Network Demasking

Reading the Data

```
library(dplyr)
netdemask = read.csv("NetworksDemaskingAllSubjects.csv", header = TRUE, sep = ",")
netdemask = netdemask %>% filter(PrimeAccuracy == "1" & TargetAccuracy == 1)
```

Raw Reaction Time

```
netdemask_rt = group_by(netdemask, subject, pathlength ) %>%
  summarise_at(vars(RTRecognisePrime, RTRecogniseTarget), mean)

netdemask_rt_agg = Rmisc::summarySE(netdemask_rt,
                        measurevar = "RTRecogniseTarget",
                        groupvars = c("pathlength"))
```

ANOVA

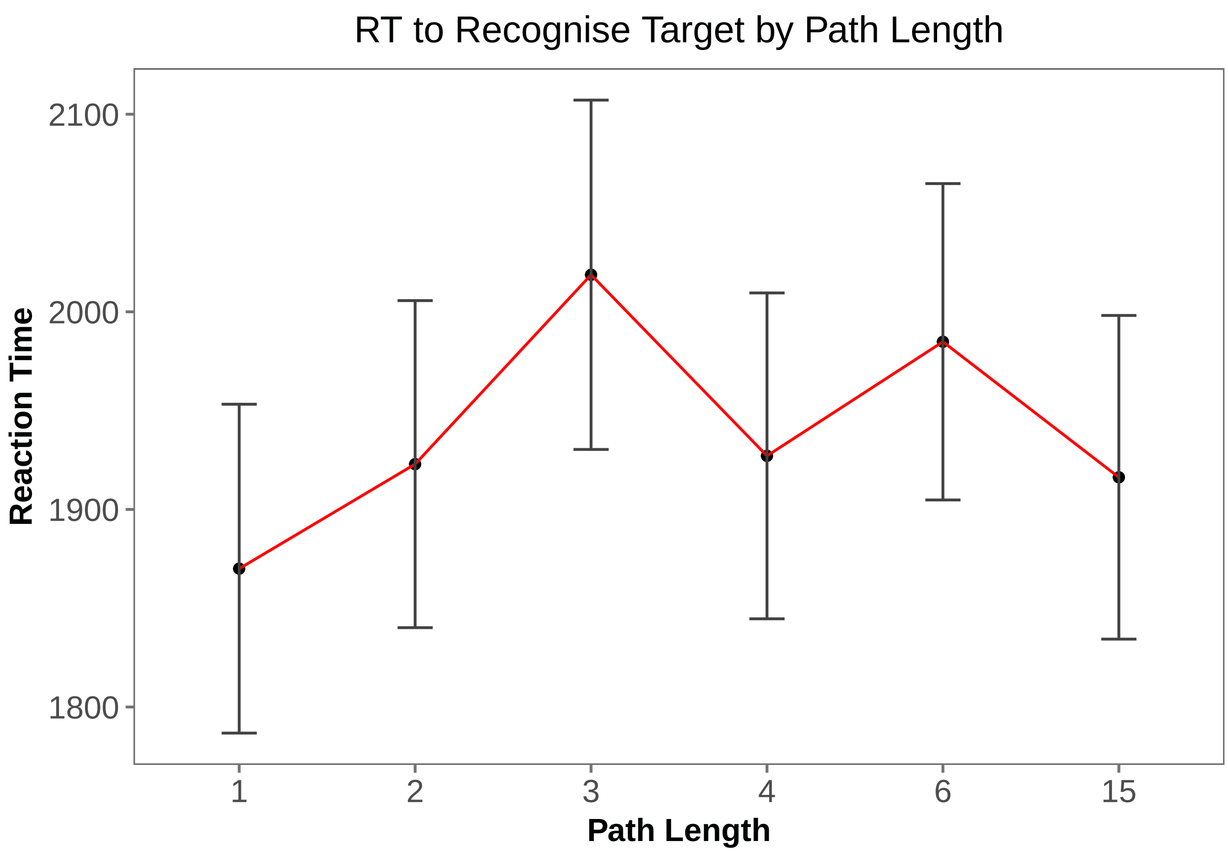
```
netdemask_rt$pathlengthfac = ordered(as.factor(as.character(netdemask_rt$pathlength)),
                            levels = c("1", "2", "3", "4", "6", "15"))
netdemask_rt$subject = as.factor(netdemask_rt$subject)
rt_aov = aov(data = netdemask_rt, RTRecogniseTarget ~ pathlengthfac +
             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

```
##
## Error: subject
##           Df   Sum Sq Mean Sq F value Pr(>F)
## Residuals 37 56497803 1526968
##
## Error: subject:pathlengthfac
##                Df  Sum Sq Mean Sq F value   Pr(>F)
## pathlengthfac   5  537034  107407   10.67 5.08e-09 ***
## Residuals     185 1862716   10069
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Plotting RTs

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

library(ggplot2)
library(ggthemes)

netdemask_rt_agg %>%
```

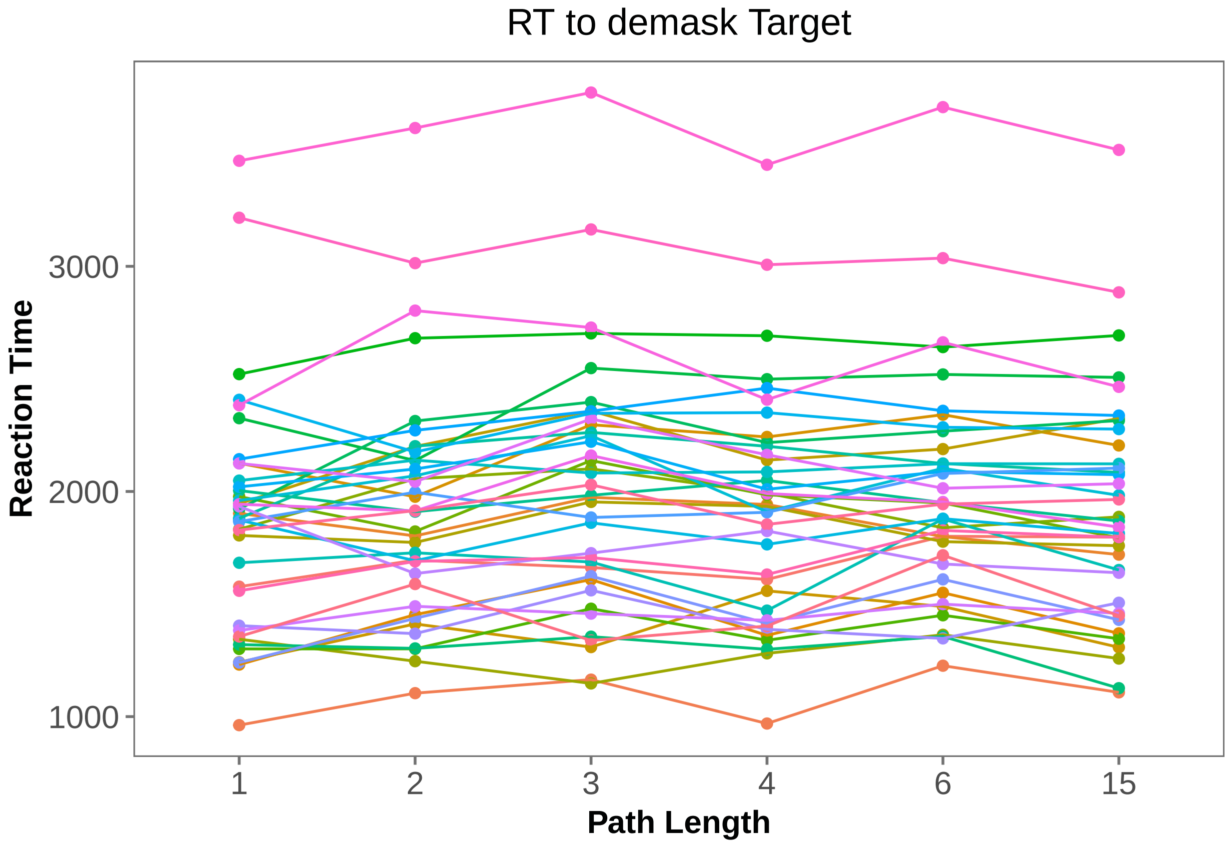
```
  ggplot(aes(x = pathlengthfac, y = RTRecogniseTarget, group = 1))+
  geom_point()+
  geom_line(color = "red")+
   geom_errorbar(aes(ymin=RTRecogniseTarget - se, ymax=RTRecogniseTarget + se),
             width=.2, color = "gray26",
             position = position_dodge(0.7))+
 theme_few()+
 # scale_x_continuous(breaks = c(1,2,3,4,5,6,10,15,20))+
    xlab("Path Length") + ylab("Reaction Time") +
  ggtitle("RT to Recognise Target by Path Length")  +
   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)

netdemask_rt %>%
  ggplot(aes(x = pathlengthfac, y = RTRecogniseTarget,
             group = subject, color = subject))+
  geom_point()+
```

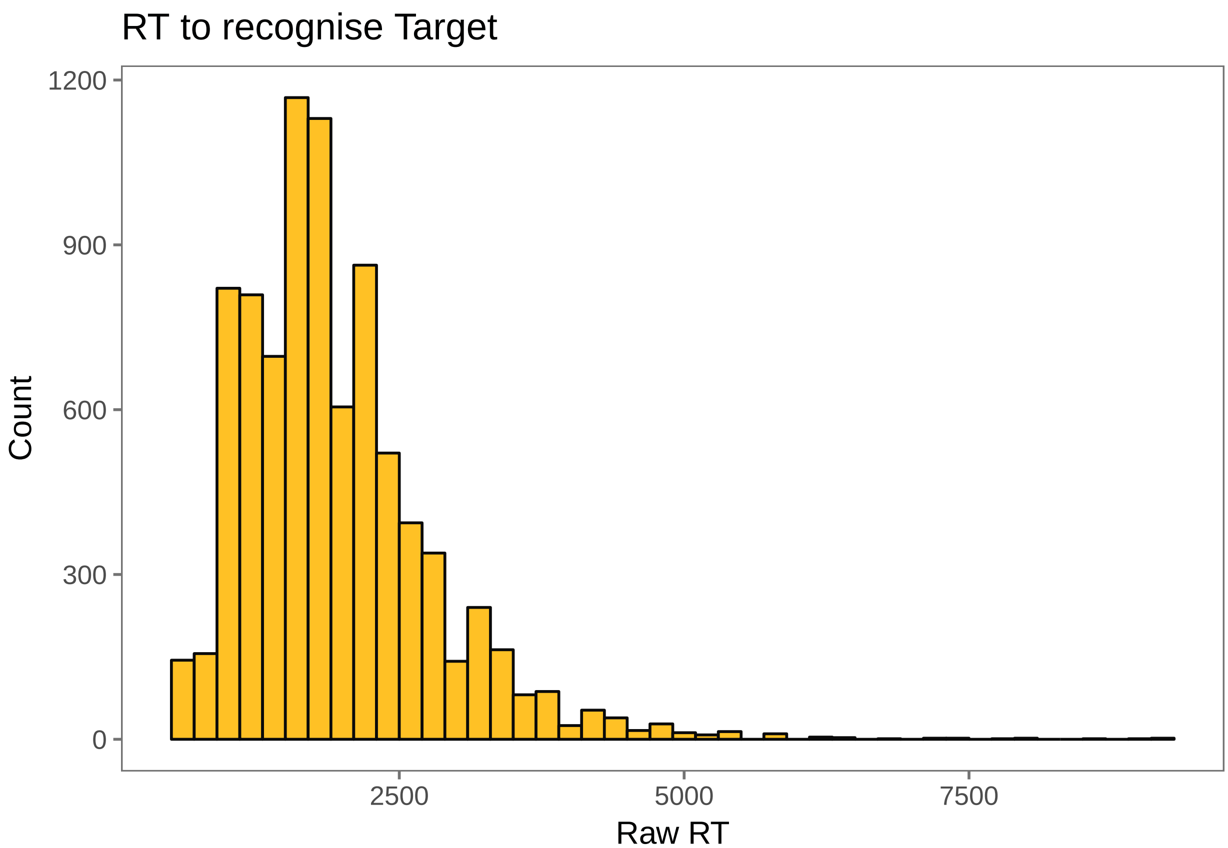
```r
  geom_line()+
 theme_few()+
  guides(color = FALSE)+
 # scale_x_continuous(breaks = c(1,2,3,4,6,15))+
    xlab("Path Length") + ylab("Reaction Time") +
  ggtitle("RT to demask Target")  +
   theme(axis.text = element_text(size = rel(1)),
          axis.title = element_text(face = "bold", size = rel(1)),
          legend.title = element_blank(),
         plot.title = element_text(hjust = .5),
         strip.text.x = element_text(face = "bold", size = rel(1.4)))
```



z-scored Reaction Time

Histogram of RT

