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

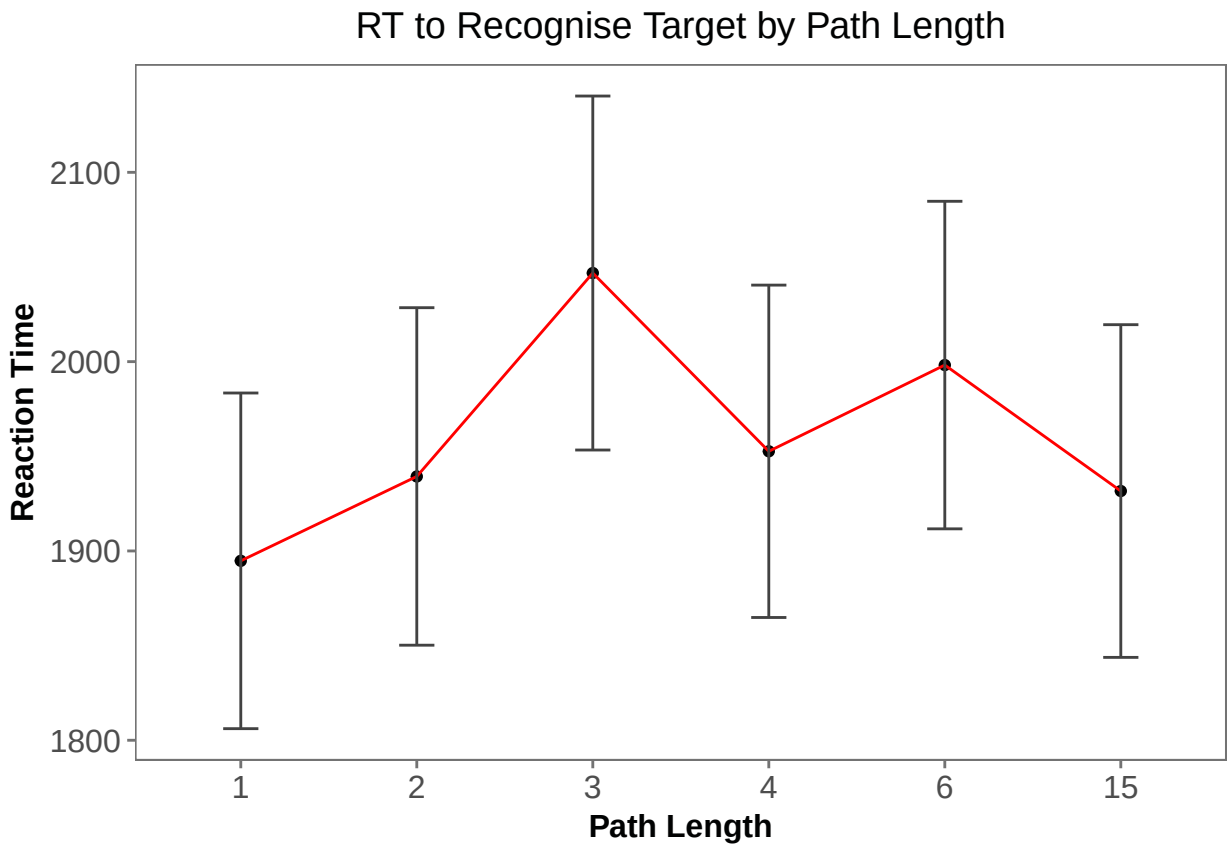
```
netdemask_rt$pathlengthfac = ordered(as.factor(as.character(netdemask_rt$pathlength)),
                             levels = c("1", "2", "3", "4", "6", "15"))
netdemask_rt$subject = as.factor(netdemask_rt$subject)
rt_aov = aov(data = netdemask_rt, RTRecogniseTarget ~ pathlengthfac +
            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

```
##
## Error: subject
##           Df   Sum Sq Mean Sq F value Pr(>F)
## Residuals 34 54769239 1610860
##
## Error: subject:pathlengthfac
##                Df  Sum Sq Mean Sq F value  Pr(>F)
## pathlengthfac   5  508542  101708   10.15 1.6e-08 ***
## Residuals     170 1703277   10019
## ---
## 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 %>%
```

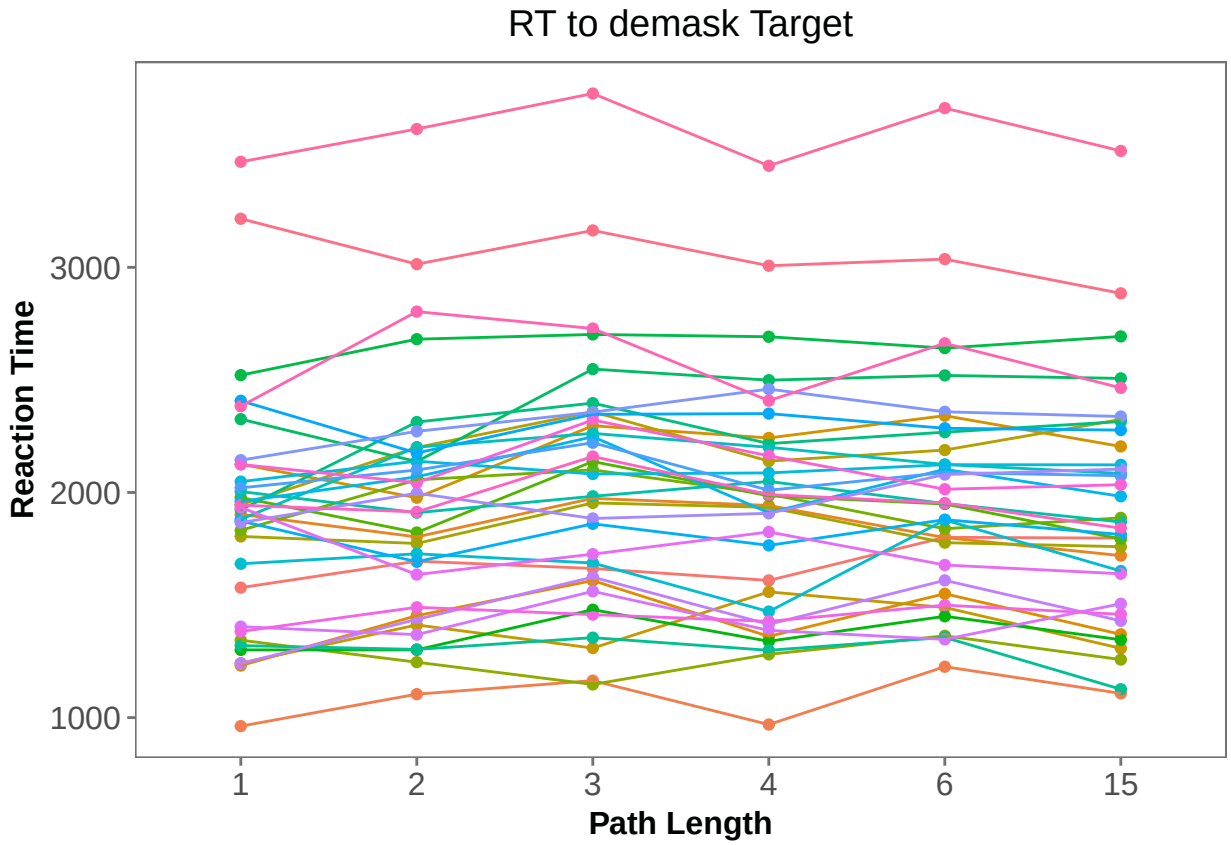
```
 ggplot(aes(x = pathlengthfac, y = RTRecogniseTarget, group = 1))+
 geom_point()+
 geom_line(color = "red")+
  geom_errorbar(aes(ymin=RTRecogniseTarget - se, ymax=RTRecogniseTarget + se),
            width=.2, color = "gray26",
            position = position_dodge(0.7))+
theme_few()+
# scale_x_continuous(breaks = 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()+
```

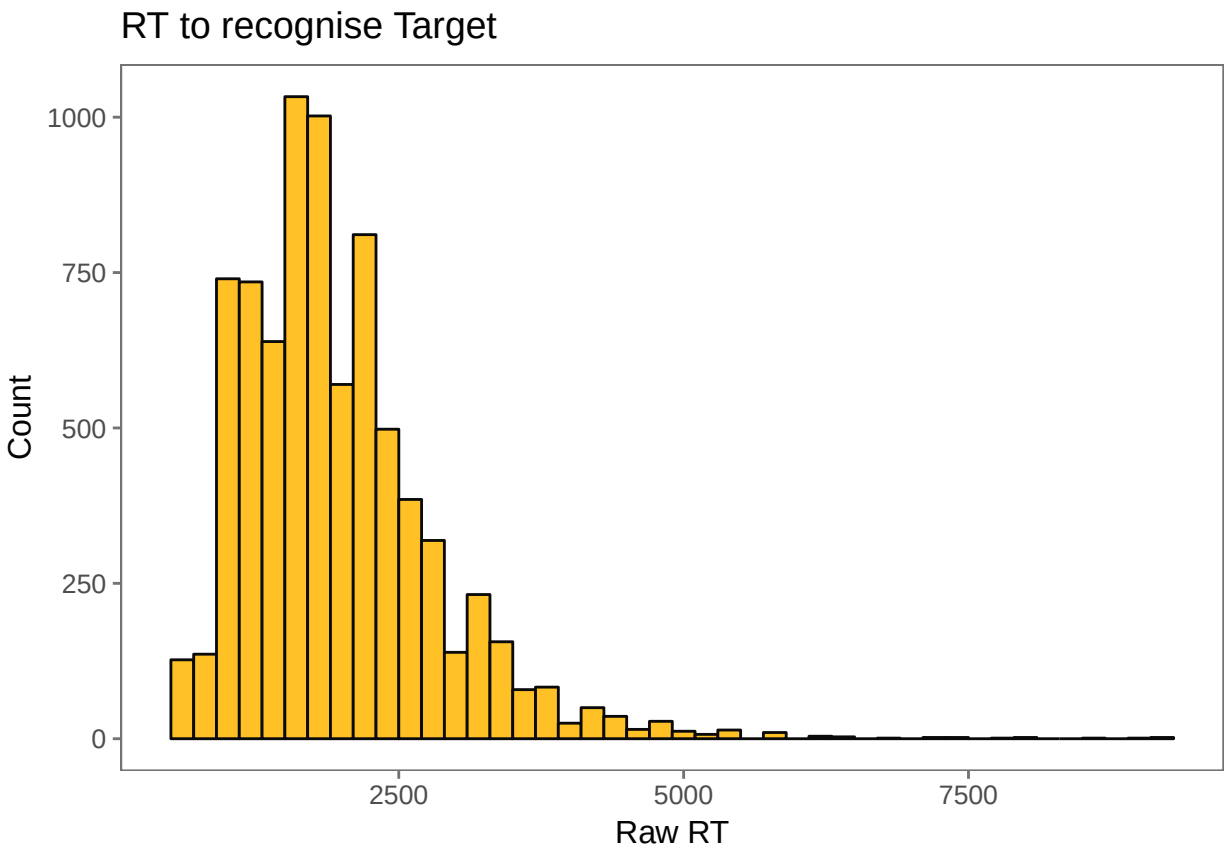
```
 geom_line()+
theme_few()+
 guides(color = FALSE)+
# scale_x_continuous(breaks = c(1,2,3,4,6,15))+
   xlab("Path Length") + ylab("Reaction Time") +
 ggtitle("RT to demask Target")  +
  theme(axis.text = element_text(size = rel(1)),
        axis.title = element_text(face = "bold", size = rel(1)),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

```
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")
```



