
> car::Anova(e3_proper_state_multinomial)
```

```
Analysis of Deviance Table (Type II tests)
```

```
Response: State
```

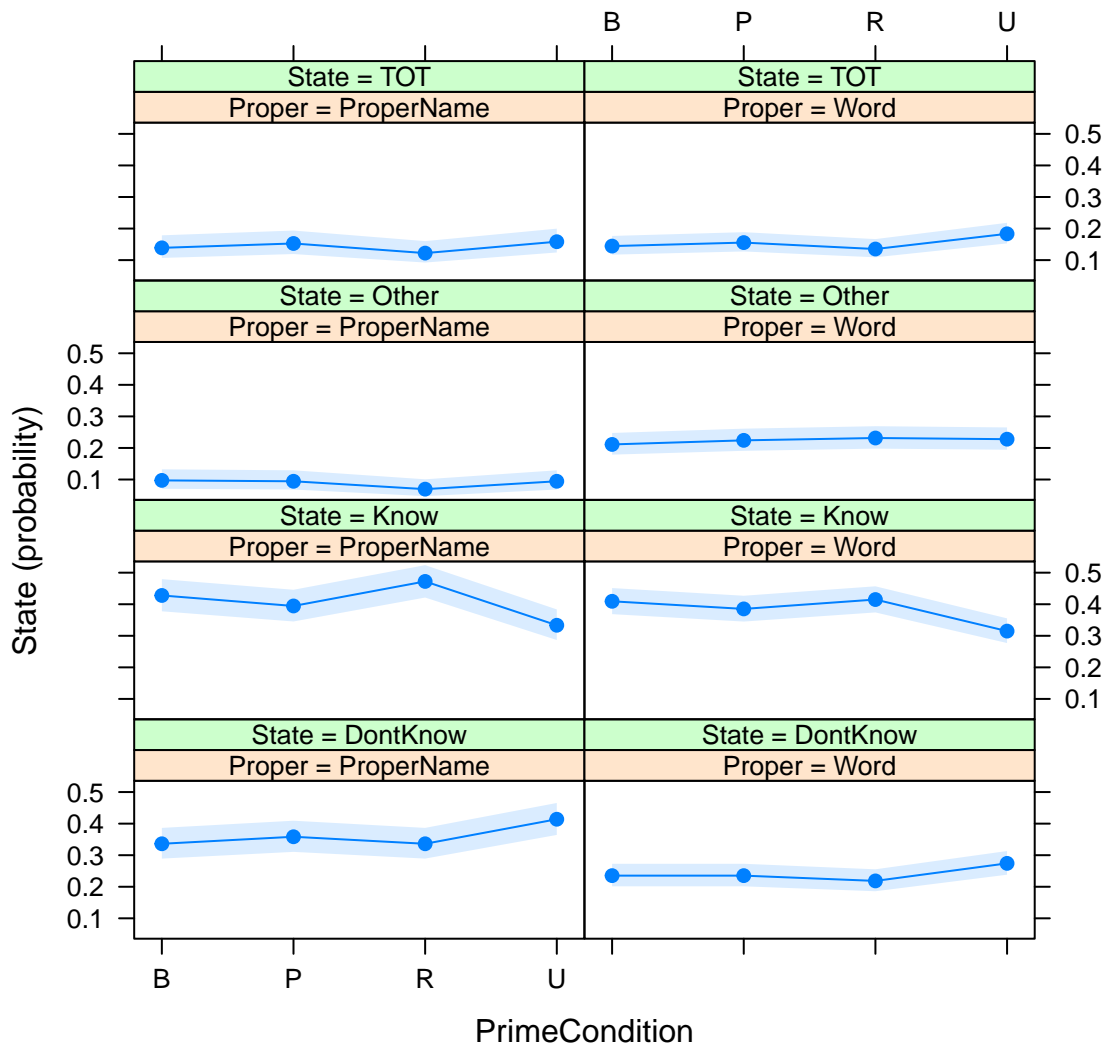
	LR	Chisq	Df	Pr(>Chisq)
Proper	145.521	3	< 2.2e-16	***
PrimeCondition	31.750	9	0.0002199	***
1 Subject	0.000	3	1.0000000	
Proper:PrimeCondition	3.581	9	0.9367561	

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> exp3_state_data <- effects::effect("Proper*PrimeCondition",
+                                   e3_proper_state_multinomial,
+   xlevels = list(PrimeCondition = c("B", "P", "R", "U"),
+                 Proper = c("ProperName", "Word")))
> plot(exp3_state_data, main = "Exp 3: WordType x PrimeCondition")
>
```

Exp 3: WordType x PrimeCondition



22.2.3 States: E2

```

> ### MULTINOMIAL LOGISTIC REGRESSION ###
>
> library(nnet)
> library(dplyr)
> e2_proper$State = ifelse(e2_proper$Question.RESP == "1", "Know",
+                          (ifelse(e2_proper$Question.RESP == "2", "DontKnow",
+                                ifelse(e2_proper$Question.RESP == "3", "Other", "TOT"))))
> e2_proper_state_multinomial = nnet::multinom(data = e2_proper,
+                                               State ~Proper*PrimeCondition*AgeGroup +

```



```
+ (1|Subject))
```

```
# weights: 72 (51 variable)
initial value 9010.913347
iter 10 value 7811.823044
iter 20 value 7553.190151
iter 30 value 7472.761587
iter 40 value 7462.492857
iter 50 value 7461.858412
final value 7461.842801
converged
```

```
> summary(e2_proper_state_multinomial)
```

Call:

```
nnet::multinom(formula = State ~ Proper * PrimeCondition * AgeGroup +
  (1 | Subject), data = e2_proper)
```

Coefficients:

	(Intercept)	Proper2	PrimeCondition1	PrimeCondition3	PrimeCondition4
Know	-0.04605436	0.04177174	0.1683075	0.09196541	-0.06624657
Other	-1.84117972	1.21532388	0.9460282	1.00788145	0.70594995
TOT	-1.29207768	0.35070157	0.7408157	0.60294131	0.57263636
	AgeGroup2 1 Subject1	Proper2:PrimeCondition1	Proper2:PrimeCondition3		
Know	0.6878776	0.04605436	-0.2758295		-0.1070695
Other	2.3265961	1.84117972	-0.6415156		-0.8031425
TOT	1.5930197	1.29207768	-0.3724929		-0.4002630
	Proper2:PrimeCondition4	Proper2:AgeGroup2	PrimeCondition1:AgeGroup2		
Know	-0.2677480		-0.15795239		-0.5803799
Other	-0.6968838		0.06042764		-0.7156478
TOT	-0.6302659		0.22201732		-0.7724448
	PrimeCondition3:AgeGroup2	PrimeCondition4:AgeGroup2			
Know	-0.5645414		-0.7978126		
Other	-1.0886618		-0.7365658		
TOT	-0.7022868		-0.5239481		
	Proper2:PrimeCondition1:AgeGroup2	Proper2:PrimeCondition3:AgeGroup2			
Know		0.7489188		0.5387847	
Other		0.6560719		0.8660903	
TOT		0.5779225		0.6131484	
	Proper2:PrimeCondition4:AgeGroup2				
Know		0.4245608			
Other		0.4771057			
TOT		0.2386811			

Std. Errors:

	(Intercept)	Proper2	PrimeCondition1	PrimeCondition3	PrimeCondition4
Know	0.05741465	0.1494824	0.1644601	0.1642517	0.1643719
Other	0.25309740	0.5597510	0.6119170	0.6026219	0.6225425

```

TOT      0.14969807 0.3684620      0.3772072      0.3828057      0.3789712
      AgeGroup2 1 | Subject1 Proper2:PrimeCondition1 Proper2:PrimeCondition3
Know      0.1708129      0.05741465      0.2140385      0.2132930
Other     0.5536606      0.25309740      0.6887814      0.6842729
TOT      0.3573848      0.14969807      0.4705378      0.4813230
      Proper2:PrimeCondition4 Proper2:AgeGroup2 PrimeCondition1:AgeGroup2
Know              0.2139691      0.2284169      0.2431238
Other              0.7028435      0.6176036      0.6786374
TOT              0.4817688      0.4420235      0.4620980
      PrimeCondition3:AgeGroup2 PrimeCondition4:AgeGroup2
Know              0.2416883      0.2449029
Other              0.6753763      0.6889354
TOT              0.4660618      0.4559898
      Proper2:PrimeCondition1:AgeGroup2 Proper2:PrimeCondition3:AgeGroup2
Know              0.3265596      0.3235821
Other              0.7718589      0.7725509
TOT              0.5793727      0.5863865
      Proper2:PrimeCondition4:AgeGroup2
Know              0.3257103
Other              0.7830637
TOT              0.5803039

Residual Deviance: 14923.69
AIC: 15019.69

```

```
> car::Anova(e2_proper_state_multinomial)
```

Analysis of Deviance Table (Type II tests)

Response: State

	LR	Chisq	Df	Pr(>Chisq)
Proper	135.67	3	< 2.2e-16	***
PrimeCondition	54.16	9	1.764e-08	***
AgeGroup	611.11	3	< 2.2e-16	***
1 Subject	0.00	3	1.000000	
Proper:PrimeCondition	8.15	9	0.518755	
Proper:AgeGroup	13.87	3	0.003086	**
PrimeCondition:AgeGroup	15.98	9	0.067324	.
Proper:PrimeCondition:AgeGroup	6.70	9	0.668549	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

> exp2_state_data <- effects::effect("Proper*AgeGroup",
+                                   e2_proper_state_multinomial,
+   xlevels = list(PrimeCondition = c("B", "P", "R", "U"),
+                 AgeGroup = c("Young", "Old"),
+   Proper = c("ProperName", "Word")))
> s2 = plot(exp2_state_data, main = "Exp 2: WordType x Age")

```

```
>
```

22.2.4 States: E1

```
> ### MULTINOMIAL LOGISTIC REGRESSION ###
>
> library(nnet)
> library(dplyr)
> e1_proper$State = ifelse(e1_proper$Question.RESP == "1", "Know",
+                          (ifelse(e1_proper$Question.RESP == "2", "DontKnow",
+                          ifelse(e1_proper$Question.RESP == "3", "Other", "TOT"))))
> e1_proper_state_multinomial = nnet::multinom(data = e1_proper,
+       State ~ Proper*PrimeCondition*AgeGroup +
+       (1|Subject))
```

```
# weights: 72 (51 variable)
initial value 10119.948836
iter 10 value 9271.488209
iter 20 value 8831.561438
iter 30 value 8673.637604
iter 40 value 8631.678731
iter 50 value 8625.341128
final value 8625.321329
converged
```

```
> summary(e1_proper_state_multinomial)
```

Call:

```
nnet::multinom(formula = State ~ Proper * PrimeCondition * AgeGroup +
  (1 | Subject), data = e1_proper)
```

Coefficients:

	(Intercept)	Proper2	PrimeCondition1	PrimeCondition3	PrimeCondition4
Know	0.2933956	-0.5276913	-0.20485606	-0.1671933	-0.6743378
Other	-1.2694970	0.7118370	-0.12381978	0.7723939	-0.7129614
TOT	-0.6352209	-0.8003760	-0.09290331	0.3149625	0.2700682
	AgeGroup2	1 Subject1	Proper2:PrimeCondition1	Proper2:PrimeCondition3	
Know	-0.18411374	-0.2933956	0.1972837	0.1789249	
Other	1.28625178	1.2694970	-0.1066329	-0.5684689	
TOT	0.07482917	0.6352209	0.5454646	0.1780069	
	Proper2:PrimeCondition4	Proper2:AgeGroup2	PrimeCondition1:AgeGroup2		
Know	0.2772082	0.9348691	0.09967292		
Other	0.6760643	0.9947307	0.16999641		
TOT	0.3082846	1.6435956	-0.02929692		
	PrimeCondition3:AgeGroup2	PrimeCondition4:AgeGroup2			
Know	0.0004757543	0.2071155			
Other	-0.2968425901	0.7570896			

```

TOT          -0.3561390090          -0.1941856
      Proper2:PrimeCondition1:AgeGroup2 Proper2:PrimeCondition3:AgeGroup2
Know          -0.3650391          -0.21424712
Other          -0.1265846          0.08027801
TOT          -0.6325761          -0.26535038
      Proper2:PrimeCondition4:AgeGroup2
Know          -0.4906433
Other          -0.9577954
TOT          -0.8187783

Std. Errors:
      (Intercept)   Proper2 PrimeCondition1 PrimeCondition3 PrimeCondition4
Know    0.05841647  0.1486476      0.1633792      0.1667778      0.1647533
Other   0.17312138  0.3889202      0.4886421      0.4224196      0.5413440
TOT     0.10002608  0.2808074      0.2794221      0.2660621      0.2530152
      AgeGroup2 1 | Subject1 Proper2:PrimeCondition1 Proper2:PrimeCondition3
Know    0.1663470   0.05841647      0.2091133      0.2124187
Other   0.3971132   0.17312138      0.5559073      0.4874986
TOT     0.2760481   0.10002608      0.3788611      0.3693655
      Proper2:PrimeCondition4 Proper2:AgeGroup2 PrimeCondition1:AgeGroup2
Know          0.2118977      0.2245530      0.2333156
Other         0.5943935      0.4530610      0.5574723
TOT           0.3520869      0.3728538      0.3892869
      PrimeCondition3:AgeGroup2 PrimeCondition4:AgeGroup2
Know          0.2372134      0.2356200
Other         0.4917936      0.6008895
TOT           0.3779072      0.3583581
      Proper2:PrimeCondition1:AgeGroup2 Proper2:PrimeCondition3:AgeGroup2
Know          0.3144418      0.3192572
Other         0.6417329      0.5763741
TOT           0.5135621      0.5061991
      Proper2:PrimeCondition4:AgeGroup2
Know          0.3170767
Other         0.6707914
TOT           0.4841503

Residual Deviance: 17250.64
AIC: 17346.64

```

```
> car::Anova(e1_proper_state_multinomial)
```

```
Analysis of Deviance Table (Type II tests)
```

```
Response: State
```

	LR	Chisq	Df	Pr(>Chisq)
Proper	213.11	3	< 2.2e-16	***
PrimeCondition	78.88	9	2.691e-13	***
AgeGroup	553.44	3	< 2.2e-16	***

```

1 | Subject                0.00  3      1.0000
Proper:PrimeCondition      9.91  9      0.3578
Proper:AgeGroup           63.50  3  1.050e-13 ***
PrimeCondition:AgeGroup    10.87  9      0.2851
Proper:PrimeCondition:AgeGroup  6.90  9      0.6477
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> exp1_state_data <- effects::effect("Proper*AgeGroup",
+                                   e1_proper_state_multinomial,
+   xlevels = list(PrimeCondition = c("B", "P", "R", "U"),
+                 AgeGroup = c("Young", "Old"),
+   Proper = c("ProperName", "Word")))
> s1 = plot(exp1_state_data, main = "Exp 1: WordType x Age")
>

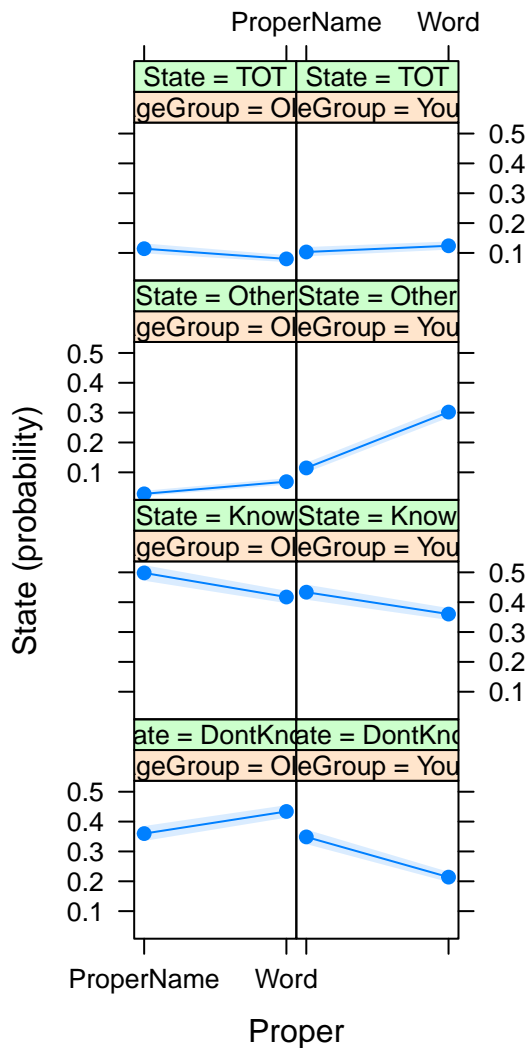
```

```

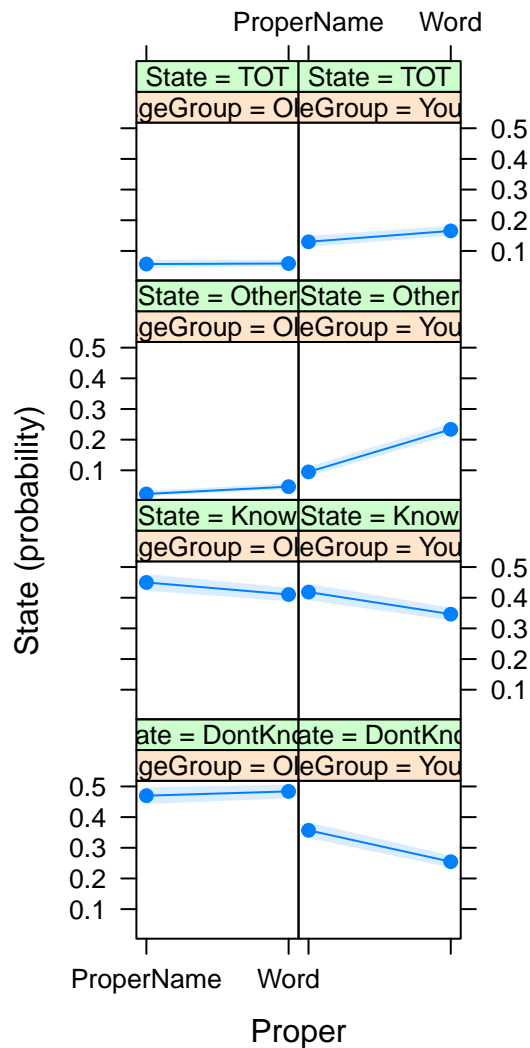
> gridExtra::grid.arrange(s1,s2, nrow = 1, ncol = 2)

```

Exp 1: WordType x Age



Exp 2: WordType x Age



22.2.5 E3: Multiple Choice Errors

```
> library(nnet)
> library(dplyr)
> e3_proper_hlm_multinomial = e3_proper %>% filter(!WhichPrime %in% c("0", "X"))
> contrasts(e3_proper_hlm_multinomial$WhichPrime) = contr.treatment(6, base = 2)
> contrasts(e3_proper_hlm_multinomial$PrimeCondition) = contr.treatment(4, base = 1)
> e3_proper_mcq_error_multinomial = nnet::multinom(data =
+                                     e3_proper_hlm_multinomial,
+                                     WhichPrime ~ Proper*PrimeCondition +
+                                     (1|Subject))
```

```
# weights: 40 (27 variable)
initial value 1687.120237
iter 10 value 1030.012554
iter 20 value 1013.430504
iter 30 value 1012.411813
iter 40 value 1012.391301
final value 1012.391176
converged
```

```
> summary(e3_proper_mcq_error_multinomial)
```

Call:

```
nnet::multinom(formula = WhichPrime ~ Proper * PrimeCondition +
  (1 | Subject), data = e3_proper_hlm_multinomial)
```

Coefficients:

	(Intercept)	Proper2	PrimeCondition2	PrimeCondition3	PrimeCondition4
P	-2.01350322	0.85959013	0.1989375	1.0567554	0.3377421
R	-0.03708411	0.08465132	0.1968128	0.8765347	0.3926347
U	-1.66622911	-1.22078817	-12.1246504	-0.3307636	-12.1355193

1 Subject1	Proper2:PrimeCondition2	Proper2:PrimeCondition3
P	2.01350322	1.53899202
R	0.03708411	0.09437654
U	1.66622911	13.51388140

	Proper2:PrimeCondition4
P	0.9707791
R	-0.1825748
U	13.9543295

Std. Errors:

	(Intercept)	Proper2	PrimeCondition2	PrimeCondition3	PrimeCondition4
P	0.50484271	1.131350	1.4285167	1.2429939	1.4299836
R	0.09629031	0.240895	0.2793878	0.2724391	0.2833179
U	0.35984541	1.236062	0.5825275	1.2422863	0.5529343

1 Subject1	Proper2:PrimeCondition2	Proper2:PrimeCondition3
P	0.50484271	1.5407976
R	0.09629031	0.3514282
U	0.35984541	0.5825061

	Proper2:PrimeCondition4
P	1.5497293
R	0.3530096
U	0.5529156

Residual Deviance: 2024.782

AIC: 2072.782