```r
library(ggplot2)
library(ggthemes)
ggplot(netdemask, aes(x = RTRecogniseTarget))+
geom_histogram(binwidth = 200, color = "gray4", fill = "goldenrod1")+
    theme_few()+
  #facet_wrap(~subject)+
  xlab("Raw RT") + ylab("Count") +
  ggtitle("RT to recognise Target")
```



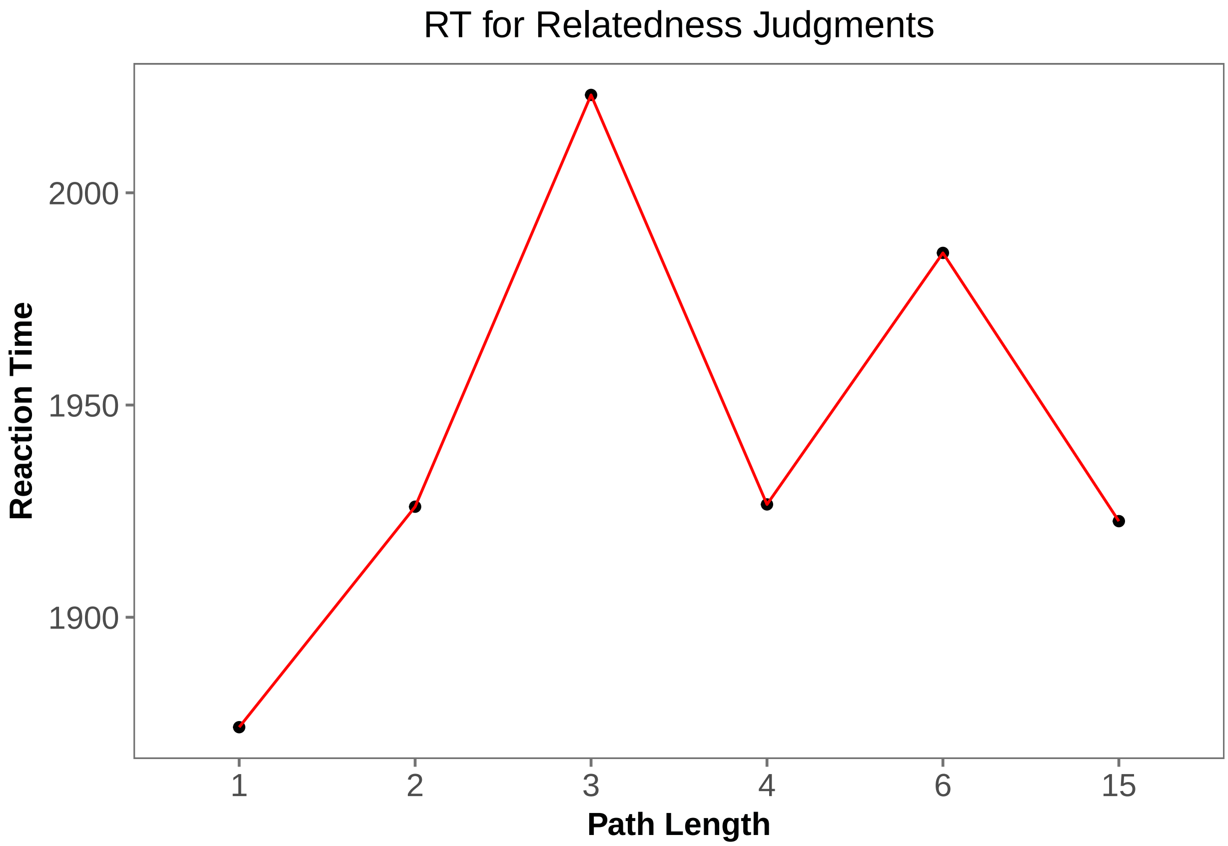
First Trim

```
library(dplyr)
netdemask_firsttrim = netdemask
```

Raw RT aggregates After Trimming

```
netdemask_rt_firsttrim = group_by(netdemask_firsttrim, subject, pathlength ) %>%
  summarise_at(vars(RTRecognisePrime,RTRecogniseTarget), mean)

netdemask_rt_agg_firsttrim = group_by(netdemask_firsttrim, pathlength ) %>%
  summarise_at(vars(RTRecognisePrime,RTRecogniseTarget), mean)

netdemask_rt_agg_firsttrim$pathlengthfac = ordered(as.factor(as.character(netdemask_rt_agg_firsttrim$pa

library(ggplot2)
library(ggthemes)

netdemask_rt_agg_firsttrim %>%
  ggplot(aes(x = pathlengthfac, y = RTRecogniseTarget, group = 1))+
  geom_point()+
  geom_line(color = "red")+
   #geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
```

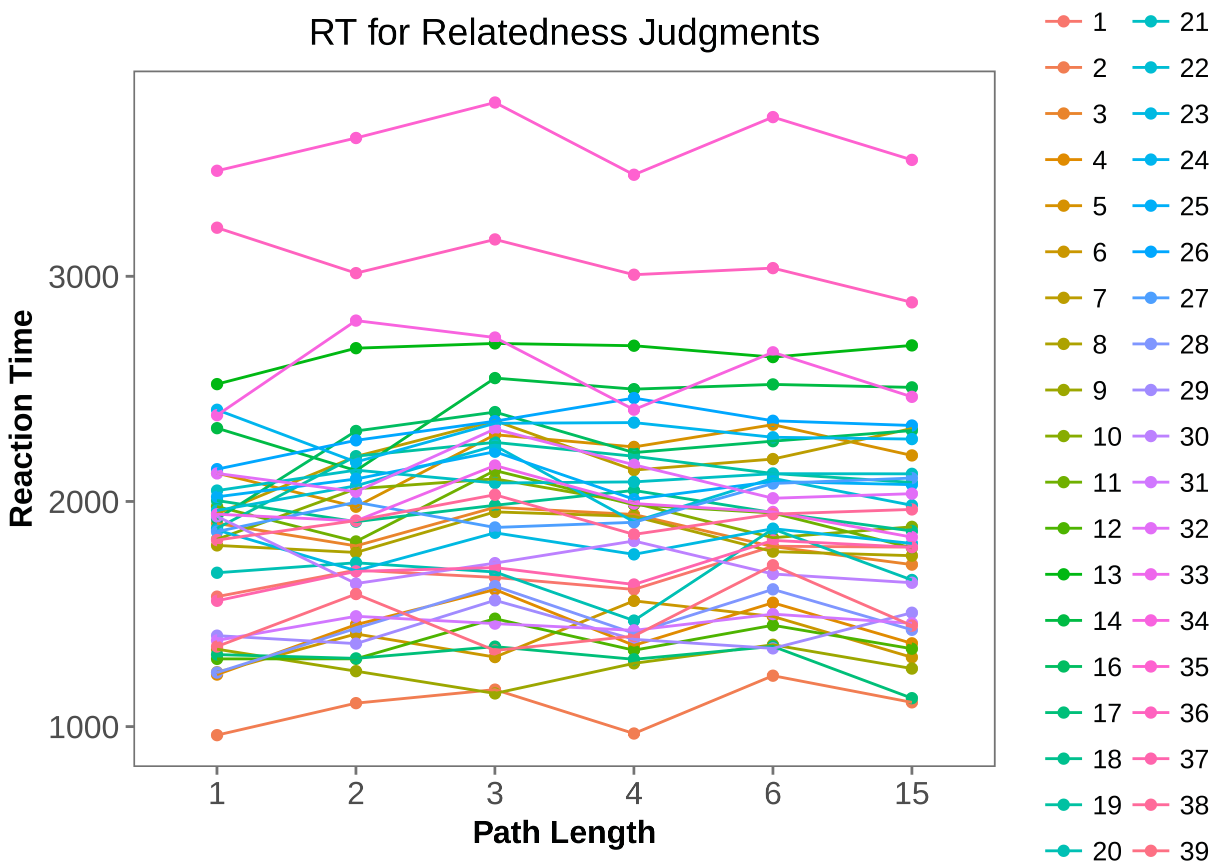
```
  #           width=.2, color = "gray26",
   #          position = position_dodge(0.7))+
 theme_few()+
  #scale_x_continuous(breaks = c(1,2,3,4,6,15))+
    xlab("Path Length") + ylab("Reaction Time") +
  ggtitle("RT for Relatedness Judgments")  +
   theme(axis.text = element_text(size = rel(1)),
          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)

