Network Demasking

Reading the Data

```
library(dplyr)
netdemask = read.csv("Compiled_NetworksE3.csv", header = TRUE, sep = ",")
## filtering out incorrect target responses
netdemask = netdemask %>% filter(TargetAccuracy == 1)
```

Raw Reaction Time

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

```
## Warning in aov(data = netdemask_rt, RTRecogniseTarget ~ pathlengthfac + :
## Error() model is singular
```

```
summary(rt_aov)
```

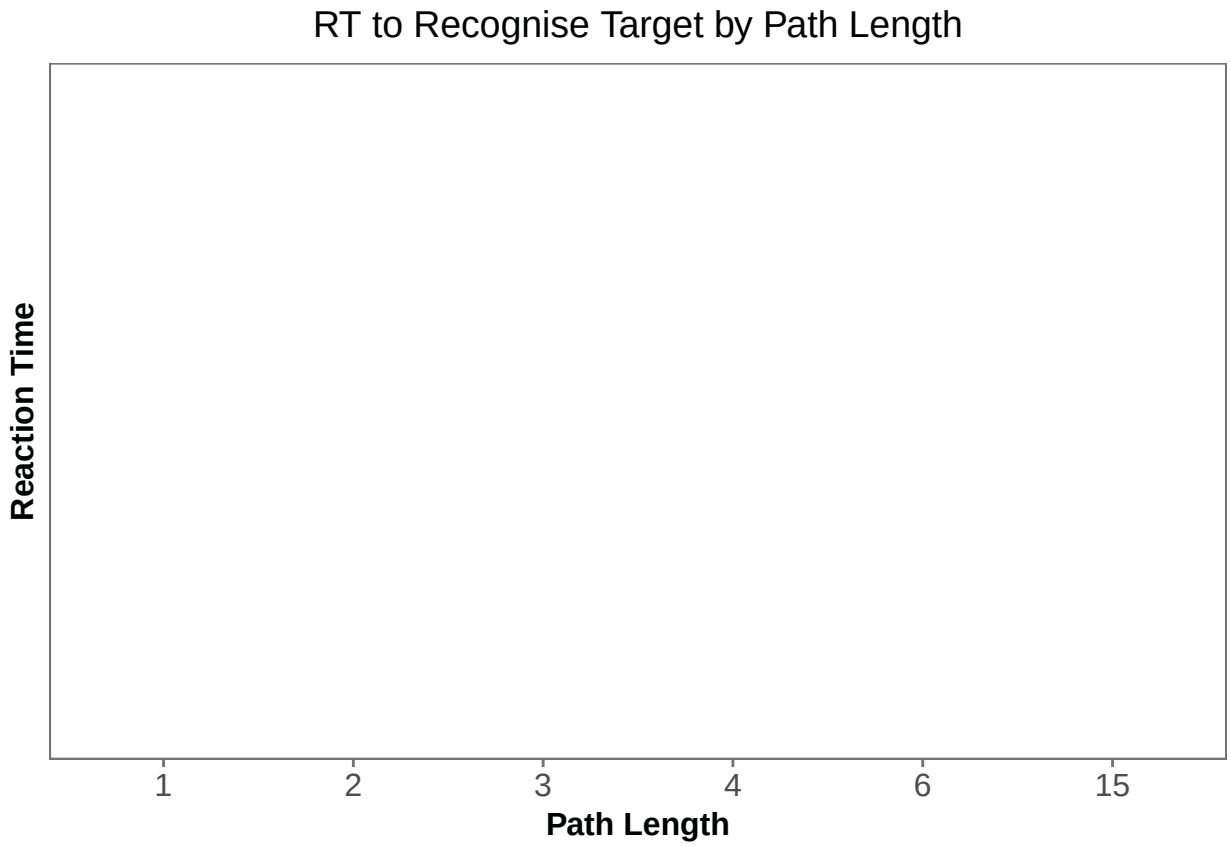
```
##
## Error: subject
##               Df   Sum Sq Mean Sq F value  Pr(>F)
## pathlengthfac  3 15947201 5315734   5.352 0.00676 **
## Residuals     21 20858985  993285
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Error: subject:pathlengthfac
##                Df  Sum Sq Mean Sq F value Pr(>F)
## pathlengthfac   5 1870134  374027   32.78 <2e-16 ***
## Residuals     107 1220925   11411
## ---
## 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)))
```

```
## Warning: Removed 6 rows containing missing values (geom_point).
```

```
## Warning: Removed 6 rows containing missing values (geom_errorbar).
```