```
> car::Anova(e3_proper_mcq_error_multinomial)
```

```
Analysis of Deviance Table (Type II tests)
```

```
Response: WhichPrime
```

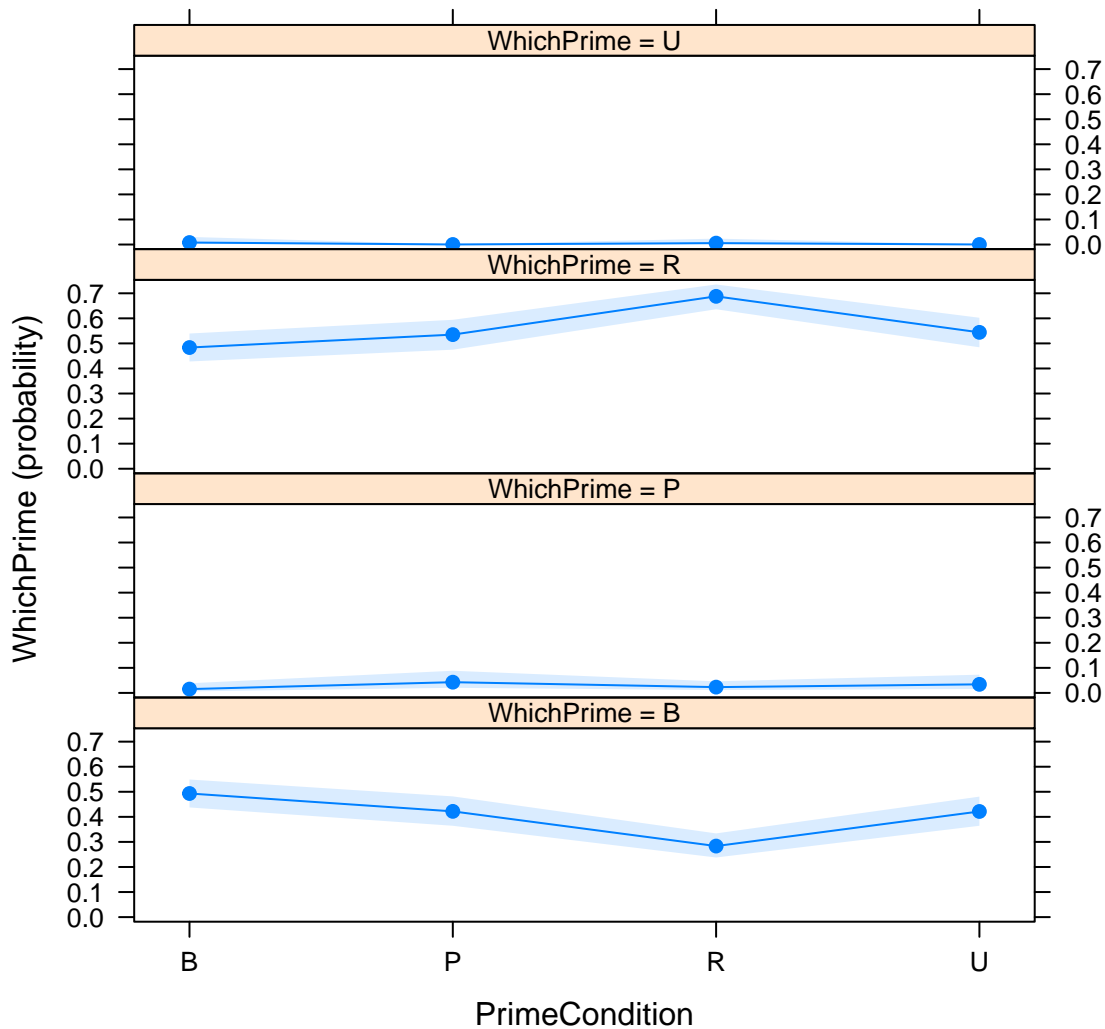
	LR	Chisq	Df	Pr(>Chisq)
Proper	15.001	3	0.001816	**
PrimeCondition	45.354	9	7.934e-07	***
1 Subject	0.000	3	1.000000	
Proper:PrimeCondition	10.052	9	0.346273	

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> exp3_plot_data <- effects::effect("PrimeCondition",
+                                 e3_proper_mcq_error_multinomial,
+   xlevels = list(PrimeCondition = c("B", "P", "R", "U"),
+                                 Proper = c("ProperName", "Word")))
> plot(exp3_plot_data)
```


PrimeCondition effect plot



22.2.6 E2:Multiple Choice Errors

```
> ### MULTINOMIAL LOGISTIC REGRESSION ###
>
> library(nnet)
> library(dplyr)
> e2_proper_hlm_multinomial = e2_proper %>% filter(!WhichPrime %in% c("0", "X"))
> contrasts(e2_proper_hlm_multinomial$WhichPrime) = contr.treatment(6, base = 2)
> contrasts(e2_proper_hlm_multinomial$PrimeCondition) = contr.treatment(4, base = 1)
> e2_proper_mcq_error_multinomial = nnet::multinom(data =
+ e2_proper_hlm_multinomial,
```

```
+          WhichPrime ~ Proper*PrimeCondition*AgeGroup +
+          (1|Subject))
```

```
# weights: 72 (51 variable)
initial value 2467.603963
iter 10 value 1814.785830
iter 20 value 1590.768991
iter 30 value 1574.569417
iter 40 value 1571.235113
iter 50 value 1570.925684
iter 60 value 1570.890476
final value 1570.889633
converged
```

```
> summary(e2_proper_mcq_error_multinomial)
```

Call:

```
nnet::multinom(formula = WhichPrime ~ Proper * PrimeCondition *
  AgeGroup + (1 | Subject), data = e2_proper_hlm_multinomial)
```

Coefficients:

	(Intercept)	Proper2	PrimeCondition2	PrimeCondition3	PrimeCondition4
P	-1.2628385	0.5798065	1.1393988	1.3017553	-0.3643811
R	0.2223507	-0.3940291	0.3207092	0.4109248	0.4491930
U	-1.6094372	-12.3351537	0.2220121	-18.1672062	1.0220784

	AgeGroup2 1 Subject1	Proper2:PrimeCondition2	Proper2:PrimeCondition3
P	0.2745237	1.2628385	-0.54804179
R	0.1944068	-0.2223507	-0.05584338
U	0.2750750	1.6094372	12.30351468

	Proper2:PrimeCondition4	Proper2:AgeGroup2	PrimeCondition2:AgeGroup2
P	0.4740931	-0.3568257	-20.8639157
R	-0.2565918	0.4867881	-0.9975065
U	12.2437876	11.6413849	-0.5732274

	PrimeCondition3:AgeGroup2	PrimeCondition4:AgeGroup2
P	-2.879018	0.6415789
R	-1.743193	-0.7800052
U	-5.610964	-1.6610867

	Proper2:PrimeCondition2:AgeGroup2	Proper2:PrimeCondition3:AgeGroup2
P	20.20367941	2.377417
R	0.06859764	0.434255
U	-11.66446548	13.526044

	Proper2:PrimeCondition4:AgeGroup2
P	-1.303064
R	0.198545
U	-11.057545

Std. Errors:

	(Intercept)	Proper2	PrimeCondition2	PrimeCondition3	PrimeCondition4
--	-------------	---------	-----------------	-----------------	-----------------

```

P    0.3674166 0.8024177      0.8887992      0.893769      1.2630501
R    0.1281028 0.3016190      0.3726858      0.386604      0.3793782
U    0.5099007 0.9142098      1.4460501      0.310408      1.2630794
  AgeGroup2 1 | Subject1 Proper2:PrimeCondition2 Proper2:PrimeCondition3
P    1.0452563      0.3674166      0.9861472      1.0222346
R    0.3821682      0.1281028      0.4383164      0.4482917
U    1.4463761      0.5099007      1.1413207      0.3104080
  Proper2:PrimeCondition4 Proper2:AgeGroup2 PrimeCondition2:AgeGroup2
P          1.3433502          1.1928018          0.3772377
R          0.4417207          0.4586494          0.5429599
U          1.0360272          0.9141904          2.0443366
  PrimeCondition3:AgeGroup2 PrimeCondition4:AgeGroup2
P          1.5404275          1.5413085
R          0.5432235          0.5220399
U          0.3104080          1.9168845
  Proper2:PrimeCondition2:AgeGroup2 Proper2:PrimeCondition3:AgeGroup2
P          0.3772377          1.7495037
R          0.6488017          0.6448865
U          1.5291683          0.3104080
  Proper2:PrimeCondition4:AgeGroup2
P          1.739643
R          0.623373
U          1.410742

Residual Deviance: 3141.779
AIC: 3237.779

```

```
> car::Anova(e2_proper_mcq_error_multinomial)
```

```
Analysis of Deviance Table (Type II tests)
```

```
Response: WhichPrime
```

	LR	Chisq	Df	Pr(>Chisq)
Proper	2.340	3	0.5048566	
PrimeCondition	15.152	9	0.0868543 .	
AgeGroup	11.465	3	0.0094586 **	
1 Subject	0.000	3	1.0000000	
Proper:PrimeCondition	9.386	9	0.4024761	
Proper:AgeGroup	9.298	3	0.0255852 *	
PrimeCondition:AgeGroup	33.017	9	0.0001326 ***	
Proper:PrimeCondition:AgeGroup	10.376	9	0.3209049	

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```

> exp2_plot_data <- effects::effect("Proper*AgeGroup",
+                                   e2_proper_mcq_error_multinomial,
+   xlevels = list(PrimeCondition = c("B", "P", "R", "U"),
+                 AgeGroup = c("Young", "Old")))

```

```
> m2= plot(exp2_plot_data, main = "Exp2: Multiple Choice Errors")
>
```

22.2.7 E1: Multiple Choice Errors

```
> ### MULTINOMIAL LOGISTIC REGRESSION ###
>
> library(nnet)
> library(dplyr)
> e1_proper_hlm_multinomial = e1_proper %>% filter(!WhichPrime %in% c("0", "X"))
> contrasts(e1_proper_hlm_multinomial$WhichPrime) = contr.treatment(6, base = 2)
> contrasts(e1_proper_hlm_multinomial$PrimeCondition) = contr.treatment(4, base = 1)
> e1_proper_mcq_error_multinomial = nnet::multinom(data =
+                                     e1_proper_hlm_multinomial,
+                                     WhichPrime ~ Proper*PrimeCondition*AgeGroup +
+                                     (1|Subject))
```

```
# weights: 72 (51 variable)
initial value 2951.420695
iter 10 value 2086.053171
iter 20 value 1856.329288
iter 30 value 1827.632429
iter 40 value 1822.067187
iter 50 value 1821.289065
iter 60 value 1821.277652
final value 1821.276515
converged
```

```
> summary(e1_proper_mcq_error_multinomial)
```

```
Call:
nnet::multinom(formula = WhichPrime ~ Proper * PrimeCondition *
  AgeGroup + (1 | Subject), data = e1_proper_hlm_multinomial)

Coefficients:
(Intercept)    Proper2 PrimeCondition2 PrimeCondition3 PrimeCondition4
P -1.71737624  0.2651641    1.35540919    -10.497523      1.1326940
R  0.06069293 -0.4118583    0.06788236     1.447261      0.2841144
U -8.38822994 -2.7389318    -6.03760719    -1.923290     15.3901737
  AgeGroup2 1 | Subject1 Proper2:PrimeCondition2 Proper2:PrimeCondition3
P -0.5547238  1.71737624          0.8825657          10.83394494
R -0.4215059 -0.06069293          0.3636624          -0.05813348
U 12.7869978  8.38822994          4.2612542          17.50560812
 Proper2:PrimeCondition4 Proper2:AgeGroup2 PrimeCondition2:AgeGroup2
P -0.2488410    1.7894176    -0.02837614
R  0.4061841    0.6665401    -0.03245388
U  3.2664944   -10.4776685     6.26640158
```

```

PrimeCondition3:AgeGroup2 PrimeCondition4:AgeGroup2
P      10.6382433          -14.6864765
R      -0.8541192           0.1929604
U       3.1616226          -14.8329444
  Proper2:PrimeCondition2:AgeGroup2 Proper2:PrimeCondition3:AgeGroup2
P      -1.6835926          -11.4677210
R       0.3723014           0.2109068
U       9.6026012          -4.6594324
  Proper2:PrimeCondition4:AgeGroup2
P      13.8354305
R      -0.4677391
U      11.2571489

Std. Errors:
(Intercept)   Proper2 PrimeCondition2 PrimeCondition3 PrimeCondition4
P    0.5081856  1.1141781      1.1865853      0.8166578      1.2580806
R    0.1233175  0.2836691      0.3701065      0.3760132      0.3796896
U    0.3587192  0.3893380      0.7144422      0.6984064      0.6968840
  AgeGroup2 1 | Subject1 Proper2:PrimeCondition2 Proper2:PrimeCondition3
P  1.4324749    0.5081856              1.2922777              0.8166537
R  0.3230279    0.1233175              0.4300317              0.4326308
U  0.6294696    0.3587192              0.4965092              0.4686026
  Proper2:PrimeCondition4 Proper2:AgeGroup2 PrimeCondition2:AgeGroup2
P      1.4052611          1.5324496          1.6683837
R      0.4383105          0.3830474          0.4837938
U      0.3937528          0.5587619          0.7144422
  PrimeCondition3:AgeGroup2 PrimeCondition4:AgeGroup2
P      0.8166249          0.3856697
R      0.4712268          0.4969598
U      0.6636088          0.9069269
  Proper2:PrimeCondition2:AgeGroup2 Proper2:PrimeCondition3:AgeGroup2
P      1.8018543          0.8953089
R      0.5811357          0.5590902
U      0.4965092          0.5516610
  Proper2:PrimeCondition4:AgeGroup2
P      0.3856671
R      0.5855723
U      0.7795470

Residual Deviance: 3642.553
AIC: 3738.553

```

```
> car::Anova(e1_proper_mcq_error_multinomial)
```

```
Analysis of Deviance Table (Type II tests)
```

```
Response: WhichPrime
```

```
LR Chisq Df Pr(>Chisq)
```

```

Proper          18.731  3  0.0003107 ***
PrimeCondition  182.733  9  < 2.2e-16 ***
AgeGroup        5.955  3  0.1137991
1 | Subject     0.000  3  1.0000000
Proper:PrimeCondition  8.062  9  0.5278638
Proper:AgeGroup    12.236  3  0.0066182 **
PrimeCondition:AgeGroup  41.276  9  4.455e-06 ***
Proper:PrimeCondition:AgeGroup  8.551  9  0.4797450
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> exp1_plot_data <- effects::effect("Proper*AgeGroup",
+                                   e1_proper_mcq_error_multinomial,
+                                   xlevels = list(PrimeCondition = c("B", "P", "R", "U"),
+                                   AgeGroup = c("Young", "Old")))
> m1 = plot(exp1_plot_data, main = "Exp1: Multiple Choice Errors")
>

```

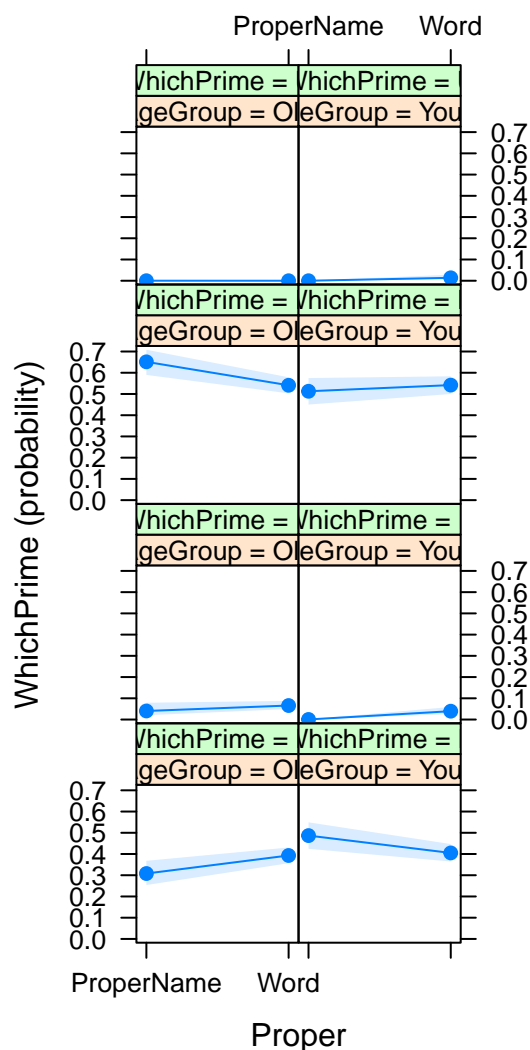
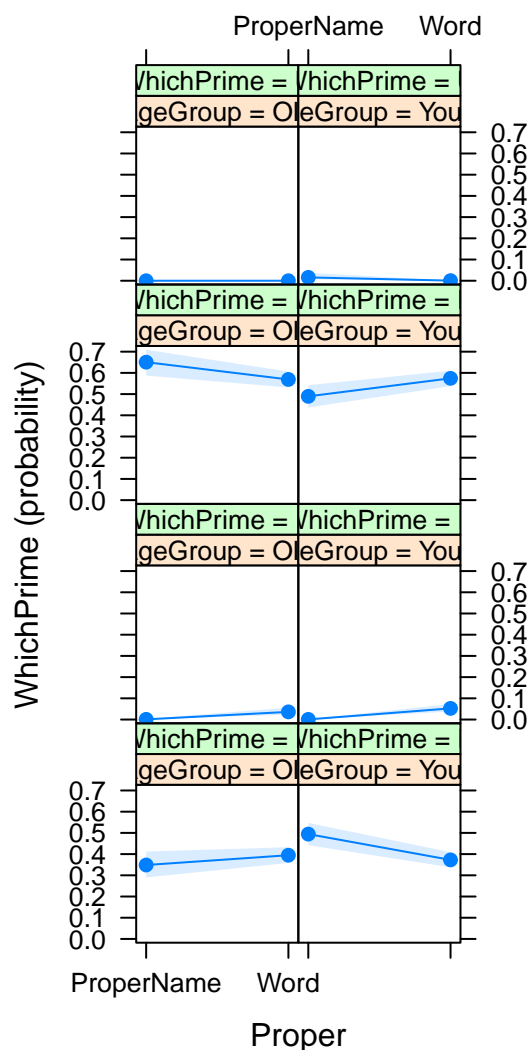
```

> gridExtra::grid.arrange(m1,m2, nrow = 1, ncol = 2)

```

Exp1: Multiple Choice Errors

Exp2: Multiple Choice Errors



23 Item Analyses

23.1 Using AGG data

```
> main = read.csv("Julie_Main5Studies.csv", header = TRUE, sep = ",")
> main$StudyNo = as.factor(main$StudyNo)
> main$PrimeCondition = as.factor(main$PrimeCondition)
> main_wide = read.csv("MainJulieagg_5studies.csv", header = TRUE, sep = ",")
> library(dplyr)
> e1_hlm = main %>% filter(StudyNo == "2" | StudyNo == "4")
```

```
> e2_hlm = main %>% filter(StudyNo == "5" | StudyNo == "6")
> e3_hlm = main %>% filter(StudyNo == "1")
```

23.2 Experiment 1

```
> e1_item_acc = group_by(e1_hlm, Target, AgeGroup, PrimeCondition) %>%
+ summarise_at(vars(Accuracy), mean)
> e1_item_state = group_by(e1_hlm, Target, AgeGroup, Question.RESP) %>%
+ summarise(StateCount = n())
> e1_item_mcqacc = group_by(e1_hlm, Target, AgeGroup, PrimeCondition) %>%
+ summarise_at(vars(McAcc), mean)
```

23.2.1 Target Accuracy

