

Semantic Networks Analysis

Abhilasha Kumar

January 16, 2018

Reading the Data

```
sem = read.csv("NetworksProject_AllSubjects.csv", header = TRUE, sep = ",")
sem$mean_len = as.double(as.character(sem$mean_len))
sem$mean_logf = as.double(as.character(sem$mean_logf))
sem$mean_ldtz = as.double(as.character(sem$mean_ldtz))
sem$mean_nmgz = as.double(as.character(sem$mean_nmgz))
sem$undirected = as.double(as.character(sem$undirected))
sem$directed = as.double(as.character(sem$directed))
sem$logp = as.double(as.character(sem$logp))

sem$mean_len = round(sem$mean_len, 2)
sem$mean_logf = round(sem$mean_logf, 2)
sem$mean_ldtz = round(sem$mean_ldtz, 2)
sem$mean_nmgz = round(sem$mean_nmgz, 2)
```

Related-Unrelated Decisions

```
library(dplyr)

sem_decision = group_by(sem, subject, pathlength, Type ) %>%
  summarize(Trials = n())

# With N = 40, there are 10 rows missing!

sem_decision[471,] = c("42406", 15, "Related", 0)
sem_decision[472,] = c("61558", 15, "Related", 0)
sem_decision[473,] = c("27113", 15, "Related", 0)
sem_decision[474,] = c("34686", 6, "Related", 0)
sem_decision[475,] = c("81385", 6, "Related", 0)
sem_decision[476,] = c("35312", 3, "Related", 0)
sem_decision[477,] = c("65089", 1, "Unrelated", 0)
sem_decision[478,] = c("65448", 4, "Related", 0)
sem_decision[479,] = c("65448", 15, "Related", 0)
sem_decision[480,] = c("84694", 6, "Related", 0)

sem_decision$Trials = as.numeric(as.character(sem_decision$Trials))

sem_decision_agg = group_by(sem_decision, pathlength, Type ) %>%
  summarise_at(vars(Trials), mean)
```

Plotting Proportions

```
sem_decision$Trials = as.numeric(as.character(sem_decision$Trials))
sem_decision_agg$Trials = as.numeric(as.character(sem_decision_agg$Trials))

sem_decision$proportion = sem_decision$Trials/40

sem_decision_agg$proportion = sem_decision_agg$Trials/(40)

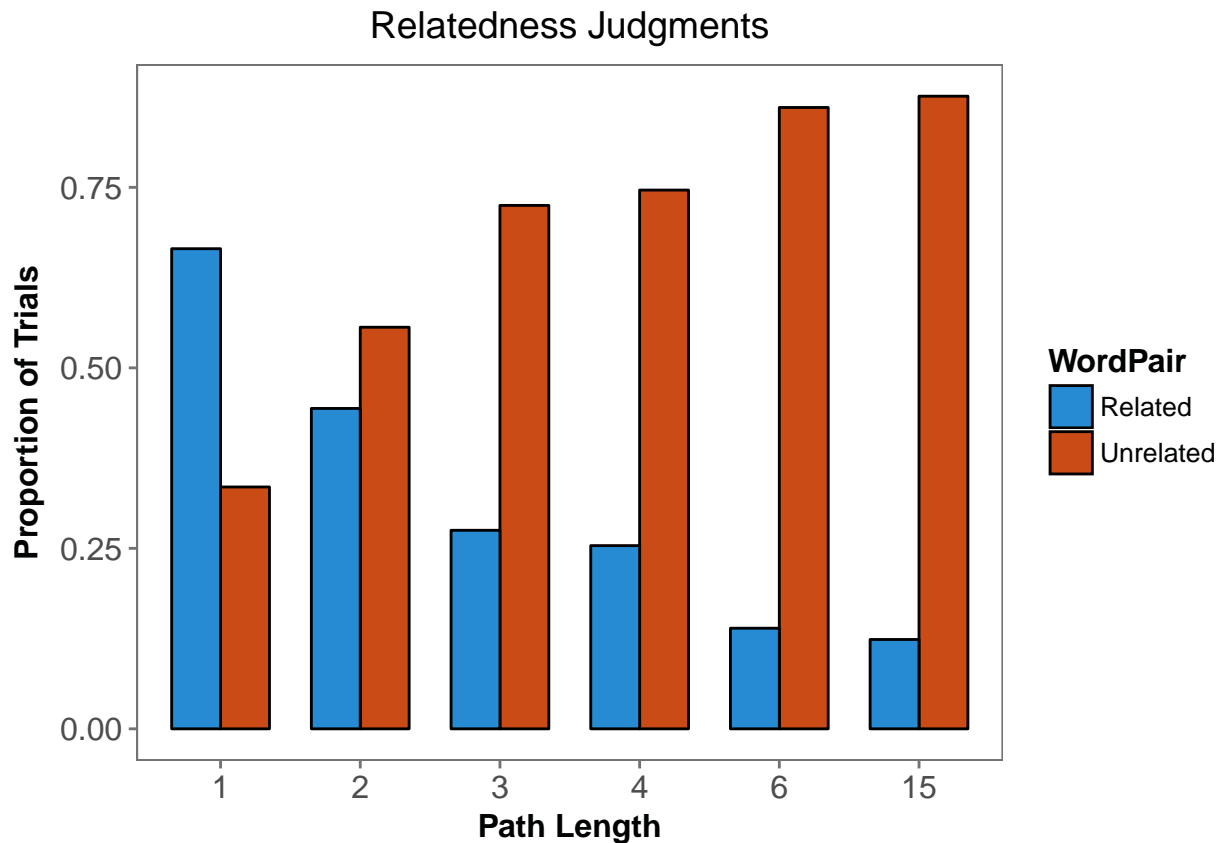
sem_decision$pathlengthfac = ordered(as.factor(as.character(sem_decision$pathlength)),
                                     levels = c("1", "2", "3", "4", "6", "15"))

sem_decision_agg$pathlengthfac = ordered(as.factor(as.character(sem_decision_agg$pathlength)),
                                         levels = c("1", "2", "3", "4", "6", "15"))

library(ggplot2)
library(ggthemes)

sem_decision_agg = sem_decision_agg[order(sem_decision_agg$pathlength,
                                          sem_decision_agg$Type),]

sem_decision_agg %>% mutate(WordPair = factor(Type,
                                              levels = unique(Type),
                                              labels = c("Related", "Unrelated")))%>%
  ggplot(aes(x = pathlengthfac, y = proportion,
            fill = WordPair, group = WordPair))+
  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_solarized()+
  xlab("Path Length") + ylab("Proportion of Trials") +
  ggtitle("Relatedness Judgments") +
  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)))
```



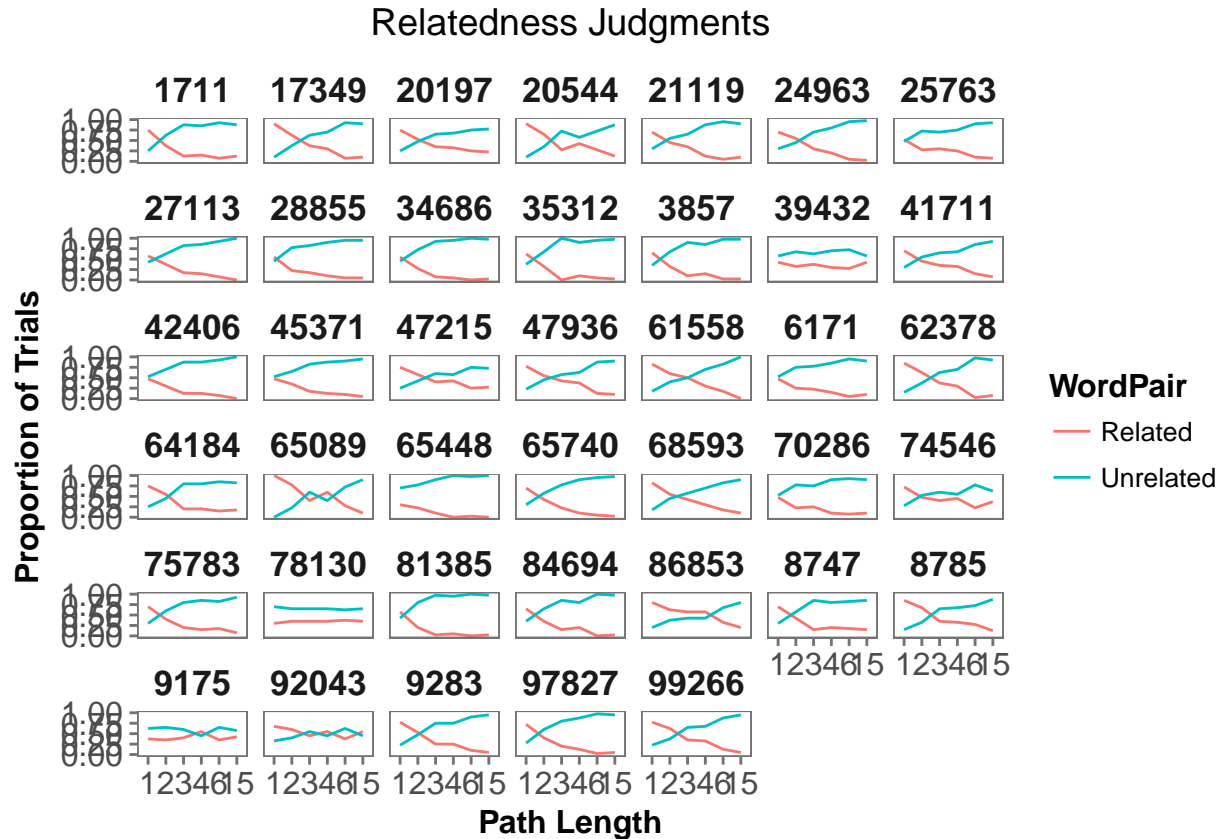
Subject-Wise

```
sem_decision$subject = as.factor(as.character(sem_decision$subject))

sem_decision = sem_decision[order(sem_decision$subject,
                                  sem_decision$pathlength,
                                  sem_decision$Type),]

sem_decision %>% mutate(WordPair = factor(Type,
                                           levels = unique(Type),
                                           labels = c("Related", "Unrelated")))%>%
  ggplot(aes(x = pathlengthfac, y = proportion,
             color = WordPair, group = WordPair))+
  geom_line()+
  # 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_solarized()+
  facet_wrap(~subject)+
  xlab("Path Length") + ylab("Proportion of Trials") +
  ggtitle("Relatedness Judgments") +
  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)))
```



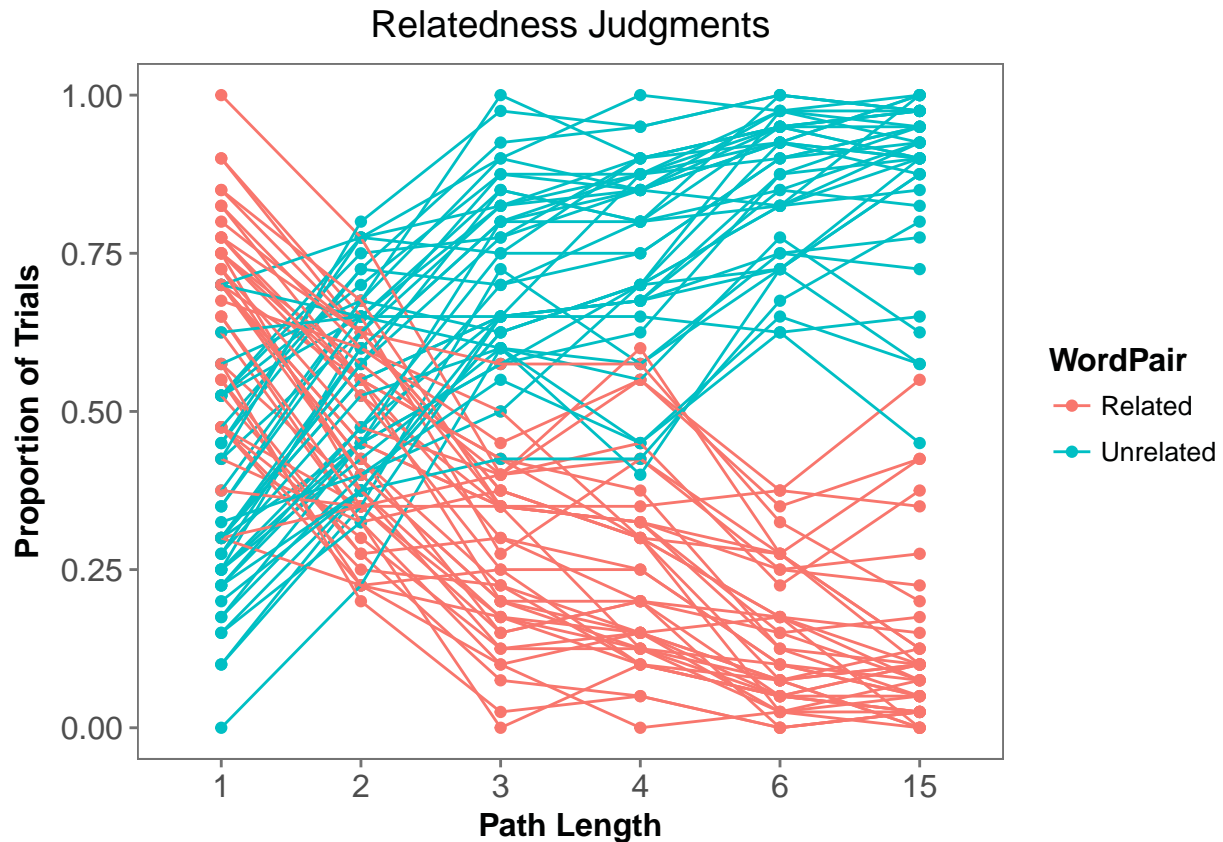
Line Plot Subject-Wise

```
sem_decision$subject = as.factor(as.character(sem_decision$subject))

sem_decision = sem_decision[order(sem_decision$subject,
                                  sem_decision$pathlength,
                                  sem_decision$Type),]

sem_decision %>% mutate(WordPair = factor(Type,
                                           levels = unique(Type),
                                           labels = c("Related", "Unrelated")))%>%
  ggplot(aes(x = pathlengthfac, y = proportion,
             color = WordPair, group=interaction(WordPair, subject))) +
  geom_point() +
  geom_line() +
  theme_few() +
  scale_fill_solarized() +
  # facet_wrap(~subject) +
  xlab("Path Length") + ylab("Proportion of Trials") +
```

```
ggtitle("Relatedness Judgments") +
  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)))
```



ANOVA

```
relunrel_aov = aov(data = sem_decision, proportion ~ pathlengthfac*Type +
  Error(subject/(pathlengthfac*Type)))
summary(relunrel_aov)
```

```
##
## Error: subject
##           Df    Sum Sq  Mean Sq F value Pr(>F)
## Residuals 39 3.634e-29 9.319e-31
##
## Error: subject:pathlengthfac
##           Df    Sum Sq  Mean Sq F value Pr(>F)
## pathlengthfac  5 1.700e-29 3.399e-30  1.523 0.184
## Residuals    195 4.353e-28 2.232e-30
##
## Error: subject:Type
##           Df Sum Sq Mean Sq F value  Pr(>F)
```

```
## Type      1 16.115  16.115   111.9 5.08e-13 ***
## Residuals 39  5.616   0.144
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Error: subject:pathlengthfac:Type
##              Df Sum Sq Mean Sq F value Pr(>F)
## pathlengthfac:Type  5 16.946    3.389    171 <2e-16 ***
## Residuals          195  3.866    0.020
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Raw Reaction Time

```
sem_rt = group_by(sem, subject, pathlength ) %>%
  summarise_at(vars(rt), mean)

sem_rt_agg = group_by(sem, pathlength ) %>%
  summarise_at(vars(rt), mean)

sem_agg_rmisc = Rmisc::summarySE(sem_rt,
  measurevar = "rt",
  groupvars = c("pathlength"))
```

ANOVA

```
sem_rt$pathlengthfac = ordered(as.factor(as.character(sem_rt$pathlength)),
  levels = c("1", "2", "3", "4", "6", "15"))
sem_rt$subject = as.factor(sem_rt$subject)
rt_aov = aov(data = sem_rt, rt ~ pathlengthfac +
  Error(subject/(pathlengthfac)))
summary(rt_aov)

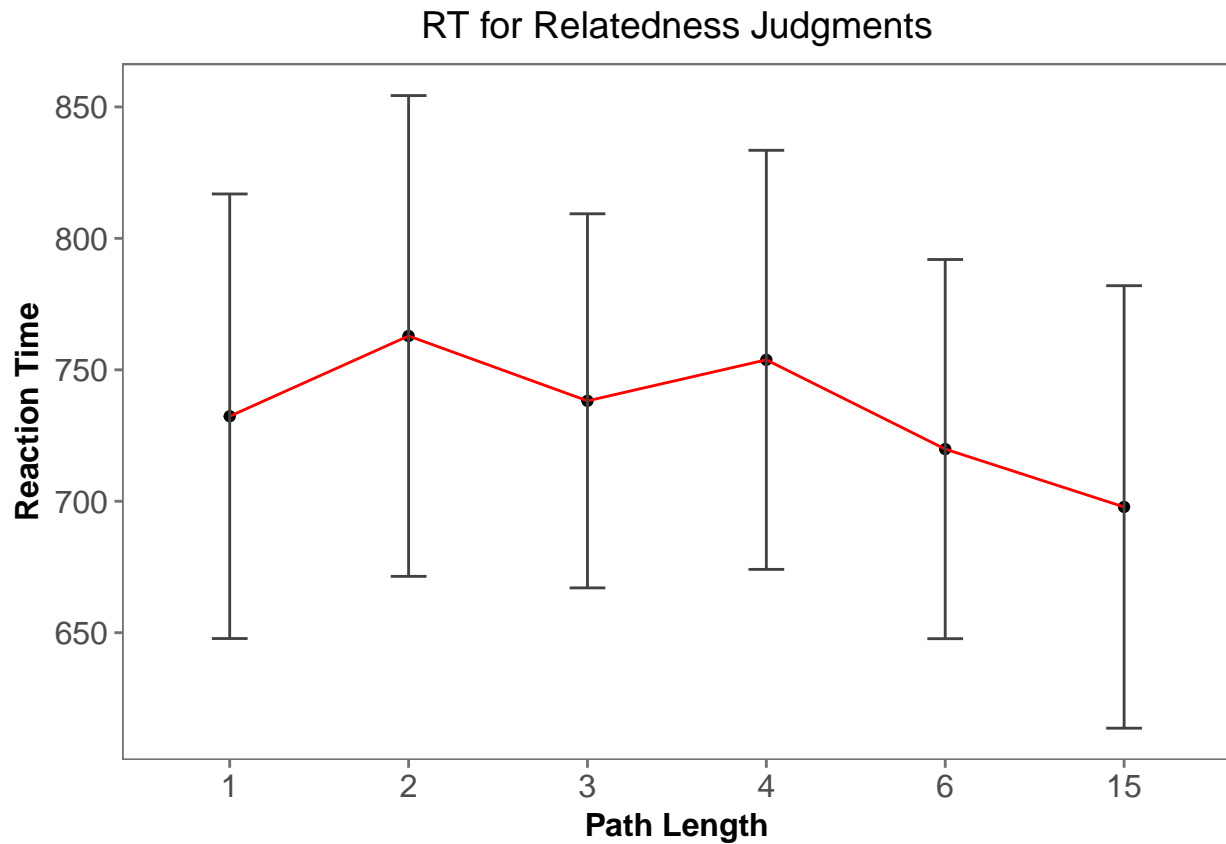
##
## Error: subject
##              Df    Sum Sq Mean Sq F value Pr(>F)
## Residuals 39 13728805   352021
##
## Error: subject:pathlengthfac
##              Df    Sum Sq Mean Sq F value    Pr(>F)
## pathlengthfac  5   110141   22028    3.52 0.00454 **
## Residuals     195 1220328    6258
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Plotting RTs

```
sem_agg_rmisc$pathlengthfac = ordered(as.factor(as.character(sem_agg_rmisc$pathlength)),
```

```
library(ggplot2)
library(ggthemes)

sem_agg_rmisc %>%
  ggplot(aes(x = pathlengthfac, y = rt, group = 1))+
  geom_point()+
  geom_line(color = "red")+
  geom_errorbar(aes(ymin=rt - ci, ymax=rt + ci),
    width=.2, color = "gray26",
    position = position_dodge(0.7))+
  theme_few()+
  # scale_x_continuous(breaks = c(1,2,3,4,5,6,10,15,20))+
  xlab("Path Length") + ylab("Reaction Time") +
  ggtitle("RT for Relatedness Judgments") +
  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)))
```