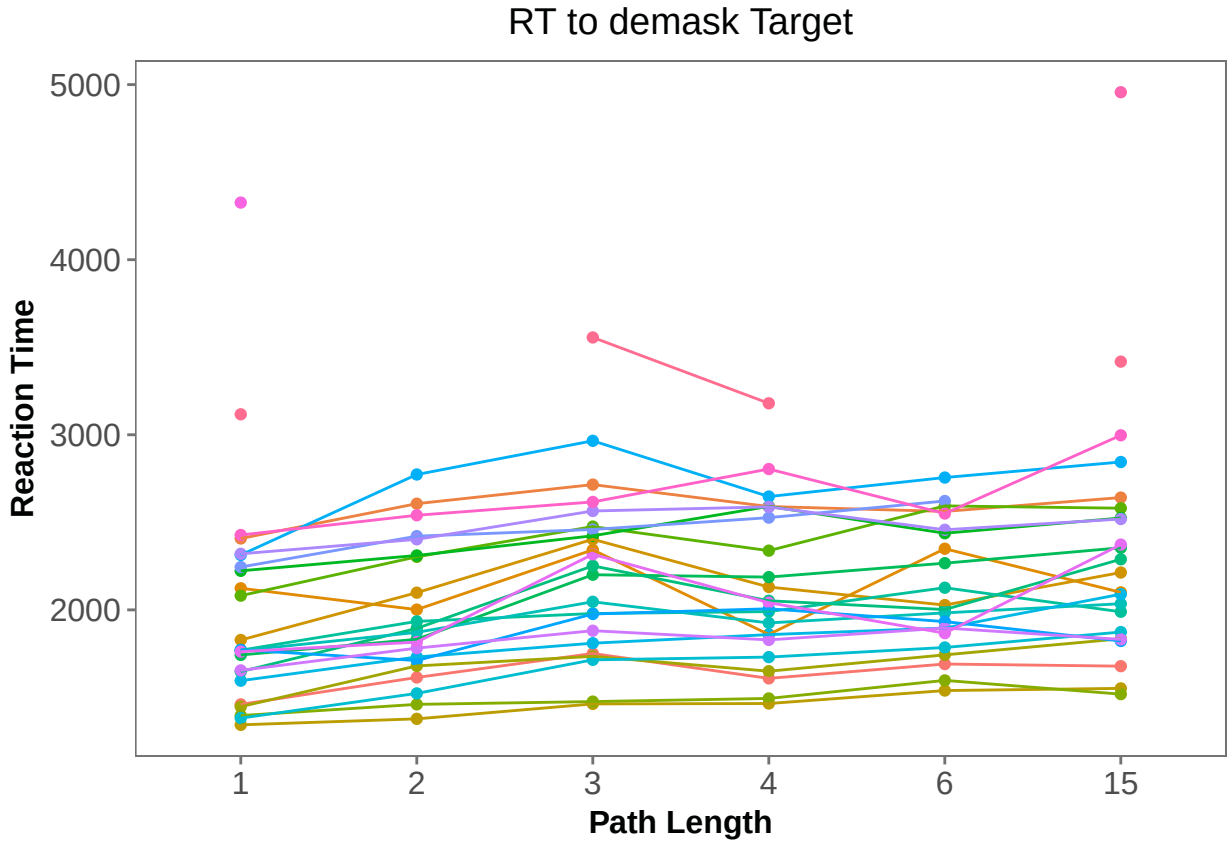


Subject-Wise

```
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)))
```

```
## Warning: Removed 13 rows containing missing values (geom_point).
```

```
## Warning: Removed 11 rows containing missing values (geom_path).
```