```
> ### TARGET RETRIEVAL ACCURACY
>
> exp1_item_acc = aov(data = e1_item_acc, Accuracy ~ AgeGroup*PrimeCondition +
+ Error (Target/(AgeGroup*PrimeCondition)))
> summary(exp1_item_acc)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99  33.22  0.3355

Error: Target:AgeGroup
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  0.002 0.00201  0.036  0.851
Residuals 99  5.585 0.05641

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3  2.119  0.7062  44.66 <2e-16 ***
Residuals      297  4.697  0.0158
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition  3  0.036 0.01189  0.828  0.48
Residuals      297  4.268 0.01437
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> exp1_target_lsm = lsmeans::lsmeans(exp1_item_acc,
+ c("AgeGroup", "PrimeCondition"))
```



```
> prime_effect = cld(exp1_target_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = c("AgeGroup"))
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
2	B - U	Old	0.0533333	0.0173738	592.6483	3.069753	0.0119826
3	B - R	Old	0.0533333	0.0173738	592.6483	3.069753	0.0119826
4	P - U	Old	0.1366667	0.0173738	592.6483	7.866242	0.0000000
5	P - R	Old	0.1366667	0.0173738	592.6483	7.866242	0.0000000
6	P - B	Old	0.0833333	0.0173738	592.6483	4.796489	0.0000122
10	P - U	Young	0.1240000	0.0173738	592.6483	7.137176	0.0000000
11	P - R	Young	0.1058889	0.0173738	592.6483	6.094739	0.0000000
12	P - B	Young	0.0865556	0.0173738	592.6483	4.981954	0.0000049

```
> ## specific t-tests
> e1_item_acc_collapsed = group_by(e1_hlm, Target, PrimeCondition) %>%
+   summarise_at(vars(Accuracy), mean)
> target_p = e1_item_acc_collapsed %>% filter(PrimeCondition == "P")
> target_r = e1_item_acc_collapsed %>% filter(PrimeCondition == "R")
> target_b = e1_item_acc_collapsed %>% filter(PrimeCondition == "B")
> target_u = e1_item_acc_collapsed %>% filter(PrimeCondition == "U")
> t.test(target_p$Accuracy, target_r$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_r$Accuracy
t = 10.011, df = 99, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.1037881 0.1551008
sample estimates:
mean of the differences
 0.1294444
```

```
> t.test(target_p$Accuracy, target_b$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_b$Accuracy
t = 5.9487, df = 99, p-value = 4.089e-08
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.0563947 0.1128451
sample estimates:
mean of the differences
 0.08461988
```

```
> t.test(target_p$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_u$Accuracy
t = 9.7028, df = 99, p-value = 4.869e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.09618115 0.14563171
sample estimates:
mean of the differences
      0.1209064
```

```
> t.test(target_b$Accuracy, target_r$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_b$Accuracy and target_r$Accuracy
t = 3.8146, df = 99, p-value = 0.0002375
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.02150844 0.06814068
sample estimates:
mean of the differences
      0.04482456
```

```
> t.test(target_b$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_b$Accuracy and target_u$Accuracy
t = 3.0836, df = 99, p-value = 0.002651
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01293676 0.05963634
sample estimates:
mean of the differences
      0.03628655
```

```
> t.test(target_r$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_r$Accuracy and target_u$Accuracy
t = -0.70346, df = 99, p-value = 0.4834
alternative hypothesis: true difference in means is not equal to 0
```

```

95 percent confidence interval:
-0.03262062  0.01554460
sample estimates:
mean of the differences
-0.008538012

```

23.2.2 State Data

```

> ## e1_item_state has many rows missing so we cannot use that dataset.
> ## we will use the Julieagg.R file to create an agg file for items.
>
> exp1_state = subset(final_statedata, final_statedata$StudyNo == '2' |
+                       final_statedata$StudyNo == '4')
> exp1_state_prime = subset(statedata_primetype_long,
+                           statedata_primetype_long$StudyNo == '2' |
+                           statedata_primetype_long$StudyNo == '4')
> exp1_state_prime$PrimeCondition = as.factor(as.character(exp1_state_prime$PrimeCondition))
> exp1_state_prime$State = as.factor(as.character(exp1_state_prime$State))
> exp1_state_prime$Target = as.factor(as.character(exp1_state_prime$Target))
> ## just state
> exp1_state_aov = aov(data = exp1_state, Trials ~ AgeGroup*State +
+                       Error(Target/(AgeGroup*State)))
> summary(exp1_state_aov)

```

```

Error: Target
      Df    Sum Sq Mean Sq F value Pr(>F)
Residuals 99 2.227e-25 2.25e-27

Error: Target:AgeGroup
      Df    Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1 2.700e-27 2.688e-27  0.331  0.566
Residuals 99 8.037e-25 8.118e-27

Error: Target:State
      Df Sum Sq Mean Sq F value Pr(>F)
State   3  18316    6105   131.9 <2e-16 ***
Residuals 297  13744      46

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:State
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:State  3   3131    1044   57.9 <2e-16 ***
Residuals      297   5353      18

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> exp1_state_lsm = lsmeans::lsmeans(exp1_state_aov, c("AgeGroup", "State"))
> prime_effect = cld(exp1_state_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = c("State"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```

contrast	State	estimate	SE	df	t.ratio	p.value
Old - Young	dontknow	4.80	0.5199675	297	9.231347	0.0000000
Old - Young	know	1.90	0.5199675	297	3.654075	0.0003051
Old - Young	other	5.95	0.5199675	297	11.443024	0.0000000

```
> ##state by prime
> exp1_stateprime_aov = aov(data = exp1_state_prime,
+                            Trials ~ AgeGroup*PrimeCondition*State +
+                            Error(Target/(AgeGroup*PrimeCondition*State)))
> summary(exp1_stateprime_aov)
```

```
Error: Target
      Df    Sum Sq   Mean Sq F value Pr(>F)
Residuals 99 6.432e-25 6.497e-27

Error: Target:AgeGroup
      Df    Sum Sq   Mean Sq F value Pr(>F)
AgeGroup 1 1.620e-27 1.619e-27  0.514  0.475
Residuals 99 3.122e-25 3.153e-27

Error: Target:PrimeCondition
      Df    Sum Sq   Mean Sq F value Pr(>F)
PrimeCondition 3 4.800e-27 1.590e-27  0.97  0.407
Residuals      297 4.868e-25 1.639e-27

Error: Target:State
      Df Sum Sq Mean Sq F value Pr(>F)
State   3   4579  1526.3   131.9 <2e-16 ***
Residuals 297   3436    11.6

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition
      Df    Sum Sq   Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition 3 5.600e-27 1.860e-27  0.928  0.428
Residuals              297 5.953e-25 2.004e-27

Error: Target:AgeGroup:State
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:State 3  782.7  260.90   57.9 <2e-16 ***
```

```

Residuals      297 1338.3      4.51
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeCondition:State
              Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition:State  9  220.8   24.533    13.52 <2e-16 ***
Residuals          891 1617.2     1.815
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition:State
              Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition:State  9   13.1     1.455    0.911  0.515
Residuals          891 1422.9     1.597

```

```

> library(ez)
> ezANOVA(data = exp1_state_prime, wid = .(Target),
+         dv = .(Trials), within = .(PrimeCondition, State),
+         between = .(AgeGroup))

```

```

$ANOVA
      Effect DFn  DFd      F      p p<.05
2      AgeGroup    1   198 -5.920075e-13 1.000000e+00
3 PrimeCondition    3   594  1.169625e-13 1.000000e+00
5           State    3   594  1.898986e+02 2.484828e-86      *
4 AgeGroup:PrimeCondition    3   594  4.010453e-13 1.000000e+00
6 AgeGroup:State    3   594  3.245945e+01 1.936966e-19      *
7 PrimeCondition:State    9  1782  1.438037e+01 1.157611e-22      *
8 AgeGroup:PrimeCondition:State    9  1782  8.527043e-01 5.674172e-01
ges
2 1.906245e-32
3 2.397542e-32
5 3.694688e-01
4 8.220776e-32
6 9.104058e-02
7 2.747865e-02
8 1.672622e-03

$`Mauchly's Test for Sphericity`
      Effect      W      p p<.05
3 PrimeCondition 0.3554475 5.199325e-42      *
4 AgeGroup:PrimeCondition 0.3554475 5.199325e-42      *
5           State 0.2667363 4.113432e-54      *
6 AgeGroup:State 0.2667363 4.113432e-54      *
7 PrimeCondition:State 0.3359210 6.431198e-24      *
8 AgeGroup:PrimeCondition:State 0.3359210 6.431198e-24      *

```

\$`Sphericity Corrections`

	Effect	GGe	p[GG]	p[GG]<.05	HFe
3	PrimeCondition	0.33333333	9.999997e-01		0.33333333
4	AgeGroup:PrimeCondition	0.33333333	9.999995e-01		0.33333333
5	State	0.5686656	2.220738e-50	*	0.5731086
6	AgeGroup:State	0.5686656	4.121646e-12	*	0.5731086
7	PrimeCondition:State	0.7978142	1.486851e-18	*	0.8308514
8	AgeGroup:PrimeCondition:State	0.7978142	5.460060e-01		0.8308514

p[HF] p[HF]<.05

3	9.999997e-01	
4	9.999995e-01	
5	9.459746e-51	*
6	3.462026e-12	*
7	3.163980e-19	*
8	5.498147e-01	

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> exp1_state_lsm = lsmeans::lsmeans(exp1_stateprime_aov, c("AgeGroup","PrimeCondition",
> prime_effect = cld(exp1_state_lsm, alpha = 0.05,
+ adjust = "tukey", details = TRUE, by = c("PrimeCondition", "AgeGroup"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```

	contrast	PrimeCondition	AgeGroup	estimate	SE	df	t.ratio	p.value
2	dontknow - other	b	Old	3.18	0.2564724	1197.602	12.3989	
3	dontknow - TOT	b	Old	2.78	0.2564724	1197.602	10.8393	
4	know - other	b	Old	3.90	0.2564724	1197.602	15.2063	
5	know - TOT	b	Old	3.50	0.2564724	1197.602	13.6466	
6	know - dontknow	b	Old	0.72	0.2564724	1197.602		
2.807319	0.0260869							
7	TOT - other	b	Young	0.84	0.2564724	1197.602		
3.275206	0.0059743							
8	dontknow - other	b	Young	1.41	0.2564724	1197.602		
5.497667	0.0000003							
10	know - other	b	Young	2.71	0.2564724	1197.602	10.5664	
11	know - TOT	b	Young	1.87	0.2564724	1197.602		
7.291232	0.0000000							
12	know - dontknow	b	Young	1.30	0.2564724	1197.602		
5.068771	0.0000028							
14	dontknow - other	p	Old	2.98	0.2564724	1197.602	11.6191	
15	dontknow - TOT	p	Old	2.83	0.2564724	1197.602	11.0343	
16	know - other	p	Old	4.03	0.2564724	1197.602	15.7131	
17	know - TOT	p	Old	3.88	0.2564724	1197.602	15.1283	
18	know - dontknow	p	Old	1.05	0.2564724	1197.602		
4.094007	0.0002640							
19	TOT - other	p	Young	0.74	0.2564724	1197.602		
2.885300	0.0207377							

20 dontknow - other p	Young		1.13	0.2564724	1197.602	
4.405931 0.0000677						
22 know - other p	Young		2.97	0.2564724	1197.602	11.5801
23 know - TOT p	Young		2.23	0.2564724	1197.602	
8.694891 0.0000000						
24 know - dontknow p	Young		1.84	0.2564724	1197.602	
7.174260 0.0000000						
26 dontknow - other r	Old		2.72	0.2564724	1197.602	10.6054
27 dontknow - TOT r	Old		2.45	0.2564724	1197.602	
9.552683 0.0000000						
28 know - other r	Old		3.49	0.2564724	1197.602	13.6077
29 know - TOT r	Old		3.22	0.2564724	1197.602	12.5549
30 know - dontknow r	Old		0.77	0.2564724	1197.602	
3.002272 0.0145194						
31 TOT - other r	Young		1.14	0.2564724	1197.602	
4.444922 0.0000567						
32 dontknow - other r	Young		1.24	0.2564724	1197.602	
4.834827 0.0000090						
34 know - other r	Young		2.50	0.2564724	1197.602	
9.747636 0.0000000						
35 know - TOT r	Young		1.36	0.2564724	1197.602	
5.302714 0.0000008						
36 know - dontknow r	Young		1.26	0.2564724	1197.602	
4.912808 0.0000061						
37 TOT - other u	Old		0.69	0.2564724	1197.602	
2.690348 0.0363528						
38 dontknow - other u	Old		2.80	0.2564724	1197.602	10.9173
39 dontknow - TOT u	Old		2.11	0.2564724	1197.602	
8.227005 0.0000000						
40 know - other u	Old		3.67	0.2564724	1197.602	14.3095
41 know - TOT u	Old		2.98	0.2564724	1197.602	11.6191
42 know - dontknow u	Old		0.87	0.2564724	1197.602	
3.392177 0.0039876						
43 TOT - other u	Young		0.97	0.2564724	1197.602	
3.782083 0.0009373						
44 dontknow - other u	Young		1.71	0.2564724	1197.602	
6.667383 0.0000000						
45 dontknow - TOT u	Young		0.74	0.2564724	1197.602	
2.885300 0.0207377						
46 know - other u	Young		1.88	0.2564724	1197.602	
7.330222 0.0000000						
47 know - TOT u	Young		0.91	0.2564724	1197.602	
3.548140 0.0022760						

```

> ### INDIVIDUAL T-TESTS FOR AGExSTATE interaction
>
> e1_young_dk = exp1_state %>% filter(AgeGroup == "Young" & State == "dontknow")
> e1_old_dk = exp1_state %>% filter(AgeGroup == "Old" & State == "dontknow")

```

```
> t.test(e1_old_dk$Trials, e1_young_dk$Trials)
```

```
Welch Two Sample t-test
```

```
data: e1_old_dk$Trials and e1_young_dk$Trials
t = 5.837, df = 197.87, p-value = 2.146e-08
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 3.178335 6.421665
sample estimates:
mean of x mean of y
 14.48      9.68
```

```
> e1_young_other = exp1_state %>% filter(AgeGroup == "Young" & State == "other")
> e1_old_other = exp1_state %>% filter(AgeGroup == "Old" & State == "other")
> t.test(e1_young_other$Trials, e1_old_other$Trials)
```

```
Welch Two Sample t-test
```

```
data: e1_young_other$Trials and e1_old_other$Trials
t = 12.087, df = 119.23, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 4.975274 6.924726
sample estimates:
mean of x mean of y
 7.88      1.93
```

```
>
```

23.2.3 Multiple Choice

```
> exp1_mcq = subset(final_mcq, final_mcq$StudyNo == '2' | final_mcq$StudyNo == '4')
> ## MULTIPLE CHOICE ACCURACY
> library(dplyr)
> exp1_mcq_acc = group_by(exp1_mcq, Target, PrimeType, AgeGroup) %>%
+   summarise_at(vars(MCQAcc), mean)
> exp1_mcq_acc_aov = aov(data = exp1_mcq_acc, MCQAcc ~ AgeGroup*PrimeType +
+   Error(Target/(AgeGroup*PrimeType)))
> summary(exp1_mcq_acc_aov)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99  29.58  0.2988

Error: Target:AgeGroup
```



```

      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  0.125  0.12500    2.406   0.124
Residuals 99  5.144  0.05195

Error: Target:PrimeType
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeType  3  1.053  0.3511    18.55 4.69e-11 ***
Residuals 297  5.623  0.0189
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeType
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeType  3  0.032  0.01060    0.477   0.699
Residuals          297  6.601  0.02223

```

```

> expl_mcqacc_lsm = lsmeans::lsmeans(expl_mcq_acc_aov, c("AgeGroup", "PrimeType"))
> prime_effect = cld(expl_mcqacc_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = c("AgeGroup"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))

```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
2	u - r	Old	0.0588889	0.0202871	590.2191	2.902776	0.0200024
4	p - r	Old	0.0777778	0.0202871	590.2191	3.833855	0.0008035
5	p - b	Old	0.0622222	0.0202871	590.2191	3.067084	0.0120858
8	u - r	Young	0.0855556	0.0202871	590.2191	4.217240	0.0001676
10	p - r	Young	0.1088889	0.0202871	590.2191	5.367396	0.0000007
11	p - b	Young	0.0644444	0.0202871	590.2191	3.176622	0.0085051

```

> ## SPECIFIC T TESTS
>
> e1_mcq_p = expl_mcq_acc %>% filter(PrimeType == "p")
> e1_mcq_r = expl_mcq_acc %>% filter(PrimeType == "r")
> e1_mcq_b = expl_mcq_acc %>% filter(PrimeType == "b")
> e1_mcq_u = expl_mcq_acc %>% filter(PrimeType == "u")
> t.test(e1_mcq_p$MCQAcc, e1_mcq_r$MCQAcc, paired = TRUE)

```

```

Paired t-test

data:  e1_mcq_p$MCQAcc and e1_mcq_r$MCQAcc
t = 6.2072, df = 199, p-value = 3.08e-09
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.0636822 0.1229845
sample estimates:
mean of the differences
 0.09333333

```

```
> t.test(e1_mcq_p$MCQAcc, e1_mcq_b$MCQAcc, paired = TRUE)
```

Paired t-test

```
data: e1_mcq_p$MCQAcc and e1_mcq_b$MCQAcc
t = 4.4499, df = 199, p-value = 1.429e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.03526759 0.09139908
sample estimates:
mean of the differences
      0.06333333
```

```
> t.test(e1_mcq_p$MCQAcc, e1_mcq_u$MCQAcc, paired = TRUE)
```

Paired t-test

```
data: e1_mcq_p$MCQAcc and e1_mcq_u$MCQAcc
t = 1.3927, df = 199, p-value = 0.1653
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-0.008781092 0.051003314
sample estimates:
mean of the differences
      0.02111111
```

```
> t.test(e1_mcq_b$MCQAcc, e1_mcq_r$MCQAcc, paired = TRUE)
```

Paired t-test

```
data: e1_mcq_b$MCQAcc and e1_mcq_r$MCQAcc
t = 2.2001, df = 199, p-value = 0.02895
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.003110736 0.056889264
sample estimates:
mean of the differences
      0.03
```

```
> t.test(e1_mcq_b$MCQAcc, e1_mcq_u$MCQAcc, paired = TRUE)
```

Paired t-test

```
data: e1_mcq_b$MCQAcc and e1_mcq_u$MCQAcc
t = -2.9048, df = 199, p-value = 0.004091
alternative hypothesis: true difference in means is not equal to 0
```

```

95 percent confidence interval:
 -0.07088545 -0.01355899
sample estimates:
mean of the differences
 -0.04222222

```

```
> t.test(e1_mcq_r$MCQAcc, e1_mcq_r$MCQAcc, paired = TRUE)
```

Paired t-test

```

data: e1_mcq_r$MCQAcc and e1_mcq_r$MCQAcc
t = NaN, df = 199, p-value = NA
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 NaN NaN
sample estimates:
mean of the differences
 0

```

```

> ### MULTIPLE CHOICE ERRORS
> ## before we do ANOVA, we need to replace NAs with 0.
> for (i in 1: nrow(exp1_mcq)){
+   if(is.na(exp1_mcq[i,9])){
+     exp1_mcq[i,9] = 0
+   }
+ }
> exp1_mcq_aov = aov(data = exp1_mcq,
+   Proportion ~ AgeGroup*PrimeType*ChosenPrime +
+   Error(Target/(AgeGroup*PrimeType*ChosenPrime)))
> summary(exp1_mcq_aov)

```

Error: Target

```

      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99  10.09  0.1019

```

Error: Target:AgeGroup

```

      Df Sum Sq Mean Sq F value  Pr(>F)
AgeGroup  1  0.337  0.3373    7.379 0.00779 **
Residuals 99  4.526  0.0457

```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Error: Target:PrimeType

```

      Df Sum Sq Mean Sq F value  Pr(>F)
PrimeType  3  0.324  0.10814    4.146 0.00671 **
Residuals 297  7.747  0.02608

```

```

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
ChosenPrime    3   89.19   29.730    117.6 <2e-16 ***
Residuals    297   75.07    0.253
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeType
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeType    3   0.112   0.03742    1.422  0.236
Residuals            297   7.814   0.02631