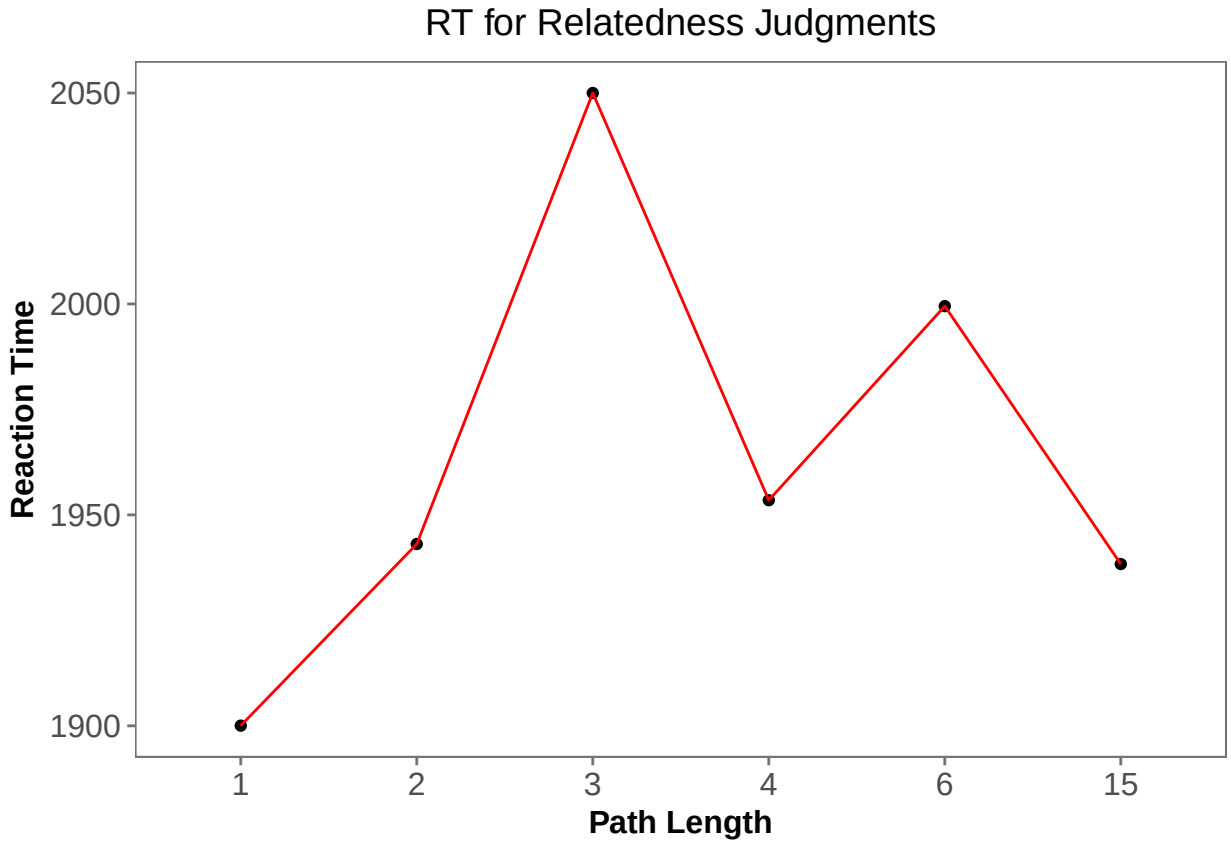
First Trim

```
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),
```