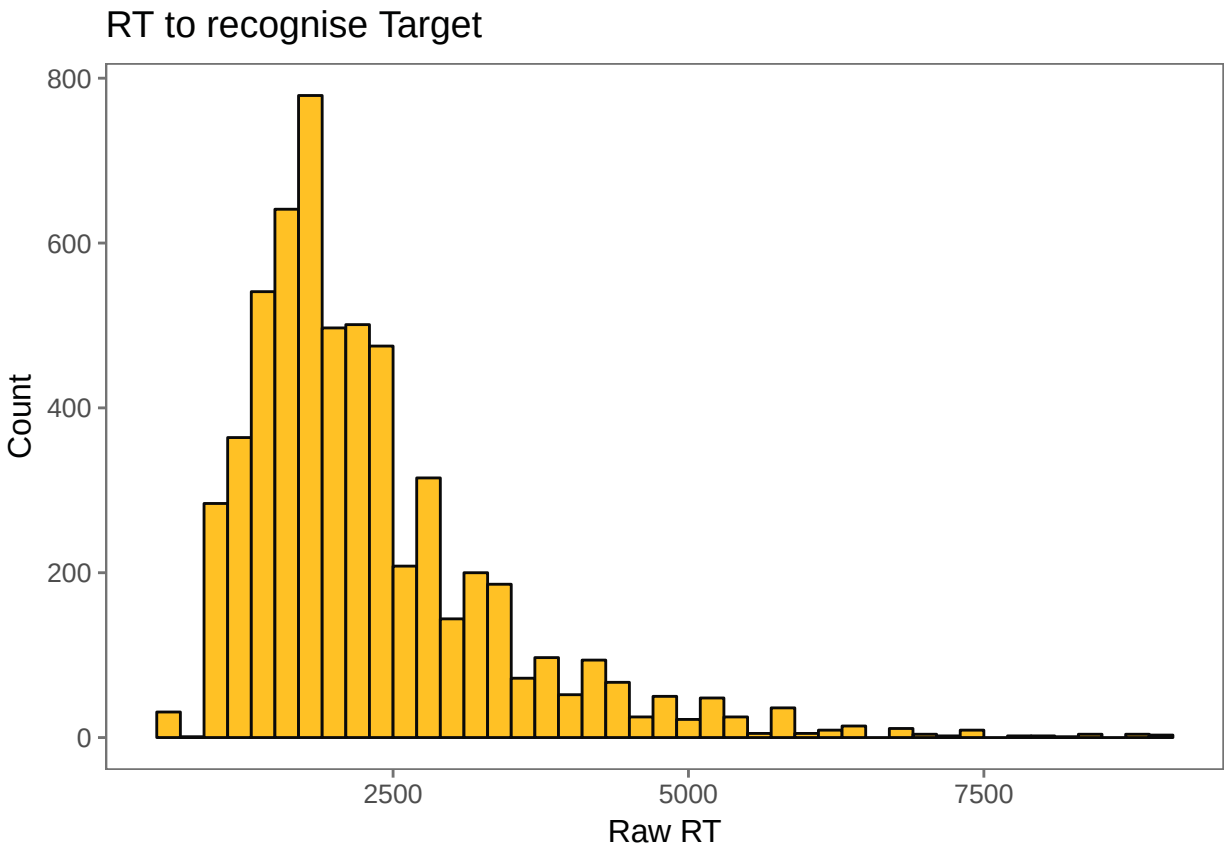


z-scored Reaction Time

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")
```

```
## Warning: Removed 22 rows containing non-finite values (stat_bin).
```

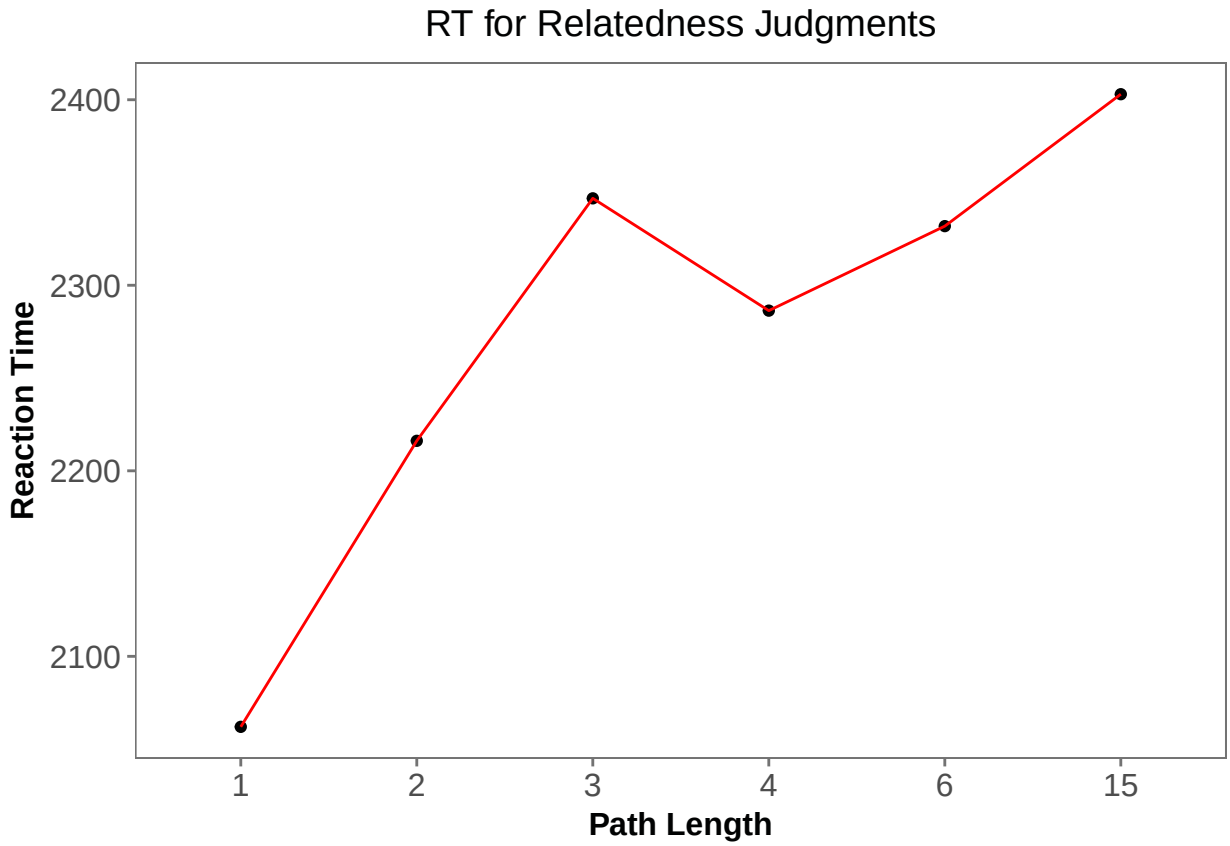


First Trim