Error: Target:AgeGroup:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:ChosenPrime    3   0.686   0.22877    2.508  0.059 .
Residuals            297  27.089   0.09121
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeType:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType:ChosenPrime    9   11.64    1.2932    20.24 <2e-16 ***
Residuals            891   56.92    0.0639
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeType:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeType:ChosenPrime    9    1.31    0.1458    2.652 0.0049 **
Residuals            891   49.00    0.0550
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> library(ez)
> ezANOVA(data = exp1_mcq, wid = .(Target),
+         dv = .(Proportion), within = .(AgeGroup, PrimeType, ChosenPrime))

```

```

$ANOVA
      Effect DFn  DFd      F      p p<.05
2      AgeGroup    1   99  7.379057 7.788985e-03 *
3      PrimeType    3  297  4.146004 6.712815e-03 *
4      ChosenPrime    3  297 117.630518 3.202508e-50 *
5      AgeGroup:PrimeType    3  297  1.422257 2.363400e-01
6      AgeGroup:ChosenPrime    3  297  2.508187 5.903367e-02
7      PrimeType:ChosenPrime    9  891  20.242645 3.462464e-31 *
8      AgeGroup:PrimeType:ChosenPrime    9  891  2.651740 4.901353e-03 *

```

```

ges
2 0.0014138740
3 0.0013598073
4 0.2723847169
5 0.0004709252
6 0.0028722556
7 0.0465756526
8 0.0054785900

```

```
$`Mauchly's Test for Sphericity`
```

	Effect	W	p	p<.05
3	PrimeType	0.911044619	1.050088e-01	
4	ChosenPrime	0.103597793	7.702015e-46	*
5	AgeGroup:PrimeType	0.913919295	1.174996e-01	
6	AgeGroup:ChosenPrime	0.192346732	6.161569e-33	*
7	PrimeType:ChosenPrime	0.002908133	3.162084e-90	*
8	AgeGroup:PrimeType:ChosenPrime	0.003686477	1.096314e-85	*

```
$`Sphericity Corrections`
```

	Effect	GGe	p[GG]	p[GG]<.05	HFe
3	PrimeType	0.9485781	7.754843e-03	*	0.9795837
4	ChosenPrime	0.5049830	1.047576e-26	*	0.5111720
5	AgeGroup:PrimeType	0.9504406	2.377424e-01		0.9815776
6	AgeGroup:ChosenPrime	0.5868437	9.128597e-02		0.5966477
7	PrimeType:ChosenPrime	0.5206715	2.176170e-17	*	0.5497057
8	AgeGroup:PrimeType:ChosenPrime	0.5073612	2.633544e-02	*	0.5349136

	p[HF]	p[HF]<.05
3	7.108311e-03	*
4	5.316999e-27	*
5	2.368692e-01	
6	9.036466e-02	
7	3.156980e-18	*
8	2.391025e-02	*

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> expl_errors_lsm = lsmeans::lsmeans(expl_mcq_aov, c("AgeGroup", "PrimeType", "ChosenPrime"))
> prime_effect = cld(expl_errors_lsm, alpha = 0.05,
+   adjust = "tukey", details = TRUE, by = c("AgeGroup", "PrimeType"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))

```

	contrast	AgeGroup	PrimeType	estimate	SE	df	t.ratio	p.value
12	r - u	0ld	b	0.3352857	0.0418508	1558.848	8.011452	0.000000
13	r - p	0ld	b	0.3204524	0.0418508	1558.848	7.657019	0.000000
14	b - u	0ld	b	0.4353254	0.0418508	1558.848	10.401841	0.000000
15	b - p	0ld	b	0.4204921	0.0418508	1558.848	10.047407	0.000000
18	r - u	0ld	p	0.2703849	0.0418508	1558.848	6.460686	0.000000

9	r - p	Old	p		0.1852500	0.0418508	1558.848	4.426438	0.000060
10	b - u	Old	p		0.2954563	0.0418508	1558.848	7.059753	0.000000
11	b - p	Old	p		0.2103214	0.0418508	1558.848	5.025505	0.000003
14	r - u	Old	r		0.1548810	0.0418508	1558.848	3.700788	0.001271
15	r - p	Old	r		0.1495476	0.0418508	1558.848	3.573351	0.002056
16	b - u	Old	r		0.5780119	0.0418508	1558.848	13.811250	0.000000
17	b - p	Old	r		0.5726786	0.0418508	1558.848	13.683813	0.000000
18	b - r	Old	r		0.4231310	0.0418508	1558.848	10.110462	0.000000
20	r - u	Old	u		0.1931905	0.0418508	1558.848	4.616171	0.000025
21	r - p	Old	u		0.1590833	0.0418508	1558.848	3.801201	0.000860
22	b - u	Old	u		0.3071667	0.0418508	1558.848	7.339564	0.000000
23	b - p	Old	u		0.2730595	0.0418508	1558.848	6.524594	0.000000
24	b - r	Old	u		0.1139762	0.0418508	1558.848	2.723393	0.033056
26	r - u	Young	b		0.3335437	0.0418508	1558.848	7.969827	0.000000
27	r - p	Young	b		0.2902817	0.0418508	1558.848	6.936109	0.000000
28	b - u	Young	b		0.4461944	0.0418508	1558.848	10.661550	0.000000
29	b - p	Young	b		0.4029325	0.0418508	1558.848	9.627833	0.000000
30	b - r	Young	b		0.1126508	0.0418508	1558.848	2.691724	0.036117
32	r - u	Young	p		0.3125714	0.0418508	1558.848	7.468708	0.000000
33	r - p	Young	p		0.2664048	0.0418508	1558.848	6.365583	0.000000
34	b - u	Young	p		0.3949286	0.0418508	1558.848	9.436583	0.000000
35	b - p	Young	p		0.3487619	0.0418508	1558.848	8.333458	0.000000
38	r - u	Young	r		0.2849524	0.0418508	1558.848	6.808767	0.000000
39	r - p	Young	r		0.2820119	0.0418508	1558.848	6.738506	0.000000
40	b - u	Young	r		0.5042262	0.0418508	1558.848	12.048184	0.000000
41	b - p	Young	r		0.5012857	0.0418508	1558.848	11.977923	0.000000
42	b - r	Young	r		0.2192738	0.0418508	1558.848	5.239417	0.000001
44	r - u	Young	u		0.2771905	0.0418508	1558.848	6.623301	0.000000
45	r - p	Young	u		0.2692857	0.0418508	1558.848	6.434422	0.000000
46	b - u	Young	u		0.3540238	0.0418508	1558.848	8.459188	0.000000
47	b - p	Young	u		0.3461190	0.0418508	1558.848	8.270308	0.000000

```
> ## SPECIFIC OLD COMPARISION T TEST
>
> e1mcq_old_r = exp1_mcq %>% filter(AgeGroup == "Old" & PrimeType == "r")
> e1mcq_old_r_r = e1mcq_old_r %>% filter(ChosenPrime == "r")
> e1mcq_old_r_p = e1mcq_old_r %>% filter(ChosenPrime == "p")
> e1mcq_old_r_b = e1mcq_old_r %>% filter(ChosenPrime == "b")
> e1mcq_old_r_u = e1mcq_old_r %>% filter(ChosenPrime == "u")
> t.test(e1mcq_old_r_r$Proportion, e1mcq_old_r_p$Proportion, paired = TRUE)
```

Paired t-test

```
data: e1mcq_old_r_r$Proportion and e1mcq_old_r_p$Proportion
t = 13.942, df = 99, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.4911778 0.6541794
```

```
sample estimates:
mean of the differences
      0.5726786
```

```
> t.test(e1mcq_old_r_r$Proportion, e1mcq_old_r_b$Proportion, paired = TRUE)
```

Paired t-test

```
data: e1mcq_old_r_r$Proportion and e1mcq_old_r_b$Proportion
t = 7.5982, df = 99, p-value = 1.723e-11
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.3126333 0.5336286
sample estimates:
mean of the differences
      0.423131
```

```
> t.test(e1mcq_old_r_r$Proportion, e1mcq_old_r_u$Proportion, paired = TRUE)
```

Paired t-test

```
data: e1mcq_old_r_r$Proportion and e1mcq_old_r_u$Proportion
t = 14.334, df = 99, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.4980003 0.6580235
sample estimates:
mean of the differences
      0.5780119
```

```
> e1mcq_young_r = exp1_mcq %>% filter(AgeGroup == "Young" & PrimeType == "r")
> e1mcq_young_r_r = e1mcq_young_r %>% filter(ChosenPrime == "r")
> ## comparing young and old
> t.test(e1mcq_young_r_r$Proportion, e1mcq_old_r_r$Proportion)
```

Welch Two Sample t-test

```
data: e1mcq_young_r_r$Proportion and e1mcq_old_r_r$Proportion
t = -1.0687, df = 197.62, p-value = 0.2865
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.16911974  0.05023879
sample estimates:
mean of x mean of y
0.5205714 0.5800119
```

23.3 Experiment 2

```
> e2_item_acc = group_by(e2_hlm, Target, AgeGroup, PrimeCondition) %>%
+   summarise_at(vars(Accuracy), mean)
> e2_item_state = group_by(e2_hlm, Target, AgeGroup, Question.RESP) %>%
+   summarise(StateCount = n())
> e2_item_mcqacc = group_by(e2_hlm, Target, AgeGroup, PrimeCondition) %>%
+   summarise_at(vars(McAcc), mean)
```

23.3.1 Target Accuracy

```
> ### TARGET RETRIEVAL ACCURACY
>
> exp2_item_acc = aov(data = e2_item_acc, Accuracy ~ AgeGroup*PrimeCondition +
+   Error (Target/(AgeGroup*PrimeCondition)))
> summary(exp2_item_acc)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99   32.1   0.3242

Error: Target:AgeGroup
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  0.471   0.4706   9.914 0.00217 **
Residuals 99   4.699   0.0475
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3  0.999   0.3330  18.75 3.67e-11 ***
Residuals      297   5.276   0.0178
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition  3  0.165   0.05487   2.595 0.0527 .
Residuals              297   6.279   0.02114
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> exp1_target_lsm = lsmeans::lsmeans(exp2_item_acc,
+   c("AgeGroup", "PrimeCondition"))
```



```
> prime_effect = cld(exp1_target_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = c("PrimeCondition"))
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	PrimeCondition	estimate	SE	df	t.ratio	p.value
2	Young - Old	P	0.0854167	0.0235474	338.7317	3.627433	0.0003303
3	Young - Old	R	0.0629167	0.0235474	338.7317	2.671914	0.0079065

```
> ## specific t-tests
> e2_item_acc_collapsed = group_by(e2_hlm, Target, PrimeCondition) %>%
+   summarise_at(vars(Accuracy), mean)
> target_p = e2_item_acc_collapsed %>% filter(PrimeCondition == "P")
> target_r = e2_item_acc_collapsed %>% filter(PrimeCondition == "R")
> target_b = e2_item_acc_collapsed %>% filter(PrimeCondition == "B")
> target_u = e2_item_acc_collapsed %>% filter(PrimeCondition == "U")
> t.test(target_p$Accuracy, target_r$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_r$Accuracy
t = 5.432, df = 99, p-value = 3.993e-07
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.04802359 0.10329994
sample estimates:
mean of the differences
      0.07566176
```

```
> t.test(target_p$Accuracy, target_b$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_b$Accuracy
t = 4.6166, df = 99, p-value = 1.173e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.03412787 0.08557802
sample estimates:
mean of the differences
      0.05985294
```

```
> t.test(target_p$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_u$Accuracy
t = 6.4934, df = 99, p-value = 3.376e-09
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.06658307 0.12518163
sample estimates:
mean of the differences
      0.09588235
```

```
> t.test(target_b$Accuracy, target_r$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_b$Accuracy and target_r$Accuracy
t = 1.2695, df = 99, p-value = 0.2072
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.008899978 0.040517625
sample estimates:
mean of the differences
      0.01580882
```

```
> t.test(target_b$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_b$Accuracy and target_u$Accuracy
t = 2.6395, df = 99, p-value = 0.009646
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.008945084 0.063113740
sample estimates:
mean of the differences
      0.03602941
```

```
> t.test(target_r$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_r$Accuracy and target_u$Accuracy
t = 1.6247, df = 99, p-value = 0.1074
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.004475157 0.044916333
sample estimates:
mean of the differences
      0.02022059
```

```
> ### age effect
> e2_item_age_collapsed = group_by(e2_hlm, Target, AgeGroup) %>%
+   summarise_at(vars(Accuracy), mean)
> target_young = e2_item_age_collapsed %>% filter(AgeGroup == "Young")
> target_old = e2_item_age_collapsed %>% filter(AgeGroup == "Old")
> t.test(target_young$Accuracy, target_old$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_young$Accuracy and target_old$Accuracy
t = 3.281, df = 99, p-value = 0.001429
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.02001686 0.08127102
sample estimates:
mean of the differences
      0.05064394
```

```
>
```

23.3.2 State Data

```
> ## e1_item_state has many rows missing so we cannot use that dataset.
> ## we will use the Julieagg.R file to create an agg file for items.
>
> exp2_state = subset(final_statedata, final_statedata$StudyNo == '5' |
+   final_statedata$StudyNo == '6')
> exp2_state_prime = subset(statedata_primetype_long, statedata_primetype_long$StudyNo ==
+   statedata_primetype_long$StudyNo == '6')
> exp2_state_prime$PrimeCondition = as.factor(as.character(exp2_state_prime$PrimeCondition))
> exp2_state_prime$State = as.factor(as.character(exp2_state_prime$State))
> exp2_state_prime$Target = as.factor(as.character(exp2_state_prime$Target))
> ## just state
> exp2_state_aov = aov(data = exp2_state, Trials ~ AgeGroup*State +
+   Error(Target/(AgeGroup*State)))
> summary(exp2_state_aov)
```

```
Error: Target
      Df  Sum Sq  Mean Sq F value Pr(>F)
Residuals 99 1.91e-25 1.929e-27

Error: Target:AgeGroup
      Df  Sum Sq  Mean Sq F value Pr(>F)
AgeGroup  1 2.900e-27 2.853e-27  0.324  0.57
Residuals 99 8.714e-25 8.802e-27
```

```
Error: Target:State
      Df Sum Sq Mean Sq F value Pr(>F)
State    3  16710    5570   145.6 <2e-16 ***
Residuals 297  11360      38
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:State
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:State    3   3328   1109.3    83.74 <2e-16 ***
Residuals      297   3934    13.2
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> exp1_state_lsm = lsmeans::lsmeans(exp1_state_aov, c("AgeGroup", "State"))
> prime_effect = cld(exp1_state_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = c("State"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```

contrast	State	estimate	SE	df	t.ratio	p.value
Old - Young	dontknow	4.80	0.5199675	297	9.231347	0.0000000
Old - Young	know	1.90	0.5199675	297	3.654075	0.0003051
Old - Young	other	5.95	0.5199675	297	11.443024	0.0000000

```
> ##state by prime
> exp2_stateprime_aov = aov(data = exp2_state_prime,
+                            Trials ~ AgeGroup*PrimeCondition*State +
+                            Error(Target/(AgeGroup*PrimeCondition*State)))
> summary(exp2_stateprime_aov)
```

```
Error: Target
      Df      Sum Sq    Mean Sq F value Pr(>F)
Residuals 99 9.085e-25 9.177e-27

Error: Target:AgeGroup
      Df      Sum Sq    Mean Sq F value Pr(>F)
AgeGroup  1 1.700e-28 1.732e-28   0.233   0.63
Residuals 99 7.361e-26 7.436e-28

Error: Target:PrimeCondition
      Df      Sum Sq    Mean Sq F value Pr(>F)
PrimeCondition  3 3.300e-27 1.113e-27   0.991   0.397
Residuals      297 3.334e-25 1.123e-27

Error: Target:State
      Df Sum Sq Mean Sq F value Pr(>F)
```

```

State      3      4177    1392.5    145.6 <2e-16 ***
Residuals 297      2840         9.6
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition
              Df      Sum Sq    Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition  3 3.220e-27  1.072e-27    1.033  0.378
Residuals                297 3.084e-25  1.038e-27

Error: Target:AgeGroup:State
              Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:State  3  832.0  277.32   83.74 <2e-16 ***
Residuals       297  983.5    3.31

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeCondition:State
              Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition:State  9  136.1  15.126   8.989 4.55e-13 ***
Residuals             891 1499.4   1.683

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition:State
              Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition:State  9   23.4   2.596   1.448  0.163
Residuals                    891 1598.1   1.794

```

```

> library(ez)
> ezANOVA(data = exp1_state_prime, wid = .(Target),
+         dv = .(Trials), within = .(PrimeCondition, State),
+         between = .(AgeGroup))

```

```

$ANOVA
              Effect DFn  DFd          F          p p<.05
2              AgeGroup    1   198 -5.920075e-13 1.000000e+00
3      PrimeCondition    3   594  1.169625e-13 1.000000e+00
5              State    3   594  1.898986e+02 2.484828e-86      *
4      AgeGroup:PrimeCondition    3   594  4.010453e-13 1.000000e+00
6      AgeGroup:State    3   594  3.245945e+01 1.936966e-19      *
7      PrimeCondition:State    9  1782  1.438037e+01 1.157611e-22      *
8 AgeGroup:PrimeCondition:State    9  1782  8.527043e-01 5.674172e-01
      ges
2 1.906245e-32
3 2.397542e-32
5 3.694688e-01
4 8.220776e-32

```

```
6 9.104058e-02
7 2.747865e-02
8 1.672622e-03
```

```
$`Mauchly's Test for Sphericity`
```

	Effect	W	p	p<.05
3	PrimeCondition	0.3554475	5.199325e-42	*
4	AgeGroup:PrimeCondition	0.3554475	5.199325e-42	*
5	State	0.2667363	4.113432e-54	*
6	AgeGroup:State	0.2667363	4.113432e-54	*
7	PrimeCondition:State	0.3359210	6.431198e-24	*
8	AgeGroup:PrimeCondition:State	0.3359210	6.431198e-24	*

```
$`Sphericity Corrections`
```

	Effect	GGe	p[GG]	p[GG]<.05	HFe
3	PrimeCondition	0.3333333	9.999997e-01		0.3333333
4	AgeGroup:PrimeCondition	0.3333333	9.999995e-01		0.3333333
5	State	0.5686656	2.220738e-50	*	0.5731086
6	AgeGroup:State	0.5686656	4.121646e-12	*	0.5731086
7	PrimeCondition:State	0.7978142	1.486851e-18	*	0.8308514
8	AgeGroup:PrimeCondition:State	0.7978142	5.460060e-01		0.8308514

```
p[HF] p[HF]<.05
```

```
3 9.999997e-01
4 9.999995e-01
5 9.459746e-51
6 3.462026e-12
7 3.163980e-19
8 5.498147e-01
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> exp1_state_lsm = lsmeans::lsmeans(exp1_stateprime_aov, c("AgeGroup","PrimeCondition",
> prime_effect = cld(exp1_state_lsm, alpha = 0.05,
+ adjust = "tukey", details = TRUE, by = c("PrimeCondition", "AgeGroup"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```

	contrast	PrimeCondition	AgeGroup	estimate	SE	df	t.ratio	p.value
12	dontknow - other	b	Old	3.18	0.2564724	1197.602	12.3989	
13	dontknow - TOT	b	Old	2.78	0.2564724	1197.602	10.8393	
14	know - other	b	Old	3.90	0.2564724	1197.602	15.2063	
15	know - TOT	b	Old	3.50	0.2564724	1197.602	13.6466	
16	know - dontknow	b	Old	0.72	0.2564724	1197.602		2.807319 0.0260869
17	TOT - other	b	Young	0.84	0.2564724	1197.602		3.275206 0.0059743
18	dontknow - other	b	Young	1.41	0.2564724	1197.602		5.497667 0.0000003

10	know - other	b		Young		2.71	0.2564724	1197.602	10.5664
11	know - TOT	b		Young		1.87	0.2564724	1197.602	
7.291232	0.0000000								
12	know - dontknow	b		Young		1.30	0.2564724	1197.602	
5.068771	0.0000028								
14	dontknow - other	p		Old		2.98	0.2564724	1197.602	11.6191
15	dontknow - TOT	p		Old		2.83	0.2564724	1197.602	11.0343
16	know - other	p		Old		4.03	0.2564724	1197.602	15.7131
17	know - TOT	p		Old		3.88	0.2564724	1197.602	15.1283
18	know - dontknow	p		Old		1.05	0.2564724	1197.602	
4.094007	0.0002640								
19	TOT - other	p		Young		0.74	0.2564724	1197.602	
2.885300	0.0207377								
20	dontknow - other	p		Young		1.13	0.2564724	1197.602	
4.405931	0.0000677								
22	know - other	p		Young		2.97	0.2564724	1197.602	11.5801
23	know - TOT	p		Young		2.23	0.2564724	1197.602	
8.694891	0.0000000								
24	know - dontknow	p		Young		1.84	0.2564724	1197.602	
7.174260	0.0000000								
26	dontknow - other	r		Old		2.72	0.2564724	1197.602	10.6054
27	dontknow - TOT	r		Old		2.45	0.2564724	1197.602	
9.552683	0.0000000								
28	know - other	r		Old		3.49	0.2564724	1197.602	13.6077
29	know - TOT	r		Old		3.22	0.2564724	1197.602	12.5549
30	know - dontknow	r		Old		0.77	0.2564724	1197.602	
3.002272	0.0145194								
31	TOT - other	r		Young		1.14	0.2564724	1197.602	
4.444922	0.0000567								
32	dontknow - other	r		Young		1.24	0.2564724	1197.602	
4.834827	0.0000090								
34	know - other	r		Young		2.50	0.2564724	1197.602	
9.747636	0.0000000								
35	know - TOT	r		Young		1.36	0.2564724	1197.602	
5.302714	0.0000008								
36	know - dontknow	r		Young		1.26	0.2564724	1197.602	
4.912808	0.0000061								
37	TOT - other	u		Old		0.69	0.2564724	1197.602	
2.690348	0.0363528								
38	dontknow - other	u		Old		2.80	0.2564724	1197.602	10.9173
39	dontknow - TOT	u		Old		2.11	0.2564724	1197.602	
8.227005	0.0000000								
40	know - other	u		Old		3.67	0.2564724	1197.602	14.3095
41	know - TOT	u		Old		2.98	0.2564724	1197.602	11.6191
42	know - dontknow	u		Old		0.87	0.2564724	1197.602	
3.392177	0.0039876								
43	TOT - other	u		Young		0.97	0.2564724	1197.602	
3.782083	0.0009373								

44	dontknow - other	u	Young		1.71	0.2564724	1197.602
6.667383		0.0000000					
45	dontknow - TOT	u	Young		0.74	0.2564724	1197.602
2.885300		0.0207377					
46	know - other	u	Young		1.88	0.2564724	1197.602
7.330222		0.0000000					
47	know - TOT	u	Young		0.91	0.2564724	1197.602
3.548140		0.0022760					