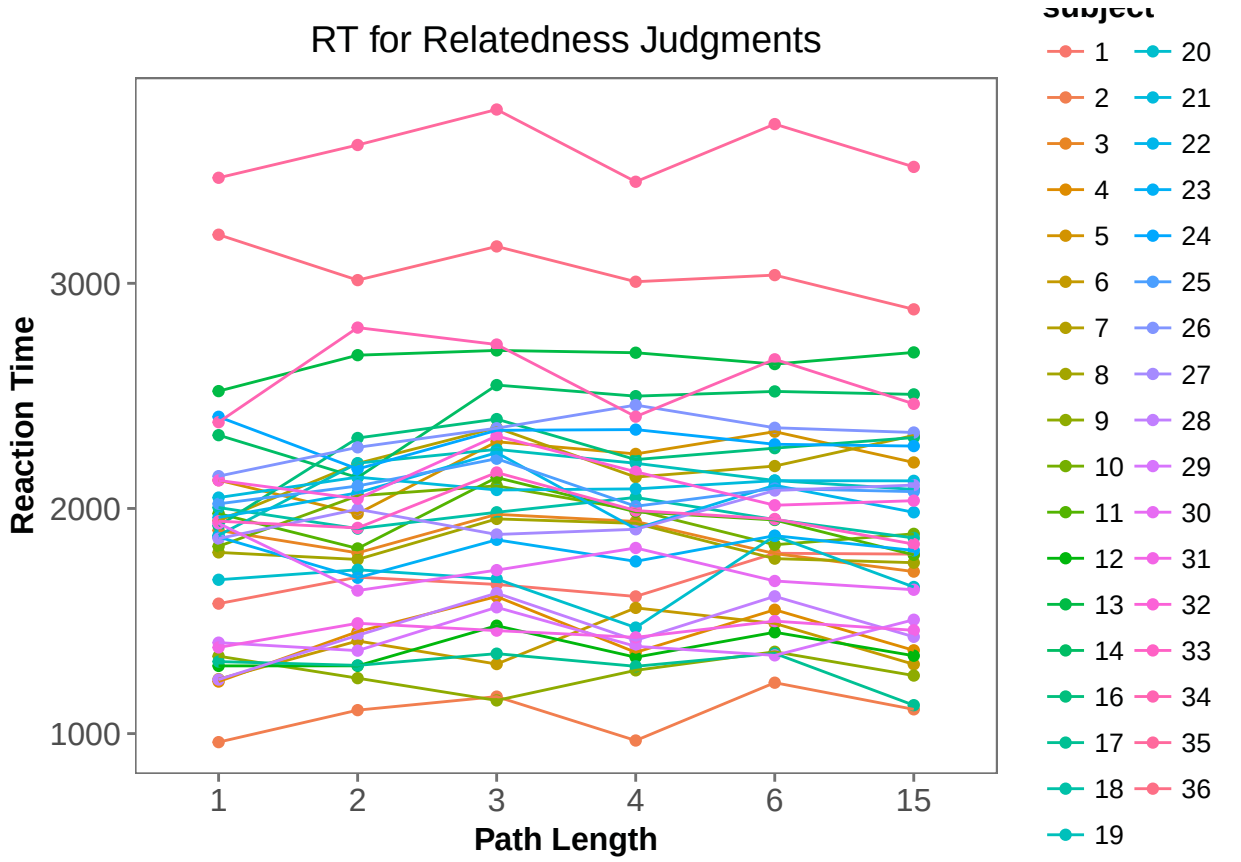
```
 #           width=.2, color = "gray26",
  #          position = position_dodge(0.7))+
theme_few()+
 #scale_x_continuous(breaks = c(1,2,3,4,6,15))+
   xlab("Path Length") + ylab("Reaction Time") +
 ggtitle("RT for Relatedness Judgments")  +
  theme(axis.text = element_text(size = rel(1)),
      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



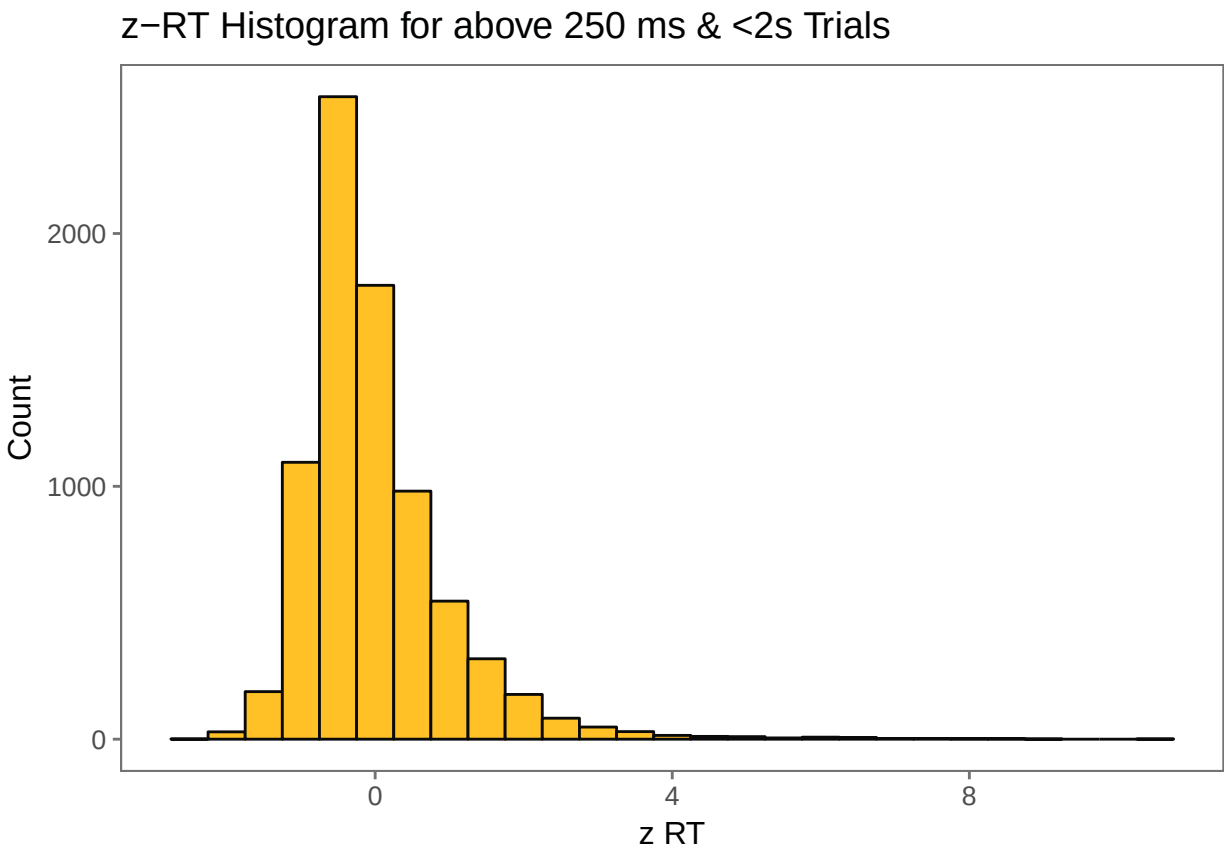
Making the z-scores

```
## 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)
```

```
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

```
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")
```

```
## 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 34 0.02048 0.0006024
##
## Error: subject:pathlengthfac
##                Df Sum Sq Mean Sq F value   Pr(>F)
## pathlengthfac   5  1.221 0.24425   8.339 4.51e-07 ***
## Residuals     170  4.979 0.02929
## ---
## 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'
```