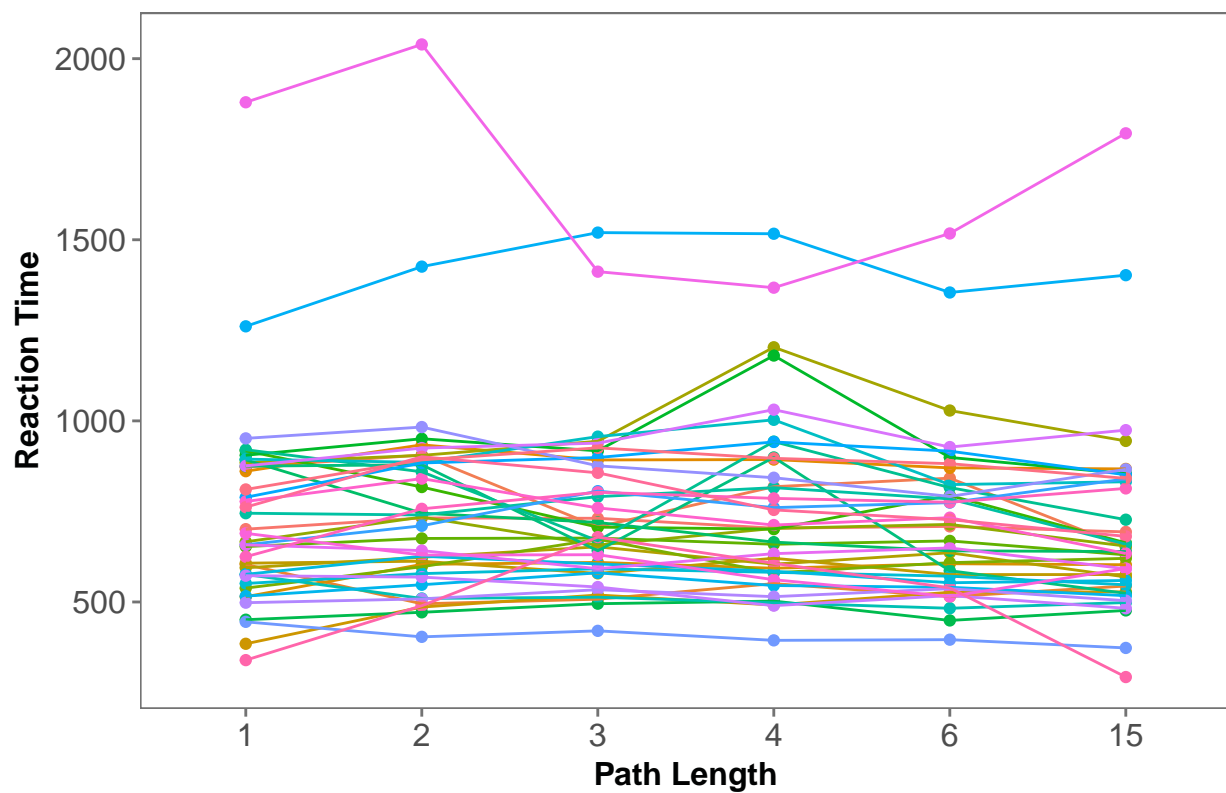


Subject-Wise

```
library(ggplot2)
library(ggthemes)
```

```
sem_rt %>%
  ggplot(aes(x = pathlengthfac, y = rt,
             group = subject, color = subject))+
  geom_point()+
  geom_line()+
  #geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
  #              width=.2, color = "gray26",
  #              position = position_dodge(0.7))+
  theme_few()+
  guides(color = FALSE)+
  # scale_x_continuous(breaks = c(1,2,3,4,6,15))+
  xlab("Path Length") + ylab("Reaction Time") +
  ggtitle("RT for Relatedness Judgments") +
  theme(axis.text = element_text(size = rel(1)),
        axis.title = element_text(face = "bold", size = rel(1)),
        legend.title = element_blank(),
        plot.title = element_text(hjust = .5),
        strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

RT for Relatedness Judgments

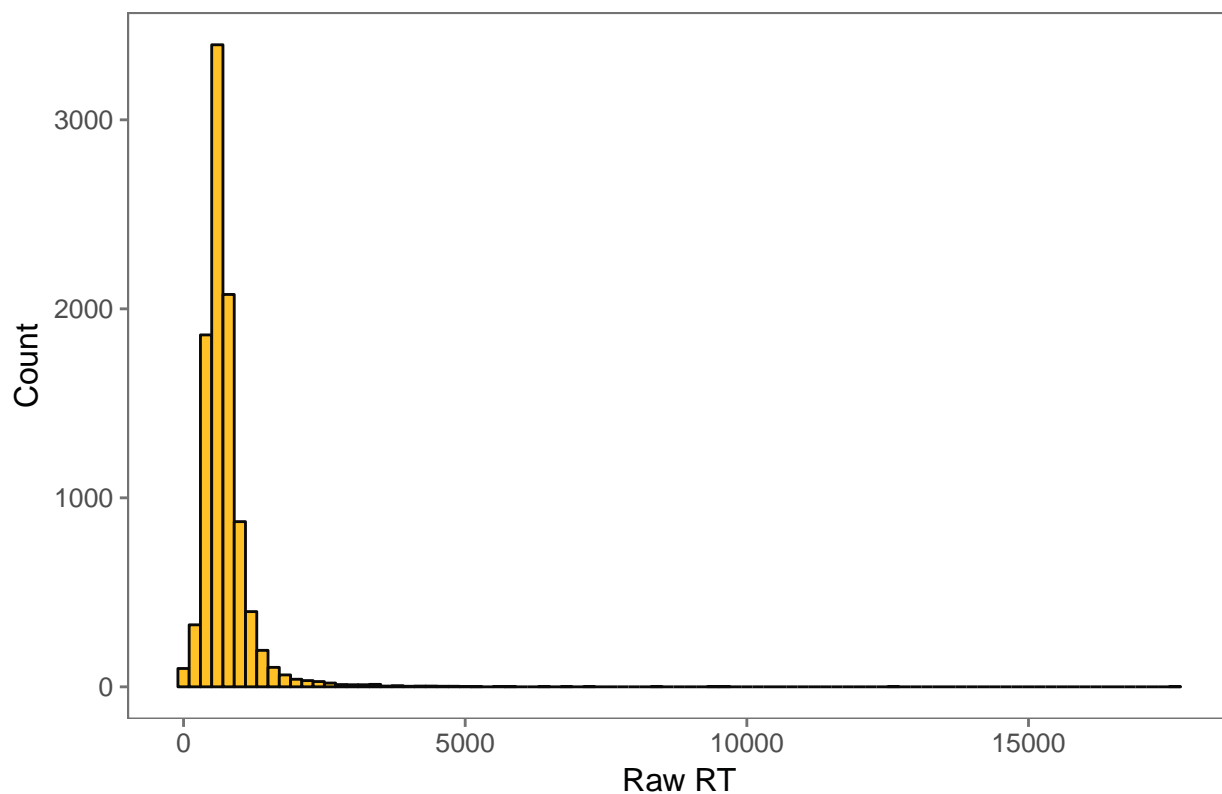


z-scored Reaction Time

Histogram of RT

```
library(ggplot2)
library(ggthemes)
ggplot(sem, aes(x = rt))+
  geom_histogram(binwidth = 200, color = "gray4", fill = "goldenrod1")+
  theme_few()+
  #facet_wrap(~subject)+
  xlab("Raw RT") + ylab("Count") +
  ggtitle("Raw RT Histogram for All Trials")
```

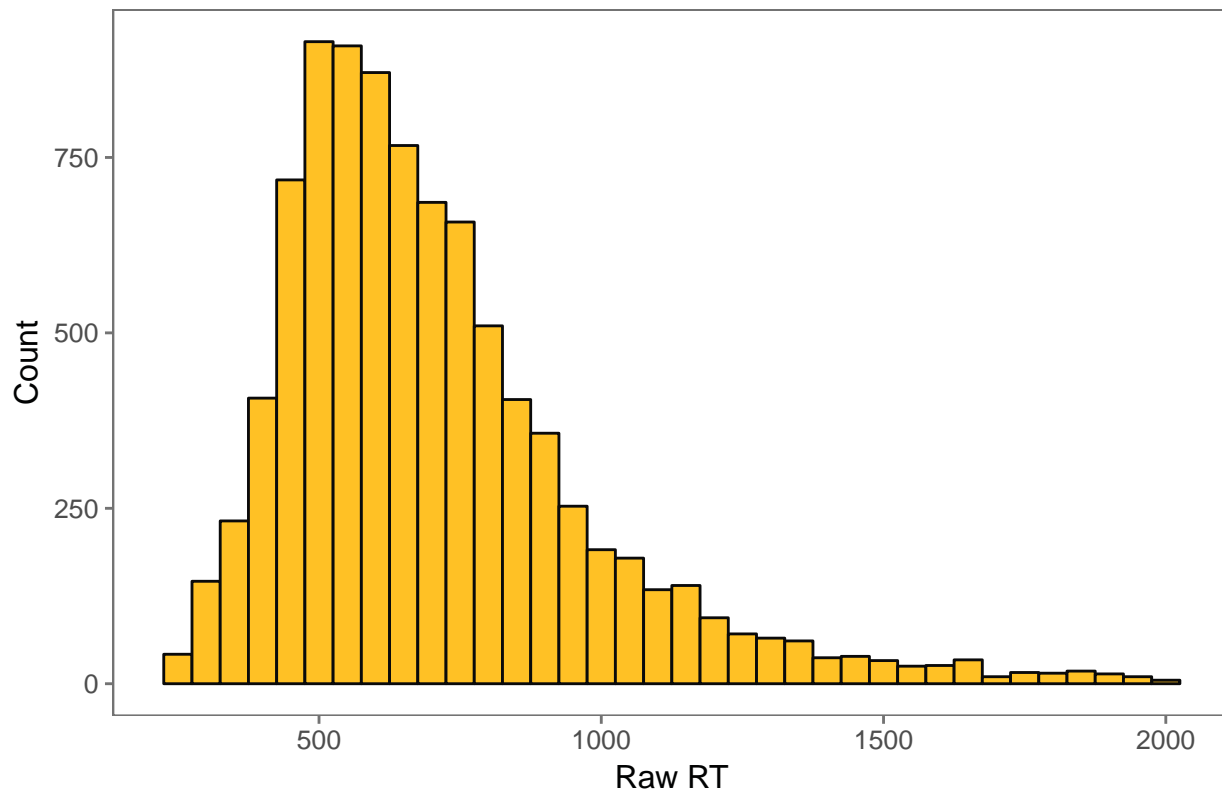
Raw RT Histogram for All Trials



First Trim

```
library(dplyr)
sem_firsttrim = sem %>% filter( rt > 250 & rt < 2000)
ggplot(sem_firsttrim, aes(x = rt))+
  geom_histogram(binwidth = 50, color = "gray4", fill = "goldenrod1")+
  theme_few()+
  #facet_wrap(~subject)+
  xlab("Raw RT") + ylab("Count") +
  ggtitle("Raw RT Histogram for Trials Above 250 ms and below 2000 ms")
```

Raw RT Histogram for Trials Above 250 ms and below 2000 ms



Raw RT aggregates After Trimming

```
sem_rt_firsttrim = group_by(sem_firsttrim, subject, pathlength ) %>%
  summarise_at(vars(rt), mean)

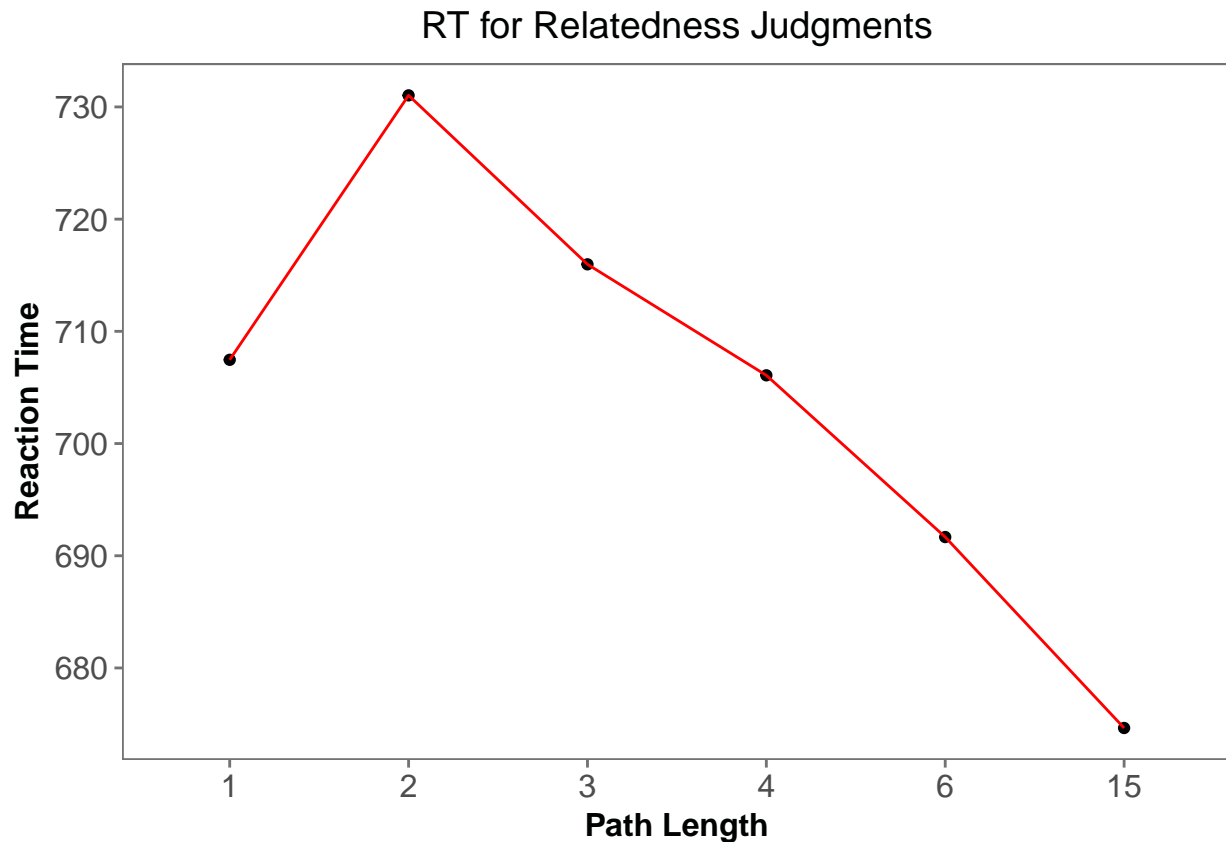
sem_rt_agg_firsttrim = group_by(sem_firsttrim, pathlength ) %>%
  summarise_at(vars(rt), mean)

sem_rt_agg_firsttrim$pathlengthfac = ordered(as.factor(as.character(sem_rt_agg_firsttrim$pathlength))),

library(ggplot2)
library(ggthemes)

sem_rt_agg_firsttrim %>%
  ggplot(aes(x = pathlengthfac, y = rt, group = 1))+
  geom_point()+
  geom_line(color = "red")+
  #geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
  #              width=.2, color = "gray26",
  #              position = position_dodge(0.7))+
  theme_few()+
  #scale_x_continuous(breaks = c(1,2,3,4,6,15))+
  xlab("Path Length") + ylab("Reaction Time") +
  ggtitle("RT for Relatedness Judgments") +
  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)))
```



Subject Raw RT again

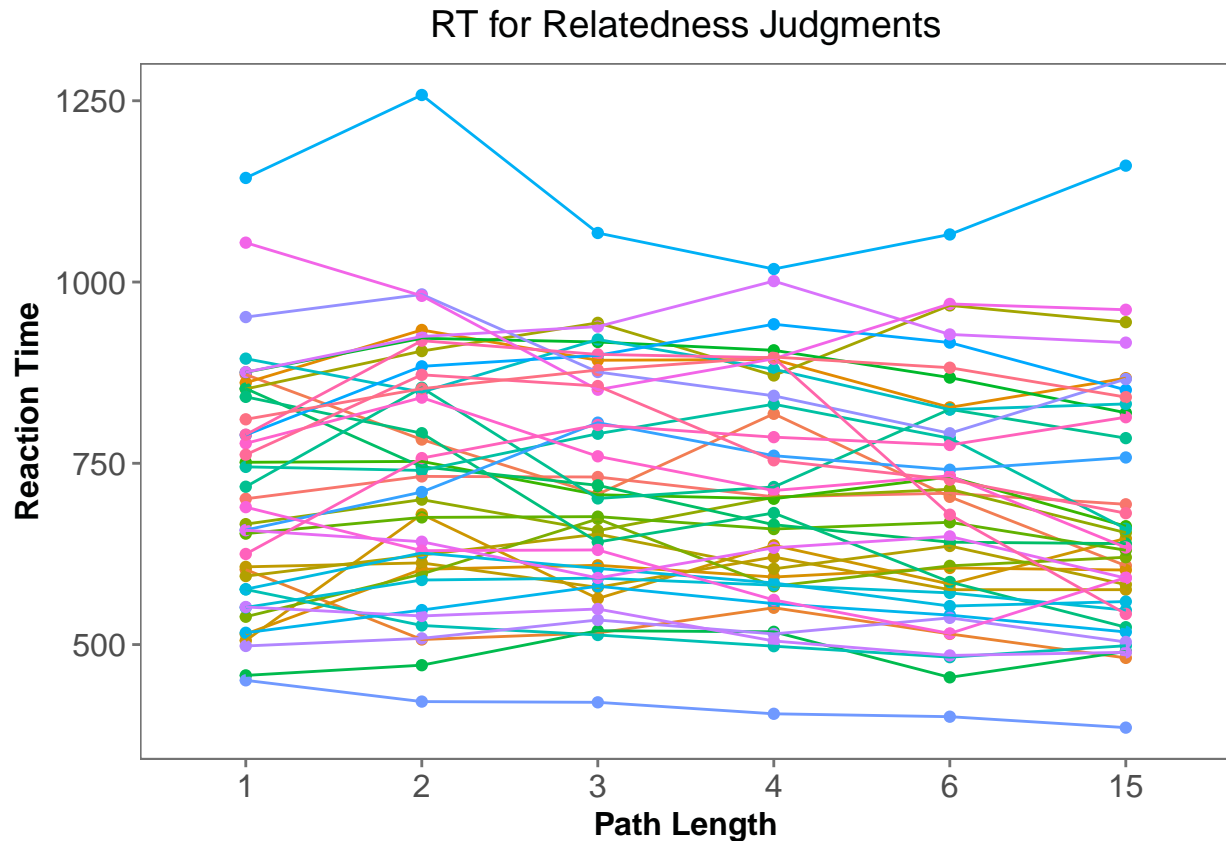
```
library(ggplot2)
library(ggthemes)

sem_rt_firsttrim$pathlengthfac = ordered(as.factor(as.character(sem_rt_firsttrim$pathlength)),
                                          levels = c("1", "2", "3", "4", "6", "15"))
sem_rt_firsttrim$subject = as.factor(sem_rt_firsttrim$subject)
sem_rt_firsttrim %>%
  ggplot(aes(x = pathlengthfac, y = rt,
             group = subject, color = subject))+
  geom_point()+
  geom_line()+
  #geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
  #              width=.2, color = "gray26",
  #              position = position_dodge(0.7))+
  theme_few()+
  guides(color = FALSE)+
  # scale_x_continuous(breaks = c(1,2,3,4,5,6,10,15,20))+
  xlab("Path Length") + ylab("Reaction Time") +
```