```
> ### INDIVIDUAL T-TESTS FOR AGExSTATE interaction
>
> e1_young_dk = exp1_state %>% filter(AgeGroup == "Young" & State == "dontknow")
> e1_old_dk = exp1_state %>% filter(AgeGroup == "Old" & State == "dontknow")
> t.test(e1_old_dk$Trials, e1_young_dk$Trials)
```

Welch Two Sample t-test

```
data: e1_old_dk$Trials and e1_young_dk$Trials
t = 5.837, df = 197.87, p-value = 2.146e-08
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 3.178335 6.421665
sample estimates:
mean of x mean of y
 14.48      9.68
```

```
> e1_young_other = exp1_state %>% filter(AgeGroup == "Young" & State == "other")
> e1_old_other = exp1_state %>% filter(AgeGroup == "Old" & State == "other")
> t.test(e1_young_other$Trials, e1_old_other$Trials)
```

Welch Two Sample t-test

```
data: e1_young_other$Trials and e1_old_other$Trials
t = 12.087, df = 119.23, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 4.975274 6.924726
sample estimates:
mean of x mean of y
 7.88      1.93
```

```
>
```

Multiple Choice


```
> exp2_mcq = subset(final_mcq, final_mcq$StudyNo == '5' | final_mcq$StudyNo == '6')
> ## MULTIPLE CHOICE ACCURACY
> library(dplyr)
> exp2_mcq_acc = group_by(exp2_mcq, Target, PrimeType, AgeGroup) %>%
+   summarise_at(vars(MCQAcc), mean)
> exp2_mcq_acc_aov = aov(data = exp2_mcq_acc, MCQAcc ~ AgeGroup*PrimeType +
+   Error(Target/(AgeGroup*PrimeType)))
> summary(exp2_mcq_acc_aov)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99  25.84    0.261

Error: Target:AgeGroup
      Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup  1   1.06   1.0603   24.59 2.95e-06 ***
Residuals 99   4.27   0.0431
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeType
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeType  3   0.370  0.12325    4.688 0.00324 **
Residuals 297   7.808  0.02629
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeType
      Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup:PrimeType  3   0.227  0.07564    3.408 0.018 *
Residuals          297   6.591  0.02219
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## SPECIFIC T TESTS
>
> e2_mcq_p = exp2_mcq_acc %>% filter(PrimeType == "p")
> e2_mcq_r = exp2_mcq_acc %>% filter(PrimeType == "r")
> e2_mcq_b = exp2_mcq_acc %>% filter(PrimeType == "b")
> e2_mcq_u = exp2_mcq_acc %>% filter(PrimeType == "u")
> e2mcq_y_p = e2_mcq_p %>% filter(AgeGroup == "Young")
> e2mcq_o_p = e2_mcq_p %>% filter(AgeGroup == "Old")
> t.test(e2mcq_y_p$MCQAcc, e2mcq_o_p$MCQAcc)
```

Welch Two Sample t-test

```
data: e2mcq_y_p$MCQAcc and e2mcq_o_p$MCQAcc
```

```
t = 3.4423, df = 197.18, p-value = 0.0007041
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.04484715 0.16515285
sample estimates:
mean of x mean of y
 0.77875   0.67375
```

```
> e2mcq_y_b = e2_mcq_b %>% filter(AgeGroup == "Young")
> e2mcq_o_b = e2_mcq_b %>% filter(AgeGroup == "Old")
> t.test(e2mcq_y_b$MCQAcc, e2mcq_o_b$MCQAcc)
```

Welch Two Sample t-test

```
data: e2mcq_y_b$MCQAcc and e2mcq_o_b$MCQAcc
t = 2.8466, df = 191.08, p-value = 0.004902
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.02878927 0.15871073
sample estimates:
mean of x mean of y
 0.78000   0.68625
```

```
> e2mcq_y_r = e2_mcq_r %>% filter(AgeGroup == "Young")
> e2mcq_o_r = e2_mcq_r %>% filter(AgeGroup == "Old")
> t.test(e2mcq_y_r$MCQAcc, e2mcq_o_r$MCQAcc)
```

Welch Two Sample t-test

```
data: e2mcq_y_r$MCQAcc and e2mcq_o_r$MCQAcc
t = 2.0973, df = 191.57, p-value = 0.03728
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.004466692 0.145533308
sample estimates:
mean of x mean of y
 0.74625   0.67125
```

```
> e2mcq_y_u = e2_mcq_u %>% filter(AgeGroup == "Young")
> e2mcq_o_u = e2_mcq_u %>% filter(AgeGroup == "Old")
> t.test(e2mcq_y_u$MCQAcc, e2mcq_o_u$MCQAcc)
```

Welch Two Sample t-test

```
data: e2mcq_y_u$MCQAcc and e2mcq_o_u$MCQAcc
t = 0.50471, df = 198, p-value = 0.6143
alternative hypothesis: true difference in means is not equal to 0
```

```

95 percent confidence interval:
-0.05087645  0.08587645
sample estimates:
mean of x mean of y
  0.68625    0.66875

```

```

> ezANOVA(data = exp2_mcq_acc, wid = .(Target),
+         dv = .(MCQAcc), within = .(PrimeType),
+         between = .(AgeGroup))

```

```

$ANOVA
      Effect DFn DFd      F      p p<.05      ges
2      AgeGroup   1 198 6.972309 0.008938249 * 0.023267635
3      PrimeType   3 594 5.084216 0.001755523 * 0.008238453
4 AgeGroup:PrimeType 3 594 3.120464 0.025601350 * 0.005072534

$`Mauchly's Test for Sphericity`
      Effect      W      p p<.05
3      PrimeType 0.9480247 0.06224892
4 AgeGroup:PrimeType 0.9480247 0.06224892

$`Sphericity Corrections`
      Effect      GGe      p[GG] p[GG]<.05      HFe      p[HF]
3      PrimeType 0.963567 0.002026939 * 0.9793647 0.001904389
4 AgeGroup:PrimeType 0.963567 0.027235021 * 0.9793647 0.026513877
p[HF]<.05
3      *
4      *

```

```

> exp2_mcqacc_lsm = lsmeans::lsmeans(exp2_mcq_acc_aov, c("AgeGroup", "PrimeType"))
> prime_effect = cld(exp2_mcqacc_lsm, alpha = 0.05,
+                   adjust = "tukey", details = TRUE, by = c("PrimeType"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.1))

```

contrast	PrimeType	estimate	SE	df	t.ratio	p.value
Young - Old	b	0.09375	0.023421	356.9997	4.002815	0.0000762
Young - Old	p	0.10500	0.023421	356.9997	4.483153	0.0000099
Young - Old	r	0.07500	0.023421	356.9997	3.202252	0.0014861

```

> ## MULTIPLE CHOICE ERRORS
>
> ## before we do ANOVA, we need to replace NAs with 0.
>
> for (i in 1:nrow(exp2_mcq)){
+   if(is.na(exp2_mcq[i,9])){
+     exp2_mcq[i,9] = 0

```

```

+   }
+
+ }
> exp2_mcq_aov = aov(data = exp2_mcq,
+                     Proportion ~ AgeGroup*PrimeType*ChosenPrime +
+                     Error(Target/(AgeGroup*PrimeType*ChosenPrime)))
> summary(exp2_mcq_aov)

```

```

Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99  10.58  0.1069

Error: Target:AgeGroup
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  0.000 0.00015  0.003  0.958
Residuals 99  5.345 0.05399

Error: Target:PrimeType
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType  3  0.135 0.04489  1.56  0.199
Residuals 297  8.543 0.02877

Error: Target:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
ChosenPrime  3  86.10 28.698 117.1 <2e-16 ***
Residuals 297  72.81  0.245
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeType
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeType  3  0.117 0.03900  1.241  0.295
Residuals 297  9.336 0.03143

Error: Target:AgeGroup:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:ChosenPrime  3  0.373 0.12426  1.36  0.255
Residuals 297 27.143 0.09139

Error: Target:PrimeType:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType:ChosenPrime  9  0.24 0.02641  0.384  0.943
Residuals 891  61.32 0.06883

Error: Target:AgeGroup:PrimeType:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeType:ChosenPrime  9  2.91 0.3239  4.644 5e-06 ***
Residuals 891  62.14 0.0697

```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> library(ez)
> ezANOVA(data = exp2_mcq, wid = .(Target),
+         dv = .(Proportion), within = .(PrimeType, ChosenPrime),
+         between = .(AgeGroup))
```

```
$ANOVA
```

	Effect	DFn	DFd	F	p	p<.05
2	AgeGroup	1	198	1.877827e-03	9.654790e-01	
3	PrimeType	3	594	1.491234e+00	2.158240e-01	
5	ChosenPrime	3	594	1.705526e+02	9.596644e-80	*
4	AgeGroup:PrimeType	3	594	1.295852e+00	2.749264e-01	
6	AgeGroup:ChosenPrime	3	594	7.384562e-01	5.293713e-01	
7	PrimeType:ChosenPrime	9	1782	3.812390e-01	9.445523e-01	
8	AgeGroup:PrimeType:ChosenPrime	9	1782	4.674359e+00	3.788925e-06	*

```
ges
```

2	5.872368e-07
3	5.232369e-04
5	2.507769e-01
4	4.547132e-04
6	1.447151e-03
7	9.233411e-04
8	1.120456e-02

```
$`Mauchly's Test for Sphericity`
```

	Effect	W	p	p<.05
3	PrimeType	0.98596364	7.337351e-01	
4	AgeGroup:PrimeType	0.98596364	7.337351e-01	
5	ChosenPrime	0.11953020	4.490883e-88	*
6	AgeGroup:ChosenPrime	0.11953020	4.490883e-88	*
7	PrimeType:ChosenPrime	0.01299866	8.661040e-149	*
8	AgeGroup:PrimeType:ChosenPrime	0.01299866	8.661040e-149	*

```
$`Sphericity Corrections`
```

	Effect	GGe	p[GG]	p[GG]<.05	HFe
3	PrimeType	0.9906290	2.161716e-01		1.0073856
4	AgeGroup:PrimeType	0.9906290	2.750172e-01		1.0073856
5	ChosenPrime	0.5236153	3.676093e-43	*	0.5270744
6	AgeGroup:ChosenPrime	0.5236153	4.482181e-01		0.5270744
7	PrimeType:ChosenPrime	0.5650868	8.646968e-01		0.5817614
8	AgeGroup:PrimeType:ChosenPrime	0.5650868	2.888667e-04	*	0.5817614

```
p[HF] p[HF]<.05
```

3	2.158240e-01	
4	2.749264e-01	
5	1.992842e-43	*
6	4.490454e-01	

```
7 8.695506e-01
8 2.441203e-04      *
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> exp2_errors_lsm = lsmeans::lsmeans(exp2_mcq_aov, c("AgeGroup", "PrimeType", "ChosenPrime"))
> prime_effect = cld(exp2_errors_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = c("PrimeType", "ChosenPrime"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```

	contrast	PrimeType	ChosenPrime	estimate	SE	df	t.ratio	p.value
1	Old - Young	b	b	0.1159167	0.0362305	1461.313	3.199423	0.001
3	Old - Young	b	r	0.0835833	0.0362305	1461.313	2.306989	0.022
9	Old - Young	r	b	0.1738333	0.0362305	1461.313	4.797984	0.000
11	Old - Young	r	r	0.0809286	0.0362305	1461.313	2.233714	0.030

```
> ## SPECIFIC OLD COMPARISION T TEST
>
> e2mcq_old_r = exp2_mcq %>% filter(AgeGroup == "Old" & PrimeType == "r")
> e2mcq_young_r = exp2_mcq %>% filter(AgeGroup == "Young" & PrimeType == "r")
> e2mcq_old_r_r = e2mcq_old_r %>% filter(ChosenPrime == "r")
> e2mcq_young_r_r = e2mcq_young_r %>% filter(ChosenPrime == "r")
> ## comparing young and old
> t.test(e2mcq_young_r_r$Proportion, e2mcq_old_r_r$Proportion)
```

Welch Two Sample t-test

```
data: e2mcq_young_r_r$Proportion and e2mcq_old_r_r$Proportion
t = -1.4314, df = 197.31, p-value = 0.1539
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.19242763 0.03057048
sample estimates:
mean of x mean of y
0.3657619 0.4466905
```

```
> e2mcq_old_b = exp2_mcq %>% filter(AgeGroup == "Old" & PrimeType == "b")
> e2mcq_young_b = exp2_mcq %>% filter(AgeGroup == "Young" & PrimeType == "b")
> e2mcq_old_b_b = e2mcq_old_b %>% filter(ChosenPrime == "b")
> e2mcq_young_b_b = e2mcq_young_b %>% filter(ChosenPrime == "b")
> ## comparing young and old
> t.test(e2mcq_young_b_b$Proportion, e2mcq_old_b_b$Proportion)
```

Welch Two Sample t-test

```
data: e2mcq_young_b_b$Proportion and e2mcq_old_b_b$Proportion
```

```

t = -2.1749, df = 197.46, p-value = 0.03082
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.22102042 -0.01081291
sample estimates:
mean of x mean of y
0.2491667 0.3650833

```

23.4 Collapsing the 4 experiments

```

> final_mcq_main4 = subset(final_mcq, final_mcq$StudyNo != '1')
> for (i in 1: nrow(final_mcq_main4)){
+   if(is.na(final_mcq_main4[i,9])){
+     final_mcq_main4[i,9] = 0
+   }
+ }
> final_mcq_main4$PrimeInstruction = ifelse(final_mcq_main4$StudyNo == "2" |
+   final_mcq_main4$StudyNo == "4",
+   "NoInstruction", "WithInstruction")
> final_mcq_main4$PrimeInstruction = as.factor(final_mcq_main4$PrimeInstruction)
> fourway_aov = aov(data = final_mcq_main4,
+   Proportion ~ AgeGroup*PrimeInstruction*PrimeType*ChosenPrime + Error(Target))
> summary(fourway_aov)

```

```

Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99   18.1   0.1828

Error: Target:AgeGroup
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1   0.162 0.16161   2.106   0.15
Residuals 99   7.598 0.07675

Error: Target:PrimeType
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType  3   0.233 0.07762   2.981 0.0317 *
Residuals 297   7.732 0.02603
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
ChosenPrime  3  175.2   58.41  133.3 <2e-16 ***
Residuals  297  130.2    0.44
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Error: Target:AgeGroup:PrimeType
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeType  3  0.099  0.03312    1.136   0.335
Residuals          297   8.659  0.02915

Error: Target:AgeGroup:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:ChosenPrime  3    0.94   0.3134    2.458   0.063 .
Residuals          297   37.87   0.1275

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeType:ChosenPrime
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeType:ChosenPrime  9    6.63   0.7362   10.85 3.99e-16 ***
Residuals          891   60.44   0.0678

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeType:ChosenPrime
      Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup:PrimeType:ChosenPrime  9    3.10   0.3447    5.244 5.6e-07 ***
Residuals          891   58.57   0.0657

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Within
      Df Sum Sq Mean Sq F value
PrimeInstruction  1    0.02   0.0195   0.372
AgeGroup:PrimeInstruction  1    0.18   0.1759   3.350
PrimeInstruction:PrimeType  3    0.23   0.0754   1.436
PrimeInstruction:ChosenPrime  3    0.04   0.0141   0.268
AgeGroup:PrimeInstruction:PrimeType  3    0.13   0.0433   0.825
AgeGroup:PrimeInstruction:ChosenPrime  3    0.12   0.0396   0.754
PrimeInstruction:PrimeType:ChosenPrime  9    5.25   0.5834  11.113
AgeGroup:PrimeInstruction:PrimeType:ChosenPrime  9    1.12   0.1250   2.380
Residuals          3168  166.31   0.0525
Pr(>F)
PrimeInstruction  0.5419
AgeGroup:PrimeInstruction  0.0673 .
PrimeInstruction:PrimeType  0.2302
PrimeInstruction:ChosenPrime  0.8483
AgeGroup:PrimeInstruction:PrimeType  0.4800
AgeGroup:PrimeInstruction:ChosenPrime  0.5197
PrimeInstruction:PrimeType:ChosenPrime <2e-16 ***
AgeGroup:PrimeInstruction:PrimeType:ChosenPrime 0.0111 *
Residuals
---

```


Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> library(ez)
> ezANOVA(data = final_mcq_main4, wid = .(Target),
+         dv = .(Proportion), within = .(AgeGroup, PrimeType, ChosenPrime),
+         between = .(PrimeInstruction)) ## IMPORTANT SPHERICITY
```

```
$ANOVA
```

	Effect	DFn	DFd	F
2	PrimeInstruction	1	198	0.18713082
3	AgeGroup	1	198	3.24171022
5	PrimeType	3	594	2.83041022
7	ChosenPrime	3	594	234.65023310
4	PrimeInstruction:AgeGroup	1	198	3.52808575
6	PrimeInstruction:PrimeType	3	594	2.74956731
8	PrimeInstruction:ChosenPrime	3	594	0.05658197
9	AgeGroup:PrimeType	3	594	1.14718318
11	AgeGroup:ChosenPrime	3	594	3.43285215
13	PrimeType:ChosenPrime	9	1782	11.09471662
10	PrimeInstruction:AgeGroup:PrimeType	3	594	1.49981212
12	PrimeInstruction:AgeGroup:ChosenPrime	3	594	0.43381939
14	PrimeInstruction:PrimeType:ChosenPrime	9	1782	8.79236462
15	AgeGroup:PrimeType:ChosenPrime	9	1782	5.52731608
16	PrimeInstruction:AgeGroup:PrimeType:ChosenPrime	9	1782	2.00366733

	p	p<.05	ges
2	6.657862e-01		3.943324e-05
3	7.330765e-02		3.260601e-04
5	3.777261e-02	*	4.697636e-04
7	2.157875e-100	*	2.612787e-01
4	6.180721e-02		3.548543e-04
6	4.207623e-02	*	4.563522e-04
8	9.823010e-01		8.527925e-05
9	3.293567e-01		2.004997e-04
11	1.679438e-02	*	1.894088e-03
13	5.411772e-17	*	1.319625e-02
10	2.135201e-01		2.621144e-04
12	7.288653e-01		2.397581e-04
14	4.889678e-13	*	1.048651e-02
15	1.548857e-07	*	6.222730e-03
16	3.543470e-02	*	2.264741e-03