netdemask_rt_firsttrim$pathlengthfac = ordered(as.factor(as.character(netdemask_rt_firsttrim$pathlength
                            levels = c("1", "2", "3", "4", "6", "15"))
netdemask_rt_firsttrim$subject = as.factor(netdemask_rt_firsttrim$subject)
netdemask_rt_firsttrim %>%
  ggplot(aes(x = pathlengthfac, y = RTRecogniseTarget,
             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)))
```



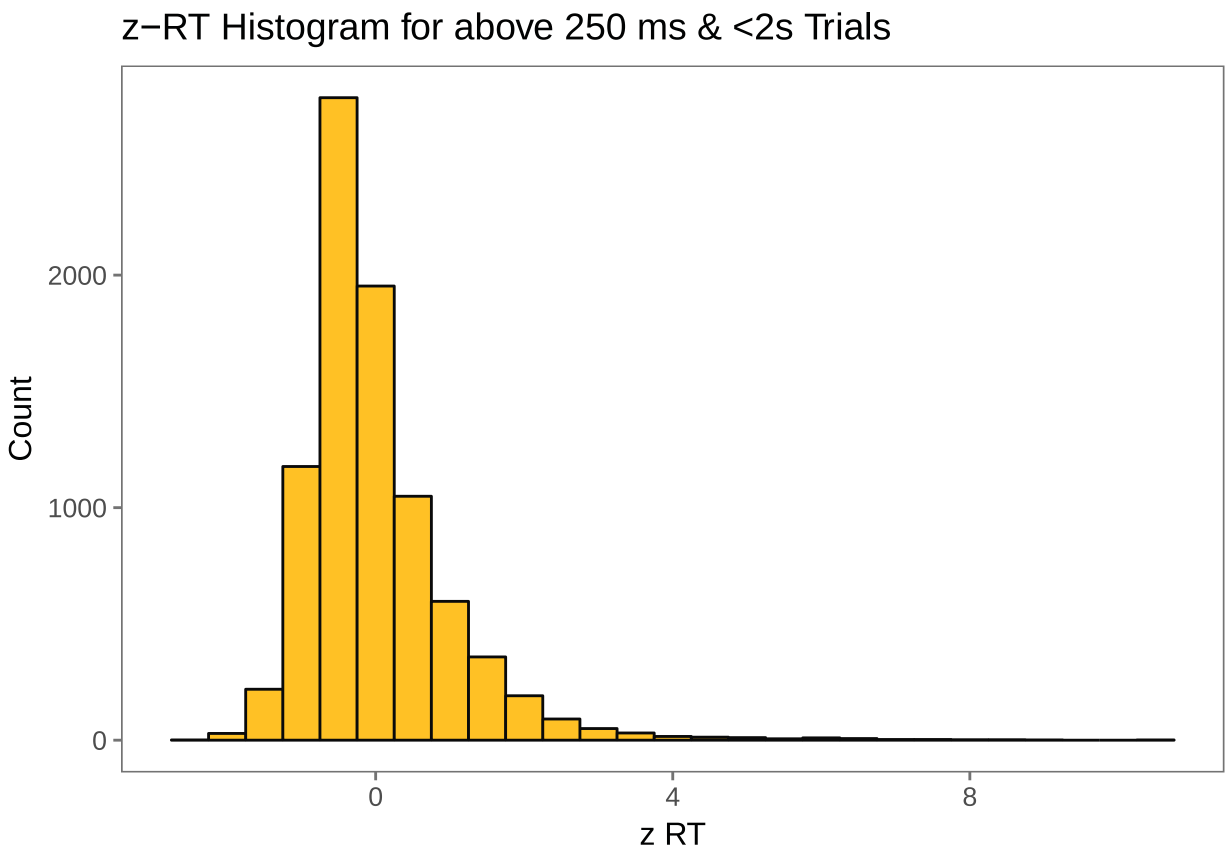
Making the z-scores

```
## aggregate per subject all IVs and DVs
meanRT = group_by(netdemask_firsttrim, subject) %>%
  summarise_at(vars(RTRecognisePrime,RTRecogniseTarget), mean)
colnames(meanRT) = c("subject", "MeanRTPrime", "MeanRTTarget")

sdRT = group_by(netdemask_firsttrim, subject) %>%
  summarise_at(vars(RTRecognisePrime,RTRecogniseTarget), sd)
```

```r
colnames(sdRT) = c("subject", "sdRTPrime", "sdRTTarget")

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

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

## person and grand-mean centered scores using original and aggregate
library(dplyr)
netdemask_z = netdemask_z %>% mutate(zRTTarget =
                            (RTRecogniseTarget - MeanRTTarget)/sdRTTarget,
                        zRTPrime =  (RTRecognisePrime - MeanRTPrime)/sdRTPrime)

## checking: subject level means should be zero

sub_pic = group_by(netdemask_z, subject) %>%
  summarise_at(vars(zRTTarget, zRTPrime), mean)
```

z-RT Distribution

```r
ggplot(netdemask_z, aes(x = zRTPrime))+
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

```
## trimming separately for prime and target
netdemask_z_trimmed_target = subset(netdemask_z, netdemask_z$zRTTarget < 3 &
                                    netdemask_z$zRTTarget > -3)

netdemask_z_trimmed_prime = subset(netdemask_z, netdemask_z$zRTPrime < 3 &
                                   netdemask_z$zRTPrime > -3)
```

Repeating z-scoring

```
library(dplyr)
## FOR TARGET
## aggregate per subject all IVs and DVs
meanRT_trim_target = group_by(netdemask_z_trimmed_target, subject) %>%
  summarise_at(vars(RTRecogniseTarget), mean)
colnames(meanRT_trim_target) = c("subject", "MeanRT_trim_target")

sdRT_trim_target = group_by(netdemask_z_trimmed_target, subject) %>%
  summarise_at(vars(RTRecogniseTarget), sd)
colnames(sdRT_trim_target) = c("subject", "sdRT_trim_target")

RT_agg_trim_target = merge(meanRT_trim_target, sdRT_trim_target, by = "subject")
```

```r
## merge aggregate info with long data
new_netdemask_z_target = merge(netdemask_z_trimmed_target,
                        RT_agg_trim_target, by = "subject", all.x = T)

## person and grand-mean centered scores using original and aggregate
library(dplyr)
new_netdemask_z_target = new_netdemask_z_target %>%
  mutate(zRTTarget_trim = (RTRecogniseTarget - MeanRT_trim_target)/sdRT_trim_target)

## checking: subject level means should be zero

sub_pic = group_by(new_netdemask_z_target, subject) %>%
  summarise_at(vars(zRTTarget_trim), mean)

## FOR PRIME


meanRT_trim_prime = group_by(netdemask_z_trimmed_prime, subject) %>%
  summarise_at(vars(RTRecognisePrime), mean)
colnames(meanRT_trim_prime) = c("subject", "MeanRT_trim_prime")

sdRT_trim_prime = group_by(netdemask_z_trimmed_prime, subject) %>%
  summarise_at(vars(RTRecognisePrime), sd)
colnames(sdRT_trim_prime) = c("subject", "sdRT_trim_prime")

RT_agg_trim_prime = merge(meanRT_trim_prime, sdRT_trim_prime, by = "subject")

## merge aggregate info with long data
new_netdemask_z_prime = merge(netdemask_z_trimmed_prime,
                        RT_agg_trim_prime, by = "subject", all.x = T)

## person and grand-mean centered scores using original and aggregate
library(dplyr)
new_netdemask_z_prime = new_netdemask_z_prime %>%
  mutate(zRTPrime_trim = (RTRecognisePrime - MeanRT_trim_prime)/sdRT_trim_prime)

## checking: subject level means should be zero

sub_pic = group_by(new_netdemask_z_prime, subject) %>%
  summarise_at(vars(zRTPrime_trim), mean)

## now we have separately z-scored RTprime and RTtarget. Need to combine.
## taking only necessary columns
new_netdemask_z_prime = new_netdemask_z_prime[,c(1,5,40)]

new_netdemask_z = merge(new_netdemask_z_target,
                             new_netdemask_z_prime,
                             by  = c("subject", "Trial"))
```

Aggregating zRT

```
z_netdemask_rt = group_by(new_netdemask_z, subject, pathlength ) %>%
  summarise_at(vars(zRTTarget_trim, zRTPrime_trim), mean)

z_rmisc = Rmisc::summarySE(new_netdemask_z,
                        measurevar = "zRTTarget_trim",
                        groupvars = c("pathlength"))
```

ANOVA

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

z_rt_aov = aov(data = z_netdemask_rt, zRTTarget_trim ~ pathlengthfac +
                     Error(subject/(pathlengthfac)))
summary(z_rt_aov)
```

```
##
## Error: subject
##           Df   Sum Sq   Mean Sq F value Pr(>F)
## Residuals 37 0.02236 0.0006043
##
## Error: subject:pathlengthfac
##                Df Sum Sq Mean Sq F value   Pr(>F)
## pathlengthfac   5  1.369 0.27379   8.572 2.54e-07 ***
## Residuals     185  5.909 0.03194
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

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

```
## The 'lsmeans' package is being deprecated.
## Users are encouraged to switch to 'emmeans'.
## See help('transition') for more information, including how
## to convert 'lsmeans' objects and scripts to work with 'emmeans'.
```

```
 library(multcomp)
```

```
## Loading required package: mvtnorm

## Loading required package: survival

## Loading required package: TH.data

## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

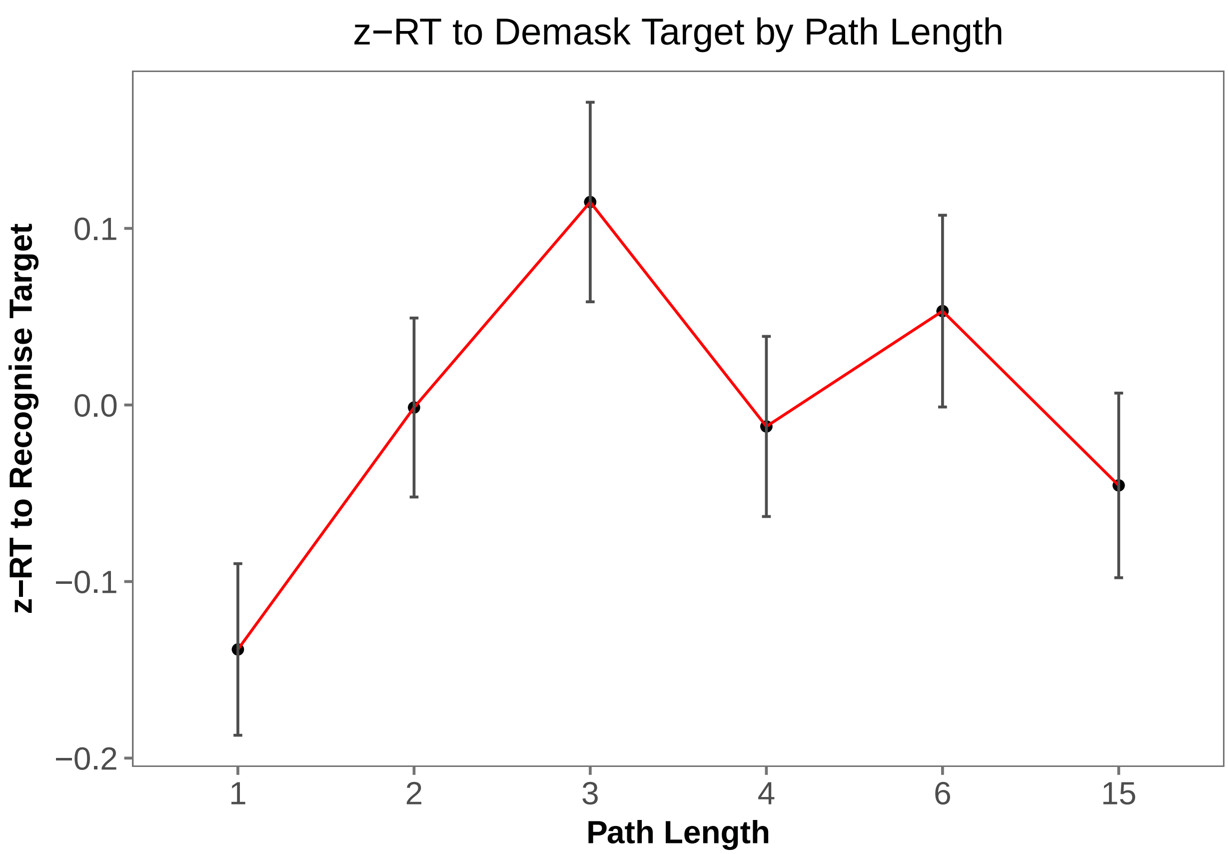
```
##
## Attaching package: 'TH.data'