```

ggtitle("RT for Relatedness Judgments") +
# facet_wrap(~subject)+
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)))

```



Making the z-scores

```

## aggregate per subject all IVs and DVs
meanRT = group_by(sem_firsttrim, subject) %>%
  summarise_at(vars(rt), mean)
colnames(meanRT) = c("subject", "MeanRT")

sdRT = group_by(sem_firsttrim, subject) %>%
  summarise_at(vars(rt), sd)
colnames(sdRT) = c("subject", "sdRT")

RT_agg = merge(meanRT, sdRT, by = "subject")

## merge aggregate info with long data
sem_z = merge(sem_firsttrim, RT_agg, by = "subject", all.x = T)

## person and grand-mean centered scores using original and aggregate

```

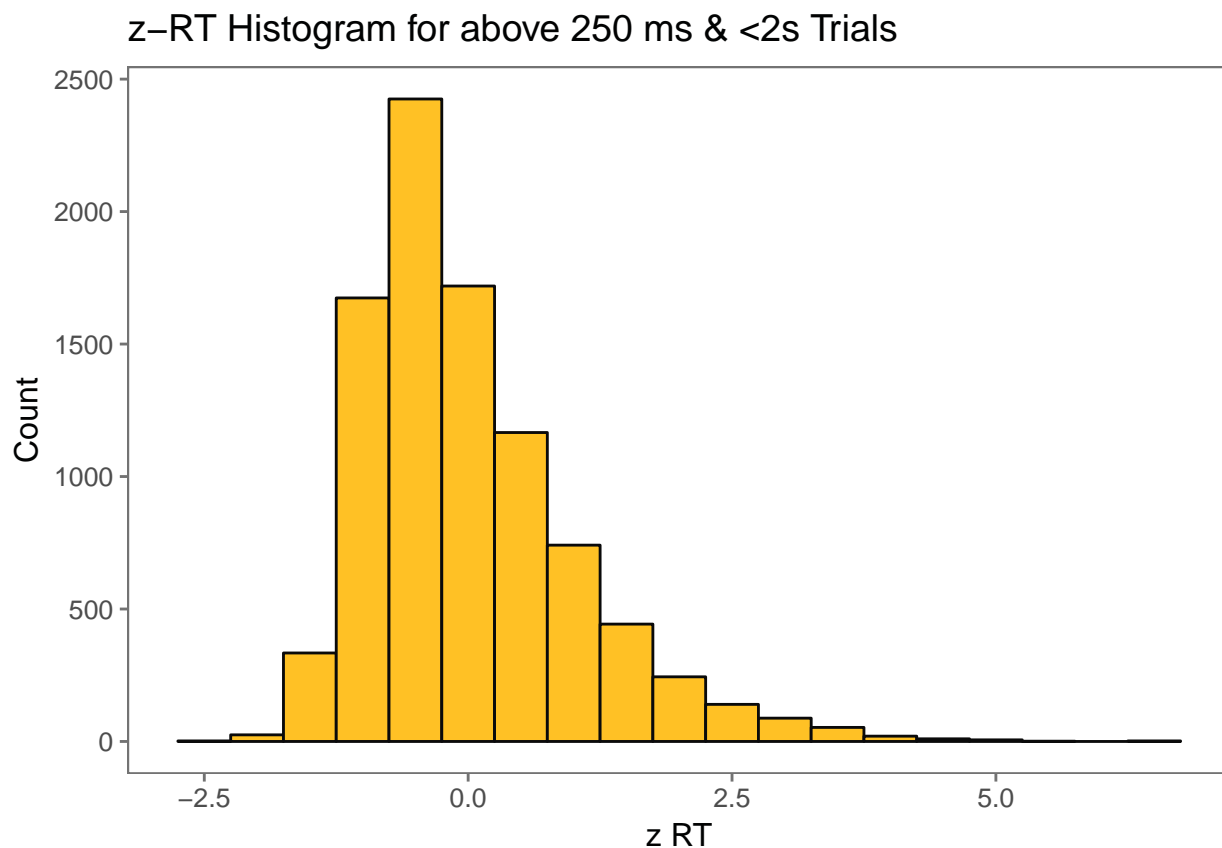
```
library(dplyr)
sem_z = sem_z %>% mutate(zRT = (rt - MeanRT)/sdRT)

## checking: subject level means should be zero

sub_pic = group_by(sem_z, subject) %>%
  summarise_at(vars(zRT), mean)
```

z-RT Distribution

```
ggplot(sem_z, aes(x = zRT)) +
  geom_histogram(binwidth = 0.5, color = "gray4", fill = "goldenrod1") +
  theme_few() +
  xlab("z RT") + ylab("Count") +
  ggtitle("z-RT Histogram for above 250 ms & <2s Trials")
```

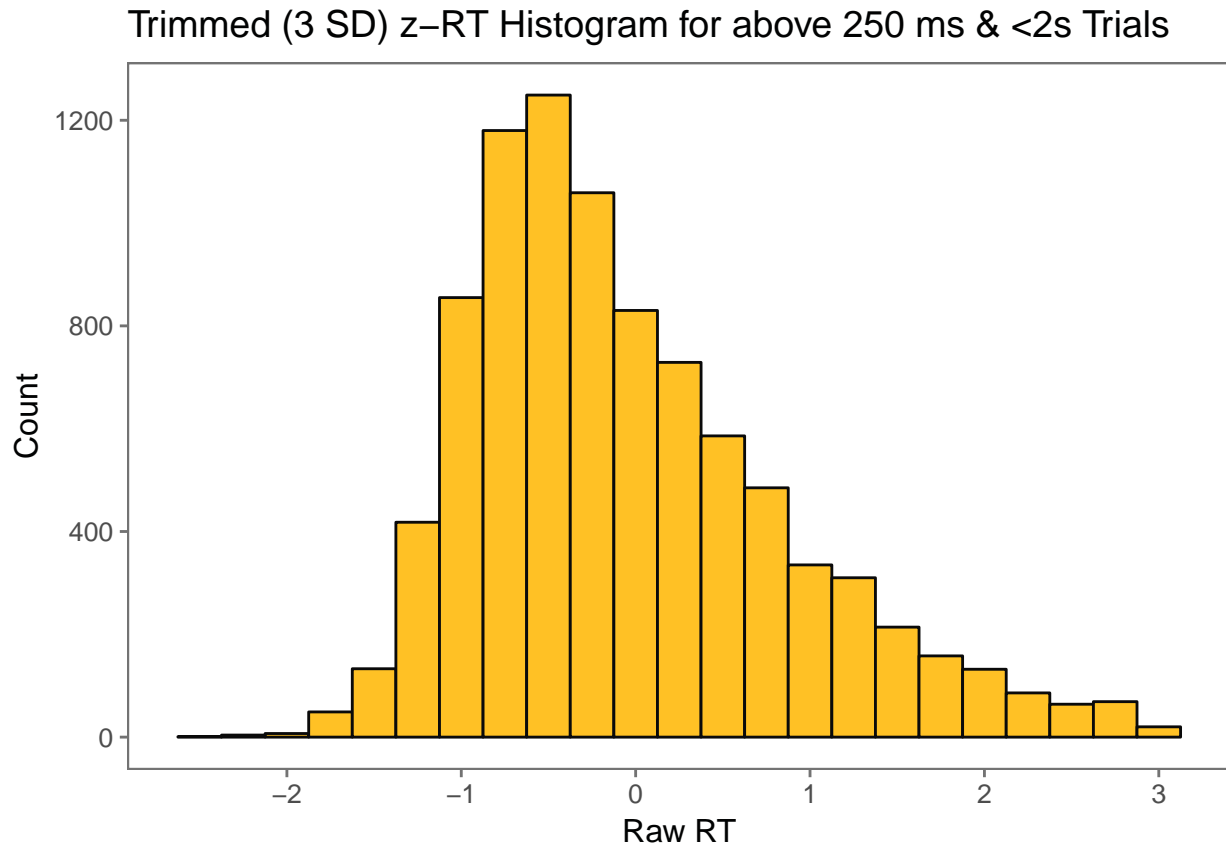


Trimming z-RT

```
sem_z_trimmed = subset(sem_z, sem_z$zRT < 3 & sem_z$zRT > -3)

ggplot(sem_z_trimmed, aes(x = zRT)) +
  geom_histogram(binwidth = 0.25, color = "gray4", fill = "goldenrod1") +
  theme_few() +
```

```
xlab("Raw RT") + ylab("Count") +
ggtitle("Trimmed (3 SD) z-RT Histogram for above 250 ms & <2s Trials")
```



Repeating z-scoring

```
library(dplyr)

## aggregate per subject all IVs and DVs
meanRT_trim = group_by(sem_z_trimmed, subject) %>%
  summarise_at(vars(rt), mean)
colnames(meanRT_trim) = c("subject", "MeanRT_trim")

sdRT_trim = group_by(sem_z_trimmed, subject) %>%
  summarise_at(vars(rt), sd)
colnames(sdRT_trim) = c("subject", "sdRT_trim")

RT_agg_trim = merge(meanRT_trim, sdRT_trim, by = "subject")

## merge aggregate info with long data
new_sem_z = merge(sem_z_trimmed, RT_agg_trim, by = "subject", all.x = T)

## person and grand-mean centered scores using original and aggregate
library(dplyr)
new_sem_z = new_sem_z %>% mutate(zRT_trim = (rt - MeanRT_trim)/sdRT_trim)
```

```
## checking: subject level means should be zero
```

```
sub_pic = group_by(new_sem_z, subject) %>%
  summarise_at(vars(zRT_trim), mean)
```

```
#write.csv(new_pic_z, file="final_pic_z.csv")
```

Aggregating zRT

```
z_sem_rt = group_by(new_sem_z, subject, pathlength ) %>%
  summarise_at(vars(zRT_trim), mean)
```

```
z_sem_rt_rel = group_by(new_sem_z, subject, pathlength, Type ) %>%
  summarise_at(vars(zRT_trim), mean)
```

```
z_sem_rt_agg = group_by(new_sem_z, pathlength ) %>%
  summarise_at(vars(zRT_trim), mean)
```

```
z_rmisc = Rmisc::summarySE(new_sem_z,
  measurevar = "zRT_trim",
  groupvars = c("pathlength"))
```

```
z_sem_rt_rel_agg = group_by(new_sem_z, pathlength, Type ) %>%
  summarise_at(vars(zRT_trim), mean)
```

```
z_sem_rt_rel_agg_rmisc = Rmisc::summarySE(new_sem_z,
  measurevar = "zRT_trim",
  groupvars = c("pathlength", "Type"))
```

ANOVA

```
z_sem_rt$pathlengthfac = ordered(as.factor(as.character(z_sem_rt$pathlength)),
  levels = c("1", "2", "3", "4", "6", "15"))
```

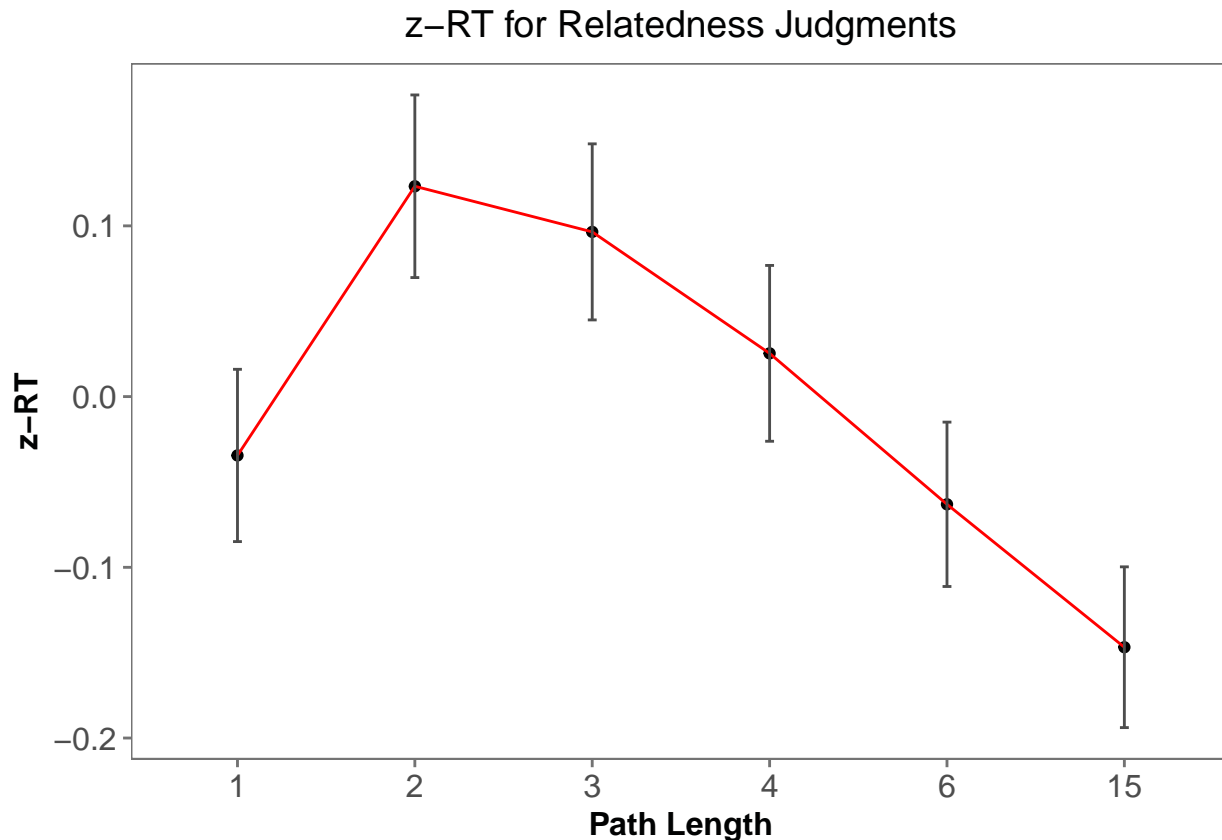
```
z_sem_rt$subject = as.factor(z_sem_rt$subject)
```

```
z_rt_aov = aov(data = z_sem_rt, zRT_trim ~ pathlengthfac +
  Error(subject/(pathlengthfac)))
summary(z_rt_aov)
```

```
##
## Error: subject
##           Df  Sum Sq   Mean Sq F value Pr(>F)
## Residuals 39 0.005412 0.0001388
##
## Error: subject:pathlengthfac
##           Df Sum Sq Mean Sq F value   Pr(>F)
## pathlengthfac  5  2.143  0.4287   9.764 2.43e-08 ***
## Residuals    195  8.561  0.0439
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Plotting RTs: collapsed

```
z_rmisc$pathlengthfac = ordered(as.factor(as.character(z_rmisc$pathlength))),  
z_rmisc$zRT_trim = as.numeric(z_rmisc$zRT_trim)  
  
library(ggplot2)  
library(ggthemes)  
  
z_rmisc %>%  
  ggplot(aes(x = pathlengthfac, y = zRT_trim, group = 1))+  
  geom_point()+  
  # geom_smooth(method = "loess")+  
  geom_line(color = "red")+  
  geom_errorbar(aes(ymin=zRT_trim - ci, ymax=zRT_trim + ci),  
    width=.05, color = "gray30",  
    position = position_dodge(0.7))+  
  theme_few()+  
  #scale_x_continuous(breaks = c(1,2,3,4,5,6,10,15,20))+  
  xlab("Path Length") + ylab("z-RT") +  
  ggtitle("z-RT for Relatedness Judgments") +  
  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)))
```



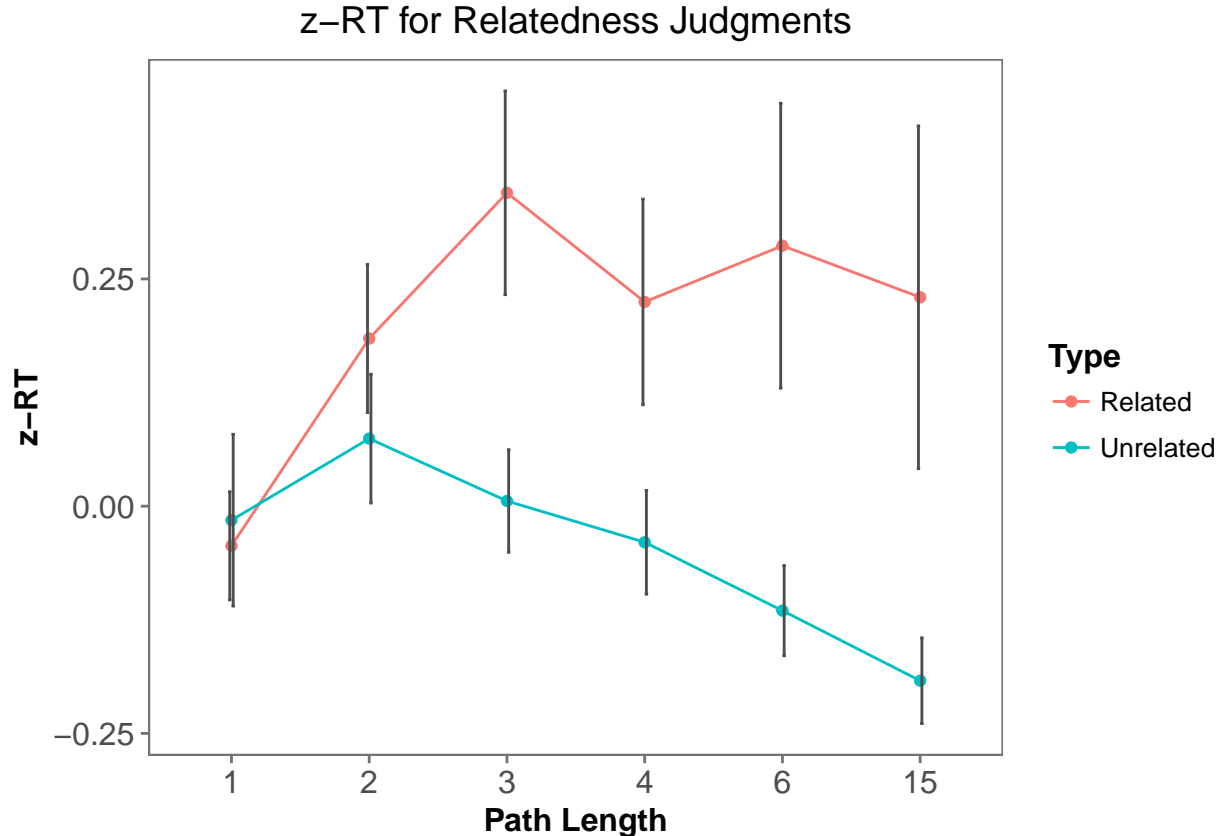
Plotting RTs: Rel-Unrel

```
z_sem_rt_rel_agg_rmisc$pathlengthfac = ordered(as.factor(as.character(z_sem_rt_rel_agg_rmisc$pathlengthh

z_sem_rt_rel_agg_rmisc$zRT_trim = as.numeric(z_sem_rt_rel_agg_rmisc$zRT_trim)

library(ggplot2)
library(ggthemes)

z_sem_rt_rel_agg_rmisc %>%
  ggplot(aes(x = pathlengthfac, y = zRT_trim, group = Type, color = Type))+
  geom_point()+
  # geom_smooth(method = "loess")+
  geom_line()+
  geom_errorbar(aes(ymin=zRT_trim - ci, ymax=zRT_trim + ci),
    width=.05, color = "gray30",
    position = position_dodge(0.05))+
  theme_few()+
  #scale_x_continuous(breaks = c(1,2,3,4,5,6,10,15,20))+
  xlab("Path Length") + ylab("z-RT") +
  ggtitle("z-RT for Relatedness Judgments") +
  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)))
```