\$`Mauchly's Test for Sphericity`

	Effect	W	p
5	PrimeType	0.96657111	2.448543e-01
6	PrimeInstruction:PrimeType	0.96657111	2.448543e-01
7	ChosenPrime	0.09594380	2.212496e-97
8	PrimeInstruction:ChosenPrime	0.09594380	2.212496e-97
9	AgeGroup:PrimeType	0.98012975	5.568938e-01

```

10      PrimeInstruction:AgeGroup:PrimeType 0.98012975 5.568938e-01
11      AgeGroup:ChosenPrime 0.19338025 1.071437e-67
12      PrimeInstruction:AgeGroup:ChosenPrime 0.19338025 1.071437e-67
13      PrimeType:ChosenPrime 0.01014665 9.654611e-159
14      PrimeInstruction:PrimeType:ChosenPrime 0.01014665 9.654611e-159
15      AgeGroup:PrimeType:ChosenPrime 0.01102039 2.032352e-155
16 PrimeInstruction:AgeGroup:PrimeType:ChosenPrime 0.01102039 2.032352e-155
    p<.05
5
6
7      *
8      *
9
10
11     *
12     *
13     *
14     *
15     *
16     *

$`Sphericity Corrections`
      Effect      GGe      p[GG]
5      PrimeType 0.9789011 3.889875e-02
6      PrimeInstruction:PrimeType 0.9789011 4.325475e-02
7      ChosenPrime 0.4944176 3.688811e-51
8      PrimeInstruction:ChosenPrime 0.4944176 8.980999e-01
9      AgeGroup:PrimeType 0.9864980 3.291018e-01
10     PrimeInstruction:AgeGroup:PrimeType 0.9864980 2.140335e-01
11     AgeGroup:ChosenPrime 0.5922821 3.876739e-02
12     PrimeInstruction:AgeGroup:ChosenPrime 0.5922821 6.246725e-01
13     PrimeType:ChosenPrime 0.5564332 1.980133e-10
14     PrimeInstruction:PrimeType:ChosenPrime 0.5564332 3.483552e-08
15     AgeGroup:PrimeType:ChosenPrime 0.5581243 4.822437e-05
16 PrimeInstruction:AgeGroup:PrimeType:ChosenPrime 0.5581243 7.537565e-02
    p[GG]<.05      HFe      p[HF] p[HF]<.05
5      * 0.9952392 3.802374e-02      *
6      * 0.9952392 4.233923e-02      *
7      * 0.4972728 1.943524e-51      *
8      0.4972728 8.991619e-01
9      1.0031067 3.293567e-01
10     1.0031067 2.135201e-01
11     * 0.5972663 3.837154e-02      *
12     0.5972663 6.263484e-01
13     * 0.5726036 1.138177e-10      *
14     * 0.5726036 2.311807e-08      *
15     * 0.5743927 3.894613e-05      *
16     0.5743927 7.325506e-02

```

23.5 Experiment 3

```
> e3_item_acc = group_by(e3_hlm, Target, PrimeCondition) %>%
+   summarise_at(vars(Accuracy), mean)
> e3_item_state = group_by(e3_hlm, Target, Question.RESP) %>%
+   summarise(StateCount = n())
> e3_item_mcqacc = group_by(e3_hlm, Target, PrimeCondition) %>%
+   summarise_at(vars(McAcc), mean)
```

23.5.1 Target Accuracy

```
> ### TARGET RETRIEVAL ACCURACY
>
> exp3_item_acc = aov(data = e3_item_acc, Accuracy ~ PrimeCondition +
+   Error (Target/PrimeCondition))
> summary(exp3_item_acc)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99  19.69   0.1989

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition   3   0.279  0.09288   5.559 0.00101 **
Residuals      297   4.962  0.01671

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## specific t-tests
> target_p = e3_item_acc %>% filter(PrimeCondition == "P")
> target_r = e3_item_acc %>% filter(PrimeCondition == "R")
> target_b = e3_item_acc %>% filter(PrimeCondition == "B")
> target_u = e3_item_acc %>% filter(PrimeCondition == "U")
> t.test(target_p$Accuracy, target_r$Accuracy, paired = TRUE)
```

```
Paired t-test

data:  target_p$Accuracy and target_r$Accuracy
t = 2.7128, df = 99, p-value = 0.007869
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01462240 0.09426649
sample estimates:
mean of the differences
 0.05444444
```

```
> t.test(target_p$Accuracy, target_b$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_b$Accuracy
t = 2.1645, df = 99, p-value = 0.03283
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.003054478 0.070278856
sample estimates:
mean of the differences
      0.03666667
```

```
> t.test(target_p$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_p$Accuracy and target_u$Accuracy
t = 3.6107, df = 99, p-value = 0.0004814
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.03203268 0.11018955
sample estimates:
mean of the differences
      0.07111111
```

```
> t.test(target_b$Accuracy, target_r$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_b$Accuracy and target_r$Accuracy
t = 1.042, df = 99, p-value = 0.3
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.01607674 0.05163230
sample estimates:
mean of the differences
      0.01777778
```

```
> t.test(target_b$Accuracy, target_u$Accuracy, paired = TRUE)
```

Paired t-test

```
data: target_b$Accuracy and target_u$Accuracy
t = 1.8723, df = 99, p-value = 0.06411
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.002058338 0.070947227
```

```
sample estimates:
mean of the differences
      0.03444444
```

```
> t.test(target_r$Accuracy, target_u$Accuracy, paired = TRUE)
```

```
      Paired t-test

data:  target_r$Accuracy and target_u$Accuracy
t = 0.96591, df = 99, p-value = 0.3364
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.01757073  0.05090406
sample estimates:
mean of the differences
      0.01666667
```

```
>
>
```

23.5.2 State Data

```
> exp3_state = subset(final_statedata, final_statedata$StudyNo == '1')
> exp3_state_prime = subset(statedata_primetype_long, statedata_primetype_long$StudyNo == '1')
> exp3_state_prime$PrimeCondition = as.factor(as.character(exp3_state_prime$PrimeCondition))
> exp3_state_prime$State = as.factor(as.character(exp3_state_prime$State))
> exp3_state_prime$Target = as.factor(as.character(exp3_state_prime$Target))
> ## just state
> exp3_state_aov = aov(data = exp3_state, Trials ~ State +
+                        Error(Target/State))
> summary(exp3_state_aov)
```

```
Error: Target
      Df  Sum Sq  Mean Sq F value Pr(>F)
Residuals 99 2.09e-26  2.111e-28

Error: Target:State
      Df Sum Sq Mean Sq F value Pr(>F)
State   3   4918  1639.3    47.15 <2e-16 ***
Residuals 297  10326    34.8

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> exp3_state_lsm = lsmeans::lsmeans(exp3_state_aov, c("State"))
> prime_effect = cld(exp3_state_lsm, alpha = 0.05,
```

```
+ adjust = "tukey", details = TRUE)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05))
```

	contrast	estimate	SE	df	t.ratio	p.value
2	dontknow - TOT	5.00	0.8338778	297	5.996082	0.00e+00
3	dontknow - other	4.29	0.8338778	297	5.144639	2.90e-06
4	know - TOT	8.69	0.8338778	297	10.421191	0.00e+00
5	know - other	7.98	0.8338778	297	9.569748	0.00e+00
6	know - dontknow	3.69	0.8338778	297	4.425109	7.97e-05

```
> ##state by prime
> exp3_stateprime_aov = aov(data = exp3_state_prime,
+ Trials ~ PrimeCondition*State +
+ Error(Target/(PrimeCondition*State)))
> summary(exp3_stateprime_aov)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99 5.699e-26 5.756e-28

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition 3 5.600e-28 1.861e-28 0.229 0.876
Residuals 297 2.417e-25 8.137e-28

Error: Target:State
      Df Sum Sq Mean Sq F value Pr(>F)
State 3 1230 409.8 47.15 <2e-16 ***
Residuals 297 2582 8.7
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeCondition:State
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition:State 9 89.9 9.984 5.115 8.99e-07 ***
Residuals 891 1739.1 1.952
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> library(ez)
> ezANOVA(data = exp3_state_prime, wid = .(Target),
+ dv = .(Trials), within = .(PrimeCondition, State))
```

\$ANOVA						
	Effect	DFn	DFd	F	p	p<.05
2	PrimeCondition	3	297	6.930670e-14	1.000000e+00	3.081031e-32

```

3          State      3 297 4.715136e+01 5.961821e-25      * 2.215266e-01
4 PrimeCondition:State  9 891 5.114953e+00 8.993080e-07      * 2.037300e-02

$`Mauchly's Test for Sphericity`
      Effect      W      p p<.05
3          State 0.3519290 1.988548e-20      *
4 PrimeCondition:State 0.3340113 6.814955e-07      *

$`Sphericity Corrections`
      Effect      GGe      p[GG] p[GG]<.05      HFe      p[HF]
2      PrimeCondition 0.6512249 1.000000e+00      0.6641758 1.000000e+00
3          State 0.6267511 1.411384e-16      * 0.6384741 7.69264e-17
4 PrimeCondition:State 0.7877646 9.835503e-06      * 0.8546854 4.61609e-06
p[HF]<.05
2
3      *
4      *

```

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> exp3_state_lsm = lsmeans::lsmeans(exp3_stateprime_aov, c("PrimeCondition", "State"))
> prime_effect = cld(exp3_state_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = c("PrimeCondition"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.1))

```

	contrast	PrimeCondition	estimate	SE	df	t.ratio	p.value
2	dontknow - TOT	b	1.20	0.2696999	722.646	4.449389	0.00005
3	dontknow - other	b	0.99	0.2696999	722.646	3.670746	0.00147
4	know - TOT	b	2.47	0.2696999	722.646	9.158327	0.00000
5	know - other	b	2.26	0.2696999	722.646	8.379684	0.00000
6	know - dontknow	b	1.27	0.2696999	722.646	4.708937	0.00001
8	dontknow - TOT	p	1.17	0.2696999	722.646	4.338155	0.00009
9	dontknow - other	p	1.01	0.2696999	722.646	3.744903	0.00111
10	know - TOT	p	2.11	0.2696999	722.646	7.823510	0.00000
11	know - other	p	1.95	0.2696999	722.646	7.230258	0.00000
12	know - dontknow	p	0.94	0.2696999	722.646	3.485355	0.00292
14	dontknow - TOT	r	1.22	0.2696999	722.646	4.523546	0.00004
15	dontknow - other	r	0.89	0.2696999	722.646	3.299964	0.00558
16	know - TOT	r	2.77	0.2696999	722.646	10.270674	0.00000
17	know - other	r	2.44	0.2696999	722.646	9.047092	0.00000
18	know - dontknow	r	1.55	0.2696999	722.646	5.747128	0.00000
20	dontknow - TOT	u	1.34	0.2696999	722.646	4.968485	0.00000
21	dontknow - other	u	1.33	0.2696999	722.646	4.931407	0.00000
22	know - TOT	u	1.41	0.2696999	722.646	5.228033	0.00000
23	know - other	u	1.40	0.2696999	722.646	5.190954	0.00000

```
>
```

Multiple Choice

```
> exp3_mcq = subset(final_mcq, final_mcq$StudyNo == '1')
> ## MULTIPLE CHOICE ACCURACY
> library(dplyr)
> exp3_mcq_acc = group_by(exp3_mcq, Target, PrimeType) %>%
+   summarise_at(vars(MCQAcc), mean)
> exp3_mcq_acc_aov = aov(data = exp3_mcq_acc, MCQAcc ~ PrimeType +
+   Error(Target/PrimeType))
> summary(exp3_mcq_acc_aov)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99  19.77  0.1997

Error: Target:PrimeType
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType  3  0.219 0.07313   3.233 0.0227 *
Residuals 297  6.719 0.02262
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## SPECIFIC T TESTS
>
> e3_mcq_p = exp3_mcq_acc %>% filter(PrimeType == "p")
> e3_mcq_r = exp3_mcq_acc %>% filter(PrimeType == "r")
> e3_mcq_b = exp3_mcq_acc %>% filter(PrimeType == "b")
> e3_mcq_u = exp3_mcq_acc %>% filter(PrimeType == "u")
> t.test(e3_mcq_r$MCQAcc, e3_mcq_u$MCQAcc, paired = TRUE) ##sig
```

Paired t-test

```
data: e3_mcq_r$MCQAcc and e3_mcq_u$MCQAcc
t = -2.4791, df = 99, p-value = 0.01486
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.10202226 -0.01131107
sample estimates:
mean of the differences
 -0.05666667
```

```
> t.test(e3_mcq_r$MCQAcc, e3_mcq_p$MCQAcc, paired = TRUE)
```

Paired t-test

```
data: e3_mcq_r$MCQAcc and e3_mcq_p$MCQAcc
t = -2.5098, df = 99, p-value = 0.0137
```



```

alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.1034554 -0.0121002
sample estimates:
mean of the differences
 -0.05777778

```

```

> ezANOVA(data = exp3_mcq_acc, wid = .(Target),
+         dv = .(MCQAcc), within = .(PrimeType))

```

```

$ANOVA
      Effect DFn DFd      F      p p<.05      ges
2 PrimeType   3 297 3.232512 0.02271627 * 0.00821472

$`Mauchly's Test for Sphericity`
      Effect      W      p p<.05
2 PrimeType 0.9448535 0.3532391

$`Sphericity Corrections`
      Effect      GGe      p[GG] p[GG]<.05      HFe      p[HF] p[HF]<.05
2 PrimeType 0.9610422 0.02436696 * 0.9929324 0.02300697 *

```

```

> exp3_mcqacc_lsm = lsmeans::lsmeans(exp3_mcq_acc_aov, c("PrimeType"))
> prime_effect = cld(exp3_mcqacc_lsm, alpha = 0.05,
+                   adjust = "tukey", details = TRUE)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.1))

```

	contrast	estimate	SE	df	t.ratio	p.value
2	u - r	0.0566667	0.0212709	297	2.664049	0.0404050
4	p - r	0.0577778	0.0212709	297	2.716285	0.0350551

```

> ## MULTIPLE CHOICE ERRORS
>
> ## before we do ANOVA, we need to replace NAs with 0.
>
> for (i in 1: nrow(exp3_mcq)){
+   if(is.na(exp3_mcq[i,9])){
+     exp3_mcq[i,9] = 0
+   }
+ }
> exp3_mcq_aov = aov(data = exp3_mcq, Proportion ~ PrimeType*ChosenPrime +
+                   Error(Target/(PrimeType*ChosenPrime)))
> summary(exp3_mcq_aov)

```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99  7.366  0.0744

Error: Target:PrimeType
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType  3  0.076 0.02525  1.414  0.239
Residuals 297  5.303 0.01785

Error: Target:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
ChosenPrime  3  57.85  19.282  99.94 <2e-16 ***
Residuals  297  57.30  0.193
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeType:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType:ChosenPrime  9  4.19  0.4660  7.141 4.83e-10 ***
Residuals  891  58.15  0.0653
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> library(ez)
> ezANOVA(data = exp3_mcq, wid = .(Target),
+         dv = .(Proportion), within = .(PrimeType, ChosenPrime))
```

```
$ANOVA
      Effect DFn DFd      F      p p<.05      ges
2      PrimeType  3 297  1.414180 2.387070e-01 0.0005909246
3      ChosenPrime  3 297 99.942258 9.598780e-45 * 0.3110603239
4 PrimeType:ChosenPrime  9 891  7.140808 4.826391e-10 * 0.0316994672

$`Mauchly's Test for Sphericity`
      Effect      W      p p<.05
2      PrimeType 0.936236742 2.658746e-01
3      ChosenPrime 0.137081845 5.311129e-40 *
4 PrimeType:ChosenPrime 0.005720887 2.537114e-77 *

$`Sphericity Corrections`
      Effect      GGe      p[GG] p[GG]<.05      HFe      p[HF]
2      PrimeType 0.9607092 2.397818e-01 0.9925756 2.389134e-01
3      ChosenPrime 0.5192956 1.581633e-24 * 0.5260856 8.184735e-25
4 PrimeType:ChosenPrime 0.5373125 2.568528e-06 * 0.5682524 1.439945e-06
p[HF]<.05
2
3      *
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> exp3_errors_lsm = lsmeans::lsmeans(exp3_mcq_aov, c("PrimeType", "ChosenPrime"))
> prime_effect = cld(exp3_errors_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = c("PrimeType"))
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.8))
```

	contrast	PrimeType	estimate	SE	df	t.ratio	p.value
	:-----	:-----	:-----	:-----	:-----	:-----	:-----
2	r - p	b	0.3957976	0.0440861	897.5443	8.977833	0.0000000
3	r - u	b	0.3916310	0.0440861	897.5443	8.883321	0.0000000
4	b - p	b	0.4233690	0.0440861	897.5443	9.603233	0.0000000
5	b - u	b	0.4192024	0.0440861	897.5443	9.508721	0.0000000
7	u - p	p	0.0555952	0.0440861	897.5443	1.261061	0.5880485
8	r - p	p	0.3229286	0.0440861	897.5443	7.324953	0.0000000
9	r - u	p	0.2673333	0.0440861	897.5443	6.063892	0.0000000
10	b - p	p	0.3957619	0.0440861	897.5443	8.977023	0.0000000
11	b - u	p	0.3401667	0.0440861	897.5443	7.715962	0.0000000
12	b - r	p	0.0728333	0.0440861	897.5443	1.652070	0.3499954
14	r - p	r	0.2431310	0.0440861	897.5443	5.514912	0.0000003
15	r - u	r	0.2247024	0.0440861	897.5443	5.096899	0.0000025
16	b - p	r	0.5817738	0.0440861	897.5443	13.196310	0.0000000
17	b - u	r	0.5633452	0.0440861	897.5443	12.778297	0.0000000
18	b - r	r	0.3386429	0.0440861	897.5443	7.681398	0.0000000
20	r - p	u	0.2939444	0.0440861	897.5443	6.667509	0.0000000
21	r - u	u	0.2677778	0.0440861	897.5443	6.073973	0.0000000
22	b - p	u	0.4051270	0.0440861	897.5443	9.189450	0.0000000
23	b - u	u	0.3789603	0.0440861	897.5443	8.595914	0.0000000
24	b - r	u	0.1111825	0.0440861	897.5443	2.521941	0.0572549

24 Comparing YA 48 ms with OA NotthePrime

```
> for (i in 1: nrow(final_mcq)){
+   if(is.na(final_mcq[i,9])){
+     final_mcq[i,9] = 0
+   }
+ }
> exp3_compare_1 = subset(final_mcq, final_mcq$StudyNo == '6' |
+                         final_mcq$StudyNo == '1')
> compare_aov_1 = aov(data = exp3_compare_1, Proportion ~ StudyNo*PrimeType*ChosenPrime
+                     Error(Target/(StudyNo*PrimeType*ChosenPrime)))
> summary(compare_aov_1)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
```

```

Residuals 99 8.959 0.09049

Error: Target:StudyNo
      Df Sum Sq Mean Sq F value Pr(>F)
StudyNo 1 0.479 0.4793 9.569 0.00257 **
Residuals 99 4.959 0.0501
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeType
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType 3 0.026 0.008629 0.353 0.787
Residuals 297 7.267 0.024468

Error: Target:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
ChosenPrime 3 99.2 33.07 130.6 <2e-16 ***
Residuals 297 75.2 0.25
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:StudyNo:PrimeType
      Df Sum Sq Mean Sq F value Pr(>F)
StudyNo:PrimeType 3 0.077 0.02551 1.12 0.341
Residuals 297 6.765 0.02278

Error: Target:StudyNo:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
StudyNo:ChosenPrime 3 0.686 0.22867 2.482 0.0611 .
Residuals 297 27.362 0.09213
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeType:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeType:ChosenPrime 9 5.18 0.5759 8.778 1.01e-12 ***
Residuals 891 58.46 0.0656
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:StudyNo:PrimeType:ChosenPrime
      Df Sum Sq Mean Sq F value Pr(>F)
StudyNo:PrimeType:ChosenPrime 9 0.66 0.07332 1.113 0.351
Residuals 891 58.72 0.06590

```

```

> exp3_compare_1 = subset(final_mcq, final_mcq$StudyNo == '6' |
+                           final_mcq$StudyNo == '1')
> exp3_compare_2 = subset(final_mcq, final_mcq$StudyNo == '1' |