## The following object is masked from 'package:MASS':
##
##     geyser
```

```
sem_lsm = lsmeans::lsmeans(z_rt_aov, c("pathlengthfac"))
prime_effect = cld(sem_lsm, alpha = 0.05,
                adjust = "tukey", details = TRUE)
library(knitr)
kable(subset(prime_effect$comparisons,prime_effect$comparisons$p.value < 0.1 ))
```

	contrast	estimate	SE	df	t.ratio	p.value
2	4 - 1	0.1213892	0.041001	185	2.960640	0.0399937
4	2 - 1	0.1321314	0.041001	185	3.222640	0.0184907
7	6 - 1	0.1865946	0.041001	185	4.550979	0.0001393
11	3 - 1	0.2493361	0.041001	185	6.081223	0.0000001
12	3 - 15	0.1593669	0.041001	185	3.886905	0.0019384
13	3 - 4	0.1279469	0.041001	185	3.120582	0.0251847
14	3 - 2	0.1172047	0.041001	185	2.858582	0.0529602

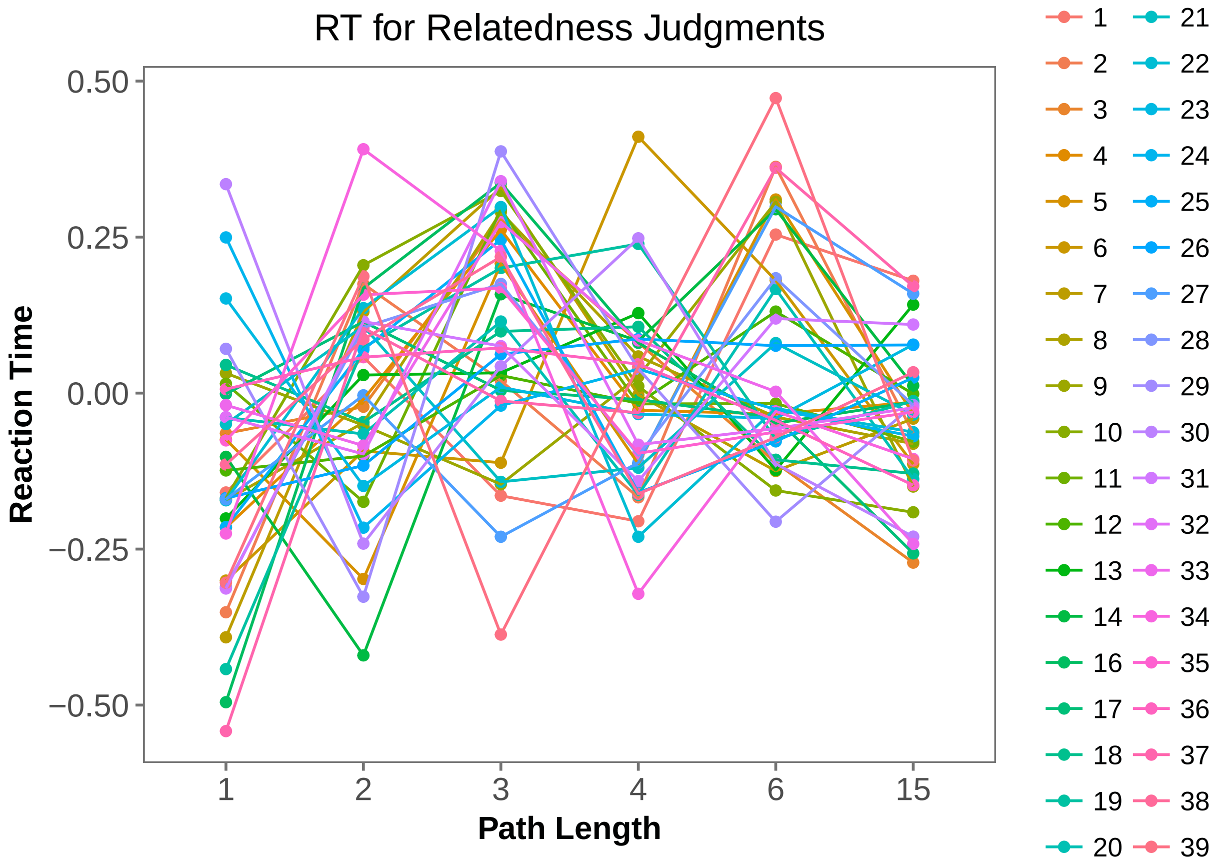
Plotting RTs: collapsed

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

z_rmisc$zRTTarget_trim = as.numeric(z_rmisc$zRTTarget_trim)

library(ggplot2)
library(ggthemes)

z_rmisc %>%
  ggplot(aes(x = pathlengthfac, y = zRTTarget_trim, group = 1))+
  geom_point()+
#  geom_smooth(method = "loess")+
geom_line(color = "red")+
   geom_errorbar(aes(ymin=zRTTarget_trim - ci, ymax=zRTTarget_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 to Recognise Target") +
  ggtitle("z-RT to Demask Target by Path Length")  +
   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 z RT

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


z_netdemask_rt %>%
  ggplot(aes(x = pathlengthfac, y = zRTTarget_trim,
             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)))
```

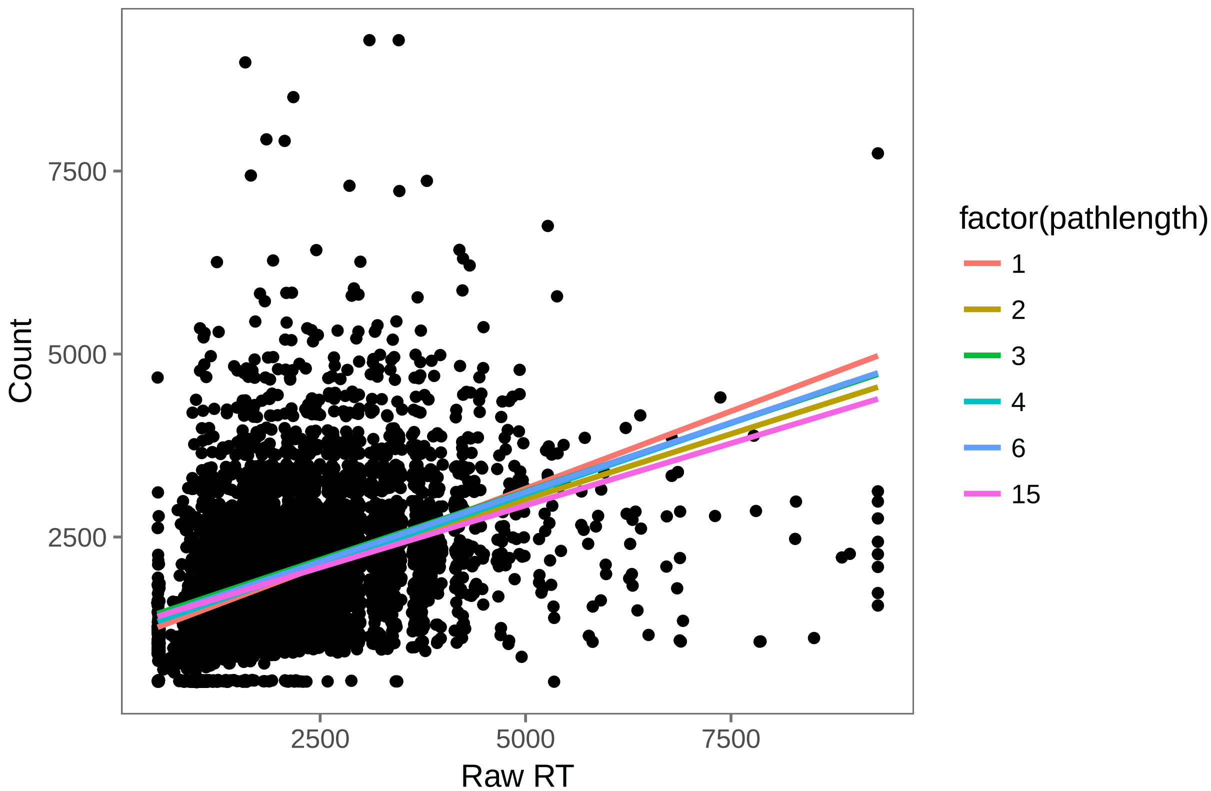


Effect of Prime on Target

Simple Scatter Plot

```
ggplot(netdemask, aes(x = RTRecognisePrime, y = RTRecogniseTarget))+
geom_point()+
  geom_smooth(method = "lm", aes(group = factor(pathlength),
                                 color = factor(pathlength)), se = FALSE)+
  theme_few()+
  #facet_wrap(~subject)+
  xlab("Raw RT") + ylab("Count") +
  ggtitle("Raw RT Histogram for All Trials")
```

Raw RT Histogram for All Trials



Linear Models

```r
library(lme4)
```

```
## Loading required package: Matrix
```

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


RTprime_model = lmer(data = new_netdemask_z,
                          zRTTarget_trim ~ zRTPrime_trim +
                            (1|subject) + (1|ItemNumber))
summary(RTprime_model)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRTTarget_trim ~ zRTPrime_trim + (1 | subject) + (1 | ItemNumber)
##    Data: new_netdemask_z
##
## REML criterion at convergence: 22430.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0479 -0.6309 -0.1472  0.5010  5.2827
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
```

```
##  ItemNumber (Intercept) 0.2108   0.4591
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7654   0.8749
## Number of obs: 8323, groups:  ItemNumber, 720; subject, 38
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)   0.007849   0.019680   0.399
## zRTPrime_trim 0.173257   0.010828  16.001
##
## Correlation of Fixed Effects:
##             (Intr)
## zRTPrim_trm -0.007
```

```
contrasts(new_netdemask_z$pathlengthfac) = contr.treatment(6, base = 3)
RTprime_model_2 = lmer(data = new_netdemask_z,
                          zRTTarget_trim ~ zRTPrime_trim + pathlengthfac +
                            (1|subject) + (1|ItemNumber))
summary(RTprime_model_2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRTTarget_trim ~ zRTPrime_trim + pathlengthfac + (1 | subject) +
##     (1 | ItemNumber)
##    Data: new_netdemask_z
##
## REML criterion at convergence: 22434.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0603 -0.6306 -0.1479  0.4991  5.2526
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.2069   0.4548
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7655   0.8749
## Number of obs: 8323, groups:  ItemNumber, 720; subject, 38
##
## Fixed effects:
##                Estimate Std. Error t value
## (Intercept)     0.12588    0.04808   2.618
## zRTPrime_trim   0.17265    0.01082  15.950
## pathlengthfac1 -0.23856    0.06778  -3.520
## pathlengthfac2 -0.11869    0.06768  -1.754
## pathlengthfac4 -0.13214    0.06793  -1.945
## pathlengthfac5 -0.05716    0.06778  -0.843
## pathlengthfac6 -0.16096    0.06790  -2.371
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ pthln1 pthln2 pthln4 pthln5
## zRTPrim_trm -0.004
## pthlngthfc1 -0.709  0.010
## pthlngthfc2 -0.710  0.008  0.504
## pthlngthfc4 -0.708 -0.007  0.502  0.503
## pthlngthfc5 -0.709 -0.006  0.503  0.504  0.502
```