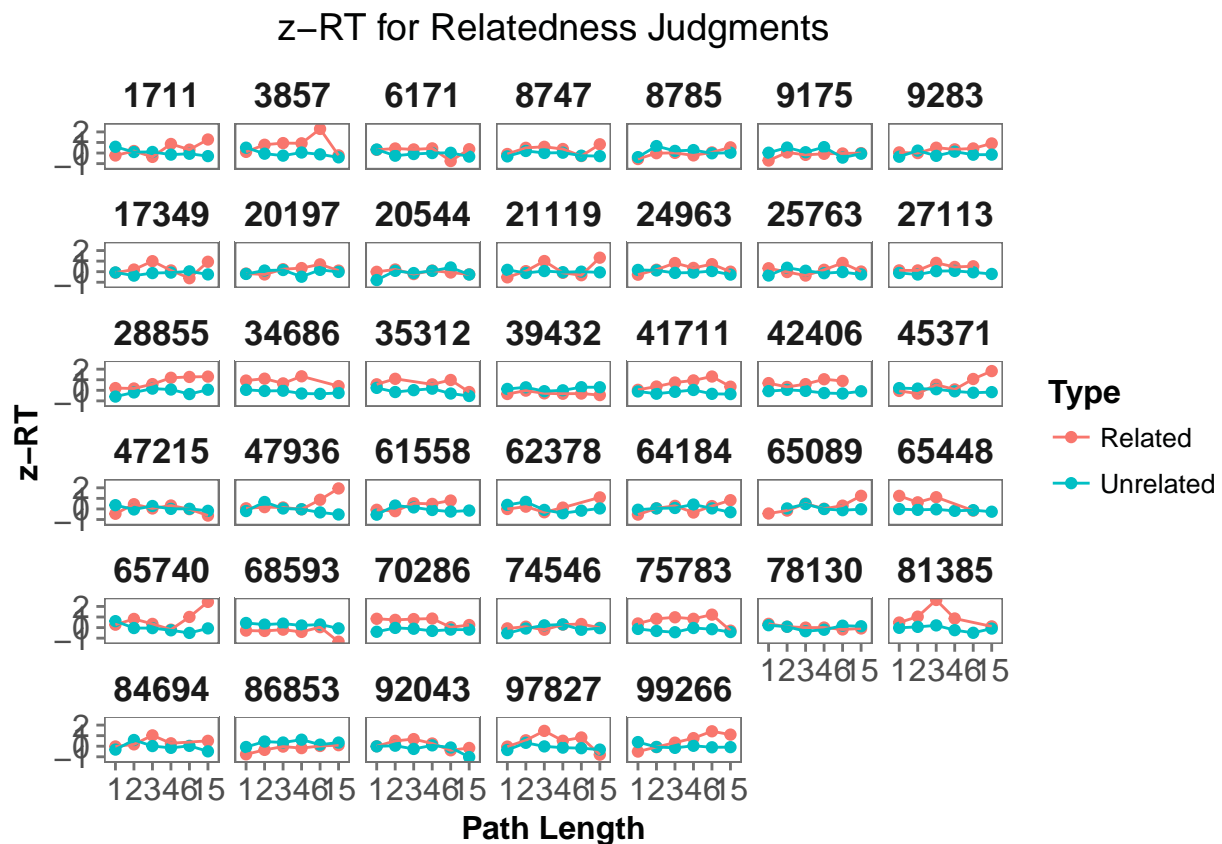


Plotting RTs: Subject Wise Rel-Unrel

```
z_sem_rt_rel$pathlengthfac = ordered(as.factor(as.character(z_sem_rt_rel$pathlength))),
z_sem_rt_rel$zRT_trim = as.numeric(z_sem_rt_rel$zRT_trim)

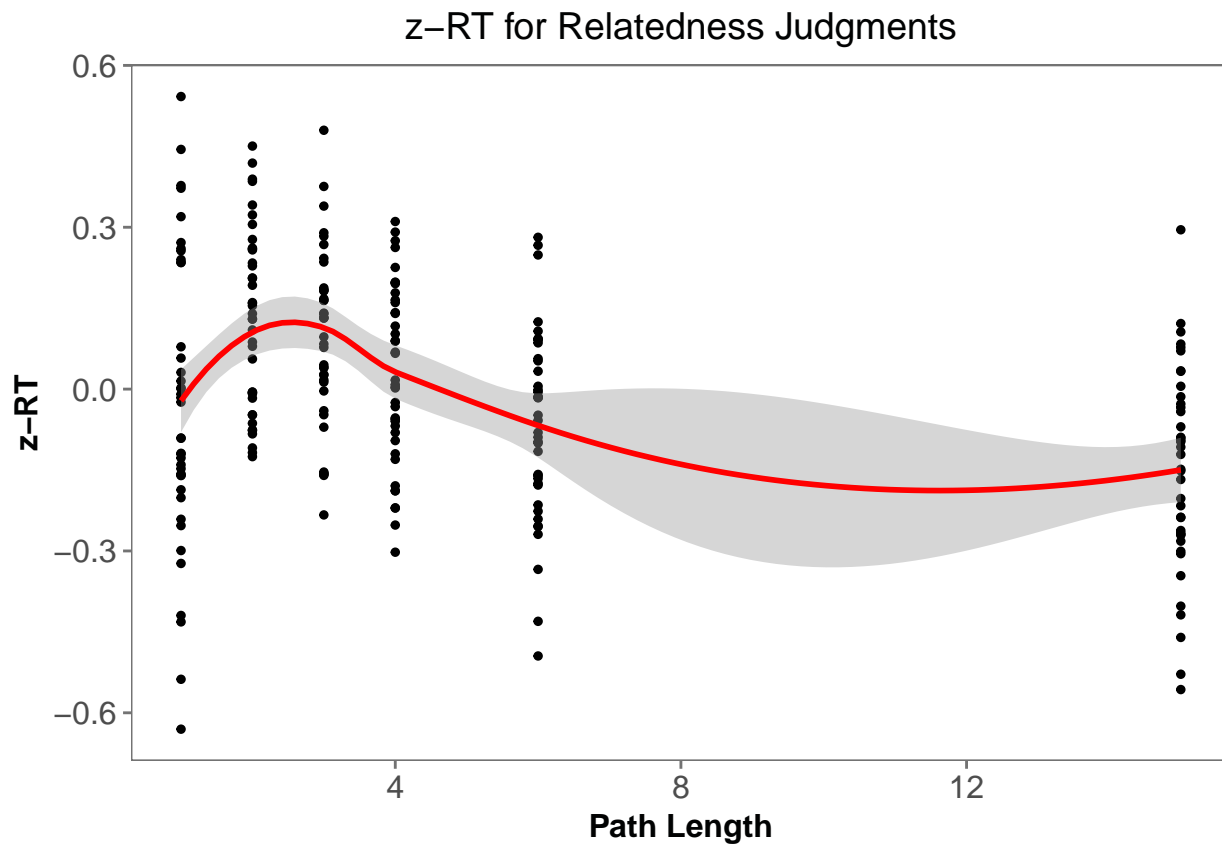
library(ggplot2)
library(ggthemes)

z_sem_rt_rel %>%
  ggplot(aes(x = pathlengthfac, y = zRT_trim, color = Type, group=interaction(Type, subject))) +
  geom_point()+
  # geom_smooth(method = "loess")+
  geom_line()+
  theme_few()+
  facet_wrap(~subject)+
  #scale_x_continuous(breaks = c(1,2,3,4,5,6,10,15,20))+
  xlab("Path Length") + ylab("z-RT") +
  ggtitle("z-RT for Relatedness Judgments") +
  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)))
```



Quadratic Trend

```
z_sem_rt %>%
  ggplot(aes(x = pathlength, y = zRT_trim))+
  geom_point(color = "black", size = 1)+
  geom_smooth(method = "loess", color = "red")+
  #geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
  #              width=.2, color = "gray26",
  #              position = position_dodge(0.7))+
  theme_few()+
  # scale_x_continuous(breaks = c(1,2,3,4,5,6,10,15,20))+
  xlab("Path Length") + ylab("z-RT") +
  ggtitle("z-RT for Relatedness Judgments") +
  # facet_wrap(~subject)+
  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)))
```



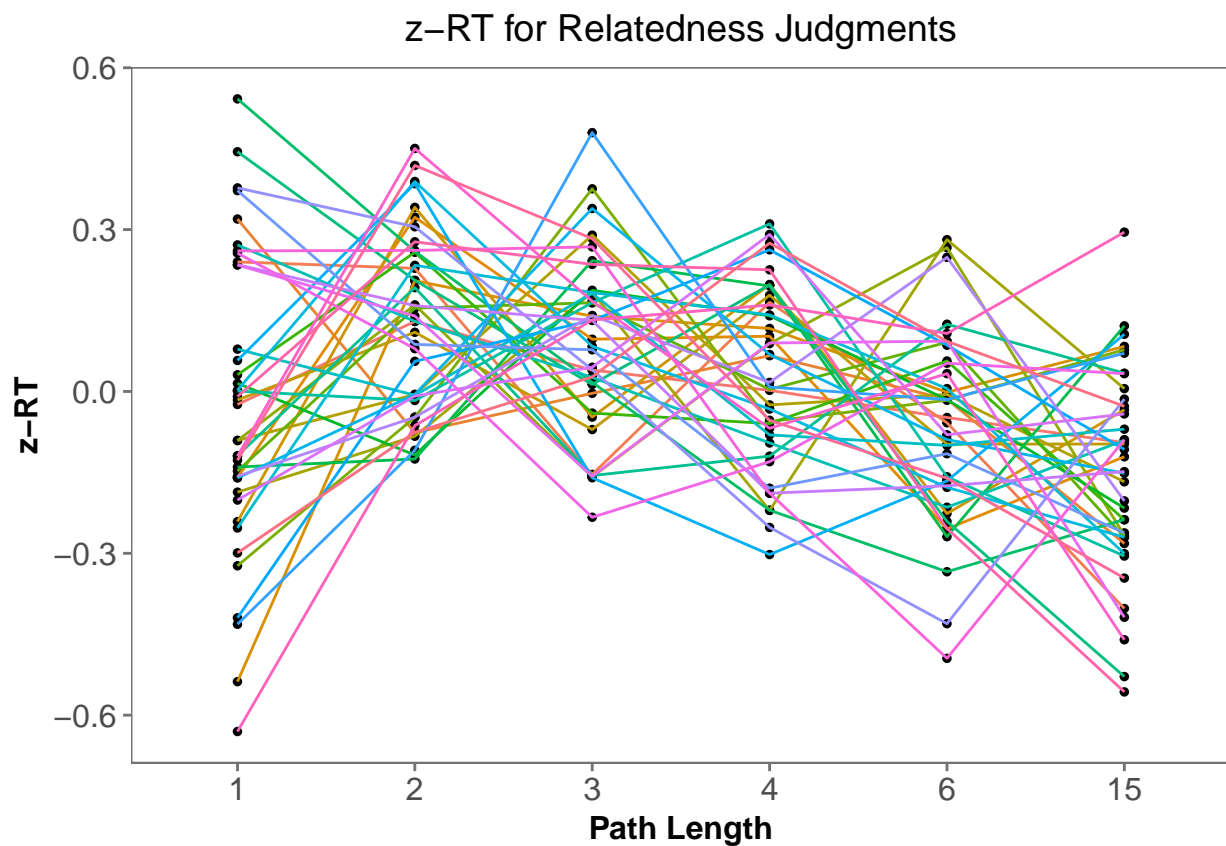
Subject-Wise

```
library(ggplot2)
library(ggthemes)
```

```

z_sem_rt %>%
  ggplot(aes(x = pathlengthfac, y = zRT_trim,
             color = subject, group = subject))+
  geom_point(color = "black", size = 1)+
  geom_line()+
  #geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
  #              width=.2, color = "gray26",
  #              position = position_dodge(0.7))+
  theme_few()+
  guides(color = FALSE)+
  # scale_x_continuous(breaks = c(1,2,3,4,5,6,10,15,20))+
  xlab("Path Length") + ylab("z-RT") +
  ggtitle("z-RT for Relatedness Judgments") +
  # facet_wrap(~subject)+
  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)))

```



Regressions

```
library(lme4)

## Loading required package: Matrix
sem$pathlengthfac = ordered(as.factor(as.character(sem$pathlength)),
                             levels = c("1", "2", "3", "4", "6", "15"))

new_sem_z$pathlengthfac = ordered(as.factor(as.character(new_sem_z$pathlength)),
                                   levels = c("1", "2", "3", "4", "6", "15"))

m0 = lmer(data = new_sem_z, rt ~ 1 + (1|subject) + (1|trial_index) +
          (1|prime_word) + (1|target_word))
summary(m0)

## Linear mixed model fit by REML ['lmerMod']
## Formula: rt ~ 1 + (1 | subject) + (1 | trial_index) + (1 | prime_word) +
##          (1 | target_word)
##      Data: new_sem_z
##
## REML criterion at convergence: 121468.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.9975 -0.5992 -0.1583  0.4368  5.8497
##
## Random effects:
##   Groups      Name      Variance Std.Dev.
## target_word (Intercept)   587.4   24.24
## prime_word  (Intercept)   914.1   30.23
## trial_index (Intercept)   124.9   11.18
## subject     (Intercept) 23886.8  154.55
## Residual                41893.2  204.68
## Number of obs: 8973, groups:
## target_word, 1918; prime_word, 1918; trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)   697.88     24.57    28.41

reghelper::ICC(m0)

## [1] 0.3784984

contrasts(sem$pathlengthfac) = contr.treatment(6, base = 1)

contrasts(new_sem_z$pathlengthfac) = contr.treatment(6, base = 2)

m1_fixed = lmer(data = new_sem_z, zRT_trim ~ pathlengthfac +
                (1|subject))
summary(m1_fixed)

## Linear mixed model fit by REML ['lmerMod']
## Formula: zRT_trim ~ pathlengthfac + (1 | subject)
```

```

## Data: new_sem_z
##
## REML criterion at convergence: 25378.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.5287 -0.7314 -0.2061  0.5651  3.9283
##
## Random effects:
##   Groups   Name                Variance Std.Dev.
## subject (Intercept) 3.626e-32 1.904e-16
## Residual          9.875e-01 9.938e-01
## Number of obs: 8973, groups: subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)    0.12320    0.02579   4.777
## pathlengthfac1 -0.15767    0.03645  -4.326
## pathlengthfac3 -0.02672    0.03632  -0.736
## pathlengthfac4 -0.09787    0.03643  -2.687
## pathlengthfac5 -0.18629    0.03641  -5.117
## pathlengthfac6 -0.27000    0.03636  -7.426
##
## Correlation of Fixed Effects:
##              (Intr) pthln1 pthln3 pthln4 pthln5
## pthlngthfc1 -0.708
## pthlngthfc3 -0.710  0.502
## pthlngthfc4 -0.708  0.501  0.503
## pthlngthfc5 -0.708  0.501  0.503  0.501
## pthlngthfc6 -0.709  0.502  0.504  0.502  0.502

```

```

m1_random = lmer(data = sem, rt ~ pathlengthfac +
                 (pathlengthfac|subject))
summary(m1_random)

```

```

## Linear mixed model fit by REML ['lmerMod']
## Formula: rt ~ pathlengthfac + (pathlengthfac | subject)
## Data: sem
##
## REML criterion at convergence: 144701.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.951 -0.351 -0.115  0.184 34.778
##
## Random effects:
##   Groups   Name                Variance Std.Dev. Corr
## subject (Intercept)    66078    257.06
##          pathlengthfac2    1530     39.11   0.56
##          pathlengthfac3   12139    110.18  -0.57  0.34
##          pathlengthfac4   13916    117.96  -0.37  0.41  0.92
##          pathlengthfac5    7966     89.25  -0.56  0.36  1.00  0.92
##          pathlengthfac6    5580     74.70  -0.17  0.71  0.84  0.70  0.84
## Residual          202173    449.64
## Number of obs: 9600, groups: subject, 40

```

```
##
## Fixed effects:
##           Estimate Std. Error t value
## (Intercept)    732.331    42.170  17.366
## pathlengthfac2   30.535    17.058   1.790
## pathlengthfac3    5.857    23.584   0.248
## pathlengthfac4   21.454    24.507   0.875
## pathlengthfac5  -12.491    21.257  -0.588
## pathlengthfac6  -34.499    19.805  -1.742
##
## Correlation of Fixed Effects:
##           (Intr) pthln2 pthln3 pthln4 pthln5
## pthlngthfc2  0.019
## pthlngthfc3 -0.531  0.406
## pthlngthfc4 -0.394  0.416  0.736
## pthlngthfc5 -0.496  0.435  0.742  0.709
## pthlngthfc6 -0.249  0.528  0.640  0.580  0.635
```

ELP Model

```
## Adding ELP covariates

elp_model = lmer(data = new_sem_z, rt ~ mean_len + mean_logf +
                 mean_ldtz +
                 (1|subject) + (1|trial_index))
fit_from_elp = broom::augment(elp_model, new_sem_z)

contrasts(fit_from_elp$pathlengthfac) = contr.treatment(6, base = 2)

m1_fixed_elp = lmer(data = fit_from_elp, .resid ~ pathlengthfac +
                   (1|subject) + (1|trial_index))
summary(m1_fixed_elp)

## Linear mixed model fit by REML ['lmerMod']
## Formula: .resid ~ pathlengthfac + (1 | subject) + (1 | trial_index)
## Data: fit_from_elp
##
## REML criterion at convergence: 121040.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.0936 -0.6140 -0.1617  0.4396  6.1244
##
## Random effects:
## Groups      Name                Variance Std.Dev.
## trial_index (Intercept) 4.923e-13 7.017e-07
## subject      (Intercept) 0.000e+00 0.000e+00
## Residual                    4.269e+04 2.066e+02
## Number of obs: 8969, groups: trial_index, 240; subject, 40
##
## Fixed effects:
##           Estimate Std. Error t value
## (Intercept)    24.926     5.362   4.649
```

```

## pathlengthfac1 -29.744      7.577 -3.925
## pathlengthfac3 -10.853      7.552 -1.437
## pathlengthfac4 -20.423      7.579 -2.695
## pathlengthfac5 -36.713      7.570 -4.850
## pathlengthfac6 -51.646      7.560 -6.832
##
## Correlation of Fixed Effects:
##          (Intr) pthln1 pthln3 pthln4 pthln5
## pthlngthfc1 -0.708
## pthlngthfc3 -0.710  0.502
## pthlngthfc4 -0.707  0.501  0.502
## pthlngthfc5 -0.708  0.501  0.503  0.501
## pthlngthfc6 -0.709  0.502  0.504  0.502  0.502

m1_all_elp = lmer(data = new_sem_z, rt ~ pathlengthfac +
                  mean_len + mean_logf + mean_ldtz +
                  (1|subject) + (1|trial_index))
summary(m1_all_elp)

## Linear mixed model fit by REML ['lmerMod']
## Formula: rt ~ pathlengthfac + mean_len + mean_logf + mean_ldtz + (1 |
##          subject) + (1 | trial_index)
## Data: new_sem_z
##
## REML criterion at convergence: 121308.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.0737 -0.6111 -0.1618  0.4401  6.1029
##
## Random effects:
## Groups      Name                Variance Std.Dev.
## trial_index (Intercept)    173.6    13.18
## subject      (Intercept) 23875.6   154.52
## Residual                    42989.3   207.34
## Number of obs: 8969, groups: trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)   691.58124    31.41168   22.017
## pathlengthfac1 -29.98213     7.62531   -3.932
## pathlengthfac3 -10.91566     7.60450   -1.435
## pathlengthfac4 -20.57194     7.62704   -2.697
## pathlengthfac5 -36.99674     7.62082   -4.855
## pathlengthfac6 -51.98456     7.61112   -6.830
## mean_len        5.82193     1.72125    3.382
## mean_logf        0.04784     1.99845    0.024
## mean_ldtz        8.40225    16.15960    0.520
##
## Correlation of Fixed Effects:
##          (Intr) pthln1 pthln3 pthln4 pthln5 pthln6 men_ln mn_lgf
## pthlngthfc1 -0.137
## pthlngthfc3 -0.154  0.502
## pthlngthfc4 -0.146  0.501  0.503
## pthlngthfc5 -0.149  0.502  0.504  0.502