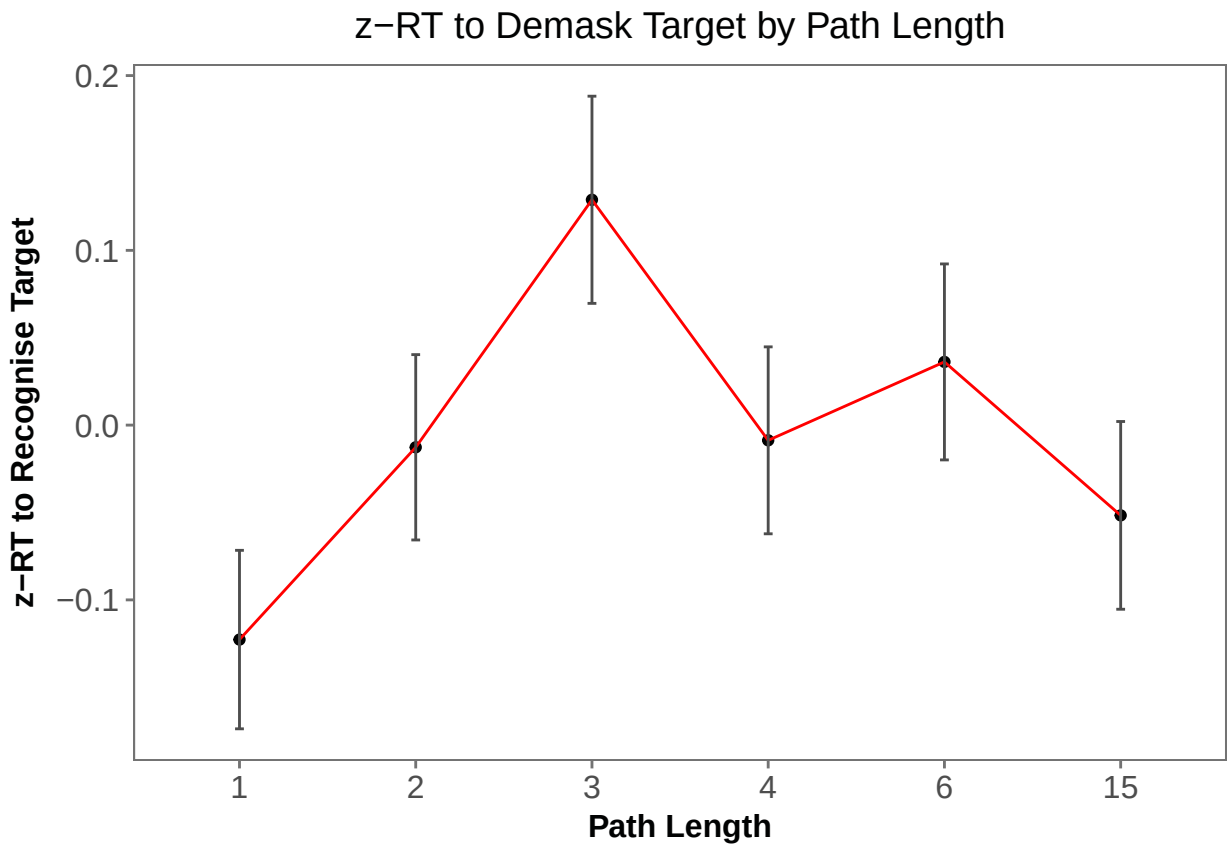
```
## The following object is masked from 'package:MASS':
##
##     geyser
```

```
 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
4	4 - 1	0.1086657	0.0409109	170	2.656156	0.0897658
7	6 - 1	0.1533685	0.0409109	170	3.748846	0.0032740
11	3 - 1	0.2484619	0.0409109	170	6.073249	0.0000001
12	3 - 15	0.1809664	0.0409109	170	4.423431	0.0002474
13	3 - 2	0.1432164	0.0409109	170	3.500693	0.0076908
14	3 - 4	0.1397962	0.0409109	170	3.417093	0.0101259

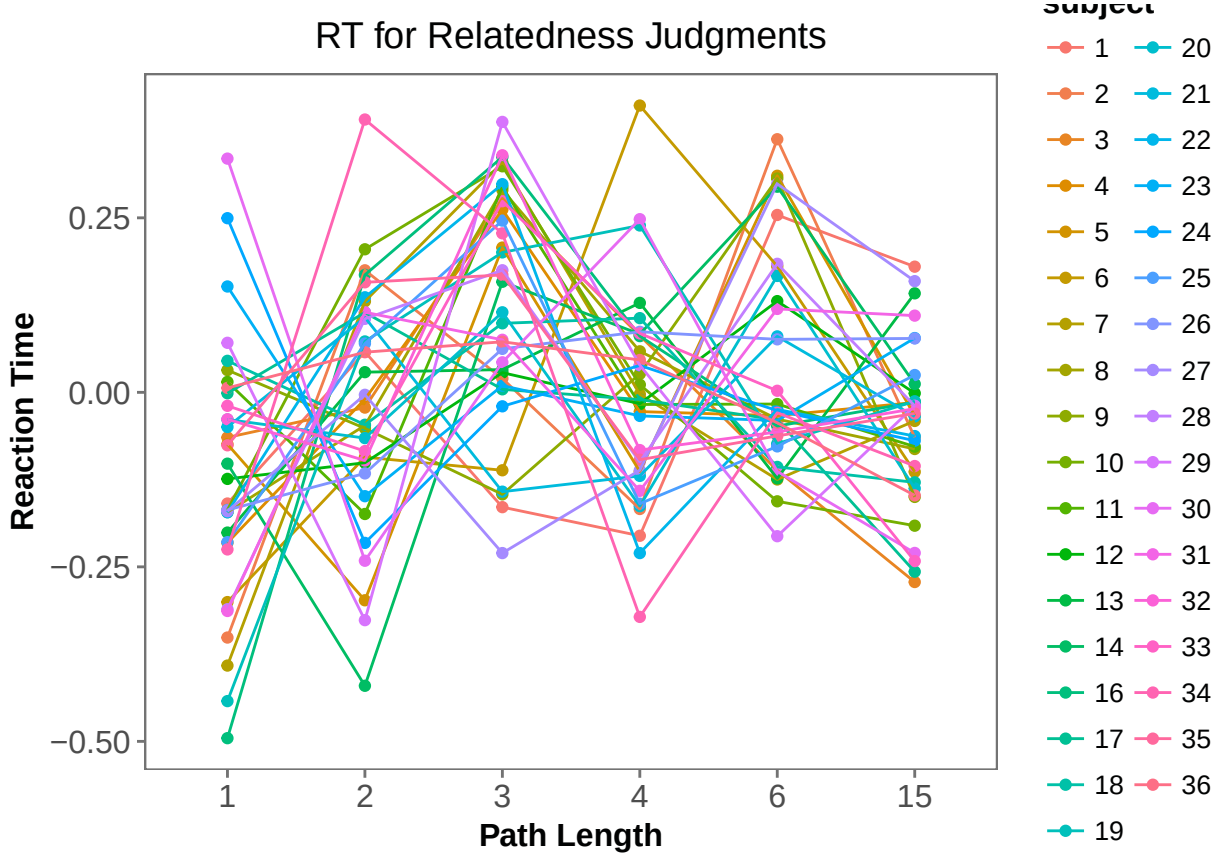
Plotting RTs: collapsed

```
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

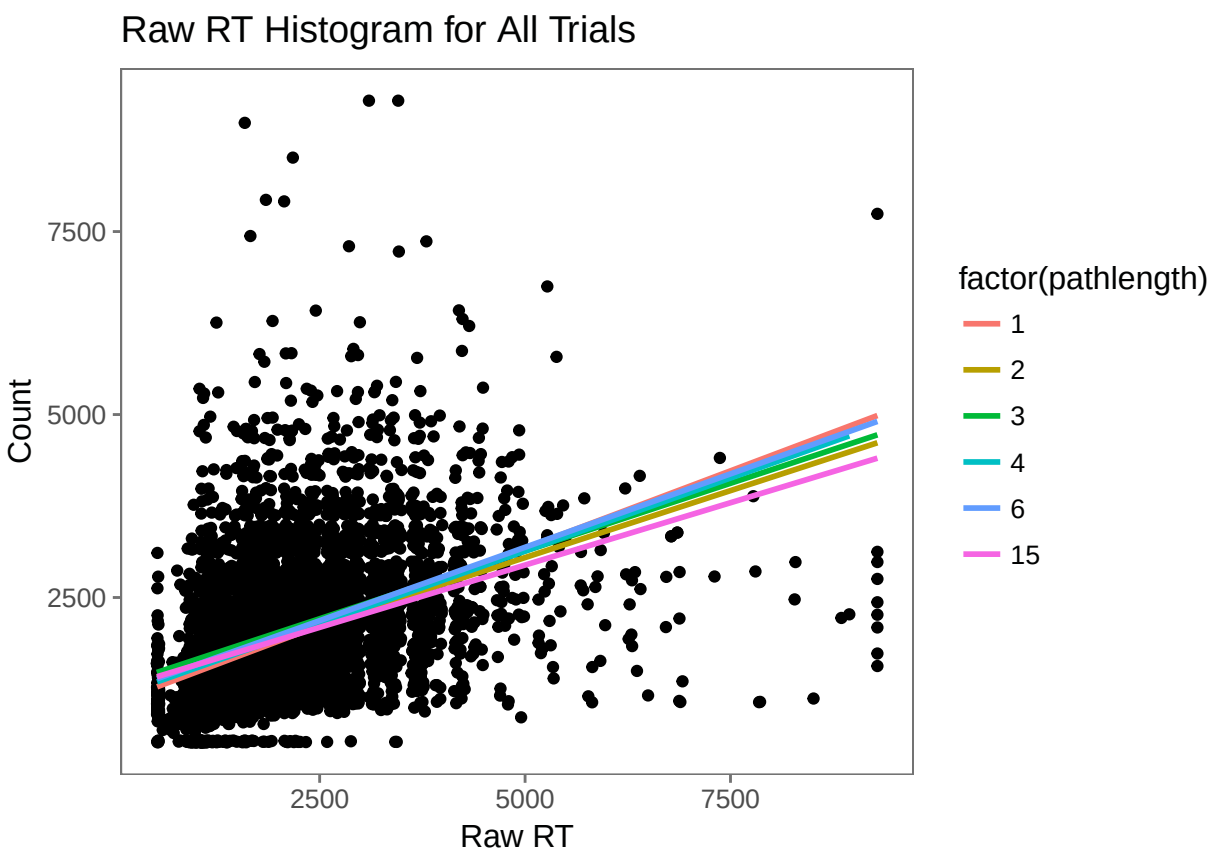
```
library(ggplot2)
library(ggthemes)


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")
```