```
## pthlngthfc6 -0.708 -0.001  0.502  0.503  0.501  0.502
```

```r
car::Anova(RTprime_model_2)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                  Chisq Df Pr(>Chisq)
## zRTPrime_trim 254.418  1    < 2e-16 ***
## pathlengthfac  14.905  5    0.01078 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
## quadratic trend

new_netdemask_z$pquad = (new_netdemask_z$pathlength)^2

RTprime_model_quad = lmer(data = new_netdemask_z,
                          zRTTarget_trim ~ zRTPrime_trim + pathlength +
                            pquad+ MeanLDTZ + MeanLength + MeanLogF +
                            (1|subject) + (1|ItemNumber))
summary(RTprime_model_quad)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRTTarget_trim ~ zRTPrime_trim + pathlength + pquad + MeanLDTZ +
##     MeanLength + MeanLogF + (1 | subject) + (1 | ItemNumber)
##    Data: new_netdemask_z
##
## REML criterion at convergence: 22321.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0256 -0.6266 -0.1503  0.4992  5.3305
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.1733   0.4162
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7652   0.8748
## Number of obs: 8310, groups:  ItemNumber, 719; subject, 38
##
## Fixed effects:
##                Estimate Std. Error t value
## (Intercept)    0.102400   0.172804   0.593
## zRTPrime_trim  0.156301   0.010862  14.390
## pathlength     0.044221   0.019221   2.301
## pquad         -0.002761   0.001126  -2.453
## MeanLDTZ       0.632505   0.140117   4.514
## MeanLength     0.061054   0.015160   4.027
## MeanLogF      -0.029441   0.016332  -1.803
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ pthlng pquad  MnLDTZ MnLngt
## zRTPrim_trm -0.025
## pathlength  -0.301 -0.015
```

```
## pquad        0.276  0.014 -0.979
## MeanLDTZ     0.210 -0.065 -0.040  0.031
## MeanLength  -0.639 -0.033  0.022 -0.016 -0.453
## MeanLogF    -0.590  0.025  0.011 -0.011  0.512 -0.080
```

```
car::Anova(RTprime_model_quad)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                   Chisq Df Pr(>Chisq)
## zRTPrime_trim 207.0671  1  < 2.2e-16 ***
## pathlength      5.2933  1    0.02141 *
## pquad           6.0166  1    0.01417 *
## MeanLDTZ       20.3772  1  6.358e-06 ***
## MeanLength     16.2203  1  5.639e-05 ***
## MeanLogF        3.2497  1    0.07144 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RTprime_model_2_2 = lmer(data = new_netdemask_z,
                          zRTTarget_trim ~ zRTPrime_trim*pathlengthfac +
                            (1|subject) + (1|ItemNumber))
summary(RTprime_model_2_2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRTTarget_trim ~ zRTPrime_trim * pathlengthfac + (1 | subject) +
##     (1 | ItemNumber)
##    Data: new_netdemask_z
##
## REML criterion at convergence: 22454.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0781 -0.6277 -0.1454  0.4987  5.2407
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.2072   0.4552
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7653   0.8748
## Number of obs: 8323, groups:  ItemNumber, 720; subject, 38
##
## Fixed effects:
##                              Estimate Std. Error t value
## (Intercept)                   0.12511    0.04811   2.600
## zRTPrime_trim                 0.21763    0.02658   8.188
## pathlengthfac1               -0.23875    0.06783  -3.520
## pathlengthfac2               -0.11762    0.06772  -1.737
## pathlengthfac4               -0.12926    0.06799  -1.901
## pathlengthfac5               -0.05676    0.06783  -0.837
## pathlengthfac6               -0.16002    0.06795  -2.355
## zRTPrime_trim:pathlengthfac1 -0.06657    0.03784  -1.759
## zRTPrime_trim:pathlengthfac2 -0.03605    0.03779  -0.954
## zRTPrime_trim:pathlengthfac4 -0.07990    0.03785  -2.111
```

```
## zRTPrime_trim:pathlengthfac5 -0.03750    0.03694  -1.015
## zRTPrime_trim:pathlengthfac6 -0.05144    0.03729  -1.379
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ pthln1 pthln2 pthln4 pthln5 pthln6 zRTP_:1
## zRTPrim_trm -0.010
## pthlngthfc1 -0.709  0.007
## pthlngthfc2 -0.710  0.007  0.504
## pthlngthfc4 -0.708  0.007  0.502  0.503
## pthlngthfc5 -0.709  0.007  0.503  0.504  0.502
## pthlngthfc6 -0.708  0.007  0.502  0.503  0.501  0.502
## zRTPrm_tr:1  0.007 -0.702  0.008 -0.005 -0.005 -0.005 -0.005
## zRTPrm_tr:2  0.007 -0.703 -0.005  0.005 -0.005 -0.005 -0.005  0.494
## zRTPrm_tr:4  0.007 -0.702 -0.005 -0.005 -0.022 -0.005 -0.005  0.493
## zRTPrm_tr:5  0.007 -0.719 -0.005 -0.005 -0.005 -0.019 -0.005  0.505
## zRTPrm_tr:6  0.007 -0.713 -0.005 -0.005 -0.005 -0.005 -0.011  0.501
##             zRTP_:2 zRTP_:4 zRTP_:5
## zRTPrim_trm
## pthlngthfc1
## pthlngthfc2
## pthlngthfc4
## pthlngthfc5
## pthlngthfc6
## zRTPrm_tr:1
## zRTPrm_tr:2
## zRTPrm_tr:4  0.494
## zRTPrm_tr:5  0.506   0.505
## zRTPrm_tr:6  0.501   0.500   0.513
```

```
car::Anova(RTprime_model_2_2)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                                 Chisq Df Pr(>Chisq)
## zRTPrime_trim                254.4804  1    < 2e-16 ***
## pathlengthfac                 14.8850  5    0.01086 *
## zRTPrime_trim:pathlengthfac    5.4421  5    0.36434
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(RTprime_model_2, RTprime_model_2_2) ## no difference interaction not reqd
```

```
## refitting model(s) with ML (instead of REML)

## Data: new_netdemask_z
## Models:
## RTprime_model_2: zRTTarget_trim ~ zRTPrime_trim + pathlengthfac + (1 | subject) +
## RTprime_model_2:     (1 | ItemNumber)
## RTprime_model_2_2: zRTTarget_trim ~ zRTPrime_trim * pathlengthfac + (1 | subject) +
## RTprime_model_2_2:     (1 | ItemNumber)
##                   Df   AIC   BIC logLik deviance  Chisq Chi Df Pr(>Chisq)
## RTprime_model_2   10 22422 22492 -11201    22402
## RTprime_model_2_2 15 22427 22532 -11198    22397 5.4394      5     0.3646
```

```r
 ## centering so that contrasts are easier
new_netdemask_z$mean_len_c = scale(new_netdemask_z$MeanLength,
                             center = TRUE, scale = FALSE)
new_netdemask_z$mean_logf_c = scale(new_netdemask_z$MeanLogF,
                              center = TRUE, scale = FALSE)
new_netdemask_z$mean_ldtz_c = scale(new_netdemask_z$MeanLDTZ,
                              center = TRUE, scale = FALSE)

RTprime_model_2_3 = lmer(data = new_netdemask_z,
                          zRTTarget_trim ~ zRTPrime_trim + pathlengthfac +
                         mean_len_c + mean_logf_c + mean_ldtz_c +
                            (1|subject) + (1|ItemNumber))
summary(RTprime_model_2_3)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRTTarget_trim ~ zRTPrime_trim + pathlengthfac + mean_len_c +
##     mean_logf_c + mean_ldtz_c + (1 | subject) + (1 | ItemNumber)
##    Data: new_netdemask_z
##
## REML criterion at convergence: 22311.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0215 -0.6282 -0.1481  0.4983  5.3014
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.1710   0.4135
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7653   0.8748
## Number of obs: 8310, groups:  ItemNumber, 719; subject, 38
##
## Fixed effects:
##                Estimate Std. Error t value
## (Intercept)     0.11560    0.04489   2.575
## zRTPrime_trim   0.15612    0.01086  14.378
## pathlengthfac1 -0.21757    0.06340  -3.431
## pathlengthfac2 -0.09702    0.06324  -1.534
## pathlengthfac4 -0.14258    0.06356  -2.243
## pathlengthfac5 -0.05217    0.06325  -0.825
## pathlengthfac6 -0.16183    0.06333  -2.555
## mean_len_c      0.06057    0.01511   4.009
## mean_logf_c    -0.03348    0.01635  -2.048
## mean_ldtz_c     0.61414    0.13984   4.392
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ pthln1 pthln2 pthln4 pthln5 pthln6 mn_ln_ mn_lg_
## zRTPrim_trm -0.001
## pthlngthfc1 -0.710  0.006
## pthlngthfc2 -0.709  0.003  0.503
## pthlngthfc4 -0.706 -0.004  0.501  0.498
## pthlngthfc5 -0.710 -0.008  0.505  0.503  0.502
## pthlngthfc6 -0.708 -0.001  0.502  0.501  0.501  0.503
## mean_len_c   0.013 -0.033 -0.017 -0.040  0.013 -0.012  0.008
```

```
## mean_logf_c -0.030  0.025  0.061 -0.030  0.048  0.039  0.020 -0.077
## mean_ldtz_c -0.045 -0.065  0.077  0.037  0.012  0.040  0.006 -0.452  0.512
```

```
car::Anova(RTprime_model_2_3)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                   Chisq Df Pr(>Chisq)
## zRTPrime_trim 206.7397  1  < 2.2e-16 ***
## pathlengthfac  15.4245  5   0.008694 **
## mean_len_c     16.0742  1  6.091e-05 ***
## mean_logf_c     4.1955  1   0.040531 *
## mean_ldtz_c    19.2869  1  1.125e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RTprime_model_3 = lmer(data = new_netdemask_z,
                          zRTTarget_trim ~ pathlengthfac +
                            (1|subject) + (1|ItemNumber))
summary(RTprime_model_3)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRTTarget_trim ~ pathlengthfac + (1 | subject) + (1 | ItemNumber)
##    Data: new_netdemask_z
##
## REML criterion at convergence: 22677.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.8433 -0.6358 -0.1504  0.5130  5.2068
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.1997   0.4469
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7922   0.8901
## Number of obs: 8323, groups:  ItemNumber, 720; subject, 38
##
## Fixed effects:
##                 Estimate Std. Error t value
## (Intercept)      0.12853    0.04766   2.697
## pathlengthfac1 -0.24941    0.06718  -3.713
## pathlengthfac2 -0.12712    0.06707  -1.895
## pathlengthfac4 -0.12475    0.06734  -1.853
## pathlengthfac5 -0.05137    0.06718  -0.765
## pathlengthfac6 -0.15982    0.06731  -2.374
##
## Correlation of Fixed Effects:
##             (Intr) pthln1 pthln2 pthln4 pthln5
## pthlngthfc1 -0.710
## pthlngthfc2 -0.711  0.504
## pthlngthfc4 -0.708  0.502  0.503
## pthlngthfc5 -0.709  0.503  0.504  0.502
## pthlngthfc6 -0.708  0.502  0.503  0.501  0.502
```