```

```

+               final_mcq$StudyNo == '5')
> compare_aov_2 = aov(data = exp3_compare_2, Proportion ~ StudyNo*PrimeType*ChosenPrime
+               Error(Target/(StudyNo*PrimeType*ChosenPrime)))
> summary(compare_aov_2)

```

Error: Target

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Residuals	99	13.49	0.1362		

Error: Target:StudyNo

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
StudyNo	1	0.496	0.4964	15.1	0.000184 ***
Residuals	99	3.254	0.0329		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeType

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeType	3	0.209	0.06977	2.942	0.0334 *
Residuals	297	7.043	0.02371		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:ChosenPrime

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
ChosenPrime	3	101.56	33.85	111.1	<2e-16 ***
Residuals	297	90.52	0.30		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:StudyNo:PrimeType

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
StudyNo:PrimeType	3	0.091	0.03048	1.222	0.302
Residuals	297	7.410	0.02495		

Error: Target:StudyNo:ChosenPrime

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
StudyNo:ChosenPrime	3	0.711	0.23690	3.278	0.0214 *
Residuals	297	21.464	0.07227		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeType:ChosenPrime

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeType:ChosenPrime	9	1.10	0.12182	1.582	0.116
Residuals	891	68.62	0.07701		

Error: Target:StudyNo:PrimeType:ChosenPrime

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
StudyNo:PrimeType:ChosenPrime	9	4.60	0.5113	8.441	3.59e-12 ***
Residuals	891	53.97	0.0606		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

25 Item Percent State Analysis

```
> j <- read.csv("MainJulie_ItemAgg.csv", header = TRUE, sep = ",")
> # j_statepercent = j[,c(1,2,3,4,5,6,74:89)]
>
> j_statepercent = j[,c(1,2,3,4,5,6, 90:105)]
> j_statepercent$value.TargetNo = as.factor(j_statepercent$value.TargetNo)
> library(tidyr)
> library(dplyr)
> statepercent <- j_statepercent %>%
+   gather(StatePrime, Percent,
+         r_know, r_dontknow, r_other, r_TOT,
+         p_know, p_dontknow, p_other, p_TOT,
+         b_know, b_dontknow, b_other, b_TOT,
+         u_know, u_dontknow, u_other, u_TOT) %>%
+   separate(StatePrime, c('Prime', 'State'), sep = "_") %>%
+   arrange(value.Target)
> colnames(statepercent) = c("AgeGroup", "StudyNo", "Target", "TargetNo",
+   "WordType", "Proper",
+   "PrimeCondition", "State", "Percent")
> statepercent$AgeGroup <- as.factor(statepercent$AgeGroup)
> statepercent$Target <- as.factor(statepercent$Target)
> statepercent$StudyNo <- as.factor(statepercent$StudyNo)
> statepercent$PrimeCondition <- as.factor(statepercent$PrimeCondition)
> statepercent$State <- as.factor(statepercent$State)
> statepercent$Percent <- as.numeric(as.character(statepercent$Percent))
> for(i in 1:nrow(statepercent)){
+   if(is.na(statepercent[i,9])) {
+     statepercent[i,9] = 0
+   }
+   else
+     statepercent[i,9] = statepercent[i,9]
+ }
> statepercent_exp1 = statepercent %>% filter(StudyNo == '2' | StudyNo == '4')
> statepercent_exp2 = statepercent %>% filter(StudyNo == '5' | StudyNo == '6')
> statepercent_exp3 = statepercent %>% filter(StudyNo == '1')
>
```

25.1 Experiment 1

25.1.1 overall

```
> e1_all_aov = aov(data = statepercent_exp1,
+                   Percent ~ AgeGroup*State*PrimeCondition +
+                   Error(Target/(AgeGroup*State*PrimeCondition)))
> summary(e1_all_aov)
```

```
Error: Target
      Df    Sum Sq   Mean Sq F value Pr(>F)
Residuals 99 9.711e-18 9.809e-20

Error: Target:AgeGroup
      Df    Sum Sq   Mean Sq F value   Pr(>F)
AgeGroup  1 1.665e-18 1.665e-18   26.16 1.54e-06 ***
Residuals 99 6.303e-18 6.370e-20
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:State
      Df Sum Sq Mean Sq F value Pr(>F)
State    3  56.53  18.844   131.9 <2e-16 ***
Residuals 297  42.42   0.143
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeCondition
      Df    Sum Sq   Mean Sq F value   Pr(>F)
PrimeCondition  3 6.56e-19 2.186e-19    4.81 0.00275 **
Residuals      297 1.35e-17 4.546e-20
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:State
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:State  3  9.663   3.221    57.9 <2e-16 ***
Residuals      297 16.522   0.056
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition
      Df    Sum Sq   Mean Sq F value   Pr(>F)
AgeGroup:PrimeCondition  3 6.800e-20 2.281e-20    0.428  0.733
Residuals              297 1.584e-17 5.333e-20

Error: Target:State:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
State:PrimeCondition  9  2.726  0.30288   13.52 <2e-16 ***
```

```

Residuals          891 19.965 0.02241
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:State:PrimeCondition
              Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:State:PrimeCondition    9  0.162  0.01796    0.911   0.515
Residuals                    891 17.567 0.01972

```

25.1.2 know

```

> e1_know = statepercent_exp1 %>% filter(State == "know")
> e1_know_aov = aov(data = e1_know,
+                   Percent ~ AgeGroup*PrimeCondition +
+                   Error(Target/(AgeGroup*PrimeCondition)))
> summary(e1_know_aov)

```

```

Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99  22.47    0.227

Error: Target:AgeGroup
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  0.557   0.5571   10.96 0.0013 **
Residuals 99   5.032   0.0508

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition    3   1.725   0.5751   24.06 5.75e-14 ***
Residuals       297   7.099   0.0239

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition    3  0.039  0.01286    0.603   0.614
Residuals                297   6.335  0.02133

```

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e1_know_aov,
+                               c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE, by = "AgeGroup")

```



```
> library(knitr)
> kable(subset(prime_effect$comparisons,prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
1	r - u	Old	0.0955556	0.0212676	592.0855	4.493009	0.0000499
2	b - u	Old	0.1133333	0.0212676	592.0855	5.328918	0.0000008
4	p - u	Old	0.1366667	0.0212676	592.0855	6.426048	0.0000000
7	r - u	Young	0.0600000	0.0212676	592.0855	2.821192	0.0254170
8	b - u	Young	0.0811111	0.0212676	592.0855	3.813833	0.0008684
10	p - u	Young	0.1133333	0.0212676	592.0855	5.328918	0.0000008

```
> target_p = e1_know %>% filter(PrimeCondition == "p")
> target_r = e1_know %>% filter(PrimeCondition == "r")
> target_b = e1_know %>% filter(PrimeCondition == "b")
> target_u = e1_know %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_r$Percent
t = -5.5638, df = 199, p-value = 8.449e-08
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.10534440 -0.05021115
sample estimates:
mean of the differences
 -0.07777778
```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = -6.5518, df = 199, p-value = 4.759e-10
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.12648424 -0.06796021
sample estimates:
mean of the differences
 -0.09722222
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
```

```
t = -8.7905, df = 199, p-value = 7.015e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.15304113 -0.09695887
sample estimates:
mean of the differences
      -0.125
```

```
> ## effect of age
> target_y = e1_know %>% filter(AgeGroup == "Young")
> target_o = e1_know %>% filter(AgeGroup == "Old")
> t.test(target_y$Percent, target_o$Percent, paired = FALSE)
```

Welch Two Sample t-test

```
data: target_y$Percent and target_o$Percent
t = -3.2266, df = 795.5, p-value = 0.001304
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.08488612 -0.02066944
sample estimates:
mean of x mean of y
0.3958333 0.4486111
```

```
>
```

25.1.3 dont know

```
> e1_dontknow = statepercent_exp1 %>% filter(State == "dontknow")
> e1_dontknow_aov = aov(data = e1_dontknow,
+                        Percent ~ AgeGroup*PrimeCondition +
+                        Error(Target/(AgeGroup*PrimeCondition)))
> summary(e1_dontknow_aov)
```

```
Error: Target
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Residuals	99	13.37	0.1351		

```
Error: Target:AgeGroup
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup	1	3.556	3.556	48.26	3.98e-10 ***
Residuals	99	7.293	0.074		

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Error: Target:PrimeCondition
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
--	----	--------	---------	---------	--------

```

PrimeCondition    3    0.736 0.24547    11.22 5.36e-07 ***
Residuals        297    6.495 0.02187
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition
              Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition    3    0.022 0.007202    0.402  0.751
Residuals                297    5.315 0.017895

```

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e1_dontknow_aov,
+                               c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = "AgeGroup")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))

```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
4	u - r	Old	0.0866667	0.0199409	588.1261	4.346179	0.0000960
5	u - p	Old	0.0766667	0.0199409	588.1261	3.844697	0.0007703
6	u - b	Old	0.0633333	0.0199409	588.1261	3.176054	0.0085223
10	u - r	Young	0.0722222	0.0199409	588.1261	3.621816	0.0018019
11	u - p	Young	0.0611111	0.0199409	588.1261	3.064614	0.0121820

```

> target_p = e1_dontknow %>% filter(PrimeCondition == "p")
> target_r = e1_dontknow %>% filter(PrimeCondition == "r")
> target_b = e1_dontknow %>% filter(PrimeCondition == "b")
> target_u = e1_dontknow %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)

```

Paired t-test

```

data: target_u$Percent and target_r$Percent
t = 5.2419, df = 199, p-value = 4.041e-07
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.0460923 0.1016855
sample estimates:
mean of the differences
 0.07388889

```

```

> t.test(target_u$Percent, target_b$Percent, paired = TRUE)

```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = 3.7438, df = 199, p-value = 0.0002372
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.02550402 0.08227376
sample estimates:
mean of the differences
      0.05388889
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = 5.5019, df = 199, p-value = 1.147e-07
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.04776233 0.10112656
sample estimates:
mean of the differences
      0.07444444
```

25.1.4 other

```
> e1_other = statepercent_exp1 %>% filter(State == "other")
> e1_other_aov = aov(data = e1_other,
+                    Percent ~ AgeGroup*PrimeCondition +
+                    Error(Target/(AgeGroup*PrimeCondition)))
> summary(e1_other_aov)
```

Error: Target

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Residuals	99	4.623	0.04669		

Error: Target:AgeGroup

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup	1	5.463	5.463	194.4	<2e-16 ***
Residuals	99	2.782	0.028		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeCondition

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
PrimeCondition	3	0.138	0.04590	4.297	0.00548 **

```

Residuals      297   3.172 0.01068
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition
              Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition    3   0.030  0.01001    0.952   0.416
Residuals                297   3.126  0.01052

```

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e1_other_aov,
+                               c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE, by = "PrimeCondition")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))

```

contrast	PrimeCondition	estimate	SE	df	t.ratio	p.value
Young - Old b		0.1633333	0.0172736	314.2215	9.455687	0
Young - Old p		0.1466667	0.0172736	314.2215	8.490821	0
Young - Old r		0.1711111	0.0172736	314.2215	9.905957	0
Young - Old u		0.1800000	0.0172736	314.2215	10.420553	0

```

>
>

```

25.1.5 TOT

```

> e1_TOT = statepercent_exp1 %>% filter(State == "TOT")
> e1_TOT_aov = aov(data = e1_TOT,
+                 Percent ~ AgeGroup*PrimeCondition +
+                 Error(Target/(AgeGroup*PrimeCondition)))
> summary(e1_TOT_aov)

```

```

Error: Target
              Df Sum Sq Mean Sq F value Pr(>F)
Residuals    99   1.956  0.01976

Error: Target:AgeGroup
              Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup      1  0.0868  0.08681    6.074 0.0154 *
Residuals    99  1.4147  0.01429
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
Error: Target:PrimeCondition
              Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition   3  0.126  0.04216     3.914 0.00916 **
Residuals       297   3.199  0.01077
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition
              Df Sum Sq Mean Sq F value    Pr(>F)
AgeGroup:PrimeCondition   3  0.0714  0.023801     2.533 0.0572 .
Residuals                 297  2.7912  0.009398
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e1_TOT_aov,
+                               c("AgeGroup","PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE)
> library(knitr)
> kable(subset(prime_effect$comparisons,prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	estimate	SE	df	t.ratio	p.value
7	Young,r - Old,p	0.0466667	0.0150385	521.5407	3.103138	0.0419683
11	Young,p - Old,p	0.0477778	0.0145748	380.8502	3.278118	0.0249941
16	Young,u - Old,p	0.0555556	0.0150385	521.5407	3.694212	0.0059239
22	Old,u - Old,p	0.0600000	0.0142020	591.2586	4.224755	0.0007249

```
> target_o_u = e1_TOT %>% filter(AgeGroup == "Old" & PrimeCondition == "u")
> target_o_p = e1_TOT %>% filter(AgeGroup == "Old" & PrimeCondition == "p")
> target_o_b = e1_TOT %>% filter(AgeGroup == "Old" & PrimeCondition == "b")
> target_o_r = e1_TOT %>% filter(AgeGroup == "Old" & PrimeCondition == "r")
> t.test(target_o_u$Percent, target_o_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_o_u$Percent and target_o_p$Percent
t = 4.3747, df = 99, p-value = 3.012e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.03278632 0.08721368
sample estimates:
mean of the differences
      0.06
```

```
> t.test(target_o_u$Percent, target_o_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_o_u$Percent and target_o_r$Percent
t = 2.1117, df = 99, p-value = 0.03723
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.001677475 0.053878080
sample estimates:
mean of the differences
      0.02777778
```

```
> t.test(target_o_u$Percent, target_o_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_o_u$Percent and target_o_b$Percent
t = 3.3704, df = 99, p-value = 0.001071
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01690821 0.06531401
sample estimates:
mean of the differences
      0.04111111
```

```
> target_p = e1_TOT %>% filter(PrimeCondition == "p")
> target_r = e1_TOT %>% filter(PrimeCondition == "r")
> target_b = e1_TOT %>% filter(PrimeCondition == "b")
> target_u = e1_TOT %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_r$Percent
t = 1.7759, df = 199, p-value = 0.07727
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-0.002023557 0.038690223
sample estimates:
mean of the differences
      0.01833333
```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = 2.621, df = 199, p-value = 0.009445
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.006465853 0.045756369
sample estimates:
mean of the differences
      0.02611111
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = 2.9963, df = 199, p-value = 0.00308
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01158554 0.05619223
sample estimates:
mean of the differences
      0.03388889
```

```
> target_y = e1_TOT %>% filter(AgeGroup == "Young")
> target_o = e1_TOT %>% filter(AgeGroup == "Old")
> t.test(target_y$Percent, target_o$Percent, paired = FALSE)
```

Welch Two Sample t-test

```
data: target_y$Percent and target_o$Percent
t = 2.692, df = 788.29, p-value = 0.007253
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.005641885 0.036024781
sample estimates:
 mean of x  mean of y
0.11638889 0.09555556
```

25.1.6 plot

```
> exp1_statepercent = Rmisc::summarySE(statepercent_exp1,
+                                     measurevar = "Percent",
+                                     groupvars = c("State", "AgeGroup", "PrimeCondition"))
> exp1_statepercent = arrange(exp1_statepercent, desc(AgeGroup))
> library(ggplot2)
```

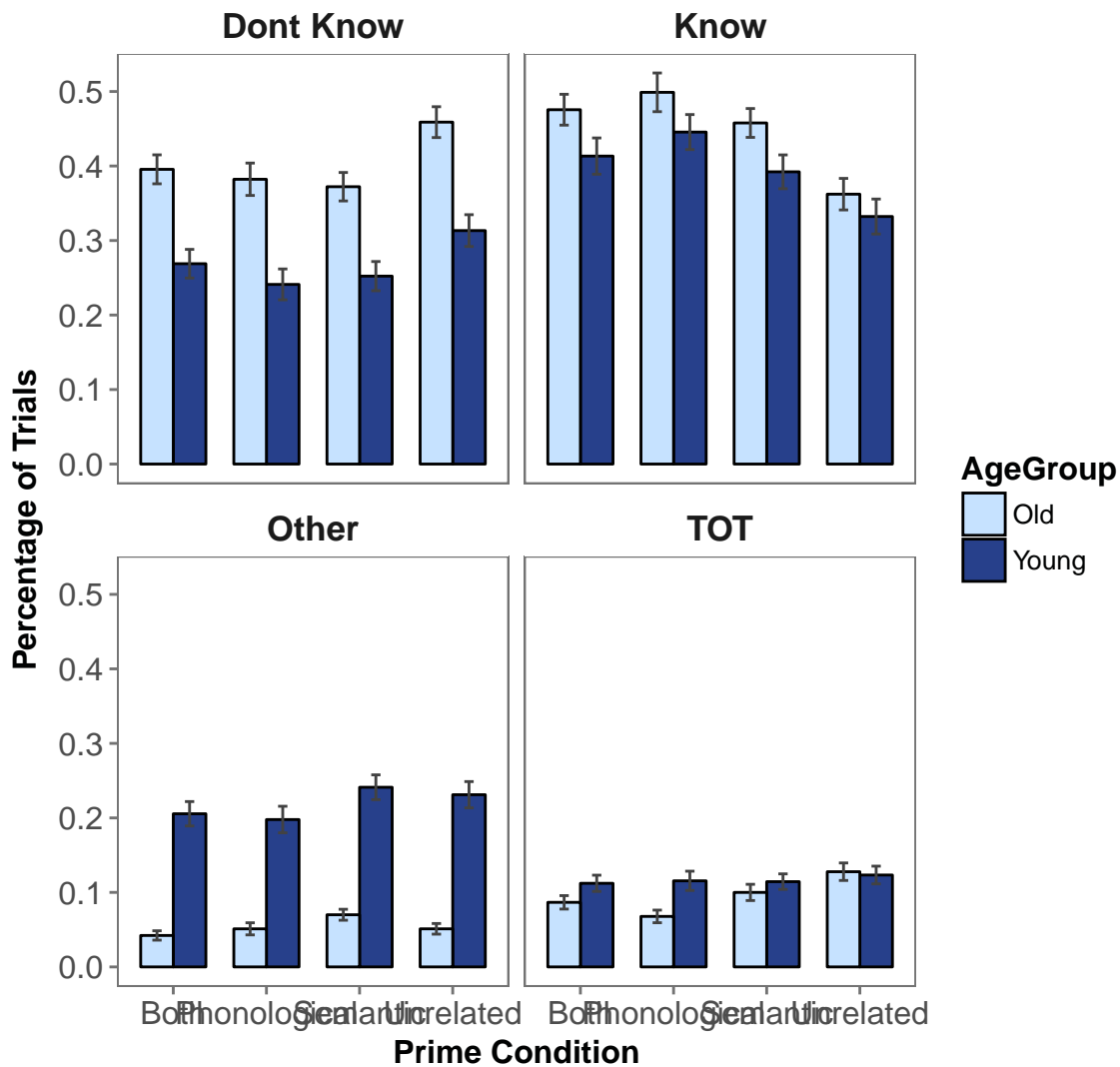


```

> library(ggthemes)
> e1_percentplot = exp1_statepercent %>%
+
+ mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                           labels = c("Both", "Phonological",
+                                       "Semantic", "Unrelated")),
+        RetrievalState = factor(State, levels = unique(State),
+                                labels = c("Dont Know", "Know", "Other", "TOT")))%>%
+
+ ggplot(aes(x = PrimeType, y = Percent,
+            group = AgeGroup, fill = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7,
+           color= "black")+
+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+                width=.2, color = "gray26",
+                position = position_dodge(0.7))+
+   theme_few()+
+   facet_wrap(~RetrievalState)+
+   scale_fill_manual(values = c("slategray1", "royalblue4"))+
+   xlab("Prime Condition") + ylab("Percentage of Trials") +
+   ggtitle("E1 Items: Young and Old Adults (No Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e1_percentplot
>

```

E1 Items: Young and Old Adults (No Instructions)