```

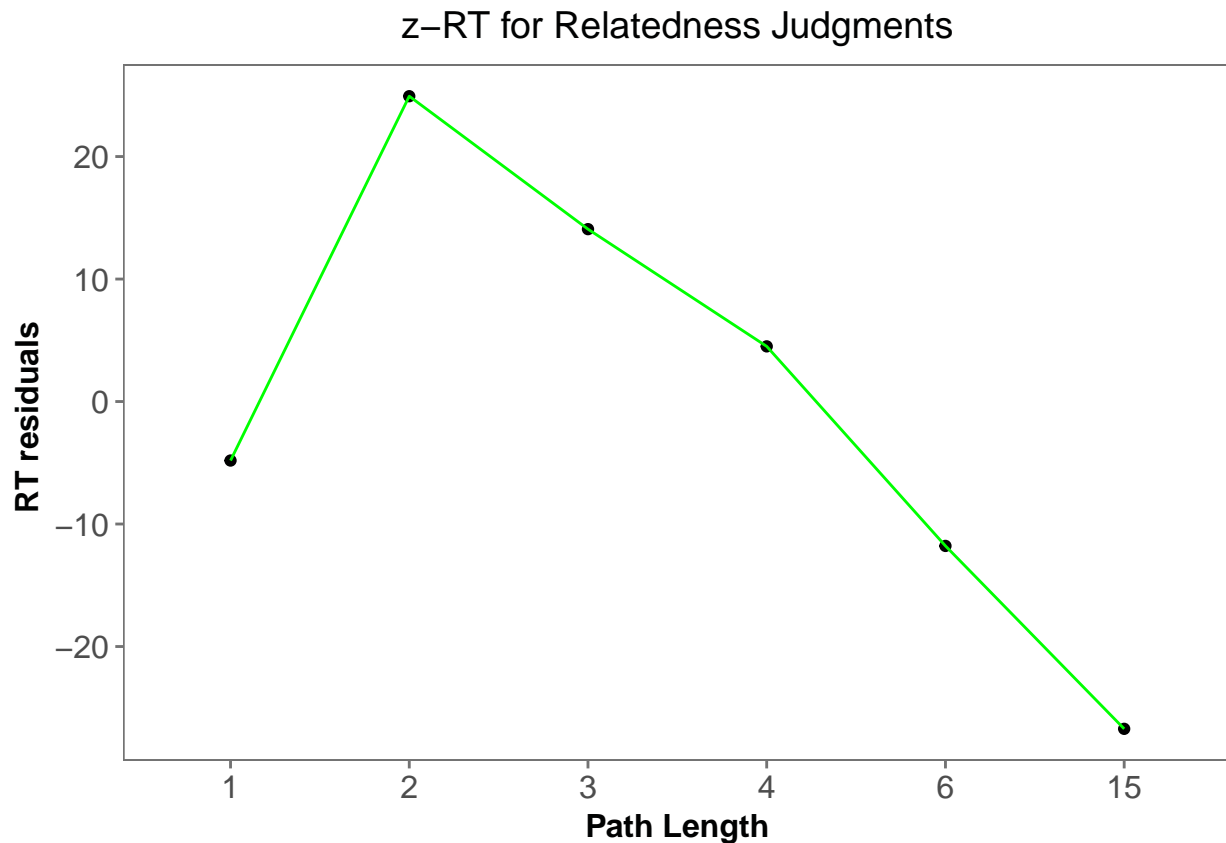


```
## pthlngthfc6 -0.152  0.503  0.504  0.503  0.503
## mean_len    -0.370  0.005  0.030  0.026  0.035  0.036
## mean_logf   -0.396  0.038  0.031  0.028  0.031  0.036 -0.112
## mean_ldtz    0.076  0.029 -0.019  0.000  0.008  0.008 -0.408  0.536
```

Plot from ELP Model

```
fixed.frame <-
  data.frame(expand.grid( pathlengthfac = c("1", "2", "3",
                                             "4", "6", "15"))) %>%
  mutate(pred = predict(m1_fixed_elp, newdata = ., re.form = NA))

fixed.frame %>%
  mutate(Pathlength = factor(pathlengthfac,
                             levels = unique(pathlengthfac),
                             labels = c("1", "2", "3",
                                           "4", "6", "15"))) %>%
  ggplot(aes(x = Pathlength, y = pred, group = 1)) +
    geom_point() +
    # geom_smooth(method = "loess") +
    geom_line(color = "green") +
    theme_few() +
    xlab("Path Length") + ylab("RT residuals ") +
    ggtitle("z-RT for Relatedness Judgments") +
    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)))
```



Concreteness Norms

```

elpnorms = read.csv("ELP_norms.csv", header = TRUE, sep = ",")
elpnorms = elpnorms[,c(1,2)]
colnames(elpnorms) = c("prime_word", "prime_concreteness")
elpnorms$prime_word = toupper(elpnorms$prime_word)
elpnorms$prime_word = paste(elpnorms$prime_word, "")

elpnorms$prime_word = as.character(elpnorms$prime_word)

sem$prime_word = as.character(sem$prime_word)
merged_sem_prime= inner_join(sem, elpnorms, by = "prime_word")
merged_sem_prime = merged_sem_prime[,c(4,7,15,37)]

colnames(elpnorms) = c("target_word", "target_concreteness")
sem$target_word = as.character(sem$target_word)
merged_sem_target= inner_join(sem, elpnorms, by = "target_word")
merged_sem_target = merged_sem_target[,c(4,7, 16,37)]

merged_concreteness = full_join(merged_sem_prime, merged_sem_target,
                                by = c("trial_index", "subject"))
merged_concreteness$mean_conc = (merged_concreteness$prime_concreteness +
                                merged_concreteness$target_concreteness) / 2

```

```
### NOW WE HAVE CONCRETENESS NORMS FOR ALL ITEMS IN THE DATASET
### NEED TO COMBINE THIS WITH ACTUAL SEM DATA
```

```
new_sem_z$prime_word = as.character(new_sem_z$prime_word)
new_sem_z$target_word = as.character(new_sem_z$target_word)
final_sem = inner_join(new_sem_z, merged_concreteness,
                        by = c("trial_index", "subject",
                              "prime_word", "target_word") )
```

Concreteness Model

```
final_sem$pathlengthfac = ordered(as.factor(as.character(final_sem$pathlength)),
                                  levels = c("1", "2", "3", "4", "6", "15"))
```

```
elp_model = lmer(data = final_sem, zRT_trim ~ mean_len + mean_logf +
                 mean_ldtz + mean_conc +
                 (1|subject) + (1|trial_index))
fit_from_elp = broom::augment(elp_model, final_sem)
```

```
contrasts(fit_from_elp$pathlengthfac) = contr.treatment(6, base = 2)
```

```
m1_fixed_elp = lmer(data = fit_from_elp, .resid ~ pathlengthfac +
                    (1|subject) + (1|trial_index))
summary(m1_fixed_elp)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: .resid ~ pathlengthfac + (1 | subject) + (1 | trial_index)
## Data: fit_from_elp
##
## REML criterion at convergence: 23002.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.5336 -0.7199 -0.2030  0.5673  4.1236
##
## Random effects:
## Groups      Name                Variance Std.Dev.
## trial_index (Intercept) 0.0000    0.0000
## subject      (Intercept) 0.0000    0.0000
## Residual                    0.9747    0.9873
## Number of obs: 8170, groups: trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)    0.12677    0.02710   4.677
## pathlengthfac1 -0.16299    0.03803  -4.286
## pathlengthfac3 -0.02289    0.03814  -0.600
## pathlengthfac4 -0.10358    0.03782  -2.739
## pathlengthfac5 -0.17736    0.03836  -4.624
## pathlengthfac6 -0.28782    0.03785  -7.604
##
## Correlation of Fixed Effects:
```

```

##          (Intr) pthln1 pthln3 pthln4 pthln5
## pthlngthfc1 -0.713
## pthlngthfc3 -0.711  0.506
## pthlngthfc4 -0.717  0.511  0.509
## pthlngthfc5 -0.707  0.503  0.502  0.506
## pthlngthfc6 -0.716  0.510  0.509  0.513  0.506

contrasts(final_sem$pathlengthfac) = contr.treatment(6, base = 2)

m1_all_elp = lme4::lmer(data = final_sem, zRT_trim ~ pathlengthfac +
                        mean_len + mean_logf + mean_ldtz + mean_conc +
                        (1|subject) + (1|trial_index) +
                        + (1|target_word))
summary(m1_all_elp)

## Linear mixed model fit by REML ['lmerMod']
## Formula: zRT_trim ~ pathlengthfac + mean_len + mean_logf + mean_ldtz +
##          mean_conc + (1 | subject) + (1 | trial_index) + +(1 | target_word)
##    Data: final_sem
##
## REML criterion at convergence: 23103.5
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6208 -0.7059 -0.1967  0.5499  3.9788
##
## Random effects:
##   Groups      Name      Variance Std.Dev.
## target_word (Intercept) 0.03846  0.19611
## trial_index (Intercept) 0.00762  0.08729
## subject      (Intercept) 0.00000  0.00000
## Residual                        0.94222  0.97068
## Number of obs: 8170, groups:
## target_word, 1741; trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)    0.469573   0.144530   3.249
## pathlengthfac1 -0.162219   0.040335  -4.022
## pathlengthfac3 -0.026076   0.040447  -0.645
## pathlengthfac4 -0.109103   0.040121  -2.719
## pathlengthfac5 -0.178195   0.040601  -4.389
## pathlengthfac6 -0.291293   0.040174  -7.251
## mean_len        0.019329   0.009508   2.033
## mean_logf       -0.017216   0.011378  -1.513
## mean_ldtz       -0.015642   0.087448  -0.179
## mean_conc       -0.085270   0.015420  -5.530
##
## Correlation of Fixed Effects:
##          (Intr) pthln1 pthln3 pthln4 pthln5 pthln6 men_ln mn_lgf mn_ldt
## pthlngthfc1 -0.128
## pthlngthfc3 -0.179  0.506
## pthlngthfc4 -0.190  0.510  0.511
## pthlngthfc5 -0.175  0.503  0.503  0.506
## pthlngthfc6 -0.162  0.510  0.509  0.514  0.508

```

```
## mean_len      -0.599 -0.007  0.039  0.029  0.046  0.031
## mean_logf     -0.660  0.017  0.026  0.041  0.034  0.027 -0.003
## mean_ldtz     -0.007  0.021 -0.018  0.013  0.008  0.006 -0.334  0.549
## mean_conc     -0.681 -0.042 -0.002  0.030 -0.011 -0.023  0.275  0.312  0.106
```

```
m1_all_elp_type = lme4::lmer(data = final_sem,
                             zRT_trim ~ pathlengthfac*Type +
                             mean_len + mean_logf + mean_ldtz + mean_conc +
                             (1|subject) + (1|trial_index) +
                             + (1|target_word))
summary(m1_all_elp_type)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## zRT_trim ~ pathlengthfac * Type + mean_len + mean_logf + mean_ldtz +
## mean_conc + (1 | subject) + (1 | trial_index) + +(1 | target_word)
## Data: final_sem
##
## REML criterion at convergence: 23051.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6059 -0.7023 -0.1857  0.5549  4.0611
##
## Random effects:
## Groups      Name                Variance Std.Dev.
## target_word (Intercept) 0.032969 0.1816
## trial_index (Intercept) 0.007192 0.0848
## subject      (Intercept) 0.000000 0.0000
## Residual                0.939188 0.9691
## Number of obs: 8170, groups:
## target_word, 1741; trial_index, 240; subject, 40
##
## Fixed effects:
##
##              Estimate Std. Error t value
## (Intercept)      0.538779   0.146095   3.688
## pathlengthfac1    -0.216892   0.053493  -4.055
## pathlengthfac3     0.157573   0.066920   2.355
## pathlengthfac4    -0.007960   0.068365  -0.116
## pathlengthfac5     0.151079   0.087122   1.734
## pathlengthfac6    -0.060296   0.094033  -0.641
## TypeUnrelated    -0.067301   0.055119  -1.221
## mean_len          0.015450   0.009409   1.642
## mean_logf        -0.017704   0.011252  -1.573
## mean_ldtz         0.010576   0.086511   0.122
## mean_conc        -0.082939   0.015248  -5.439
## pathlengthfac1:TypeUnrelated 0.119753   0.079858   1.500
## pathlengthfac3:TypeUnrelated -0.237445   0.082019  -2.895
## pathlengthfac4:TypeUnrelated -0.115904   0.082758  -1.401
## pathlengthfac5:TypeUnrelated -0.355639   0.098491  -3.611
## pathlengthfac6:TypeUnrelated -0.231945   0.104178  -2.226
##
##
## Correlation matrix not shown by default, as p = 16 > 12.
## Use print(x, correlation=TRUE) or
```

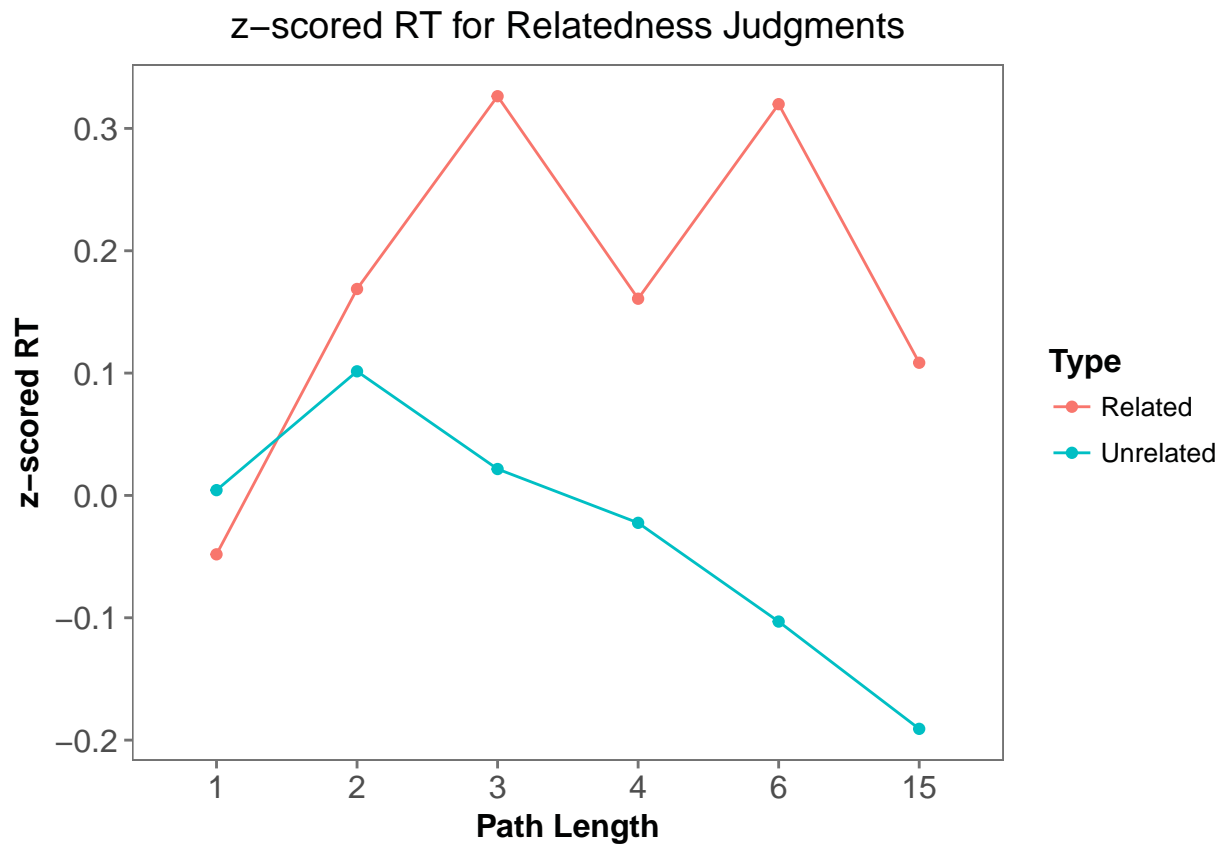
```
##   vcov(x)      if you need it
```

Plot

```
mean_length = mean(final_sem$mean_len, na.rm = TRUE)
mean_logfreq = mean(final_sem$mean_logf, na.rm = TRUE)
mean_lexdec = mean(final_sem$mean_ldtz, na.rm = TRUE)
mean_concreteness = mean(final_sem$mean_conc, na.rm = TRUE)

fixed.frame <-
  data.frame(expand.grid( pathlengthfac = c("1","2", "3",
                                           "4", "6", "15"),
                          Type= c("Related", "Unrelated"),
                          mean_len = mean_length,
                          mean_logf = mean_logfreq,
                          mean_ldtz = mean_lexdec,
                          mean_conc = mean_concreteness)) %>%
  mutate(pred = predict(m1_all_elp_type, newdata = ., re.form = NA))

fixed.frame %>%
  mutate(Pathlength = factor(pathlengthfac,
                             levels = unique(pathlengthfac),
                             labels = c("1","2", "3",
                                         "4", "6", "15")))%>%
  ggplot(aes(x = Pathlength, y = pred, group = Type, color = Type))+
  geom_point()+
  geom_line()+
  # geom_smooth(method = "loess")+
  #geom_line(color = "green")+
  theme_few()+
  xlab("Path Length") + ylab("z-scored RT") +
  ggtitle("z-scored RT for Relatedness Judgments") +
  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)))
```

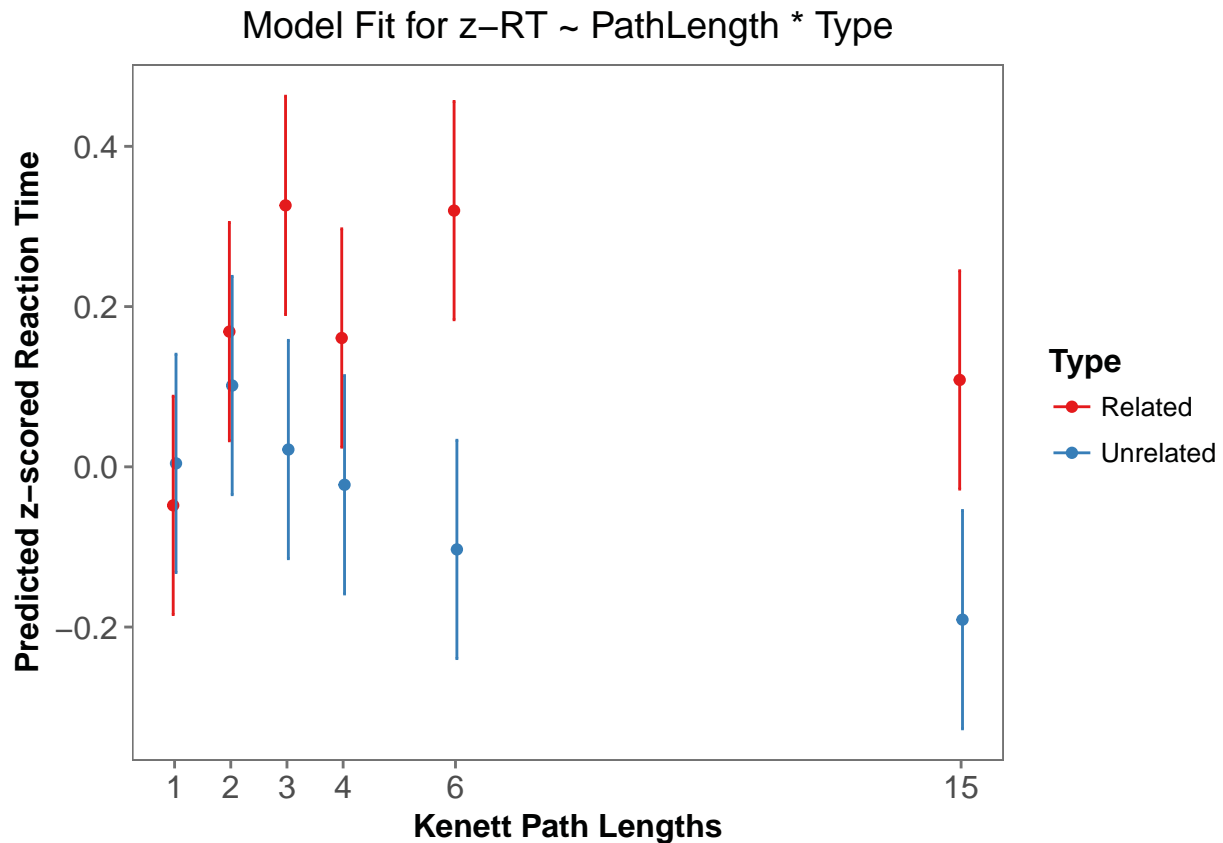


SjPlot Model

```
x = sjPlot::plot_model(m1_all_elp_type, type = "int")
```

Note: uncertainty of the random effects parameters are not taken into account for confidence intervals

```
x + theme_few() +
  xlab("Kenett Path Lengths") +
  ylab("Predicted z-scores Reaction Time") +
  ggtitle("Model Fit for z-RT ~ PathLength * Type") +
  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)))
```