```r
car::Anova(RTprime_model_3)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                 Chisq Df Pr(>Chisq)
## pathlengthfac 16.645  5   0.005224 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(RTprime_model_3, RTprime_model_2)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: new_netdemask_z
## Models:
## RTprime_model_3: zRTTarget_trim ~ pathlengthfac + (1 | subject) + (1 | ItemNumber)
## RTprime_model_2: zRTTarget_trim ~ zRTPrime_trim + pathlengthfac + (1 | subject) +
## RTprime_model_2:     (1 | ItemNumber)
##                 Df   AIC   BIC logLik deviance  Chisq Chi Df Pr(>Chisq)
## RTprime_model_3  9 22670 22733 -11326    22652
## RTprime_model_2 10 22422 22492 -11201    22402 249.69      1  < 2.2e-16
##
## RTprime_model_3
## RTprime_model_2 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(RTprime_model, RTprime_model_2)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: new_netdemask_z
## Models:
## RTprime_model: zRTTarget_trim ~ zRTPrime_trim + (1 | subject) + (1 | ItemNumber)
## RTprime_model_2: zRTTarget_trim ~ zRTPrime_trim + pathlengthfac + (1 | subject) +
## RTprime_model_2:     (1 | ItemNumber)
##                 Df   AIC   BIC logLik deviance  Chisq Chi Df Pr(>Chisq)
## RTprime_model    5 22427 22462 -11209    22417
## RTprime_model_2 10 22422 22492 -11201    22402 14.875      5    0.01091 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Contrasts

```r
groups <- read.table('groupsdemasking.csv',
                     sep=',',header=TRUE,stringsAsFactors=FALSE)
groups
```

```
##   Group pathlength pathlengthfac1 pathlengthfac2 pathlengthfac4
## 1     1          1              1              0              0
## 2     2          2              0              1              0
## 3     3          3              0              0              0
## 4     4          4              0              0              1
## 5     5          6              0              0              0
```

```
## 6     6         15               0              0              0
##   pathlengthfac5 pathlengthfac6
## 1              0              0
## 2              0              0
## 3              0              0
## 4              0              0
## 5              1              0
## 6              0              1
```

```
dummy_codes <- as.matrix(groups[,3:7])
dummy_codes
```

```
##      pathlengthfac1 pathlengthfac2 pathlengthfac4 pathlengthfac5
## [1,]              1              0              0              0
## [2,]              0              1              0              0
## [3,]              0              0              0              0
## [4,]              0              0              1              0
## [5,]              0              0              0              1
## [6,]              0              0              0              0
##      pathlengthfac6
## [1,]              0
## [2,]              0
## [3,]              0
## [4,]              0
## [5,]              0
## [6,]              1
```

```
fixed_effects <- matrix(fixef(RTprime_model_2))
fixed_effects
```

```
##             [,1]
## [1,]  0.12588075
## [2,]  0.17264901
## [3,] -0.23855777
## [4,] -0.11869046
## [5,] -0.13214181
## [6,] -0.05716032
## [7,] -0.16096061
```

```
means_matrix <- matrix(rep(0,42),ncol=7,nrow=6)
means_matrix[,1] <- 1
means_matrix[,2] <- 0
means_matrix[,3:7] <- dummy_codes[,1:5]
means_matrix
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7]
## [1,]    1    0    1    0    0    0    0
## [2,]    1    0    0    1    0    0    0
## [3,]    1    0    0    0    0    0    0
## [4,]    1    0    0    0    1    0    0
## [5,]    1    0    0    0    0    1    0
## [6,]    1    0    0    0    0    0    1
```

```
means <- means_matrix %*% fixed_effects
print(cbind(means,groups[,2]))
```

```
##             [,1] [,2]
```

```
## [1,] -0.112677021    1
## [2,]  0.007190288    2
## [3,]  0.125880750    3
## [4,] -0.006261064    4
## [5,]  0.068720433    6
## [6,] -0.035079864   15
```

```r
contrast_matrix <- matrix(c(
  1,-1,0,0,0,0,
  1,0,-1,0,0,0,
  0,1,-1,0,0,0,
  0,0,1,-1,0,0,
  0,0,0,1,-1,0,
  0,0,0,0,1,-1,
  0,0,1,0,-1,0,
  0,0,1,0,0,-1), nrow=8,ncol=6,byrow=TRUE)
row.names(contrast_matrix) <- c("path 1 vs. path 2 ",
                                "path 1 vs. path 3 ",
                                "path 2 vs. path 3",
                                "path 3 vs. path 4",
                                "path 4 vs. path 6",
                                "path 6 vs. path 15",
                                "path 3 vs. path 6 ",
                                "path 3 vs. path 15  ")

matrix_for_glht <-contrast_matrix %*%  means_matrix
matrix_for_glht
```

```
##                       [,1] [,2] [,3] [,4] [,5] [,6] [,7]
## path 1 vs. path 2        0    0    1   -1    0    0    0
## path 1 vs. path 3        0    0    1    0    0    0    0
## path 2 vs. path 3        0    0    0    1    0    0    0
## path 3 vs. path 4        0    0    0    0   -1    0    0
## path 4 vs. path 6        0    0    0    0    1   -1    0
## path 6 vs. path 15       0    0    0    0    0    1   -1
## path 3 vs. path 6        0    0    0    0    0   -1    0
## path 3 vs. path 15       0    0    0    0    0    0   -1
```

```r
matrix_for_glht <-contrast_matrix %*%  means_matrix
matrix_for_glht
```

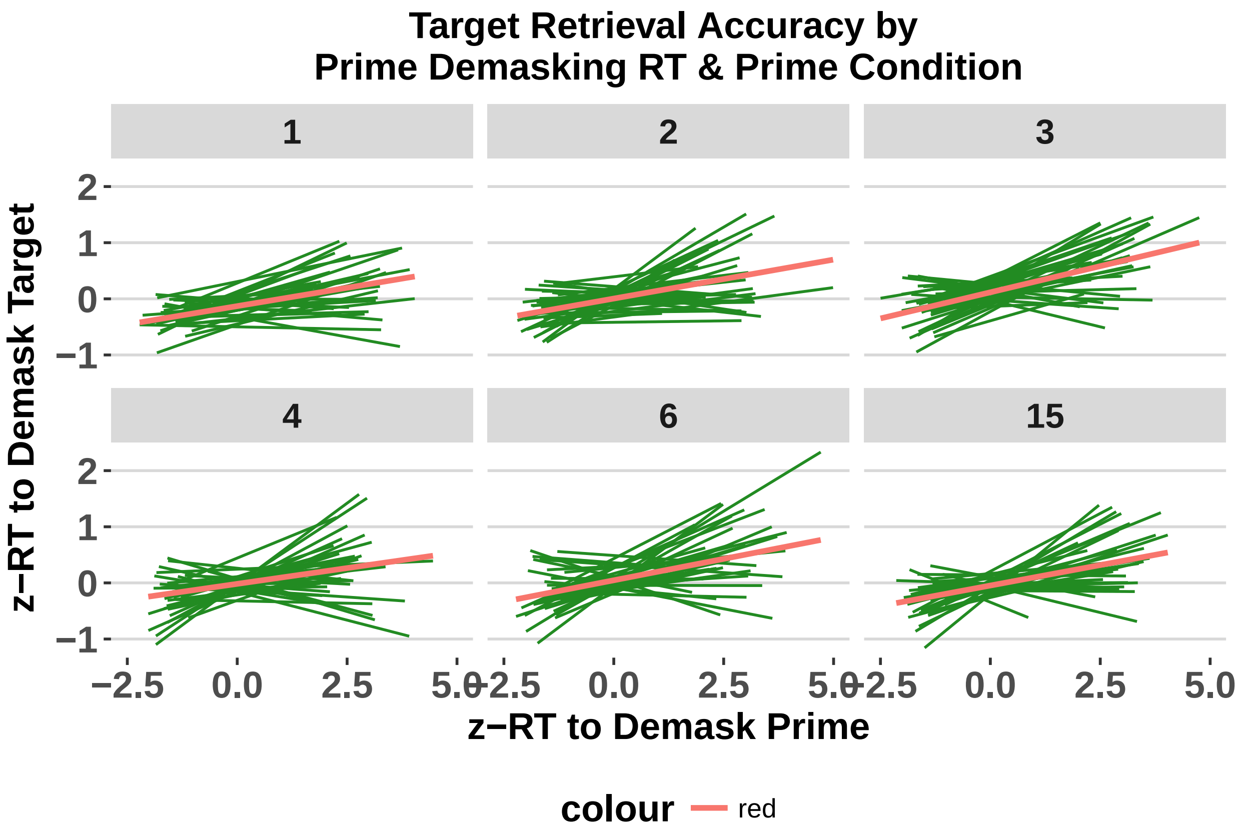
```
##                       [,1] [,2] [,3] [,4] [,5] [,6] [,7]
## path 1 vs. path 2        0    0    1   -1    0    0    0
## path 1 vs. path 3        0    0    1    0    0    0    0
## path 2 vs. path 3        0    0    0    1    0    0    0
## path 3 vs. path 4        0    0    0    0   -1    0    0
## path 4 vs. path 6        0    0    0    0    1   -1    0
## path 6 vs. path 15       0    0    0    0    0    1   -1
## path 3 vs. path 6        0    0    0    0    0   -1    0
## path 3 vs. path 15       0    0    0    0    0    0   -1
```

```r
glht_sem <- multcomp::glht(RTprime_model_2,
                           linfct = matrix_for_glht,
              alternative = "two.sided", rhs = 0)
summary(glht_sem)
```

```
##
##   Simultaneous Tests for General Linear Hypotheses
##
## Fit: lmer(formula = zRTTarget_trim ~ zRTPrime_trim + pathlengthfac +
##     (1 | subject) + (1 | ItemNumber), data = new_netdemask_z)
##
## Linear Hypotheses:
##                            Estimate Std. Error z value Pr(>|z|)
## path 1 vs. path 2  == 0   -0.11987    0.06745  -1.777  0.36529
## path 1 vs. path 3  == 0   -0.23856    0.06778  -3.520  0.00367 **
## path 2 vs. path 3 == 0    -0.11869    0.06768  -1.754  0.37929
## path 3 vs. path 4 == 0     0.13214    0.06793   1.945  0.27146
## path 4 vs. path 6 == 0    -0.07498    0.06771  -1.107  0.80471
## path 6 vs. path 15 == 0    0.10380    0.06768   1.534  0.52377
## path 3 vs. path 6  == 0    0.05716    0.06778   0.843  0.92574
## path 3 vs. path 15   == 0  0.16096    0.06790   2.371  0.10905
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- single-step method)
```