Linear Models

```
library(lme4)
```

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

```
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: 20698.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0553 -0.6314 -0.1494  0.5054  5.3029
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
```

```
##  ItemNumber (Intercept) 0.2111   0.4594
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7658   0.8751
## Number of obs: 7665, groups:  ItemNumber, 720; subject, 35
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)   0.007787   0.019893   0.391
## zRTPrime_trim 0.171334   0.011299  15.164
##
## 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: 20703.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0641 -0.6301 -0.1482  0.5058  5.2669
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.2073   0.4553
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7659   0.8751
## Number of obs: 7665, groups:  ItemNumber, 720; subject, 35
##
## Fixed effects:
##                Estimate Std. Error t value
## (Intercept)     0.14014    0.04861   2.883
## zRTPrime_trim   0.17085    0.01129  15.126
## pathlengthfac1 -0.23867    0.06853  -3.483
## pathlengthfac2 -0.14239    0.06843  -2.081
## pathlengthfac4 -0.14161    0.06873  -2.060
## pathlengthfac5 -0.08913    0.06853  -1.301
## pathlengthfac6 -0.18146    0.06863  -2.644
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ pthln1 pthln2 pthln4 pthln5
## zRTPrim_trm -0.003
## pthlngthfc1 -0.709  0.007
## pthlngthfc2 -0.710  0.009  0.504
## pthlngthfc4 -0.707 -0.009  0.502  0.502
## pthlngthfc5 -0.709 -0.006  0.503  0.504  0.502
```

```
## pthlngthfc6 -0.708 -0.002  0.502  0.503  0.501  0.502
```

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

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                 Chisq Df Pr(>Chisq)
## zRTPrime_trim 228.810  1    < 2e-16 ***
## pathlengthfac  14.166  5    0.01459 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## 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: 20598.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0298 -0.6305 -0.1470  0.5020  5.3496
##
## Random effects:
##  Groups     Name        Variance  Std.Dev.
##  ItemNumber (Intercept) 1.750e-01 4.184e-01
##  subject    (Intercept) 1.188e-18 1.090e-09
##  Residual               7.656e-01 8.750e-01
## Number of obs: 7653, groups:  ItemNumber, 719; subject, 35
##
## Fixed effects:
##                Estimate Std. Error t value
## (Intercept)    0.126574   0.175449   0.721
## zRTPrime_trim  0.153870   0.011339  13.570
## pathlength     0.037650   0.019520   1.929
## pquad         -0.002431   0.001143  -2.127
## MeanLDTZ       0.621943   0.142202   4.374
## MeanLength     0.059375   0.015395   3.857
## MeanLogF      -0.029594   0.016585  -1.784
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ pthlng pquad  MnLDTZ MnLngt
## zRTPrim_trm -0.028
## pathlength  -0.301 -0.013
```

```
## pquad        0.275  0.012 -0.979
## MeanLDTZ     0.210 -0.067 -0.040  0.031
## MeanLength  -0.639 -0.033  0.021 -0.015 -0.452
## MeanLogF    -0.590  0.027  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 184.1450  1  < 2.2e-16 ***
## pathlength      3.7202  1   0.053759 .
## pquad           4.5249  1   0.033405 *
## MeanLDTZ       19.1290  1  1.222e-05 ***
## MeanLength     14.8736  1   0.000115 ***
## MeanLogF        3.1838  1   0.074371 .
## ---
## 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: 20723.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.1038 -0.6304 -0.1497  0.5031  5.2601
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.2077   0.4558
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7657   0.8751
## Number of obs: 7665, groups:  ItemNumber, 720; subject, 35
##
## Fixed effects:
##                                Estimate Std. Error t value
## (Intercept)                     0.13965    0.04865   2.871
## zRTPrime_trim                   0.21121    0.02773   7.618
## pathlengthfac1                 -0.23894    0.06859  -3.484
## pathlengthfac2                 -0.14148    0.06850  -2.065
## pathlengthfac4                 -0.13873    0.06880  -2.016
## pathlengthfac5                 -0.08938    0.06859  -1.303
## pathlengthfac6                 -0.18074    0.06869  -2.631
## zRTPrime_trim:pathlengthfac1 -0.06378    0.03957  -1.612
## zRTPrime_trim:pathlengthfac2 -0.03045    0.03958  -0.769
## zRTPrime_trim:pathlengthfac4 -0.07683    0.03944  -1.948
```

```
## zRTPrime_trim:pathlengthfac5 -0.02367    0.03845  -0.616
## zRTPrime_trim:pathlengthfac6 -0.04965    0.03886  -1.277
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ pthln1 pthln2 pthln4 pthln5 pthln6 zRTP_:1
## zRTPrim_trm -0.007
## pthlngthfc1 -0.709  0.005
## pthlngthfc2 -0.710  0.005  0.504
## pthlngthfc4 -0.707  0.005  0.502  0.502
## pthlngthfc5 -0.709  0.005  0.503  0.504  0.501
## pthlngthfc6 -0.708  0.005  0.502  0.503  0.501  0.502
## zRTPrm_tr:1  0.005 -0.701  0.006 -0.004 -0.004 -0.004 -0.004
## zRTPrm_tr:2  0.005 -0.701 -0.004  0.009 -0.004 -0.004 -0.004  0.491
## zRTPrm_tr:4  0.005 -0.703 -0.004 -0.004 -0.022 -0.004 -0.004  0.493
## zRTPrm_tr:5  0.005 -0.721 -0.004 -0.004 -0.004 -0.016 -0.004  0.505
## zRTPrm_tr:6  0.005 -0.713 -0.004 -0.004 -0.004 -0.004 -0.010  0.500
##             zRTP_:2 zRTP_:4 zRTP_:5
## zRTPrim_trm
## pthlngthfc1
## pthlngthfc2
## pthlngthfc4
## pthlngthfc5
## pthlngthfc6
## zRTPrm_tr:1
## zRTPrm_tr:2
## zRTPrm_tr:4  0.493
## zRTPrm_tr:5  0.505   0.507
## zRTPrm_tr:6  0.500   0.502   0.515
```

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

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                               Chisq Df Pr(>Chisq)
## zRTPrime_trim              228.8709  1    < 2e-16 ***
## pathlengthfac               14.1434  5    0.01472 *
## zRTPrime_trim:pathlengthfac  5.1296  5    0.40026
## ---
## 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 20691 20761 -10336    20671
## RTprime_model_2_2 15 20696 20800 -10333    20666 5.1258      5     0.4007
```

```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: 20588
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0228 -0.6300 -0.1493  0.5040  5.3146
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.1726   0.4154
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7656   0.8750
## Number of obs: 7653, groups:  ItemNumber, 719; subject, 35
##
## Fixed effects:
##                Estimate Std. Error t value
## (Intercept)     0.12991    0.04557   2.851
## zRTPrime_trim   0.15375    0.01133  13.565
## pathlengthfac1 -0.21771    0.06437  -3.382
## pathlengthfac2 -0.12083    0.06421  -1.882
## pathlengthfac4 -0.15206    0.06457  -2.355
## pathlengthfac5 -0.08417    0.06421  -1.311
## pathlengthfac6 -0.18208    0.06427  -2.833
## mean_len_c      0.05928    0.01534   3.865
## mean_logf_c    -0.03326    0.01660  -2.004
## mean_ldtz_c     0.60059    0.14189   4.233
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ pthln1 pthln2 pthln4 pthln5 pthln6 mn_ln_ mn_lg_
## zRTPrim_trm  0.001
## pthlngthfc1 -0.710  0.003
## pthlngthfc2 -0.709  0.004  0.503
## pthlngthfc4 -0.705 -0.006  0.501  0.497
## pthlngthfc5 -0.710 -0.007  0.505  0.503  0.501
## pthlngthfc6 -0.708 -0.002  0.502  0.501  0.500  0.503
## mean_len_c   0.012 -0.033 -0.016 -0.040  0.012 -0.012  0.008
```

```
## mean_logf_c -0.030  0.027  0.060 -0.031  0.048  0.038  0.020 -0.077
## mean_ldtz_c -0.045 -0.068  0.077  0.037  0.013  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 184.0107  1  < 2.2e-16 ***
## pathlengthfac  14.4486  5  0.0129974 *
## mean_len_c     14.9344  1  0.0001113 ***
## mean_logf_c     4.0153  1  0.0450899 *
## mean_ldtz_c    17.9152  1   2.31e-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: 20921.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.6700 -0.6357 -0.1507  0.5120  5.2049
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.1997   0.4469
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7924   0.8902
## Number of obs: 7665, groups:  ItemNumber, 720; subject, 35
##
## Fixed effects:
##                Estimate Std. Error t value
## (Intercept)     0.14190    0.04818   2.945
## pathlengthfac1 -0.24655    0.06791  -3.631
## pathlengthfac2 -0.15153    0.06781  -2.235
## pathlengthfac4 -0.13268    0.06811  -1.948
## pathlengthfac5 -0.08363    0.06791  -1.232
## pathlengthfac6 -0.17963    0.06802  -2.641
##
## Correlation of Fixed Effects:
##             (Intr) pthln1 pthln2 pthln4 pthln5
## pthlngthfc1 -0.709
## pthlngthfc2 -0.710  0.504
## pthlngthfc4 -0.707  0.502  0.503
## pthlngthfc5 -0.709  0.503  0.504  0.502
## pthlngthfc6 -0.708  0.503  0.503  0.501  0.503
```