Quadratic Model

```
## TESTING QUADRATIC TREND
m3_linear = lmer(data = final_sem, zRT_trim ~ pathlength +
                 mean_len + mean_logf + mean_ldtz + mean_conc +
                 (1|subject) + (1|trial_index) +
                 (1|target_word))
summary(m3_linear)

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## zRT_trim ~ pathlength + mean_len + mean_logf + mean_ldtz + mean_conc +
## (1 | subject) + (1 | trial_index) + (1 | target_word)
## Data: final_sem
##
## REML criterion at convergence: 23116.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.7039 -0.7082 -0.1944  0.5466  3.9646
##
## Random effects:
## Groups      Name                Variance Std.Dev.
## target_word (Intercept) 0.041520 0.20377
## trial_index (Intercept) 0.007236 0.08506
```



```

## subject      (Intercept) 0.000000 0.00000
## Residual      0.942721 0.97094
## Number of obs: 8170, groups:
## target_word, 1741; trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  4.464e-01  1.425e-01   3.133
## pathlength  -1.636e-02  2.474e-03  -6.614
## mean_len     1.846e-02  9.547e-03   1.933
## mean_logf    -1.656e-02  1.143e-02  -1.449
## mean_ldtz     9.075e-05  8.786e-02   0.001
## mean_conc    -8.899e-02  1.547e-02  -5.753
##
## Correlation of Fixed Effects:
##              (Intr) pthlng men_ln mn_lgf mn_ldt
## pathlength  -0.109
## mean_len     -0.605  0.031
## mean_logf    -0.666  0.015 -0.005
## mean_ldtz    -0.007  0.000 -0.333  0.550
## mean_conc    -0.693 -0.012  0.274  0.312  0.107

m3_quad = lmer(data = final_sem, zRT_trim ~ pathlength + I((pathlength)^2) +
              mean_len + mean_logf + mean_ldtz + mean_conc +
              (1|subject) + (1|trial_index) +
              + (1|target_word))
summary(m3_quad)

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## zRT_trim ~ pathlength + I((pathlength)^2) + mean_len + mean_logf +
## mean_ldtz + mean_conc + (1 | subject) + (1 | trial_index) +
## +(1 | target_word)
## Data: final_sem
##
## REML criterion at convergence: 23128.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6919 -0.7088 -0.1953  0.5490  3.9573
##
## Random effects:
## Groups      Name             Variance Std.Dev.
## target_word (Intercept) 0.041432 0.20355
## trial_index (Intercept) 0.007273 0.08528
## subject      (Intercept) 0.000000 0.00000
## Residual      0.942805 0.97098
## Number of obs: 8170, groups:
## target_word, 1741; trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)      0.4166791  0.1470053   2.834
## pathlength      -0.0065184  0.0122664  -0.531
## I((pathlength)^2) -0.0005875  0.0007171  -0.819

```

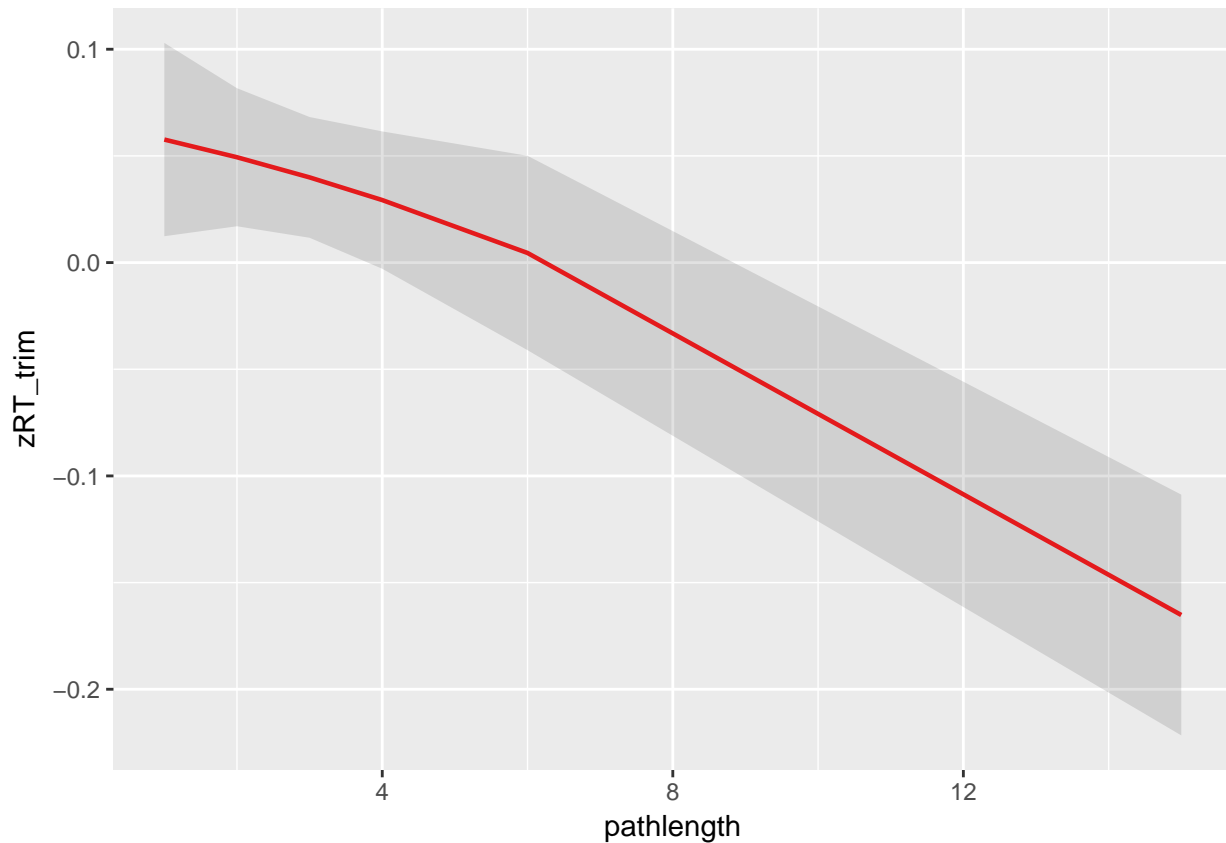
```

## mean_len          0.0188756  0.0095593   1.975
## mean_logf        -0.0162806  0.0114353  -1.424
## mean_ldtz        -0.0004100  0.0878500  -0.005
## mean_conc        -0.0883897  0.0154853  -5.708
##
## Correlation of Fixed Effects:
##          (Intr) pthlng I(( )^2 men_ln mn_lgf mn_ldt
## pathlength  -0.262
## I((pthl)^2)  0.246 -0.979
## mean_len    -0.599  0.058 -0.053
## mean_logf   -0.652  0.032 -0.030 -0.003
## mean_ldtz   -0.005 -0.007  0.007 -0.333  0.549
## mean_conc   -0.683  0.044 -0.047  0.276  0.313  0.107
anova(m3_linear, m3_quad)

## refitting model(s) with ML (instead of REML)
## Data: final_sem
## Models:
## m3_linear: zRT_trim ~ pathlength + mean_len + mean_logf + mean_ldtz + mean_conc +
## m3_linear:      (1 | subject) + (1 | trial_index) + +(1 | target_word)
## m3_quad: zRT_trim ~ pathlength + I((pathlength)^2) + mean_len + mean_logf +
## m3_quad:      mean_ldtz + mean_conc + (1 | subject) + (1 | trial_index) +
## m3_quad:      +(1 | target_word)
##          Df   AIC   BIC logLik deviance  Chisq Chi Df Pr(>Chisq)
## m3_linear 10 23094 23164 -11537    23074
## m3_quad   11 23096 23173 -11537    23074 0.6739     1    0.4117
sjPlot::sjp.lm(m3_quad, type = "poly", poly.term = "pathlength")

## `sjp.lm()` will become deprecated in the future. Please use `plot_model()` instead.

```

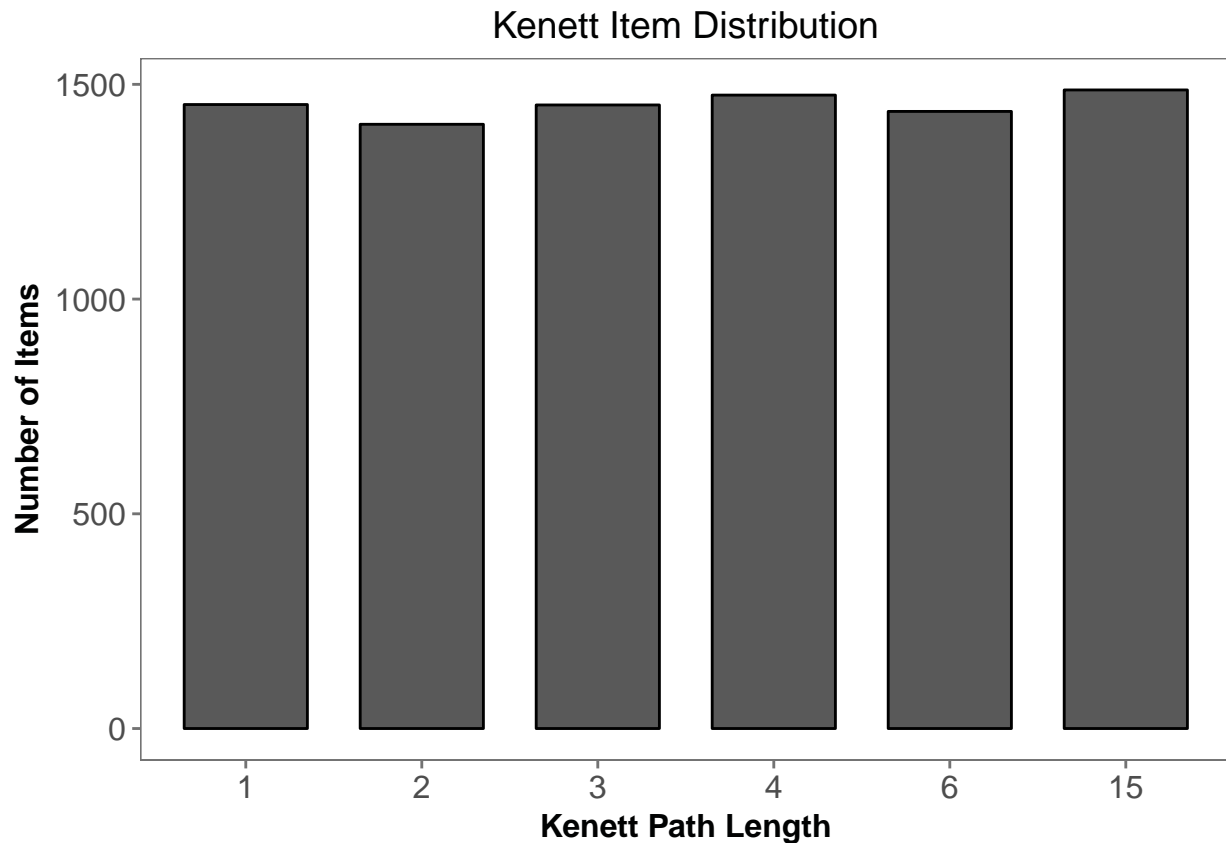


Other Networks

Kenett Path Length

```
items_kenett = group_by(final_sem, pathlength) %>%
  summarise(items = n())

items_kenett$pathlength = as.factor(items_kenett$pathlength)
ggplot(items_kenett, aes(x = pathlength, y = items))+
  geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
  theme_few()+
  xlab("Kenett Path Length") + ylab("Number of Items") +
  ggtitle("Kenett Item Distribution") +
  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)))
```



Undirected

```

items_undirected = group_by(final_sem, undirected) %>%
  summarise(items = n())

items_undirected_subject = group_by(final_sem, subject, undirected) %>%
  summarise(items = n())

undirected_rmisc = Rmisc::summarySE(items_undirected_subject,
  measurevar = "items",
  groupvars = c("undirected"))

final_sem$undirectedfac = ordered(as.factor(as.character(final_sem$undirected)),
  levels = c("1", "2", "3", "4"))
contrasts(final_sem$undirectedfac) = contr.treatment(4, base = 2)

m_undirected = lmer(data = final_sem, zRT_trim ~ undirectedfac +
  mean_len + mean_logf + mean_ldtz + mean_conc +
  (1|subject) + (1|trial_index) +
  + (1|target_word))
summary(m_undirected)

## Linear mixed model fit by REML ['lmerMod']
## Formula: zRT_trim ~ undirectedfac + mean_len + mean_logf + mean_ldtz +

```

```

##      mean_conc + (1 | subject) + (1 | trial_index) + +(1 | target_word)
##      Data: final_sem
##
## REML criterion at convergence: 23086.9
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.7227 -0.7133 -0.2025  0.5444  4.0369
##
## Random effects:
##      Groups             Name             Variance Std.Dev.
## target_word (Intercept) 0.037414 0.19343
## trial_index (Intercept) 0.006589 0.08117
## subject      (Intercept) 0.000000 0.00000
## Residual                0.942936 0.97105
## Number of obs: 8170, groups:
## target_word, 1741; trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)    0.482695   0.141761   3.405
## undirectedfac1 -0.283504   0.043206  -6.562
## undirectedfac3 -0.141262   0.025055  -5.638
## undirectedfac4 -0.307011   0.049505  -6.202
## mean_len        0.018833   0.009468   1.989
## mean_logf       -0.011706   0.011363  -1.030
## mean_ldtz        0.045660   0.087680   0.521
## mean_conc       -0.099461   0.015398  -6.459
##
## Correlation of Fixed Effects:
##              (Intr) undrc1 undrc3 undrc4 men_ln mn_lgf mn_ldt
## undirctdfc1 -0.059
## undirctdfc3 -0.122  0.303
## undirctdfc4 -0.074  0.150  0.266
## mean_len    -0.602  0.016  0.015  0.012
## mean_logf   -0.660 -0.049  0.007 -0.022 -0.006
## mean_ldtz   -0.001  0.003 -0.017 -0.102 -0.333  0.548
## mean_conc   -0.697  0.084  0.039  0.019  0.275  0.307  0.106

m_undirected_type = lmer(data = final_sem, zRT_trim ~ undirectedfac*Type +
                        mean_len + mean_logf + mean_ldtz + mean_conc +
                        (1|subject) + (1|trial_index) +
                        + (1|target_word))
summary(m_undirected_type)

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## zRT_trim ~ undirectedfac * Type + mean_len + mean_logf + mean_ldtz +
##      mean_conc + (1 | subject) + (1 | trial_index) + +(1 | target_word)
##      Data: final_sem
##
## REML criterion at convergence: 23025
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max

```

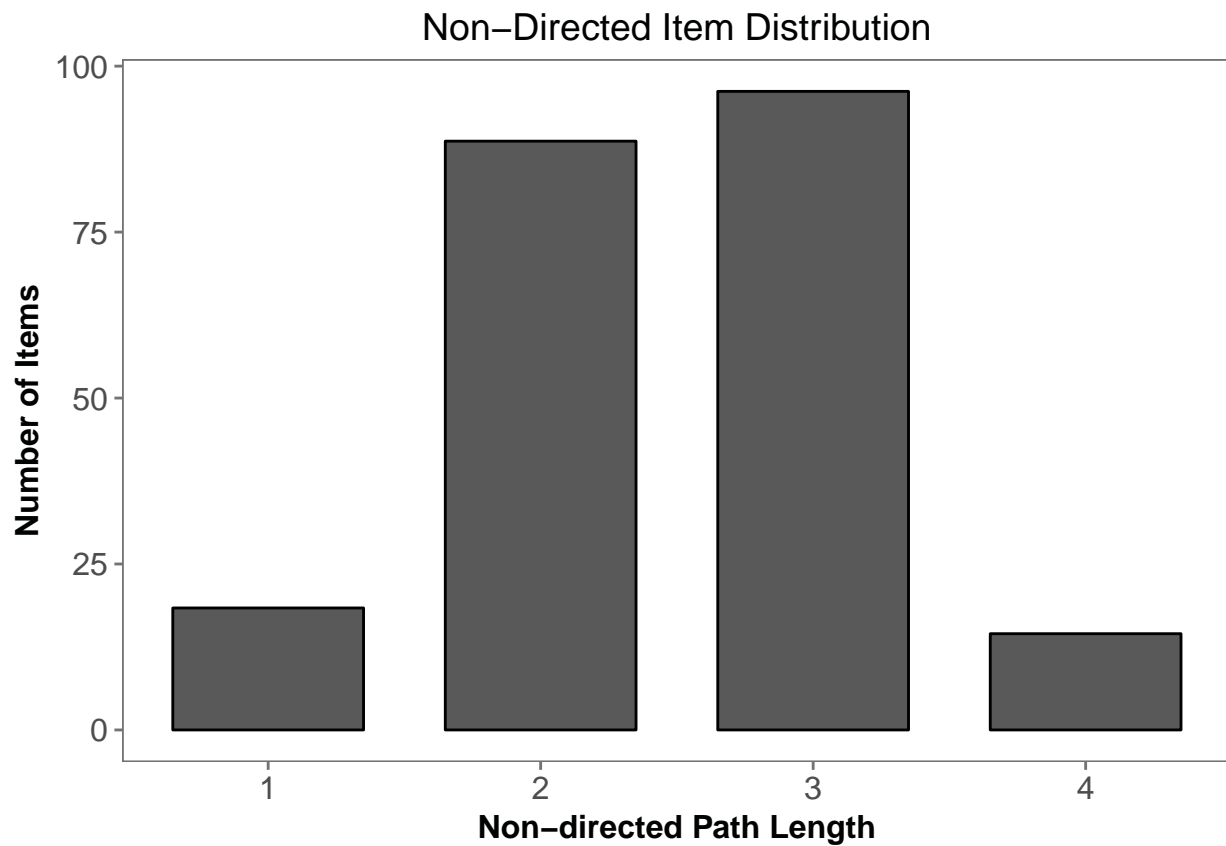
```

## -2.8171 -0.7129 -0.1926  0.5499  4.1372
##
## Random effects:
##   Groups      Name      Variance Std.Dev.
## target_word (Intercept) 0.03249  0.18024
## trial_index (Intercept) 0.00668  0.08173
## subject      (Intercept) 0.00000  0.00000
## Residual                0.93855  0.96879
## Number of obs: 8170, groups:
## target_word, 1741; trial_index, 240; subject, 40
##
## Fixed effects:
##
##              Estimate Std. Error t value
## (Intercept)      0.61911    0.14234   4.350
## undirectedfac1    -0.43169    0.05122  -8.428
## undirectedfac3      0.03379    0.05175   0.653
## undirectedfac4    -0.26933    0.18560  -1.451
## TypeUnrelated    -0.17416    0.03484  -4.999
## mean_len          0.01372    0.00939   1.461
## mean_logf        -0.01334    0.01124  -1.187
## mean_ldtz         0.05194    0.08675   0.599
## mean_conc        -0.09692    0.01522  -6.367
## undirectedfac1:TypeUnrelated 0.39650    0.09385   4.225
## undirectedfac3:TypeUnrelated -0.14392    0.05902  -2.439
## undirectedfac4:TypeUnrelated 0.03091    0.19154   0.161
##
## Correlation of Fixed Effects:
##              (Intr) undrc1 undrc3 undrc4 TypUnr men_ln mn_lgf mn_ldt mn_cnc
## undirctdfc1 -0.117
## undirctdfc3 -0.113  0.268
## undirctdfc4 -0.035  0.073  0.075
## TypeUnreltd -0.175  0.381  0.375  0.104
## mean_len    -0.603  0.033  0.004 -0.007  0.070
## mean_logf   -0.651 -0.028  0.000 -0.006  0.008 -0.006
## mean_ldtz    0.000  0.011  0.004 -0.033 -0.002 -0.333  0.548
## mean_conc   -0.687  0.075  0.045  0.014  0.009  0.274  0.307  0.106
## undrctd1:TU  0.070 -0.510 -0.140 -0.039 -0.371 -0.009 -0.027 -0.020 -0.007
## undrctd3:TU  0.092 -0.228 -0.863 -0.063 -0.587 -0.011  0.001 -0.013 -0.029
## undrctd4:TU  0.029 -0.070 -0.070 -0.962 -0.181  0.005 -0.001  0.007 -0.010
##
##              un1:TU un3:TU
## undirctdfc1
## undirctdfc3
## undirctdfc4
## TypeUnreltd
## mean_len
## mean_logf
## mean_ldtz
## mean_conc
## undrctd1:TU
## undrctd3:TU  0.219
## undrctd4:TU  0.068  0.108

