

Semantic Networks Analysis

Abhilasha Kumar

January 16, 2018

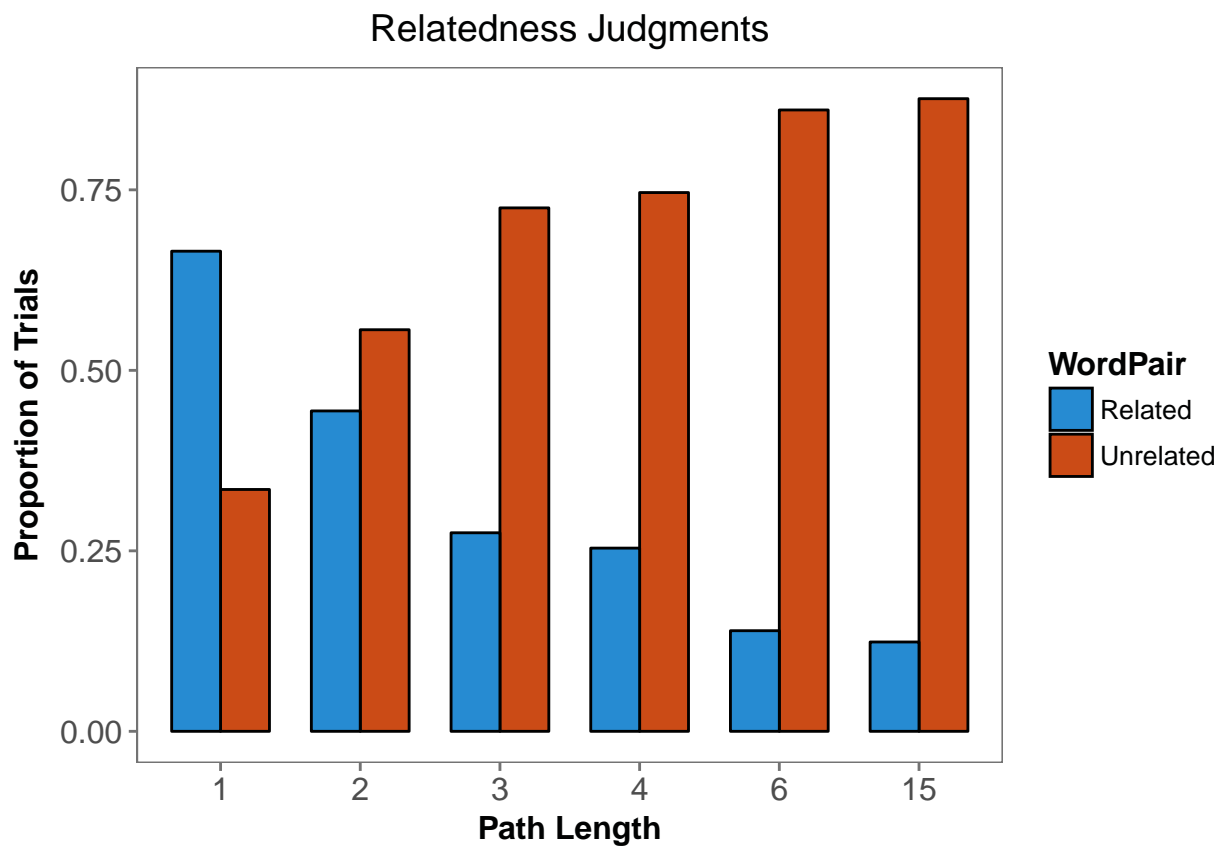
Reading the Data

```
## Warning: NAs introduced by coercion
## Warning: NAs introduced by coercion
## Warning: NAs introduced by coercion
## Warning: NAs introduced by coercion
## Warning: NAs introduced by coercion
## Warning: NAs introduced by coercion
## Warning: NAs introduced by coercion
```