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

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                Chisq Df Pr(>Chisq)
## pathlengthfac 15.408  5   0.008754 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
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 20914 20976 -10448    20896
## RTprime_model_2 10 20691 20761 -10336    20671 224.54      1  < 2.2e-16
##
## RTprime_model_3
## RTprime_model_2 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
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 20695 20730 -10343    20685
## RTprime_model_2 10 20691 20761 -10336    20671 14.144      5    0.01472 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Contrasts

```
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,]  0.14014295
## [2,]  0.17085220
## [3,] -0.23867133
## [4,] -0.14239423
## [5,] -0.14161024
## [6,] -0.08912765
## [7,] -0.18146350
```

```
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,] -0.098528388    1
## [2,] -0.002251287    2
## [3,]  0.140142946    3
## [4,] -0.001467294    4
## [5,]  0.051015301    6
## [6,] -0.041320555   15
```

```
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
```

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

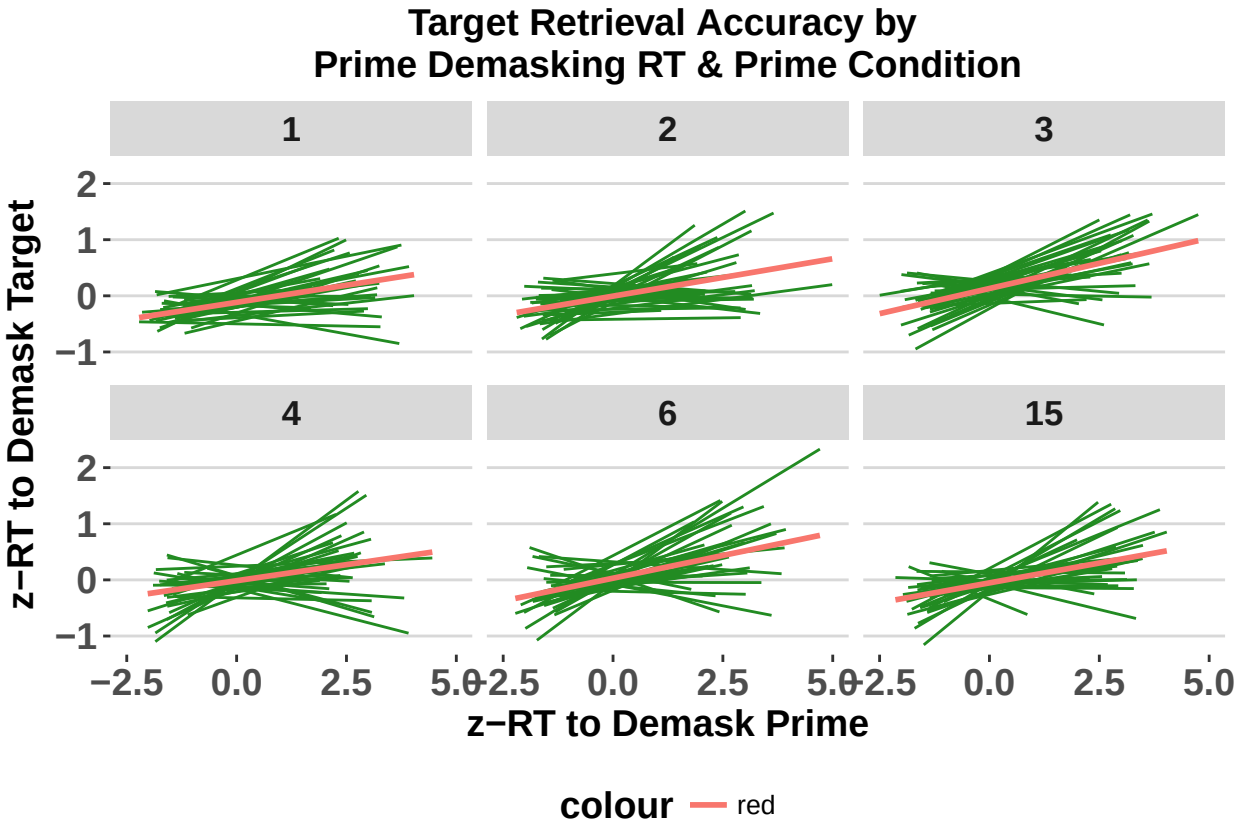
```
##                      [,1] [,2] [,3] [,4] [,5] [,6] [,7]
## path 1 vs. path 2       0    0    1   -1    0    0    0
## path 1 vs. path 3       0    0    1    0    0    0    0
## path 2 vs. path 3       0    0    0    1    0    0    0
## path 3 vs. path 4       0    0    0    0   -1    0    0
##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