```

Plot Item Distribution

```
ggplot(undirected_rmisc, aes(x = undirected, y = items))+  
  geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+  
  theme_few()+  
  xlab("Non-directed Path Length") + ylab("Number of Items") +  
  ggtitle("Non-Directed Item Distribution") +  
  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)))
```



Plot Undirected

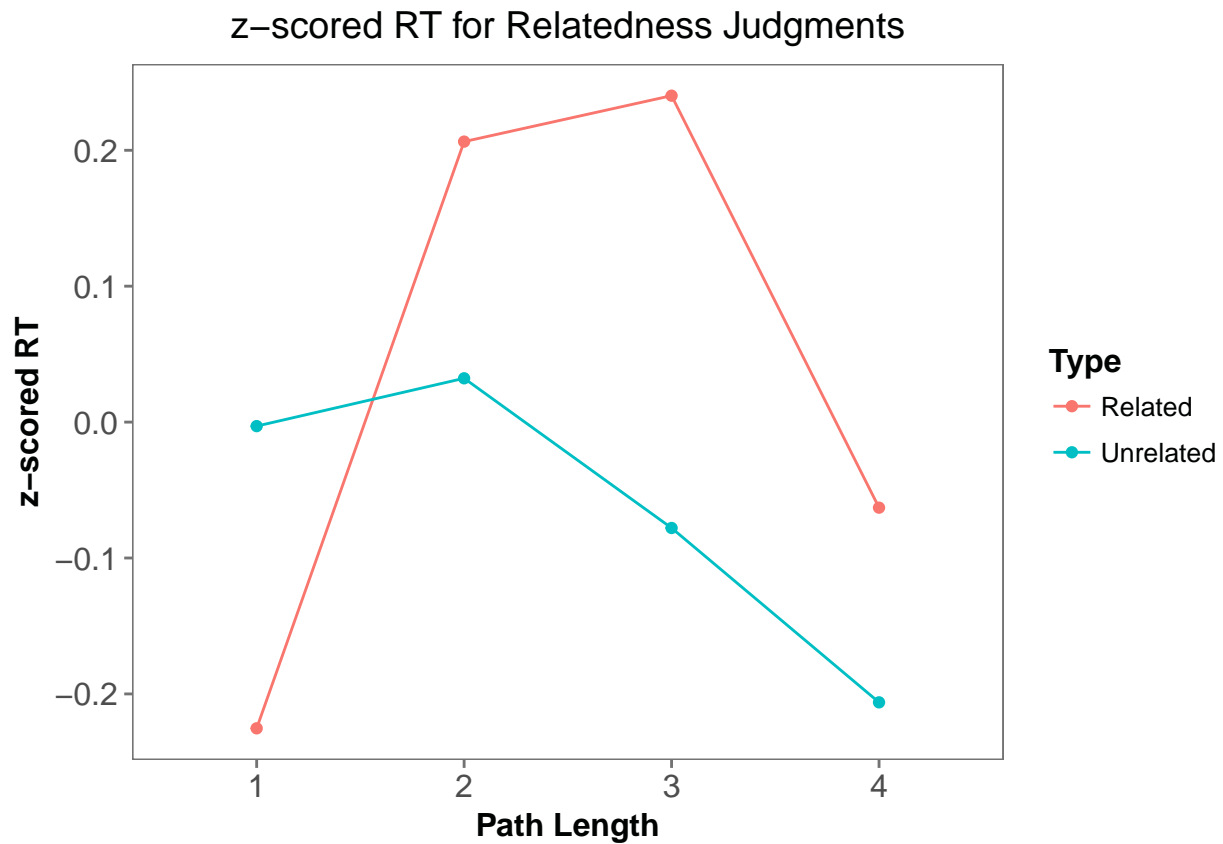
```
mean_length = mean(final_sem$mean_len, na.rm = TRUE)  
mean_logfreq = mean(final_sem$mean_logf, na.rm = TRUE)  
mean_lexdec = mean(final_sem$mean_ldtz, na.rm = TRUE)  
mean_concreteness = mean(final_sem$mean_conc, na.rm = TRUE)  
  
fixed.frame <-  
  data.frame(expand.grid( undirectedfac = c("1", "2", "3", "4"),  
                          Type = c("Related", "Unrelated"),  
                          mean_len = mean_length,
```

```

      mean_logf = mean_logfreq,
      mean_ldtz = mean_lexdec,
      mean_conc = mean_concreteness)) %>%
mutate(pred = predict(m_undirected_type, newdata = ., re.form = NA))

fixed.frame %>%
  mutate(Pathlength = factor(undirectedfac,
    levels = unique(undirectedfac),
    labels = c("1", "2", "3", "4"))) %>%
ggplot(aes(x = Pathlength, y = pred, group = Type, color = Type)) +
  geom_point() +
  # geom_smooth(method = "loess") +
  geom_line() +
  theme_few() +
  xlab("Path Length") + ylab("z-scored RT") +
  ggtitle("z-scored RT for Relatedness Judgments") +
  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)))

```

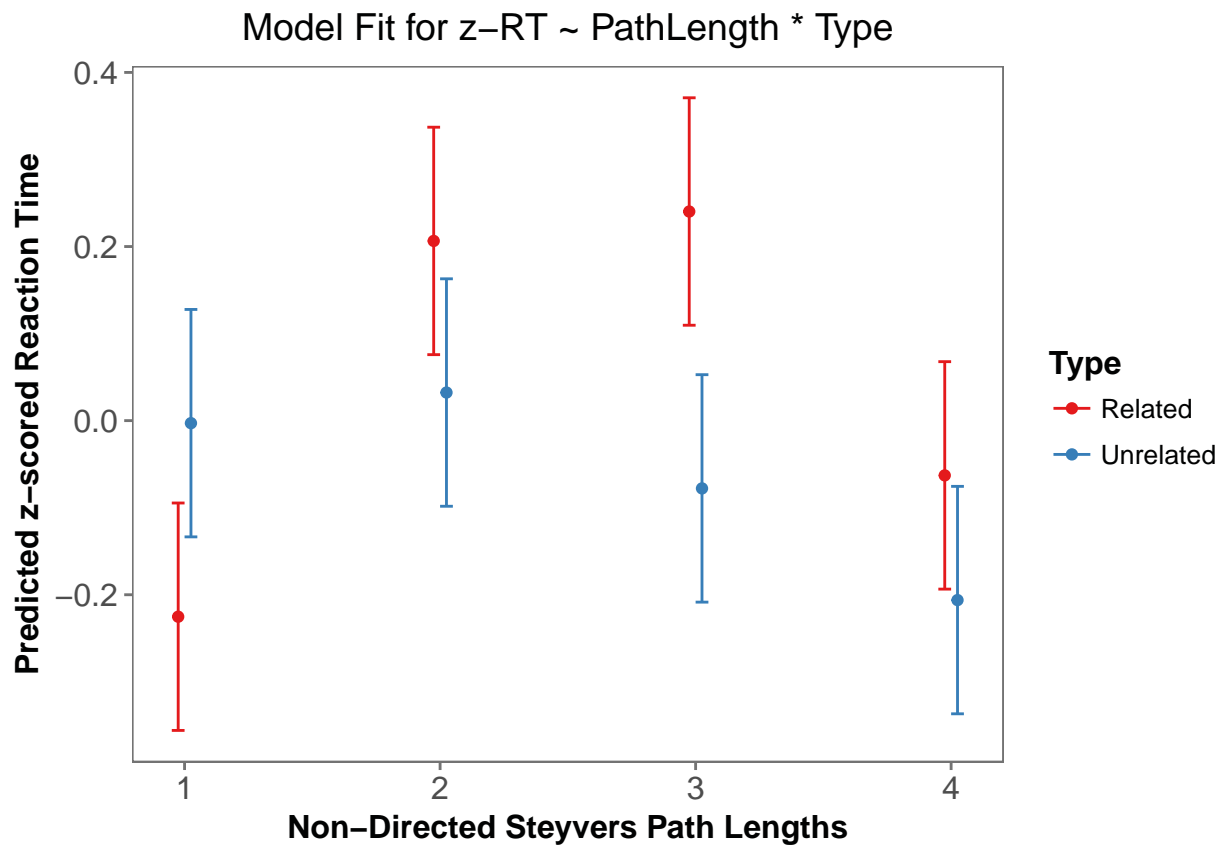


SjPlot Model

```
x = sjPlot::plot_model(m_undirected_type, type = "int")
```

Note: uncertainty of the random effects parameters are not taken into account for confidence intervals.

```
x + theme_few() +  
  xlab("Non-Directed Steyvers Path Lengths") +  
  ylab("Predicted z-scored Reaction Time") +  
  ggtitle("Model Fit for z-RT ~ PathLength * Type") +  
  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)))
```



Directed

```
final_sem$newdirected = ifelse(final_sem$directed == "Inf" |  
                               final_sem$directed == "NA", NA,  
                               final_sem$directed)  
  
final_sem$directedcollapsed = ifelse((final_sem$newdirected == "5" |  
                                       final_sem$newdirected == "6" |  
                                       final_sem$newdirected == "7" |
```

```

                                final_sem$newdirected == "8"), "H",
                                final_sem$newdirected)

items_directed = group_by(final_sem, newdirected) %>%
  summarise(items = n())

items_directed_subject = group_by(final_sem, subject, newdirected) %>%
  summarise(items = n())

directed_rmisc = Rmisc::summarySE(items_directed_subject,
                                   measurevar = "items",
                                   groupvars = c("newdirected"))

final_sem$directedfac =
  ordered(as.factor(as.character(final_sem$newdirected))),
          levels = c("1", "2", "3", "4", "5",
                     "6", "7", "8"))
contrasts(final_sem$directedfac) = contr.treatment(8, base = 2)

final_sem$collapsedfac =
  ordered(as.factor(as.character(final_sem$directedcollapsed))),
          levels = c("1", "2", "3", "4", "H"))
contrasts(final_sem$collapsedfac) = contr.treatment(5, base = 2)

m_directed = lme4::lmer(data = final_sem, zRT_trim ~ collapsedfac +
                        mean_len + mean_logf + mean_ldtz + mean_conc +
                        (1|subject) + (1|trial_index) +
                        + (1|target_word))
summary(m_directed)

## Linear mixed model fit by REML ['lmerMod']
## Formula: zRT_trim ~ collapsedfac + mean_len + mean_logf + mean_ldtz +
##          mean_conc + (1 | subject) + (1 | trial_index) + +(1 | target_word)
## Data: final_sem
##
## REML criterion at convergence: 22250.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6966 -0.7093 -0.2074  0.5562  3.9095
##
## Random effects:
##   Groups      Name      Variance Std.Dev.
## target_word (Intercept) 0.041829 0.20452
## trial_index (Intercept) 0.006879 0.08294
## subject     (Intercept) 0.000000 0.00000
## Residual                0.941108 0.97011
## Number of obs: 7866, groups:
## target_word, 1673; trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)   0.479590   0.150491   3.187

```

```

## collapsedfac1 -0.262369  0.056515 -4.642
## collapsedfac3  0.021607  0.039350  0.549
## collapsedfac4 -0.081550  0.036610 -2.228
## collapsedfac5 -0.144159  0.038750 -3.720
## mean_len      0.018071  0.009767  1.850
## mean_logf     -0.013135  0.011725 -1.120
## mean_ldtz      0.044319  0.090534  0.490
## mean_conc     -0.102677  0.015856 -6.475
##
## Correlation of Fixed Effects:
##      (Intr) cllps1 cllps3 cllps4 cllps5 men_ln mn_lgf mn_ldt
## collapsedfc1 -0.120
## collapsedfc3 -0.145  0.402
## collapsedfc4 -0.198  0.434  0.620
## collapsedfc5 -0.218  0.410  0.587  0.641
## mean_len     -0.596  0.016 -0.014  0.003 -0.012
## mean_logf    -0.660 -0.032  0.001  0.021  0.043  0.006
## mean_ldtz     0.014 -0.003 -0.004 -0.019 -0.057 -0.339  0.536
## mean_conc    -0.691  0.076 -0.004  0.036  0.055  0.281  0.319  0.100

m_directed_type = lme4::lmer(data = final_sem, zRT_trim ~ collapsedfac*Type +
                             mean_len + mean_logf + mean_ldtz + mean_conc +
                             (1|subject) + (1|trial_index) +
                             + (1|target_word))
summary(m_directed_type)

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## zRT_trim ~ collapsedfac * Type + mean_len + mean_logf + mean_ldtz +
##      mean_conc + (1 | subject) + (1 | trial_index) + +(1 | target_word)
##      Data: final_sem
##
## REML criterion at convergence: 22180.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.7188 -0.7057 -0.1923  0.5503  4.0091
##
## Random effects:
##      Groups      Name      Variance Std.Dev.
## target_word (Intercept) 0.035422 0.18821
## trial_index (Intercept) 0.007105 0.08429
## subject      (Intercept) 0.000000 0.00000
## Residual                0.936022 0.96748
## Number of obs: 7866, groups:
## target_word, 1673; trial_index, 240; subject, 40
##
## Fixed effects:
##
##              Estimate Std. Error t value
## (Intercept)      0.52979   0.15103   3.508
## collapsedfac1    -0.35396   0.06567  -5.390
## collapsedfac3      0.11403   0.05881   1.939
## collapsedfac4      0.09724   0.05863   1.659
## collapsedfac5      0.15122   0.06548   2.309
## TypeUnrelated    -0.05323   0.05755  -0.925

```

```
## mean_len          0.01285    0.00966    1.331
## mean_logf        -0.01449    0.01157   -1.253
## mean_ldtz         0.04968    0.08939    0.556
## mean_conc        -0.09814    0.01565   -6.269
## collapsedfac1:TypeUnrelated  0.36441    0.12092    3.014
## collapsedfac3:TypeUnrelated -0.11054    0.07824   -1.413
## collapsedfac4:TypeUnrelated -0.19897    0.07585   -2.623
## collapsedfac5:TypeUnrelated -0.33433    0.08177   -4.089

##
## Correlation matrix not shown by default, as p = 14 > 12.
## Use print(x, correlation=TRUE) or
##   vcov(x)      if you need it
```

Plot Item Distribution

```
items_directed$newdirected = as.factor(items_directed$newdirected)
ggplot(directed_rmisc, aes(x = newdirected, y = items))+
  geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
  theme_few()+
  xlab("Directed Path Length") + ylab("Number of Items") +
  ggtitle("Directed Item Distribution") +
  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)))

## Warning: Removed 1 rows containing missing values (geom_bar).
```



Plot Directed

```
mean_length = mean(final_sem$mean_len, na.rm = TRUE)
mean_logfreq = mean(final_sem$mean_logf, na.rm = TRUE)
mean_lexdec = mean(final_sem$mean_ldtz, na.rm = TRUE)
mean_concreteness = mean(final_sem$mean_conc, na.rm = TRUE)

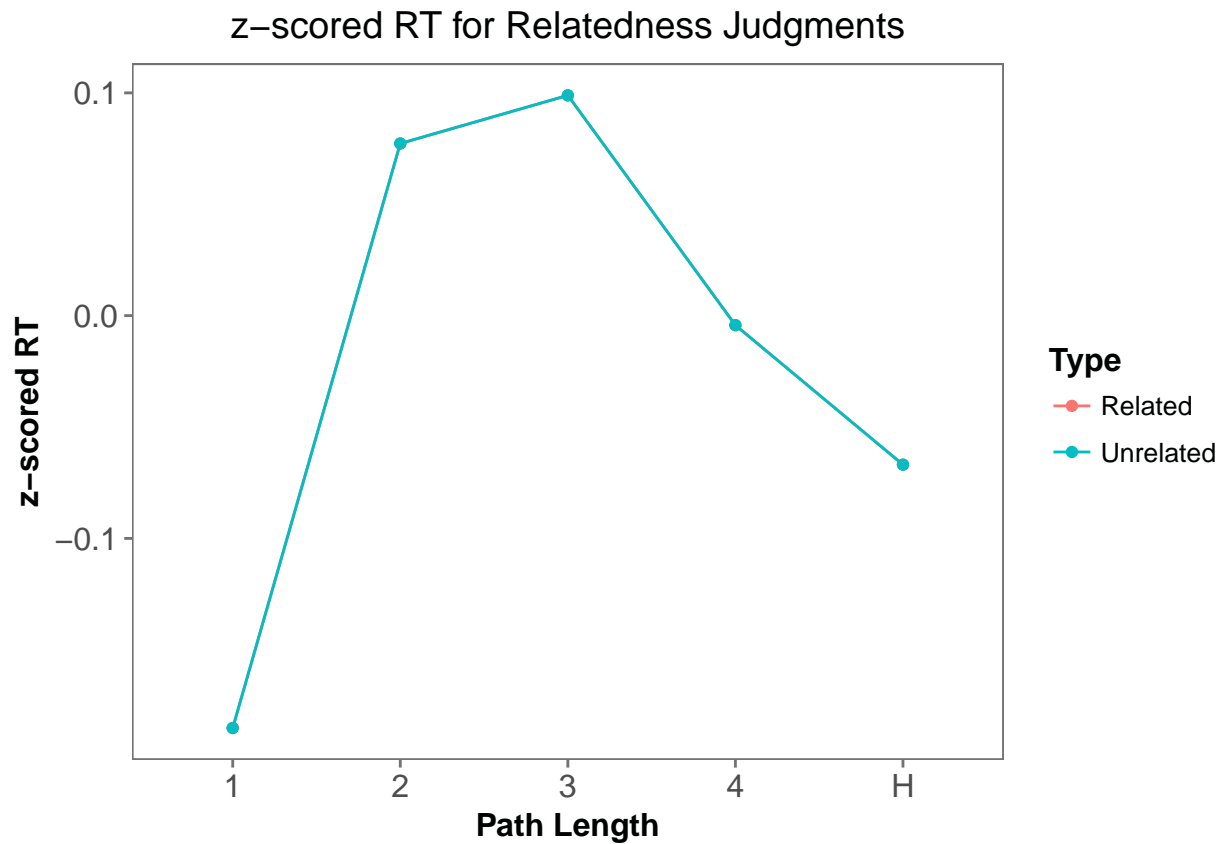
fixed.frame <-
  data.frame(expand.grid( collapsedfac = c("1", "2", "3", "4", "H"),
    Type = c("Related", "Unrelated"),
    mean_len = mean_length,
    mean_logf = mean_logfreq,
    mean_ldtz = mean_lexdec,
    mean_conc = mean_concreteness)) %>%
  mutate(pred = predict(m_directed, newdata = ., re.form = NA))

# fixed.frame <-
#   data.frame(expand.grid( newdirected =
#     seq(min(final_sem$newdirected, na.rm = TRUE),
#       max(final_sem$newdirected, na.rm = TRUE),
#       1),
#     mean_len = mean_length,
#     mean_logf = mean_logfreq,
#     mean_ldtz = mean_lexdec,
#     mean_conc = mean_concreteness)) %>%
#   mutate(pred = predict(m_directed, newdata = ., re.form = NA))
```

```

fixed.frame %>%
  mutate(Pathlength = factor(collapsedfac,
    levels = unique(collapsedfac),
    labels = c("1", "2", "3", "4", "H"))) %>%
  ggplot(aes(x = collapsedfac, y = pred, group = Type, color = Type)) +
  geom_point() +
  # geom_smooth(method = "loess") +
  geom_line() +
  theme_few() +
  xlab("Path Length") + ylab("z-scored RT") +
  ggtitle("z-scored RT for Relatedness Judgments") +
  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)))