```
library(dplyr)
netdemask_firsttrim = netdemask %>% filter(RTRecogniseTarget > 250 &
                                             RTRecogniseTarget < 7000)
```

Raw RT aggregates After Trimming

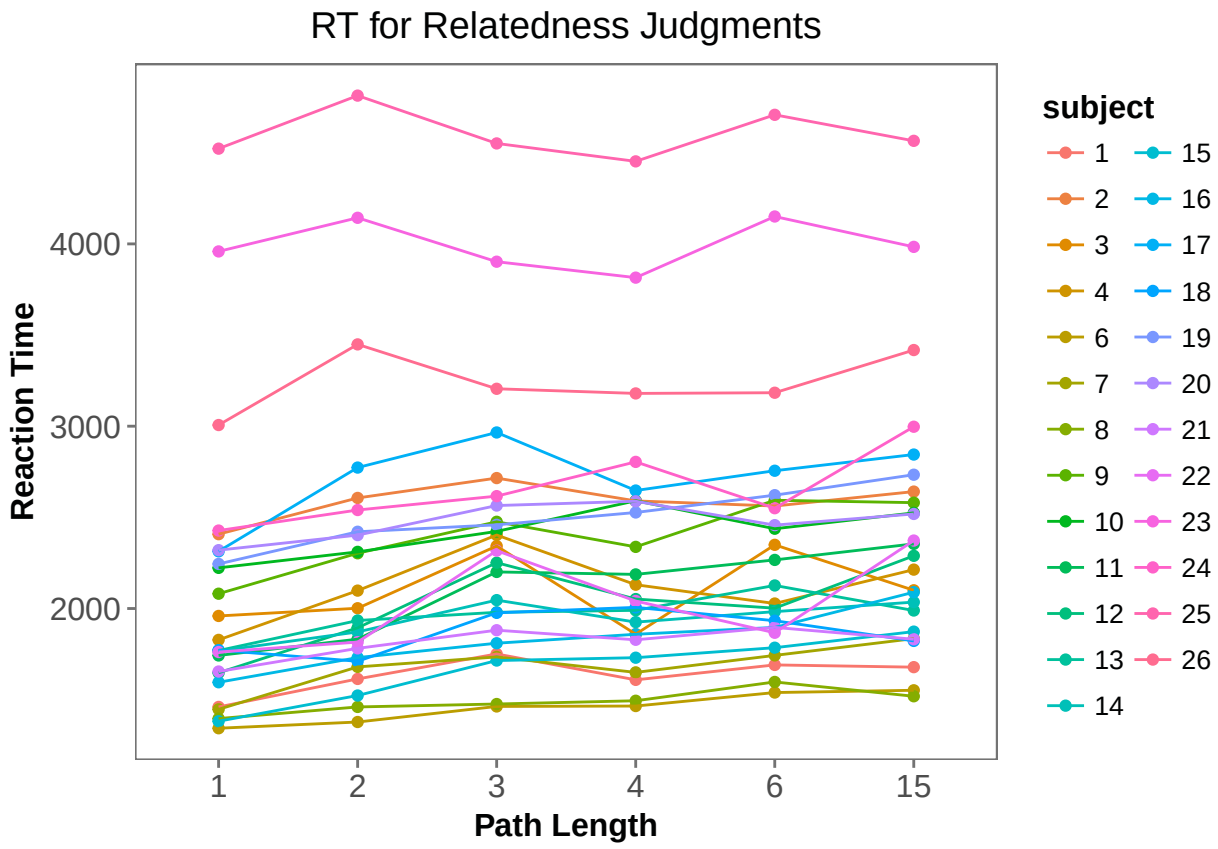
```
netdemask_rt_firsttrim = group_by(netdemask_firsttrim, subject, pathlength ) %>%
  summarise_at(vars(RTRecogniseTarget), mean)

netdemask_rt_agg_firsttrim = group_by(netdemask_firsttrim, pathlength ) %>%
  summarise_at(vars(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)))
```



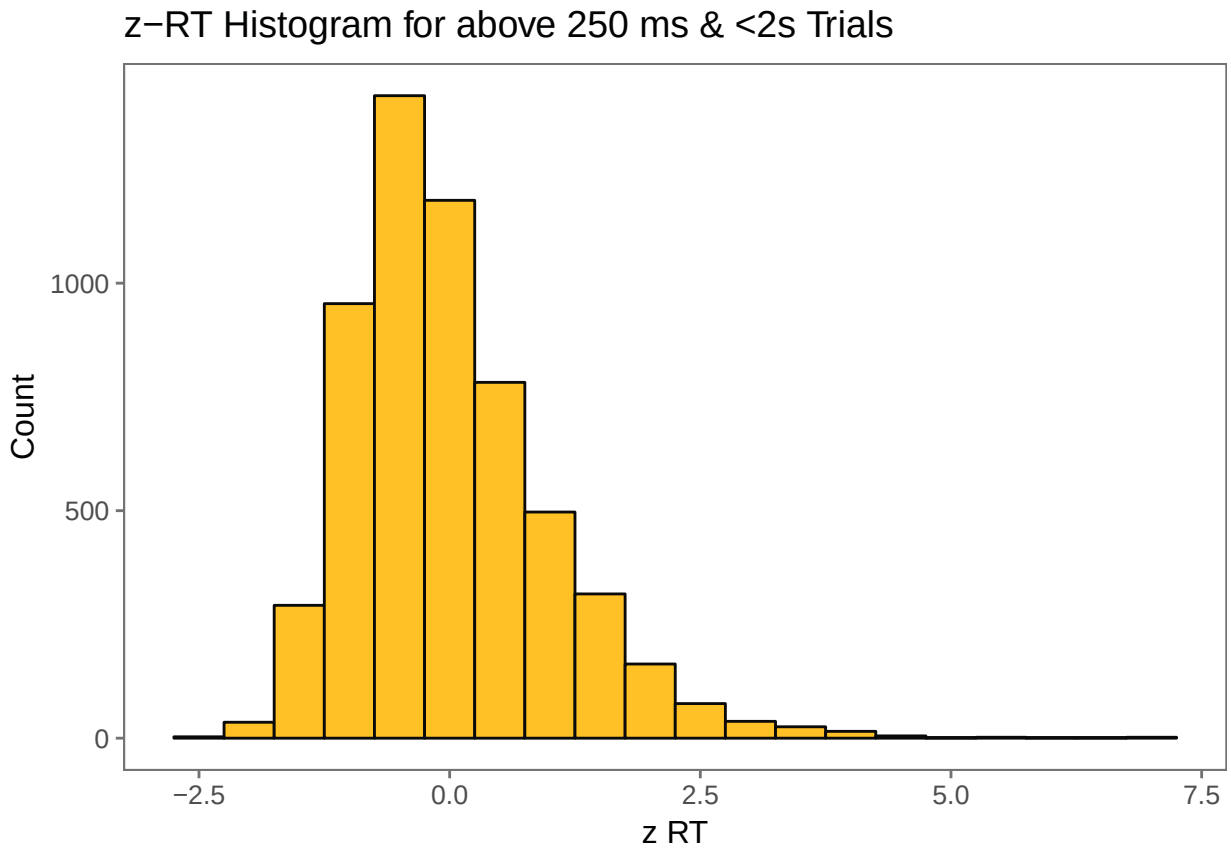
Making the z-scores

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

sdRT = group_by(netdemask_firsttrim, subject) %>%
  summarise_at(vars(RTRecogniseTarget), sd)
colnames(sdRT) = c("subject", "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)

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

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

z-RT Distribution