25.2 Experiment 2

25.2.1 overall

```
> e2_all_aov = aov(data = statepercent_exp2,
+                   Percent ~ AgeGroup*State*PrimeCondition +
+                   Error(Target/(AgeGroup*State*PrimeCondition)))
> summary(e2_all_aov)
```

```
Error: Target
      Df    Sum Sq   Mean Sq F value Pr(>F)
```

```

Residuals 99 1.42e-26 1.434e-28

Error: Target:AgeGroup
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup 1 2.70e-30 2.707e-30 0.233 0.63
Residuals 99 1.15e-27 1.162e-29

Error: Target:State
      Df Sum Sq Mean Sq F value Pr(>F)
State 3 65.27 21.757 145.6 <2e-16 ***
Residuals 297 44.38 0.149
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition 3 2.400e-30 8.110e-31 0.239 0.869
Residuals 297 1.009e-27 3.397e-30

Error: Target:AgeGroup:State
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:State 3 13.00 4.333 83.74 <2e-16 ***
Residuals 297 15.37 0.052
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition 3 4.400e-29 1.466e-29 0.922 0.43
Residuals 297 4.723e-27 1.590e-29

Error: Target:State:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
State:PrimeCondition 9 2.127 0.23634 8.989 4.55e-13 ***
Residuals 891 23.428 0.02629
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:State:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:State:PrimeCondition 9 0.365 0.04057 1.448 0.163
Residuals 891 24.971 0.02803

```

25.2.2 know

```

> e2_know = statepercent_exp2 %>% filter(State == "know")
> e2_know_aov = aov(data = e2_know,

```

```
+ Percent ~ AgeGroup*PrimeCondition +
+ Error(Target/(AgeGroup*PrimeCondition)))
> summary(e2_know_aov)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99  21.96  0.2218

Error: Target:AgeGroup
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  0.538  0.5382  9.386 0.00282 **
Residuals 99  5.677  0.0573
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3  1.199  0.3996  14.37 9.09e-09 ***
Residuals      297  8.258  0.0278
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition  3  0.203  0.06763  2.131 0.0964 .
Residuals              297  9.426  0.03174
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e2_know_aov,
+                               c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                     adjust = "tukey", details = TRUE, by = "AgeGroup")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
7	b - u	Young	0.09750	0.0244014	591.4209	3.995673	0.0004215
8	r - u	Young	0.10375	0.0244014	591.4209	4.251805	0.0001445
10	p - u	Young	0.14875	0.0244014	591.4209	6.095962	0.0000000

```
> target_p = e2_know %>% filter(PrimeCondition == "p")
> target_r = e2_know %>% filter(PrimeCondition == "r")
> target_b = e2_know %>% filter(PrimeCondition == "b")
```

```
> target_u = e2_know %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_r$Percent
t = -4.6637, df = 199, p-value = 5.69e-06
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.10760193 -0.04364807
sample estimates:
mean of the differences
 -0.075625
```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = -4.2944, df = 199, p-value = 2.736e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.10670319 -0.03954681
sample estimates:
mean of the differences
 -0.073125
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = -5.9969, df = 199, p-value = 9.326e-09
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.13952693 -0.07047307
sample estimates:
mean of the differences
 -0.105
```

```
> target_y = e2_know %>% filter(AgeGroup == "Young")
> target_o = e2_know %>% filter(AgeGroup == "Old")
> t.test(target_y$Percent, target_o$Percent, paired = FALSE)
```

Welch Two Sample t-test

```
data: target_y$Percent and target_o$Percent
```

```

t = -3.0318, df = 789.97, p-value = 0.00251
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.0854618 -0.0182882
sample estimates:
mean of x mean of y
 0.373750  0.425625

```

```
>
```

25.2.3 dont know

```

> e2_dontknow = statepercent_exp2 %>% filter(State == "dontknow")
> e2_dontknow_aov = aov(data = e2_dontknow,
+                        Percent ~ AgeGroup*PrimeCondition +
+                        Error(Target/(AgeGroup*PrimeCondition)))
> summary(e2_dontknow_aov)

```

```

Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99  17.52   0.1769

Error: Target:AgeGroup
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1  6.730   6.730  110.6 <2e-16 ***
Residuals 99  6.022   0.061
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3  0.825  0.27502   9.761 3.68e-06 ***
Residuals      297  8.368  0.02818
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:AgeGroup:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition  3  0.131  0.04367   1.49  0.217
Residuals              297  8.703  0.02930

```

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e2_dontknow_aov,
+                               c("AgeGroup","PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,

```

```
+ adjust = "tukey", details = TRUE, by = "AgeGroup")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	AgeGroup	estimate	SE	df	t.ratio	p.value
10	u - b	Young	0.10500	0.0239748	593.7718	4.379599	0.0000827
11	u - r	Young	0.10125	0.0239748	593.7718	4.223185	0.0001632
12	u - p	Young	0.07250	0.0239748	593.7718	3.024009	0.0138267

```
> target_p = e2_dontknow %>% filter(PrimeCondition == "p")
> target_r = e2_dontknow %>% filter(PrimeCondition == "r")
> target_b = e2_dontknow %>% filter(PrimeCondition == "b")
> target_u = e2_dontknow %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_r$Percent
t = 3.7815, df = 199, p-value = 0.0002061
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.03170229 0.10079771
sample estimates:
mean of the differences
      0.06625
```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = 4.7493, df = 199, p-value = 3.9e-06
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.04751431 0.11498569
sample estimates:
mean of the differences
      0.08125
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = 4.2031, df = 199, p-value = 3.975e-05
alternative hypothesis: true difference in means is not equal to 0
```

```

95 percent confidence interval:
 0.03815345 0.10559655
sample estimates:
mean of the differences
      0.071875

```

```
>
```

25.2.4 other

```

> e2_other = statepercent_exp2 %>% filter(State == "other")
> e2_other_aov = aov(data = e2_other,
+                    Percent ~ AgeGroup*PrimeCondition +
+                    Error(Target/(AgeGroup*PrimeCondition)))
> summary(e2_other_aov)

```

```

Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99   2.863   0.02892

Error: Target:AgeGroup
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1   3.903     3.903   201.2 <2e-16 ***
Residuals 99   1.920     0.019

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3   0.052   0.01726    1.448   0.229
Residuals      297   3.540   0.01192

Error: Target:AgeGroup:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup:PrimeCondition  3   0.0196   0.00653    0.625   0.599
Residuals              297   3.1035   0.01045

```

```

> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e2_other_aov,
+                               c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE, by = "PrimeCondition")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))

```


contrast	PrimeCondition	estimate	SE	df	t.ratio	p.value
Young - Old	b	0.14875	0.0159279	362.2474	9.338975	0
Young - Old	p	0.13625	0.0159279	362.2474	8.554187	0
Young - Old	r	0.12500	0.0159279	362.2474	7.847878	0
Young - Old	u	0.14875	0.0159279	362.2474	9.338975	0

```
> target_y = e2_other %>% filter(AgeGroup == "Young")
> target_o = e2_other %>% filter(AgeGroup == "Old")
> t.test(target_y$Percent, target_o$Percent, paired = FALSE)
```

Welch Two Sample t-test

```
data: target_y$Percent and target_o$Percent
t = 16.458, df = 543.84, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.123015 0.156360
sample estimates:
mean of x mean of y
0.1775000 0.0378125
```

25.2.5 TOT

```
> e2_TOT = statepercent_exp2 %>% filter(State == "TOT")
> e2_TOT_aov = aov(data = e2_TOT,
+                   Percent ~ AgeGroup*PrimeCondition +
+                   Error(Target/(AgeGroup*PrimeCondition)))
> summary(e2_TOT_aov)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99   2.034  0.02055

Error: Target:AgeGroup
      Df Sum Sq Mean Sq F value Pr(>F)
AgeGroup  1   1.829   1.8288   103.5 <2e-16 ***
Residuals 99   1.749   0.0177

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3   0.051  0.01714    1.561  0.199
Residuals      297   3.261  0.01098
```

```
Error: Target:AgeGroup:PrimeCondition
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
AgeGroup:PrimeCondition	3	0.012	0.00388	0.308	0.819
Residuals	297	3.738	0.01259		

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e2_TOT_aov,
+                               c("AgeGroup", "PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE, by = "PrimeCondition")
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

contrast	PrimeCondition	estimate	SE	df	t.ratio	p.value
Young - Old b		0.08375	0.016648	386.2576	5.030648	8e-07
Young - Old p		0.09625	0.016648	386.2576	5.781491	0e+00
Young - Old r		0.09750	0.016648	386.2576	5.856576	0e+00
Young - Old u		0.10500	0.016648	386.2576	6.307081	0e+00

```
> target_p = e2_TOT %>% filter(PrimeCondition == "p")
> target_r = e2_TOT %>% filter(PrimeCondition == "r")
> target_b = e2_TOT %>% filter(PrimeCondition == "b")
> target_u = e2_TOT %>% filter(PrimeCondition == "u")
> t.test(target_p$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_p$Percent and target_r$Percent
t = -1.6533, df = 199, p-value = 0.09984
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.037002019 0.003252019
sample estimates:
mean of the differences
 -0.016875
```

```
> t.test(target_p$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_p$Percent and target_b$Percent
t = -1.734, df = 199, p-value = 0.08447
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
```

```
-0.042745149  0.002745149
sample estimates:
mean of the differences
           -0.02
```

```
> t.test(target_p$Percent, target_u$Percent, paired = TRUE)
```

Paired t-test

```
data: target_p$Percent and target_u$Percent
t = -1.7081, df = 199, p-value = 0.08917
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.039049332  0.002799332
sample estimates:
mean of the differences
           -0.018125
```

25.2.6 plot

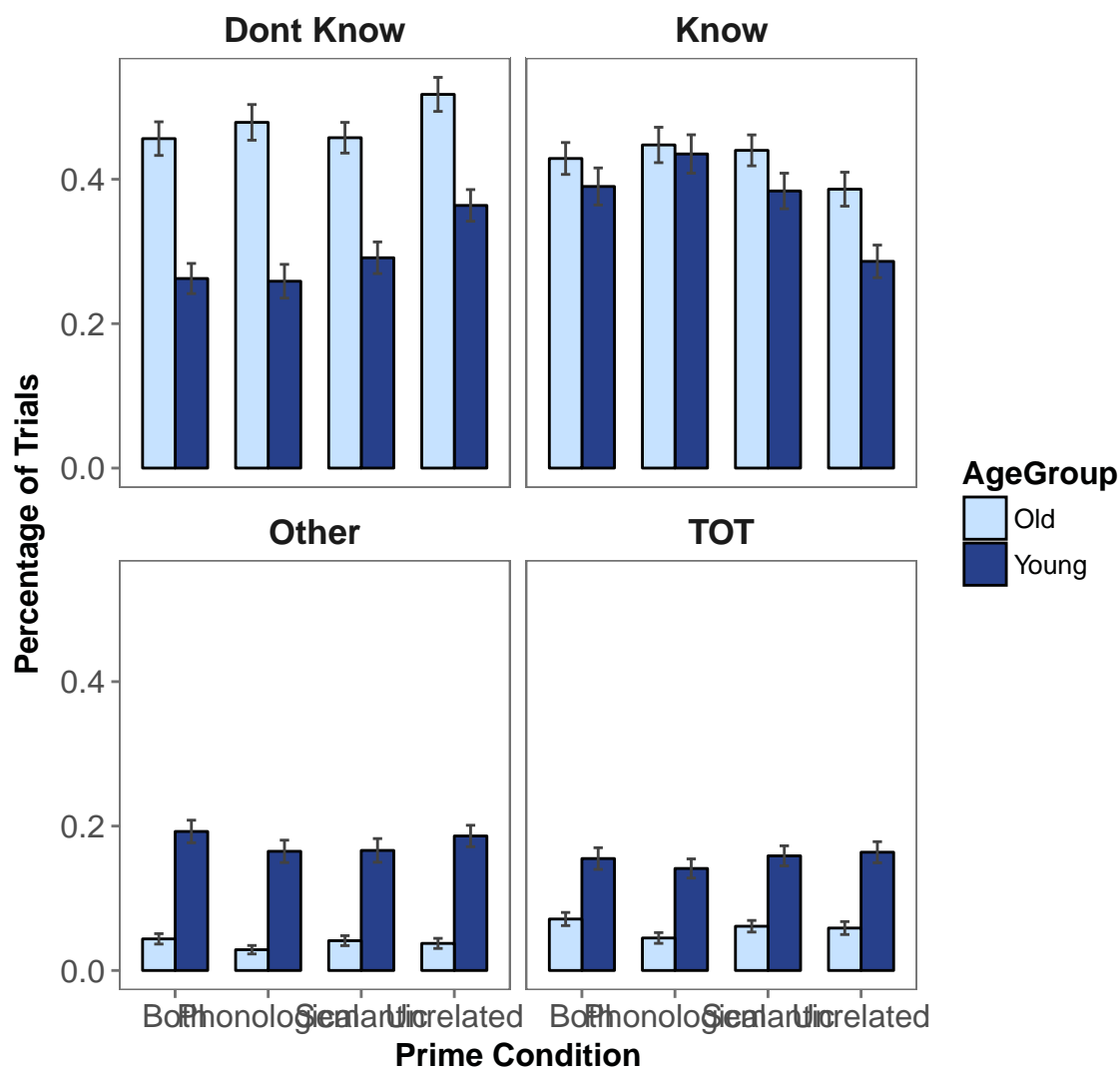
```
> exp2_statepercent= Rmisc::summarySE(statepercent_exp2,
+                                     measurevar = "Percent",
+                                     groupvars = c("State", "AgeGroup", "PrimeCondition"))
> library(ggplot2)
> library(ggthemes)
> e2_percentplot = exp2_statepercent %>%
+ mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                           labels = c("Both", "Phonological",
+                                       "Semantic", "Unrelated")),
+        RetrievalState = factor(State, levels = unique(State),
+                                labels = c("Dont Know", "Know", "Other", "TOT")))%>%
+
+ ggplot(aes(x = PrimeType, y = Percent,
+            group = AgeGroup, fill = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7,
+           color= "black")+
+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+               width=.2, color = "gray26",
+               position = position_dodge(0.7))+
+   theme_few()+
+   facet_wrap(~RetrievalState)+
+   scale_fill_manual(values = c("slategray1", "royalblue4"))+
+   xlab("Prime Condition") + ylab("Percentage of Trials") +
+   ggtitle("E2 Items: Young and Old Adults (With Instructions)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
```

```

+       plot.title = element_text(hjust = .5),
+       strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e2_percentplot
>

```

E2 Items: Young and Old Adults (With Instructions)



25.3 Experiment 3

25.3.1 know

```

> e3_know = statepercent_exp3 %>% filter(State == "know")
> e3_know_aov = aov(data = e3_know,

```

```
+               Percent ~ PrimeCondition +
+               Error(Target/PrimeCondition))
> summary(e3_know_aov)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99  12.78   0.1291

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value    Pr(>F)
PrimeCondition   3   0.758 0.25271   9.816 3.42e-06 ***
Residuals      297   7.646 0.02574
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmeans)
> library(multcomp)
> target_lsm = lsmeans::lsmeans(e3_know_aov,
+                               c("PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE)
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	estimate	SE	df	t.ratio	p.value
1	p - u	0.0666667	0.0226913	297	2.937981	0.0185958
2	b - u	0.0944444	0.0226913	297	4.162140	0.0002408
4	r - u	0.1155556	0.0226913	297	5.092501	0.0000037

```
> target_p = e3_know %>% filter(PrimeCondition == "p")
> target_r = e3_know %>% filter(PrimeCondition == "r")
> target_b = e3_know %>% filter(PrimeCondition == "b")
> target_u = e3_know %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_r$Percent
t = -5.8043, df = 99, p-value = 7.806e-08
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.15505885 -0.07605226
sample estimates:
mean of the differences
 -0.1155556
```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = -4.1528, df = 99, p-value = 6.969e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.13956997 -0.04931892
sample estimates:
mean of the differences
 -0.09444444
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = -3.0308, df = 99, p-value = 0.003113
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.11031271 -0.02302062
sample estimates:
mean of the differences
 -0.06666667
```

25.3.2 dont know

```
> e3_dontknow = statepercent_exp3 %>% filter(State == "dontknow")
> e3_dontknow_aov = aov(data = e3_dontknow,
+                        Percent ~ PrimeCondition +
+                        Error(Target/PrimeCondition))
> summary(e3_dontknow_aov)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99    12.3   0.1243

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3   0.243  0.08107    3.887 0.0095 **
Residuals    297   6.195  0.02086

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> options(contrasts = c('contr.sum', 'contr.poly'))
> library(lsmmeans)
> library(multcomp)
> target_lsm = lsmmeans::lsmmeans(e3_dontknow_aov,
+                                c("PrimeCondition"))
> prime_effect = cld(target_lsm, alpha = 0.05,
+                    adjust = "tukey", details = TRUE)
> library(knitr)
> kable(subset(prime_effect$comparisons, prime_effect$comparisons$p.value < 0.05 ))
```

	contrast	estimate	SE	df	t.ratio	p.value
4	u - r	0.0644444	0.0204249	297	3.155193	0.0095257
5	u - b	0.0544444	0.0204249	297	2.665594	0.0402372

```
> target_p = e3_dontknow %>% filter(PrimeCondition == "p")
> target_r = e3_dontknow %>% filter(PrimeCondition == "r")
> target_b = e3_dontknow %>% filter(PrimeCondition == "b")
> target_u = e3_dontknow %>% filter(PrimeCondition == "u")
> t.test(target_u$Percent, target_r$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_r$Percent
t = 3.1466, df = 99, p-value = 0.002182
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.02380655 0.10508234
sample estimates:
mean of the differences
 0.06444444
```

```
> t.test(target_u$Percent, target_b$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_b$Percent
t = 2.4848, df = 99, p-value = 0.01464
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01096802 0.09792086
sample estimates:
mean of the differences
 0.05444444
```

```
> t.test(target_u$Percent, target_p$Percent, paired = TRUE)
```

Paired t-test

```
data: target_u$Percent and target_p$Percent
t = 2.2249, df = 99, p-value = 0.02836
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.00492793 0.08618318
sample estimates:
mean of the differences
 0.04555556
```

25.3.3 other

```
> e3_other = statepercent_exp3 %>% filter(State == "other")
> e3_other_aov = aov(data = e3_other,
+                   Percent ~ PrimeCondition +
+                   Error(Target/PrimeCondition))
> summary(e3_other_aov)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99   4.499   0.04545

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3   0.006   0.001842    0.156   0.926
Residuals    297   3.516   0.011839
```

```
>
```

25.3.4 TOT

```
> e3_TOT = statepercent_exp3 %>% filter(State == "TOT")
> e3_TOT_aov = aov(data = e3_TOT,
+                 Percent ~ PrimeCondition +
+                 Error(Target/PrimeCondition))
> summary(e3_TOT_aov)
```

```
Error: Target
      Df Sum Sq Mean Sq F value Pr(>F)
Residuals 99   2.29   0.02313

Error: Target:PrimeCondition
      Df Sum Sq Mean Sq F value Pr(>F)
PrimeCondition  3   0.102   0.03416    2.466   0.0624 .
```



```
Residuals      297    4.114  0.01385
```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

25.3.5 plot

```
> exp3_statepercent= Rmisc::summarySE(statepercent_exp3,
+                                     measurevar = "Percent",
+                                     groupvars = c("State","AgeGroup", "PrimeCondition"))
> library(ggplot2)
> library(ggthemes)
> e3_percentplot = exp3_statepercent %>%
+ mutate(PrimeType = factor(PrimeCondition, levels = unique(PrimeCondition),
+                           labels = c("Both", "Phonological",
+                                       "Semantic", "Unrelated")),
+        RetrievalState = factor(State, levels = unique(State),
+                                labels = c("Dont Know", "Know", "Other", "TOT")))%>%
+
+ ggplot(aes(x = PrimeType, y = Percent,
+            group = AgeGroup, fill = AgeGroup))+
+   geom_bar(stat = "identity", position = "dodge", width = 0.7,
+           color= "black")+
+   geom_errorbar(aes(ymin=Percent - se, ymax=Percent + se),
+                width=.2, color = "gray26",
+                position = position_dodge(0.7))+
+   theme_few()+
+   facet_wrap(~RetrievalState)+
+   scale_fill_manual(values = c("royalblue4"))+
+   xlab("Prime Condition") + ylab("Percentage of Trials") +
+   ggtitle("E3 Items: Young (48 ms)") +
+   theme(axis.text = element_text(size = rel(1)),
+         axis.title = element_text(face = "bold", size = rel(1)),
+         legend.title = element_text(face = "bold", size = rel(1)),
+         plot.title = element_text(hjust = .5),
+         strip.text.x = element_text(face = "bold", size = rel(1.4)))
> e3_percentplot
>
```

E3 Items: Young (48 ms)