```
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   -0.09628    0.06821  -1.411  0.60825
## path 1 vs. path 3  == 0   -0.23867    0.06853  -3.483  0.00408 **
## path 2 vs. path 3 == 0    -0.14239    0.06843  -2.081  0.20822
## path 3 vs. path 4 == 0     0.14161    0.06873   2.060  0.21724
## path 4 vs. path 6 == 0    -0.05248    0.06851  -0.766  0.94912
## path 6 vs. path 15 == 0    0.09234    0.06841   1.350  0.65093
## path 3 vs. path 6  == 0    0.08913    0.06853   1.301  0.68400
## path 3 vs. path 15   == 0  0.18146    0.06863   2.644  0.05393 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- single-step method)
```

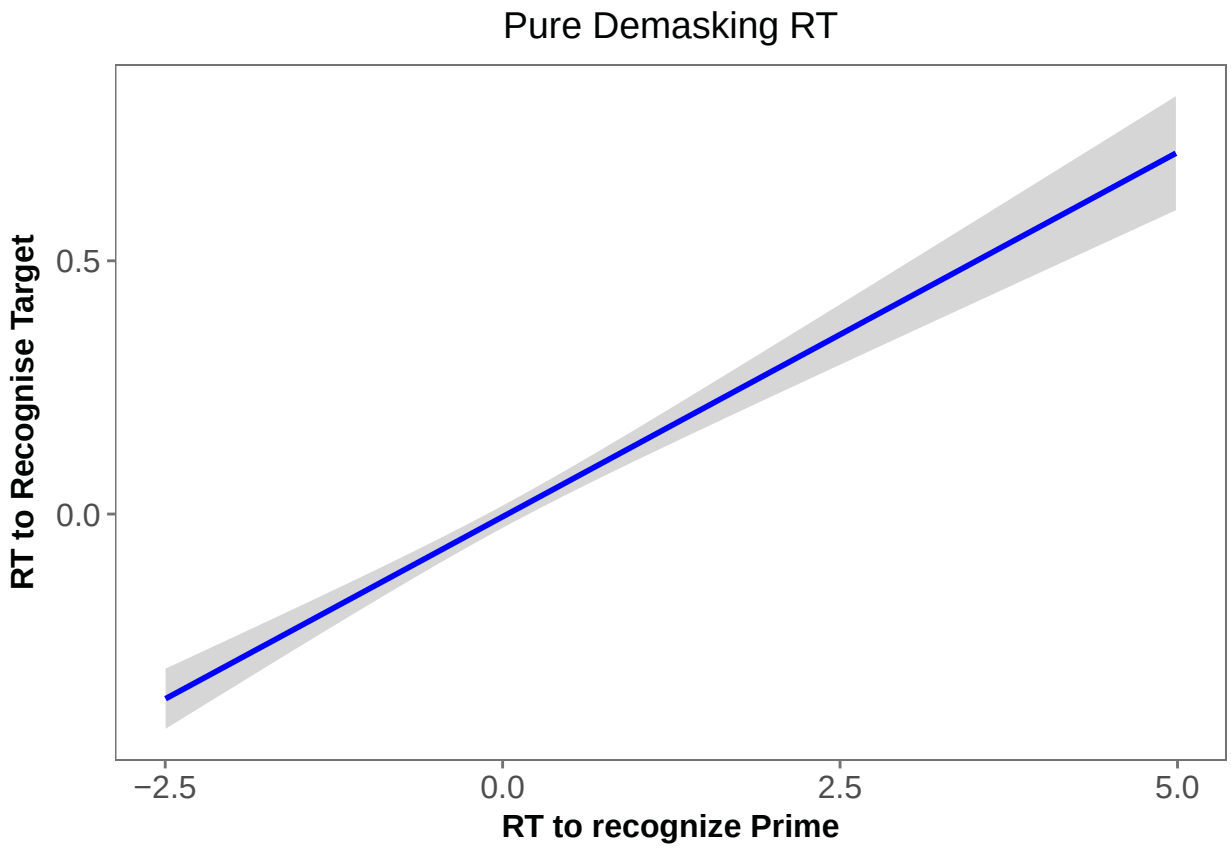
Plot

```
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)
```

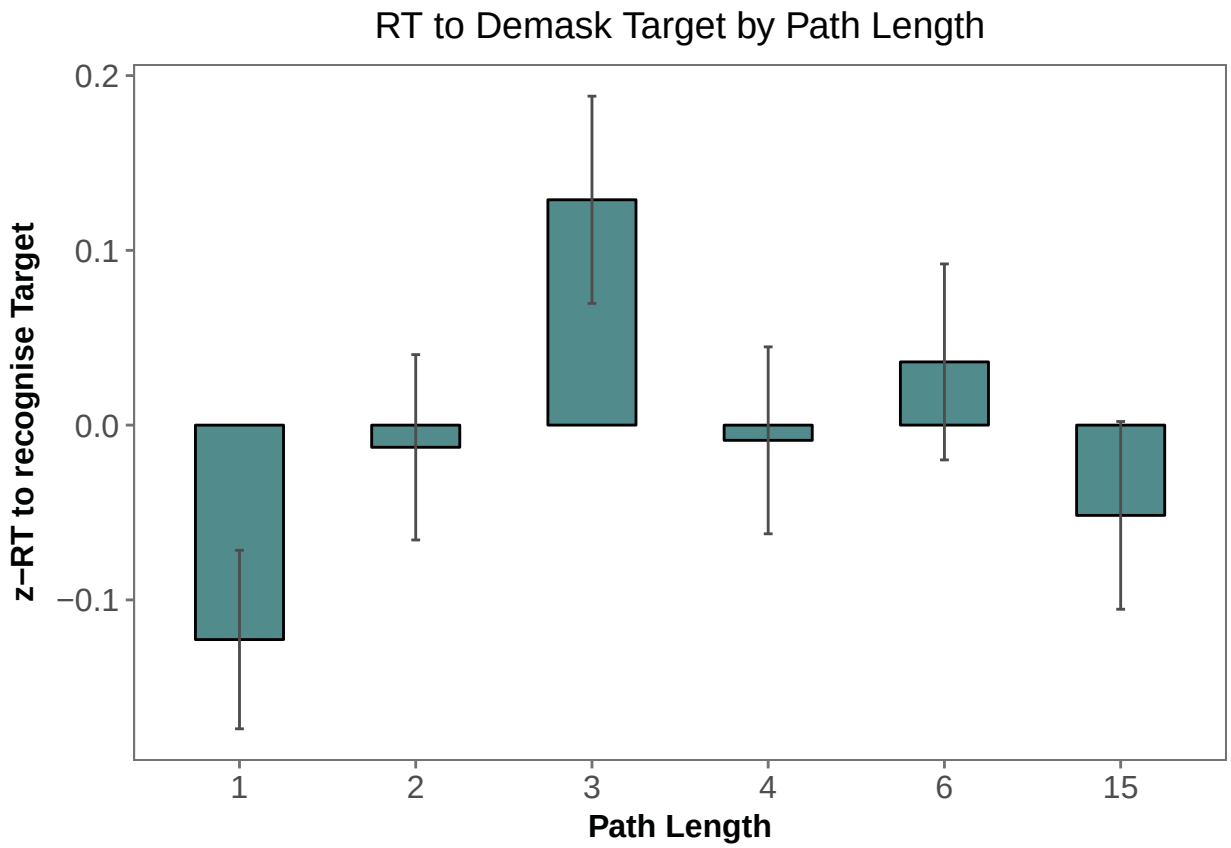


Main effects

```
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: 20678.9
##
```

```
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0575 -0.6313 -0.1498  0.5052  5.2960
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.2102   0.4584
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7663   0.8754
## Number of obs: 7654, groups:  ItemNumber, 719; subject, 35
##
## Fixed effects:
##                Estimate Std. Error t value
## (Intercept)     0.05376    0.07491   0.718
## zRTPrime_trim   0.17057    0.01131  15.080
## undirectedfac1 -0.19708    0.10242  -1.924
## undirectedfac2 -0.04412    0.08099  -0.545
## undirectedfac3 -0.02580    0.08079  -0.319
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ undrc1 undrc2
## zRTPrim_trm -0.021
## undirctdfc1 -0.732  0.028
## 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 227.4136  1     <2e-16 ***
## undirectedfac   5.5055  3     0.1383
## ---
## 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: 20679.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0614 -0.6312 -0.1516  0.5055  5.2921
##
## Random effects:
```