```
ggplot(netdemask_z, aes(x = zRTTarget))+
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)
```

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)


new_netdemask_z = new_netdemask_z_target
```

Aggregating zRT

```
z_netdemask_rt = group_by(new_netdemask_z, subject, pathlength ) %>%
  summarise_at(vars(zRTTarget_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 24 0.001251 5.212e-05
```

```
##
## Error: subject:pathlengthfac
##                Df Sum Sq Mean Sq F value Pr(>F)
## pathlengthfac   5  4.559  0.9119   32.76 <2e-16 ***
## Residuals     120  3.340  0.0278
## ---
## 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)
```

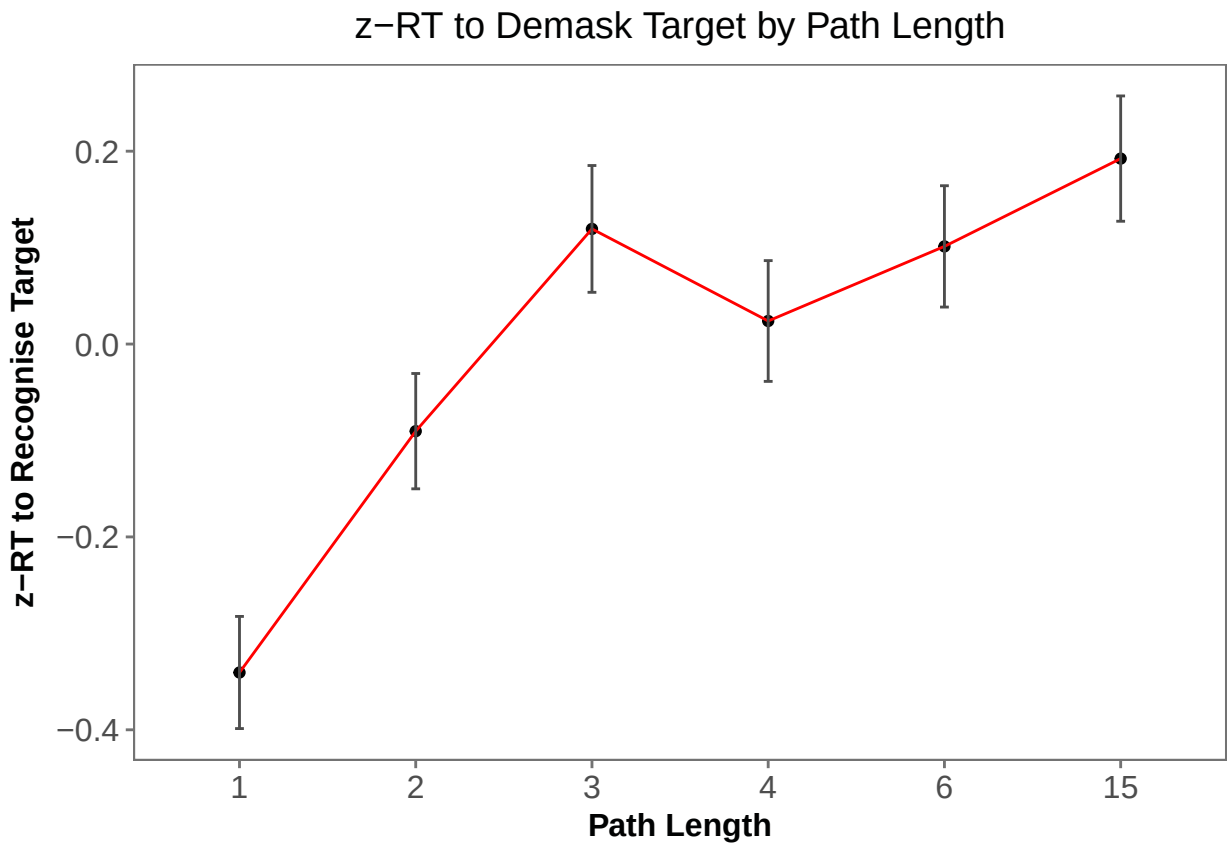
```
## Loading required package: mvtnorm
## Loading required package: survival
## Loading required package: TH.data
## Loading required package: MASS