```



SjPlot Model

```

x = sjPlot::plot_model(m_directed_type, type = "int")

```

Note: uncertainty of the random effects parameters are not taken into account for confidence intervals

```

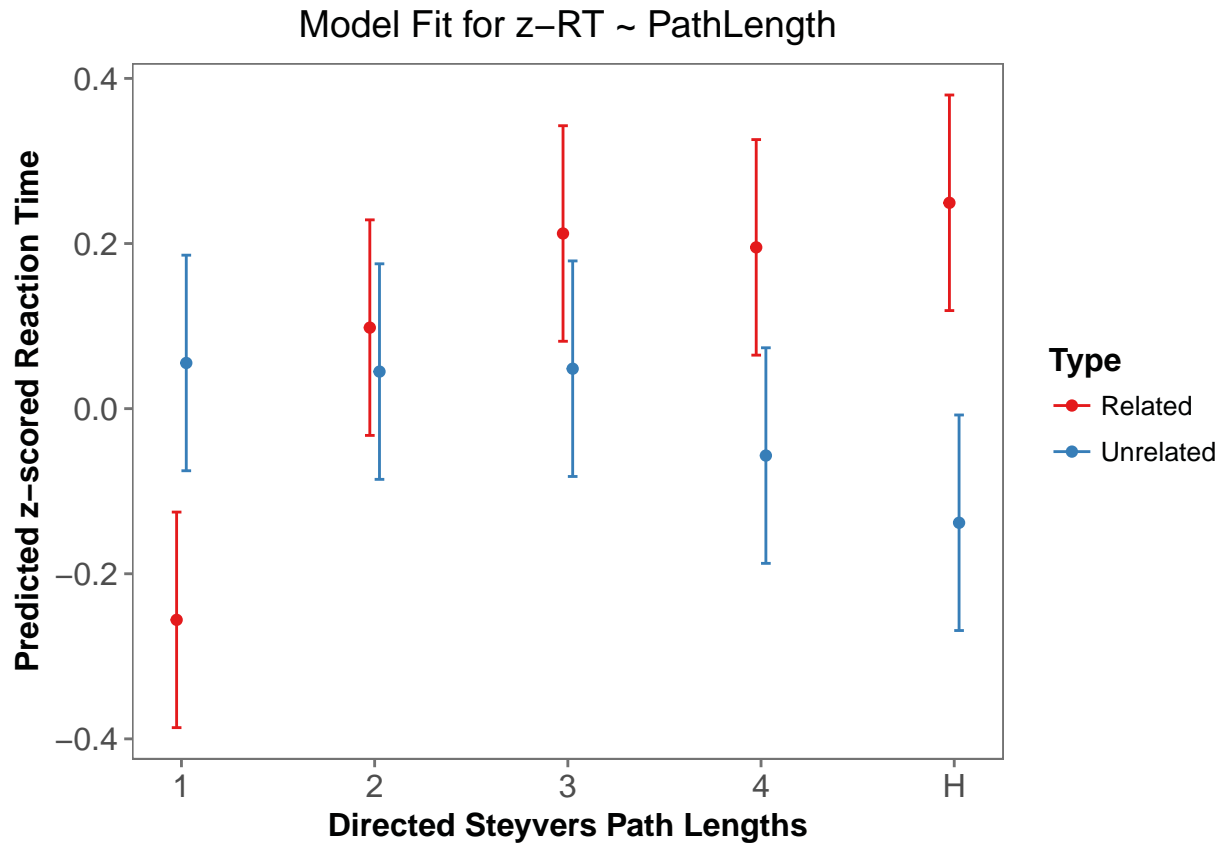
x + theme_few() +
  xlab("Directed Steyvers Path Lengths") +

```

```

ylab("Predicted z-scored Reaction Time") +
ggtitle("Model Fit for z-RT ~ PathLength") +
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)))

```



Log P

```

final_sem$logp = round(final_sem$logp, 2)

final_sem$newlogp = ifelse(final_sem$logp == "Inf" |
                          final_sem$logp == "NA", NA,
                          final_sem$logp)
final_sem$roundedlogp = round(final_sem$newlogp, 0)

items_logp = group_by(final_sem, roundedlogp) %>%
  summarise(items = n())

items_logp_subject = group_by(final_sem, subject, roundedlogp) %>%
  summarise(items = n())

logp_rmisc = Rmisc::summarySE(items_logp_subject,
                              measurevar = "items",

```

```

groupvars = c("roundedlogp"))

m_logp = lmer(data = final_sem, zRT_trim ~ newlogp +
              mean_len + mean_logf + mean_ldtz + mean_conc +
              (1|subject) + (1|trial_index) +
              + (1|target_word))
summary(m_logp)

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## zRT_trim ~ newlogp + mean_len + mean_logf + mean_ldtz + mean_conc +
## (1 | subject) + (1 | trial_index) + +(1 | target_word)
## Data: final_sem
##
## REML criterion at convergence: 21105.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6738 -0.7055 -0.2085  0.5527  3.9507
##
## Random effects:
## Groups      Name                Variance Std.Dev.
## target_word (Intercept) 0.046259 0.21508
## trial_index (Intercept) 0.008487 0.09213
## subject      (Intercept) 0.000000 0.00000
## Residual                    0.942873 0.97102
## Number of obs: 7444, groups:
## target_word, 1673; trial_index, 240; subject, 38
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  0.396958   0.158327   2.507
## newlogp      -0.002093   0.002005  -1.044
## mean_len      0.017918   0.010154   1.765
## mean_logf     -0.013591   0.012133  -1.120
## mean_ldtz     0.046904   0.094134   0.498
## mean_conc    -0.089047   0.016390  -5.433
##
## Correlation of Fixed Effects:
##              (Intr) newlgp men_ln mn_lgf mn_ldt
## newlogp      -0.281
## mean_len     -0.578 -0.037
## mean_logf    -0.661  0.059  0.005
## mean_ldtz     0.029 -0.088 -0.343  0.535
## mean_conc    -0.689  0.069  0.272  0.331  0.106

m_logp_type = lmer(data = final_sem, zRT_trim ~ newlogp*Type +
                  mean_len + mean_logf + mean_ldtz + mean_conc +
                  (1|subject) + (1|trial_index) +
                  + (1|target_word))
summary(m_logp_type)

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

```



```

## Formula: zRT_trim ~ newlogp * Type + mean_len + mean_logf + mean_ldtz +
##      mean_conc + (1 | subject) + (1 | trial_index) + (1 | target_word)
##      Data: final_sem
##
## REML criterion at convergence: 21042.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6996 -0.7057 -0.1962  0.5447  3.9531
##
## Random effects:
##      Groups      Name      Variance Std.Dev.
## target_word (Intercept) 0.039887 0.19972
## trial_index (Intercept) 0.008394 0.09162
## subject      (Intercept) 0.000000 0.00000
## Residual                0.938255 0.96864
## Number of obs: 7444, groups:
## target_word, 1673; trial_index, 240; subject, 38
##
## Fixed effects:
##
##              Estimate Std. Error t value
## (Intercept)      0.255111   0.159016   1.604
## newlogp           0.017775   0.003117   5.702
## TypeUnrelated     0.273450   0.065135   4.198
## mean_len          0.012525   0.010059   1.245
## mean_logf         -0.013984   0.011975  -1.168
## mean_ldtz         0.065216   0.092956   0.702
## mean_conc         -0.085761   0.016179  -5.301
## newlogp:TypeUnrelated -0.027875   0.003967  -7.026
##
## Correlation of Fixed Effects:
##              (Intr) newlgp TypUnr men_ln mn_lgf mn_ldt mn_cnc
## newlogp      -0.291
## TypeUnreltd -0.186  0.589
## mean_len     -0.570 -0.050  0.027
## mean_logf    -0.649  0.033 -0.005  0.005
## mean_ldtz     0.026 -0.039  0.016 -0.343  0.534
## mean_conc    -0.672  0.047 -0.027  0.267  0.331  0.106
## nwlgp:TypUn  0.175 -0.742 -0.913  0.010  0.005 -0.021  0.009

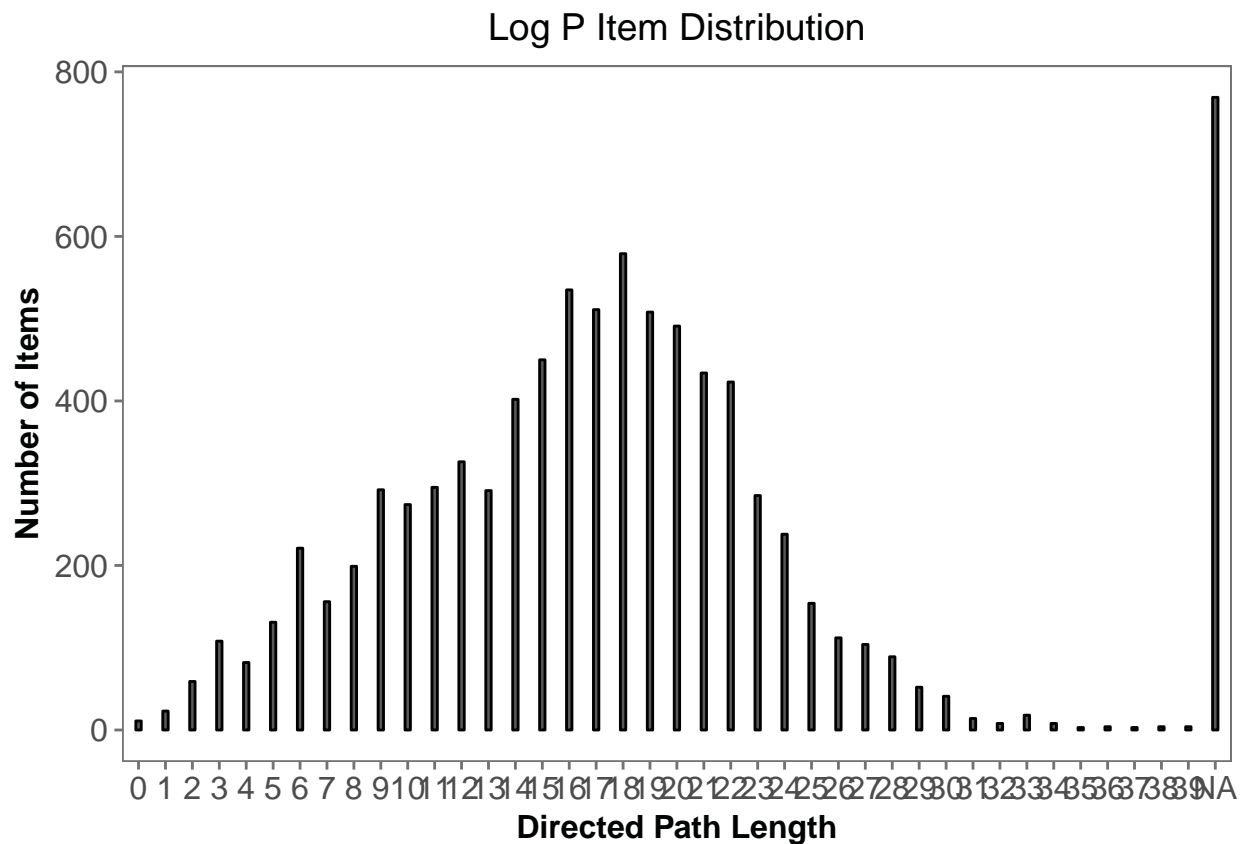
```

Plot Item Distribution

```

items_logp$roundedlogp = as.factor(items_logp$roundedlogp)
ggplot(items_logp, aes(x = roundedlogp, y = items))+
  geom_bar(stat = "identity", position = "dodge", width = 0.2, color= "black")+
  theme_few()+
  xlab("Directed Path Length") + ylab("Number of Items") +
  ggtitle("Log P Item Distribution") +
  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)))

```



Plot LogP

```

mean_length = mean(final_sem$mean_len, na.rm = TRUE)
mean_logfreq = mean(final_sem$mean_logf, na.rm = TRUE)
mean_lexdec = mean(final_sem$mean_ldtz, na.rm = TRUE)
mean_concreteness = mean(final_sem$mean_conc, na.rm = TRUE)

fixed.frame <-
  data.frame(expand.grid( newlogp =
    seq(min(final_sem$newlogp, na.rm = TRUE),
        max(final_sem$newlogp, na.rm = TRUE),
        2),
    Type = c("Related", "Unrelated"),
    mean_len = mean_length,
    mean_logf = mean_logfreq,
    mean_ldtz = mean_lexdec,
    mean_conc = mean_concreteness)) %>%
  mutate(pred = predict(m_logp_type, newdata = ., re.form = NA))

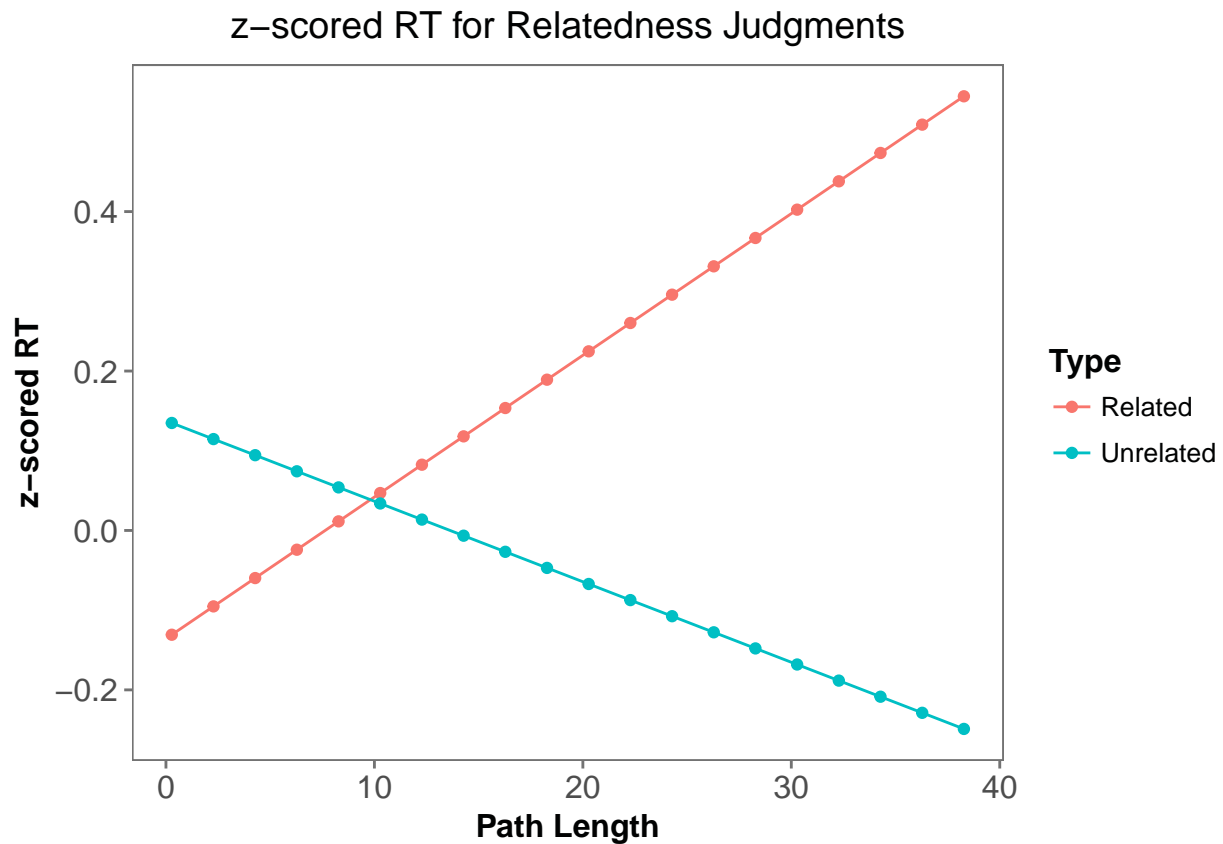
fixed.frame %>%
  # mutate(Pathlength = factor(directedfac,
  #   levels = unique(directedfac),
  #   labels = c("1", "2", "3", "4",

```

```

# "5", "6", "7", "8")))%>%
ggplot(aes(x = newlogp, y = pred, group = Type, color = Type))+
  geom_point()+
  # geom_smooth(method = "loess")+
  geom_line()+
  theme_few()+
  xlab("Path Length") + ylab("z-scored RT") +
  ggtitle("z-scored RT for Relatedness Judgments") +
  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)))

```



z-scores for Path Lengths

```

library(dplyr)
z_pathlength1 = final_sem %>% filter(pathlength == "1")
z_pathlength2 = final_sem %>% filter(pathlength == "2")
z_pathlength3 = final_sem %>% filter(pathlength == "3")
z_pathlength4 = final_sem %>% filter(pathlength == "4")
z_pathlength6 = final_sem %>% filter(pathlength == "6")
z_pathlength15 = final_sem %>% filter(pathlength == "15")

```

Creating separate z-scores

```
sem_firsttrim1 = sem_firsttrim %>% filter(pathlength == "15")

## aggregate per subject all IVs and DVs
meanRT = group_by(sem_firsttrim1, subject) %>%
  summarise_at(vars(rt), mean)
colnames(meanRT) = c("subject", "MeanRT")

sdRT = group_by(sem_firsttrim1, subject) %>%
  summarise_at(vars(rt), sd)
colnames(sdRT) = c("subject", "sdRT")

RT_agg = merge(meanRT, sdRT, by = "subject")

## merge aggregate info with long data
sem_z_1 = merge(sem_firsttrim1, RT_agg, by = "subject", all.x = T)

## person and grand-mean centered scores using original and aggregate
library(dplyr)
sem_z_1 = sem_z_1 %>% mutate(zRT = (rt - MeanRT)/sdRT)

## checking: subject level means should be zero

sub_pic = group_by(sem_z_1, subject) %>%
  summarise_at(vars(zRT), mean)
```

“““