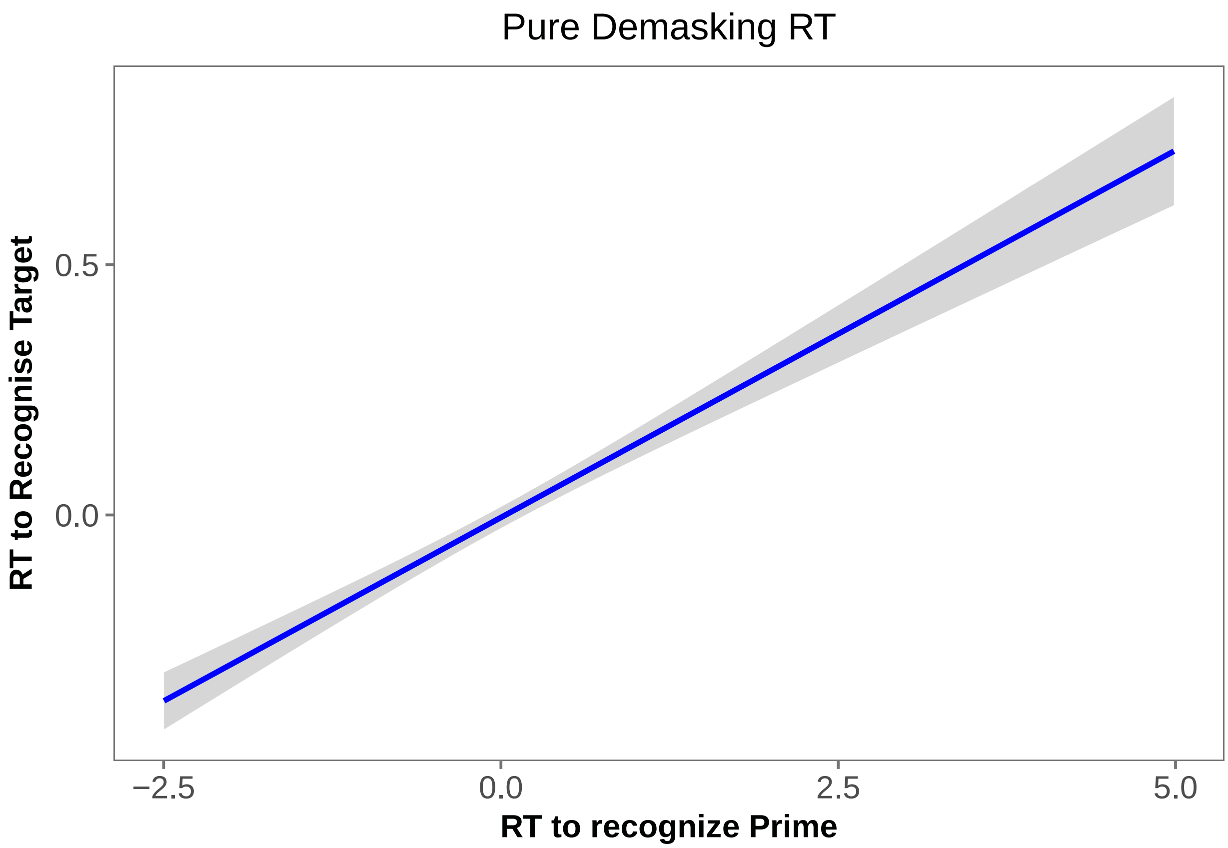
Plot

```r
primeplot = new_netdemask_z %>%
  ggplot(aes(x = zRTPrime_trim, y = zRTTarget_trim,
             group = factor(subject))) +
  geom_smooth(method = "lm", se = FALSE, color = "forestgreen", size = 0.5)+
    xlab("z-RT to Demask Prime") + ylab ("z-RT to Demask Target")+
  ggtitle("Target Retrieval Accuracy by \nPrime Demasking RT & Prime Condition")+
theme_hc() +
  facet_wrap(~pathlengthfac)+
    theme(axis.text = element_text(face = "bold", size = rel(1.2)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
          legend.title = element_text(face = "bold", size = rel(1.2)),
                   strip.text.x = element_text(face = "bold", size = rel(1.4)),
          plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))

primeplot + stat_smooth(aes(group = pathlengthfac, color = "red"), method = "lm", se = FALSE)
```

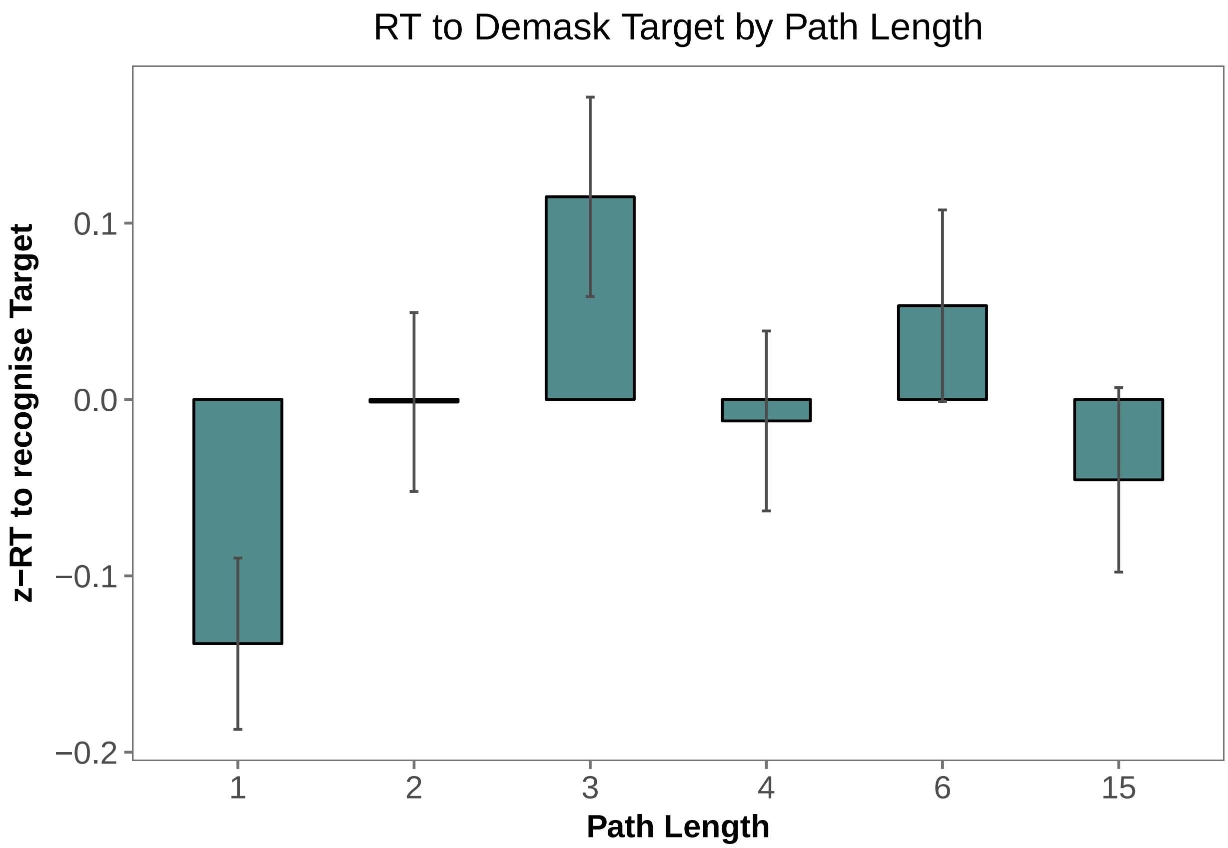


Main effects

```
new_netdemask_z %>%
  ggplot(aes(x = zRTPrime_trim, y = zRTTarget_trim))+
#  geom_smooth(method = "loess")+
  geom_smooth(size = 1, color = "blue", method = "lm")+
 theme_few()+
  #scale_x_continuous(breaks = c(1,2,3,4,5,6,10,15,20))+
    xlab("RT to recognize Prime") + ylab("RT to Recognise Target") +
  ggtitle("Pure Demasking RT")  +
   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)))
```




```
path_group = Rmisc::summarySE(new_netdemask_z,
                        measurevar = "zRTTarget_trim",
                        groupvars = c("pathlengthfac"))

path_group %>%
  ggplot(aes(x = pathlengthfac, y = zRTTarget_trim))+
#  geom_smooth(method = "loess")+
  geom_bar(stat = "identity", position = "dodge", width = 0.5,
           color = "black", fill = "darkslategray4")+
  geom_errorbar(aes(ymin=zRTTarget_trim - ci,
                    ymax=zRTTarget_trim + ci),
             width=.05, color = "gray30",
             position = position_dodge(0))+
 theme_few()+
  #scale_x_continuous(breaks = c(1,2,3,4,5,6,10,15,20))+
    xlab("Path Length") + ylab("z-RT to recognise Target") +
  ggtitle("RT to Demask Target by Path Length")  +
   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)))
```



Other Networks

Steyvers Non Directed

```
library(lme4)
new_netdemask_z$Undirected = as.double(as.character(new_netdemask_z$Undirected))
new_netdemask_z$Directed = as.double(as.character(new_netdemask_z$Directed))

new_netdemask_z$undirectedfac = ordered(as.factor(as.character(new_netdemask_z$Undirected)),


contrasts(new_netdemask_z$undirectedfac) = contr.treatment(4, base = 4)
RTprime_undirected = lmer(data = new_netdemask_z,
                          zRTTarget_trim ~ zRTPrime_trim + undirectedfac +
                            (1|subject) + (1|ItemNumber))
summary(RTprime_undirected)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRTTarget_trim ~ zRTPrime_trim + undirectedfac + (1 | subject) +
##     (1 | ItemNumber)
##    Data: new_netdemask_z
##
## REML criterion at convergence: 22407.5
##
```

```
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0503 -0.6324 -0.1450  0.4984  5.2751
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.2094   0.4576
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7659   0.8752
## Number of obs: 8311, groups:  ItemNumber, 719; subject, 38
##
## Fixed effects:
##                Estimate Std. Error t value
## (Intercept)     0.05910    0.07403   0.798
## zRTPrime_trim   0.17239    0.01084  15.905
## undirectedfac1 -0.21721    0.10118  -2.147
## undirectedfac2 -0.05125    0.08004  -0.640
## undirectedfac3 -0.02717    0.07984  -0.340
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ undrc1 undrc2
## zRTPrim_trm -0.021
## undirctdfc1 -0.732  0.026
## undirctdfc2 -0.925  0.022  0.677
## undirctdfc3 -0.927  0.013  0.678  0.858
```

```
car::Anova(RTprime_undirected)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                  Chisq Df Pr(>Chisq)
## zRTPrime_trim 252.9744  1    < 2e-16 ***
## undirectedfac   6.9179  3    0.07456 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RTprime_undirected_quad = lmer(data = new_netdemask_z,
                          zRTTarget_trim ~ zRTPrime_trim + Undirected +
                            I(Undirected^2)+
                            (1|subject) + (1|ItemNumber))
summary(RTprime_undirected_quad)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRTTarget_trim ~ zRTPrime_trim + Undirected + I(Undirected^2) +
##     (1 | subject) + (1 | ItemNumber)
##    Data: new_netdemask_z
##
## REML criterion at convergence: 22407.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0540 -0.6302 -0.1451  0.4973  5.2714
##
## Random effects:
```

```
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.2092   0.4574
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7659   0.8752
## Number of obs: 8311, groups:  ItemNumber, 719; subject, 38
##
## Fixed effects:
##                  Estimate Std. Error t value
## (Intercept)     -0.32831    0.16731  -1.962
## zRTPrime_trim    0.17241    0.01084  15.907
## Undirected       0.23165    0.13819   1.676
## I(Undirected^2) -0.03557    0.02742  -1.297
##
## Correlation of Fixed Effects:
##              (Intr) zRTPr_ Undrct
## zRTPrim_trm  0.010
## Undirected  -0.970 -0.005
## I(Undrct^2)  0.912 -0.001 -0.982
```