##
## Attaching package: 'MASS'
## The following object is masked from 'package:dplyr':
##
##     select
##
## Attaching package: 'TH.data'
## The following 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
1	2 - 1	0.2499884	0.0471859	120	5.297948	0.0000079
2	4 - 1	0.3594404	0.0471859	120	7.617540	0.0000000
4	6 - 1	0.4388879	0.0471859	120	9.301253	0.0000000
5	6 - 2	0.1888995	0.0471859	120	4.003305	0.0014898
7	3 - 1	0.4542780	0.0471859	120	9.627413	0.0000000
8	3 - 2	0.2042896	0.0471859	120	4.329464	0.0004396
11	15 - 1	0.5283363	0.0471859	120	11.196914	0.0000000
12	15 - 2	0.2783479	0.0471859	120	5.898965	0.0000005
13	15 - 4	0.1688959	0.0471859	120	3.579373	0.0064736

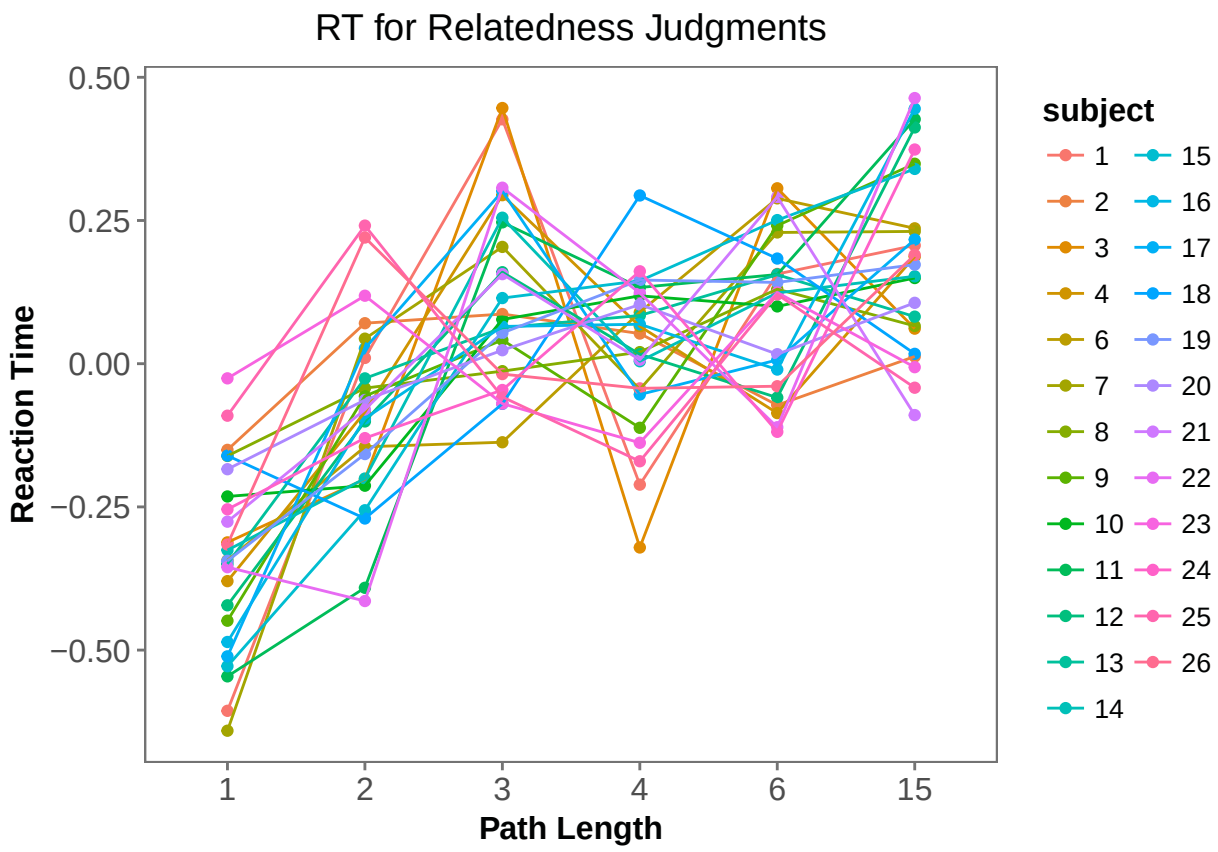
Plotting RTs: collapsed

```
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)))
```



Other Networks

Steyvers Non Directed