```
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.2100   0.4583
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.7663   0.8754
## Number of obs: 7654, groups:  ItemNumber, 719; subject, 35
##
## Fixed effects:
##                 Estimate Std. Error t value
## (Intercept)     -0.29589    0.16939  -1.747
## zRTPrime_trim    0.17060    0.01131  15.083
## Undirected       0.21062    0.13989   1.506
## I(Undirected^2) -0.03263    0.02776  -1.176
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ Undrct
## zRTPrim_trm  0.012
## Undirected  -0.970 -0.006
## I(Undrct^2)  0.912  0.000 -0.982
```

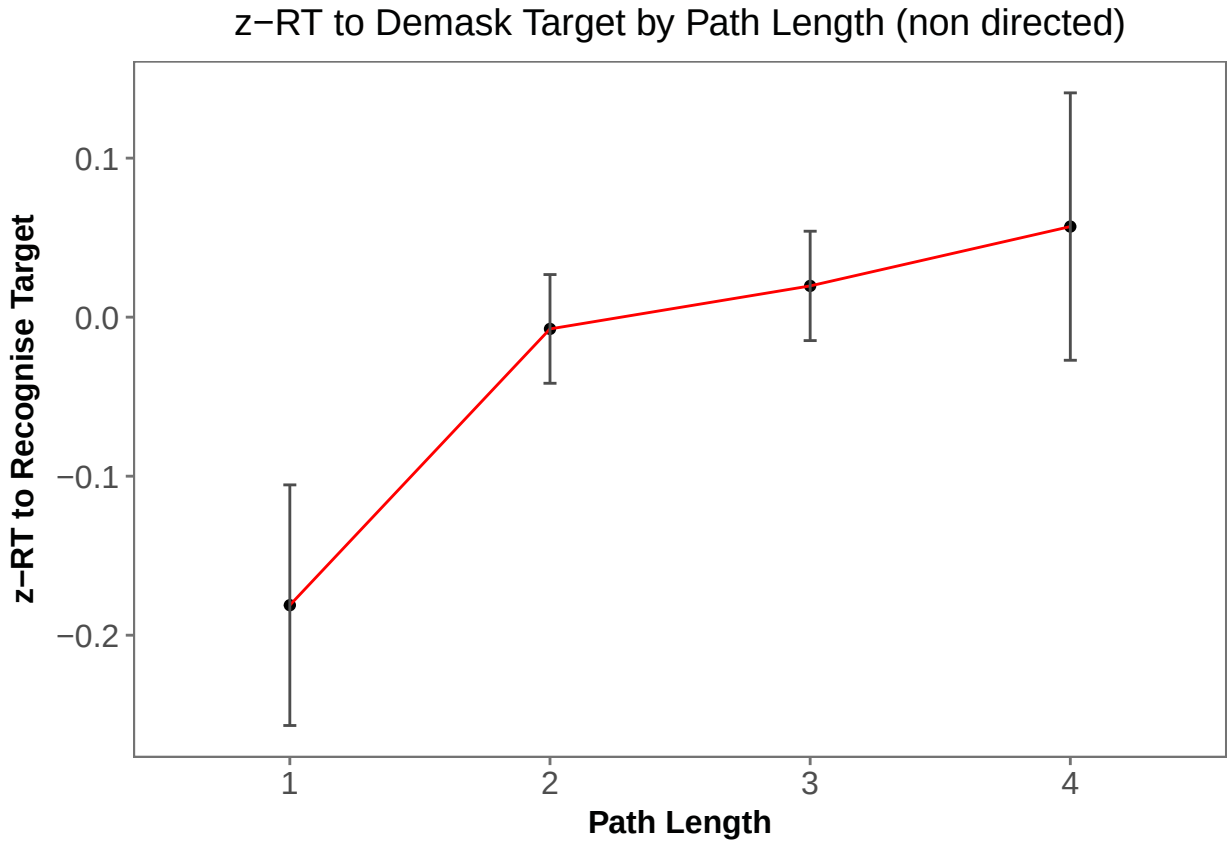
```
car::Anova(RTprime_undirected_quad)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                    Chisq Df Pr(>Chisq)
## zRTPrime_trim   227.5073  1     <2e-16 ***
## Undirected        2.2670  1     0.1322
## I(Undirected^2)   1.3822  1     0.2397
## ---
## 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)),
```

```
                            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: 19524.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0730 -0.6327 -0.1470  0.4986  5.3485
##
## Random effects:
##  Groups     Name        Variance  Std.Dev.
##  ItemNumber (Intercept) 1.806e-01 4.249e-01
##  subject    (Intercept) 1.717e-18 1.310e-09
##  Residual               7.530e-01 8.678e-01
## Number of obs: 7297, groups:  ItemNumber, 683; subject, 35
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)    0.04691    0.04430   1.059
## zRTPrime_trim  0.16834    0.01143  14.734
## directedfac1  -0.35113    0.09427  -3.725
## directedfac2  -0.23745    0.06554  -3.623
## directedfac3  -0.06795    0.06003  -1.132
## directedfac4  -0.04771    0.05595  -0.853
## directedfac6   0.35072    0.09699   3.616
## directedfac7   0.10771    0.29428   0.366
## directedfac8   0.18616    0.35799   0.520
##
## Correlation of Fixed Effects:
##             (Intr) zRTPr_ drctd1 drctd2 drctd3 drctd4 drctd6 drctd7
## zRTPrim_trm -0.004
## directedfc1 -0.470  0.011
## directedfc2 -0.676  0.003  0.318
## directedfc3 -0.738 -0.004  0.347  0.499
## directedfc4 -0.792 -0.001  0.372  0.535  0.584
## directedfc6 -0.457  0.013  0.215  0.309  0.337  0.362
## directedfc7 -0.151  0.012  0.071  0.102  0.111  0.119  0.069
```

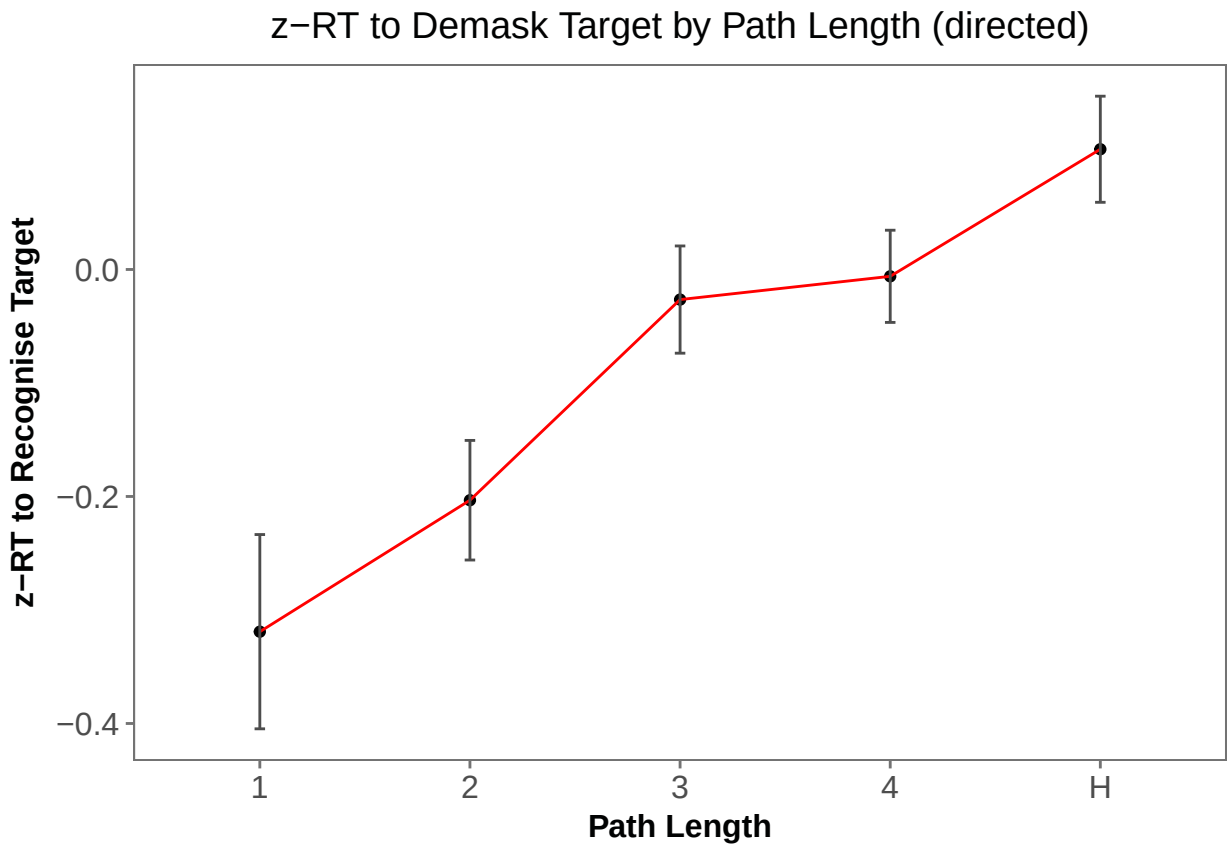
```
## directedfc8 -0.124 -0.004  0.058  0.084  0.091  0.098  0.056  0.019
```

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

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                  Chisq Df Pr(>Chisq)
## zRTPrime_trim 217.082  1  < 2.2e-16 ***
## directedfac    51.027  7  9.076e-09 ***
## ---
## 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)")  +
```

```
  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)))
```