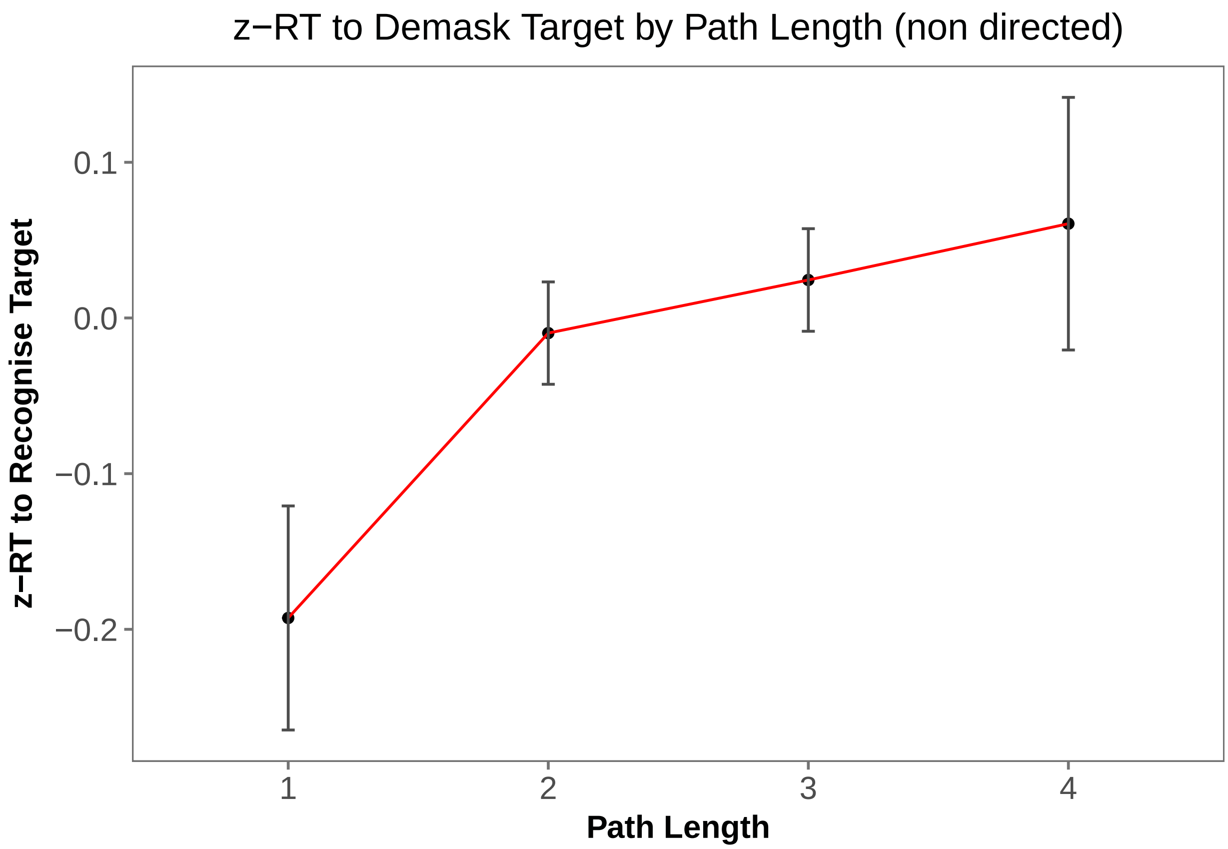
```
car::Anova(RTprime_undirected_quad)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                    Chisq Df Pr(>Chisq)
## zRTPrime_trim   253.0402  1    < 2e-16 ***
## Undirected        2.8101  1    0.09367 .
## I(Undirected^2)   1.6823  1    0.19462
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Plot

```
z_rmisc_undirected = Rmisc::summarySE(new_netdemask_z,
                        measurevar = "zRTTarget_trim",
                        groupvars = c("Undirected"))
z_rmisc_undirected = z_rmisc_undirected %>% filter(Undirected != "NA")
z_rmisc_undirected$undirectedfac = ordered(as.factor(as.character(z_rmisc_undirected$Undirected)),

z_rmisc_undirected$zRTTarget_trim = as.numeric(z_rmisc_undirected$zRTTarget_trim)

library(ggplot2)
library(ggthemes)

z_rmisc_undirected %>%
  ggplot(aes(x = undirectedfac, y = zRTTarget_trim, group = 1))+
  geom_point()+
#  geom_smooth(method = "loess")+
geom_line(color = "red")+
   geom_errorbar(aes(ymin=zRTTarget_trim - ci, ymax=zRTTarget_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 to Recognise Target") +
  ggtitle("z-RT to Demask Target by Path Length (non directed)")  +
   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)))
```



Steyvers Directed

```
library(lme4)
new_netdemask_z$newdirected = ifelse(new_netdemask_z$Directed == "Inf" |
                              new_netdemask_z$Directed == "NA", NA,
                              new_netdemask_z$Directed)

new_netdemask_z$directedcollapsed = ifelse((new_netdemask_z$newdirected == "5" |
                                      new_netdemask_z$newdirected == "6" |
                                       new_netdemask_z$newdirected == "7" |
                                       new_netdemask_z$newdirected == "8"), "H",
                              new_netdemask_z$newdirected)


new_netdemask_z$directedfac =
  ordered(as.factor(as.character(new_netdemask_z$newdirected)),
```

```r
                            levels = c("1", "2", "3", "4", "5",
                                       "6", "7", "8"))
contrasts(new_netdemask_z$directedfac) = contr.treatment(8, base = 5)

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


RTprime_directed = lmer(data = new_netdemask_z,
                          zRTTarget_trim ~ zRTPrime_trim + directedfac +
                            (1|subject) + (1|ItemNumber))
summary(RTprime_directed)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRTTarget_trim ~ zRTPrime_trim + directedfac + (1 | subject) +
##     (1 | ItemNumber)
##    Data: new_netdemask_z
##
## REML criterion at convergence: 21158.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0696 -0.6301 -0.1415  0.4931  5.3326
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.1765   0.4202
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7515   0.8669
## Number of obs: 7931, groups:  ItemNumber, 683; subject, 38
##
## Fixed effects:
##                Estimate Std. Error t value
## (Intercept)     0.04916    0.04343   1.132
## zRTPrime_trim   0.17116    0.01093  15.665
## directedfac1   -0.36572    0.09244  -3.956
## directedfac2   -0.25215    0.06426  -3.924
## directedfac3   -0.07463    0.05883  -1.268
## directedfac4   -0.04897    0.05484  -0.893
## directedfac6    0.37440    0.09500   3.941
## directedfac7    0.13197    0.28802   0.458
## directedfac8    0.17246    0.35053   0.492
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ drctd1 drctd2 drctd3 drctd4 drctd6 drctd7
## zRTPrim_trm -0.004
## directedfc1 -0.470  0.012
## directedfc2 -0.676  0.005  0.318
## directedfc3 -0.738 -0.004  0.347  0.499
## directedfc4 -0.792 -0.002  0.372  0.535  0.585
## directedfc6 -0.457  0.013  0.215  0.309  0.337  0.362
## directedfc7 -0.151  0.008  0.071  0.102  0.111  0.119  0.069
```

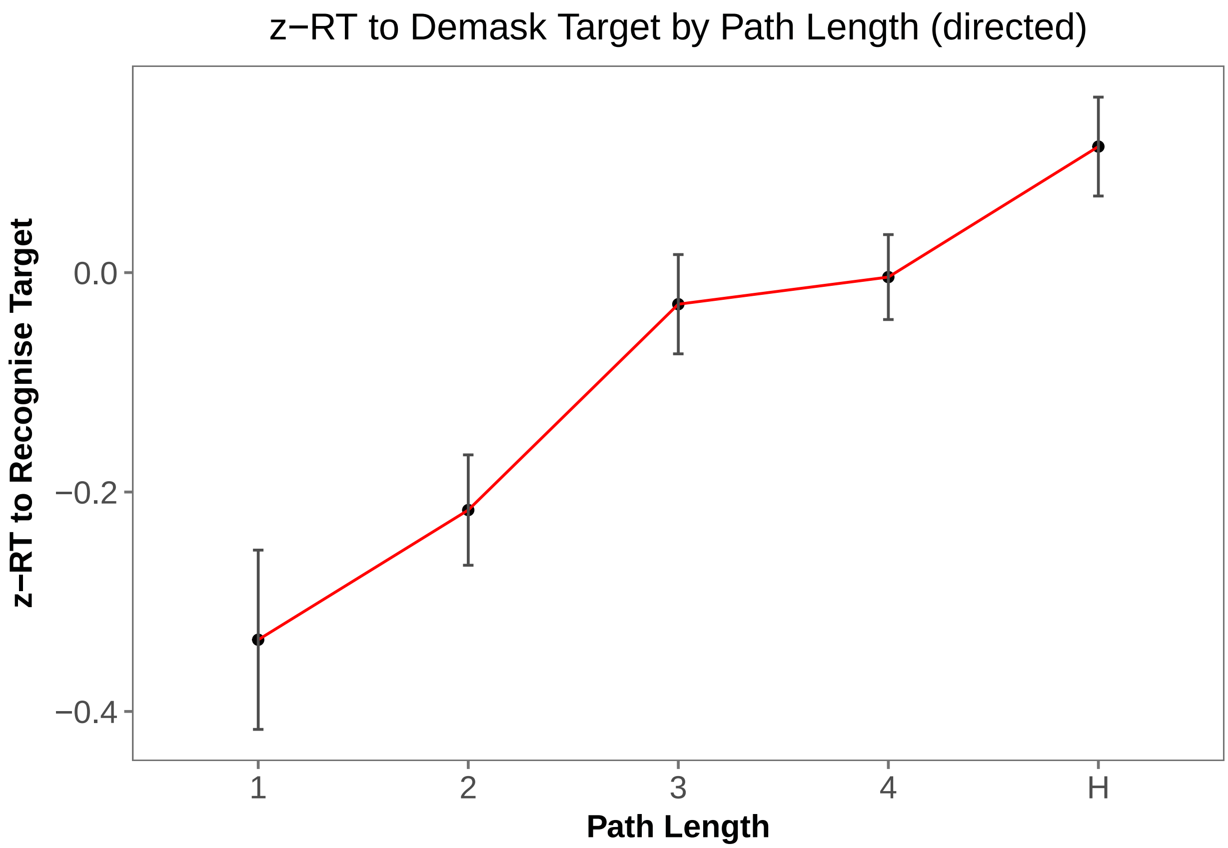
```
## directedfc8 -0.124 -0.002  0.058  0.084  0.091  0.098  0.057  0.019
```

```
car::Anova(RTprime_directed)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                  Chisq Df Pr(>Chisq)
## zRTPrime_trim 245.378  1  < 2.2e-16 ***
## directedfac    59.584  7  1.828e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Plot Collapsed

```
z_rmisc_directed = Rmisc::summarySE(new_netdemask_z,
                        measurevar = "zRTTarget_trim",
                        groupvars = c("collapsedfac"))
z_rmisc_directed = z_rmisc_directed %>% filter(collapsedfac != "NA")
z_rmisc_directed$collapsedfac2 = ordered(as.factor(as.character(z_rmisc_directed$collapsedfac)),

z_rmisc_directed$zRTTarget_trim = as.numeric(z_rmisc_directed$zRTTarget_trim)

library(ggplot2)
library(ggthemes)

z_rmisc_directed %>%
  ggplot(aes(x = collapsedfac2, y = zRTTarget_trim, group = 1))+
  geom_point()+
#  geom_smooth(method = "loess")+
geom_line(color = "red")+
   geom_errorbar(aes(ymin=zRTTarget_trim - ci, ymax=zRTTarget_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 to Recognise Target") +
  ggtitle("z-RT to Demask Target by Path Length (directed)")  +
   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 Not Collapsed

```
z_rmisc_directed = Rmisc::summarySE(new_netdemask_z,
                        measurevar = "zRTTarget_trim",
                        groupvars = c("directedfac"))
z_rmisc_directed = z_rmisc_directed %>% filter(directedfac != "NA")
z_rmisc_directed$collapsedfac2 = ordered(as.factor(as.character(z_rmisc_directed$directedfac)),

z_rmisc_directed$zRTTarget_trim = as.numeric(z_rmisc_directed$zRTTarget_trim)

library(ggplot2)
library(ggthemes)

z_rmisc_directed %>%
  ggplot(aes(x = collapsedfac2, y = zRTTarget_trim, group = 1))+
  geom_point()+
#  geom_smooth(method = "loess")+
geom_line(color = "red")+
   geom_errorbar(aes(ymin=zRTTarget_trim - ci, ymax=zRTTarget_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 to Recognise Target") +
  ggtitle("z-RT to Demask Target by Path Length (directed)")  +
```

```r
  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)))
```