```
library(lme4)
```

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

```
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 ~ undirectedfac +
                            (1|subject) + (1|ItemNumber))
summary(RTprime_undirected)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRTTarget_trim ~ undirectedfac + (1 | subject) + (1 | ItemNumber)
##    Data: new_netdemask_z
##
## REML criterion at convergence: 15677.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.8121 -0.6945 -0.1422  0.5756  3.9875
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.1141   0.3378
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.8484   0.9211
## Number of obs: 5731, groups:  ItemNumber, 240; subject, 25
##
## Fixed effects:
##                Estimate Std. Error t value
## (Intercept)     0.39127    0.09142   4.280
## undirectedfac1 -0.75338    0.12434  -6.059
## undirectedfac2 -0.50956    0.09950  -5.121
## undirectedfac3 -0.25989    0.09898  -2.626
##
## Correlation of Fixed Effects:
##             (Intr) undrc1 undrc2
## undirctdfc1 -0.735
## undirctdfc2 -0.919  0.675
## undirctdfc3 -0.924  0.679  0.849
```

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

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
```

```
##                Chisq Df Pr(>Chisq)
## undirectedfac 57.759  3  1.769e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

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

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRTTarget_trim ~ Undirected + I(Undirected^2) + (1 | subject) +
##     (1 | ItemNumber)
##    Data: new_netdemask_z
##
## REML criterion at convergence: 15677.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.8124 -0.6947 -0.1425  0.5753  3.9874
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.1135   0.3369
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.8484   0.9211
## Number of obs: 5731, groups:  ItemNumber, 240; subject, 25
##
## Fixed effects:
##                   Estimate Std. Error t value
## (Intercept)     -0.596004   0.205345  -2.902
## Undirected       0.230624   0.170281   1.354
## I(Undirected^2)  0.003995   0.033861   0.118
##
## Correlation of Fixed Effects:
##             (Intr) Undrct
## Undirected  -0.969
## I(Undrct^2)  0.909 -0.981
```

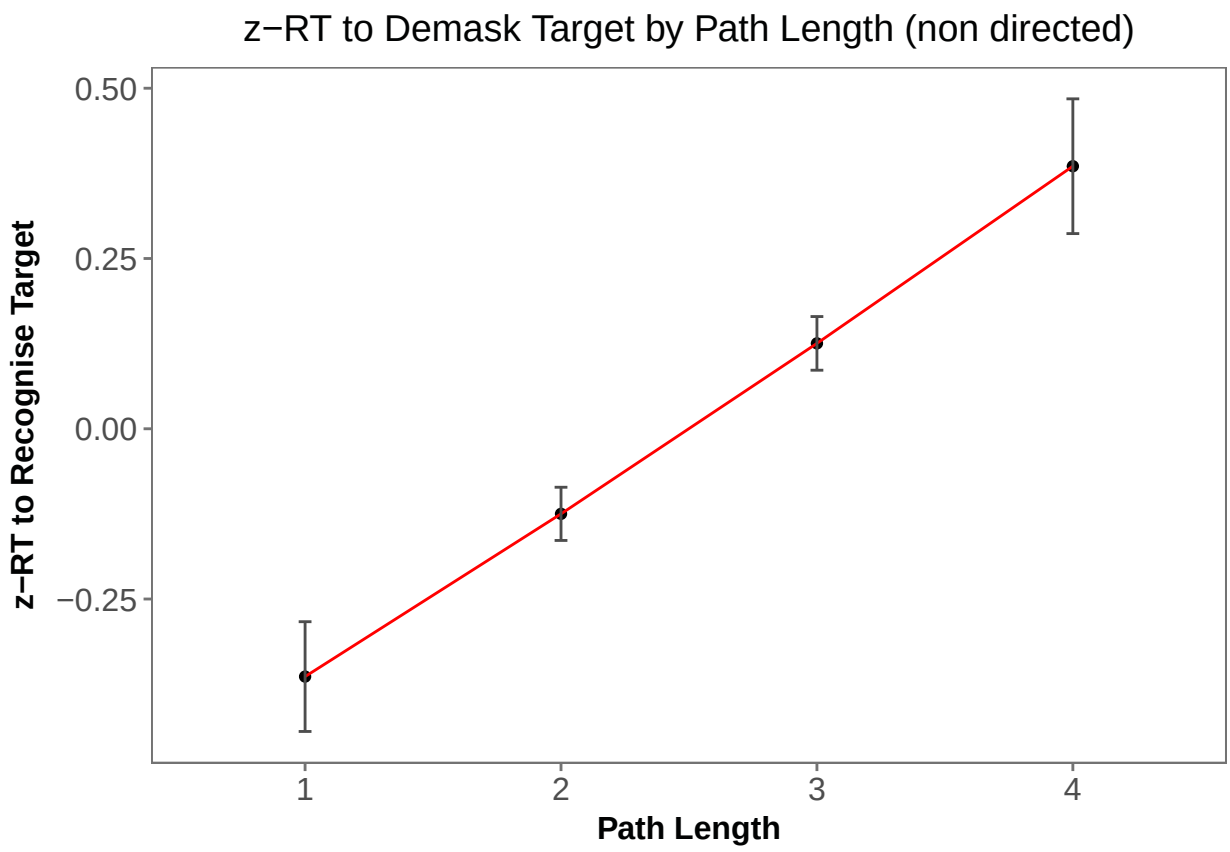
```
car::Anova(RTprime_undirected_quad)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##                   Chisq Df Pr(>Chisq)
## Undirected      1.8343  1     0.1756
## I(Undirected^2) 0.0139  1     0.9061
```

Plot

```
z_rmisc_undirected = Rmisc::summarySE(new_netdemask_z,
                        measurevar = "zRTTarget_trim",
                        groupvars = c("Undirected"))
```

```r
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 ~ directedfac +
                            (1|subject) + (1|ItemNumber))
summary(RTprime_directed)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRTTarget_trim ~ directedfac + (1 | subject) + (1 | ItemNumber)
##    Data: new_netdemask_z
##
## REML criterion at convergence: 15169.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.7964 -0.6983 -0.1492  0.5834  3.9866
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  ItemNumber (Intercept) 0.1048   0.3238
##  subject    (Intercept) 0.0000   0.0000
##  Residual               0.8424   0.9178
## Number of obs: 5560, groups:  ItemNumber, 240; subject, 25
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)   0.10505    0.04177   2.515
## directedfac1 -0.37914    0.08501  -4.460
## directedfac2 -0.29164    0.05606  -5.202
## directedfac3 -0.15993    0.05109  -3.130
```

```
## directedfac4 -0.07908    0.04708  -1.680
## directedfac6  0.16047    0.07221   2.222
## directedfac7  0.30734    0.18308   1.679
## directedfac8  0.36010    0.24168   1.490
##
## Correlation of Fixed Effects:
##             (Intr) drctd1 drctd2 drctd3 drctd4 drctd6 drctd7
## directedfc1 -0.448
## directedfc2 -0.621  0.367
## directedfc3 -0.659  0.349  0.513
## directedfc4 -0.703  0.374  0.513  0.576
## directedfc6 -0.419  0.228  0.340  0.312  0.372
## directedfc7 -0.204  0.099  0.137  0.149  0.209  0.096
## directedfc8 -0.154  0.079  0.143  0.145  0.121  0.072  0.033
```

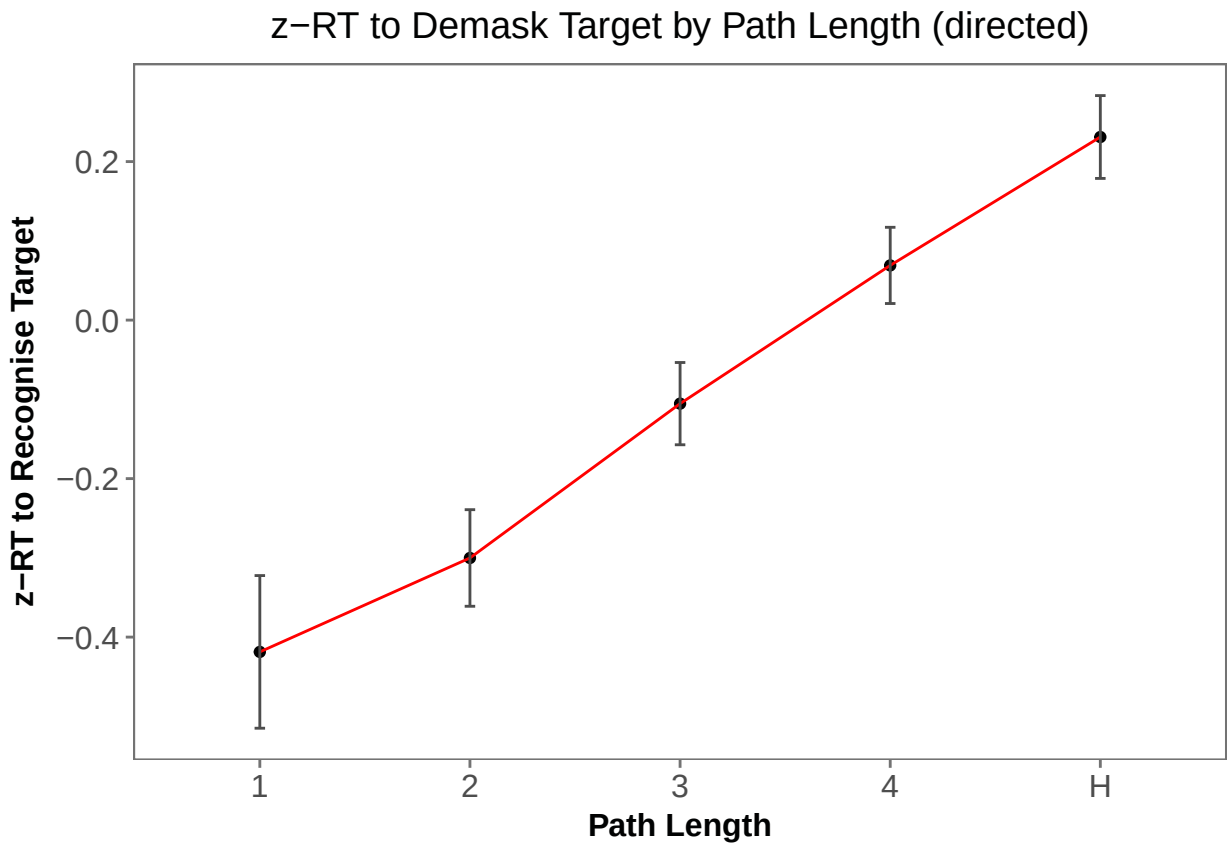
```
car::Anova(RTprime_directed)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: zRTTarget_trim
##              Chisq Df Pr(>Chisq)
## directedfac 67.907  7  3.908e-12 ***
## ---
## 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)))
```