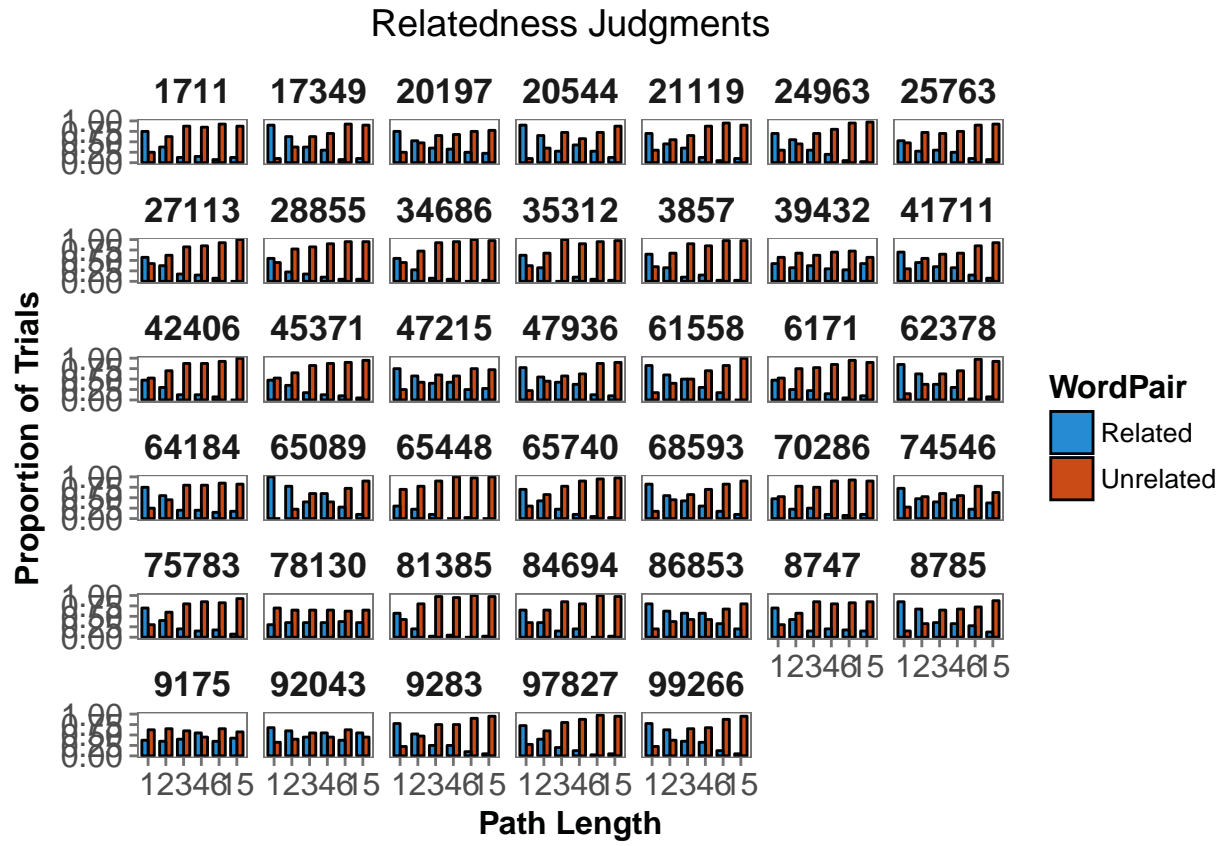
Related-Unrelated Decisions

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##   filter, lag
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

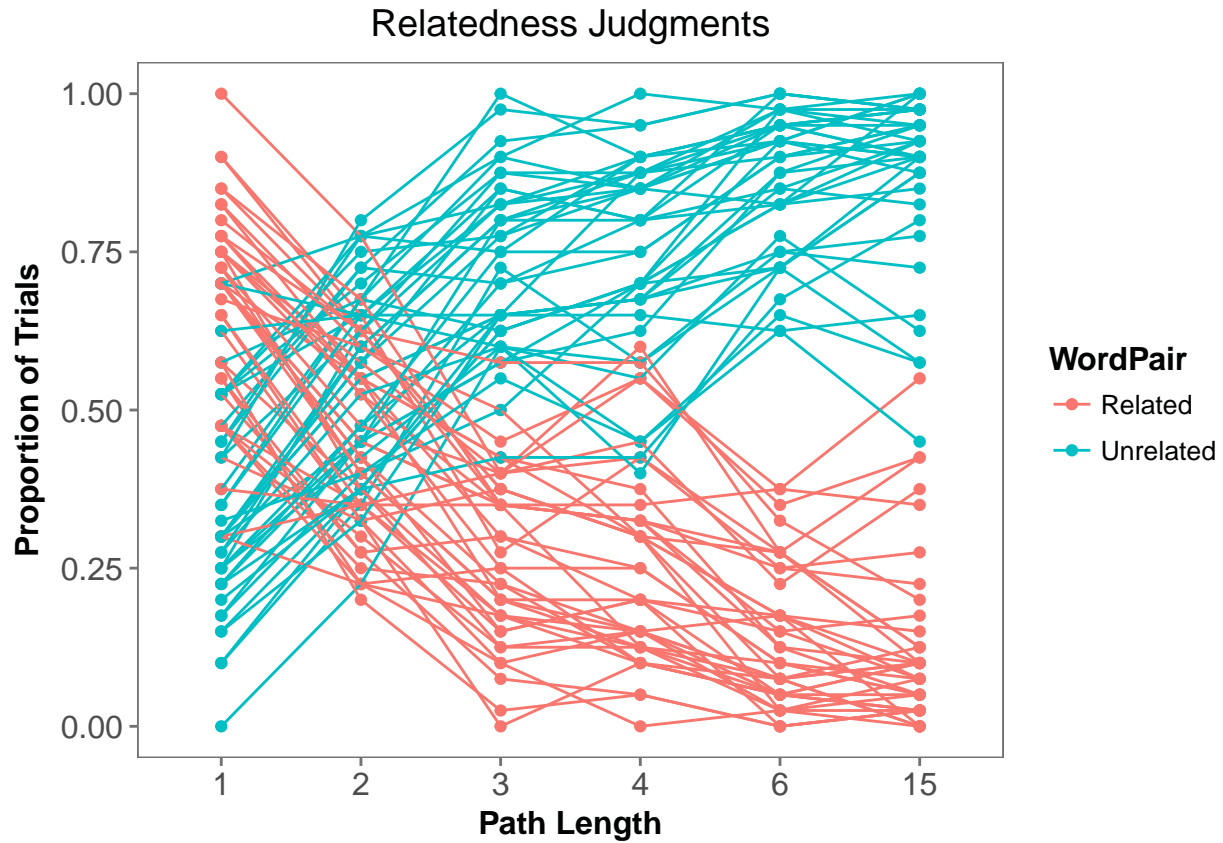
Plotting Proportions



Subject-Wise



Line Plot Subject-Wise



ANOVA

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

```
##
## Error: subject
##           Df    Sum Sq  Mean Sq F value Pr(>F)
## Residuals 39 3.634e-29 9.319e-31
##
## Error: subject:pathlengthfac
##           Df    Sum Sq  Mean Sq F value Pr(>F)
## pathlengthfac  5 1.700e-29 3.399e-30  1.523  0.184
## Residuals    195 4.353e-28 2.232e-30
##
## Error: subject:Type
##           Df Sum Sq Mean Sq F value  Pr(>F)
## Type       1 16.115  16.115  111.9 5.08e-13 ***
## Residuals 39  5.616   0.144
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

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

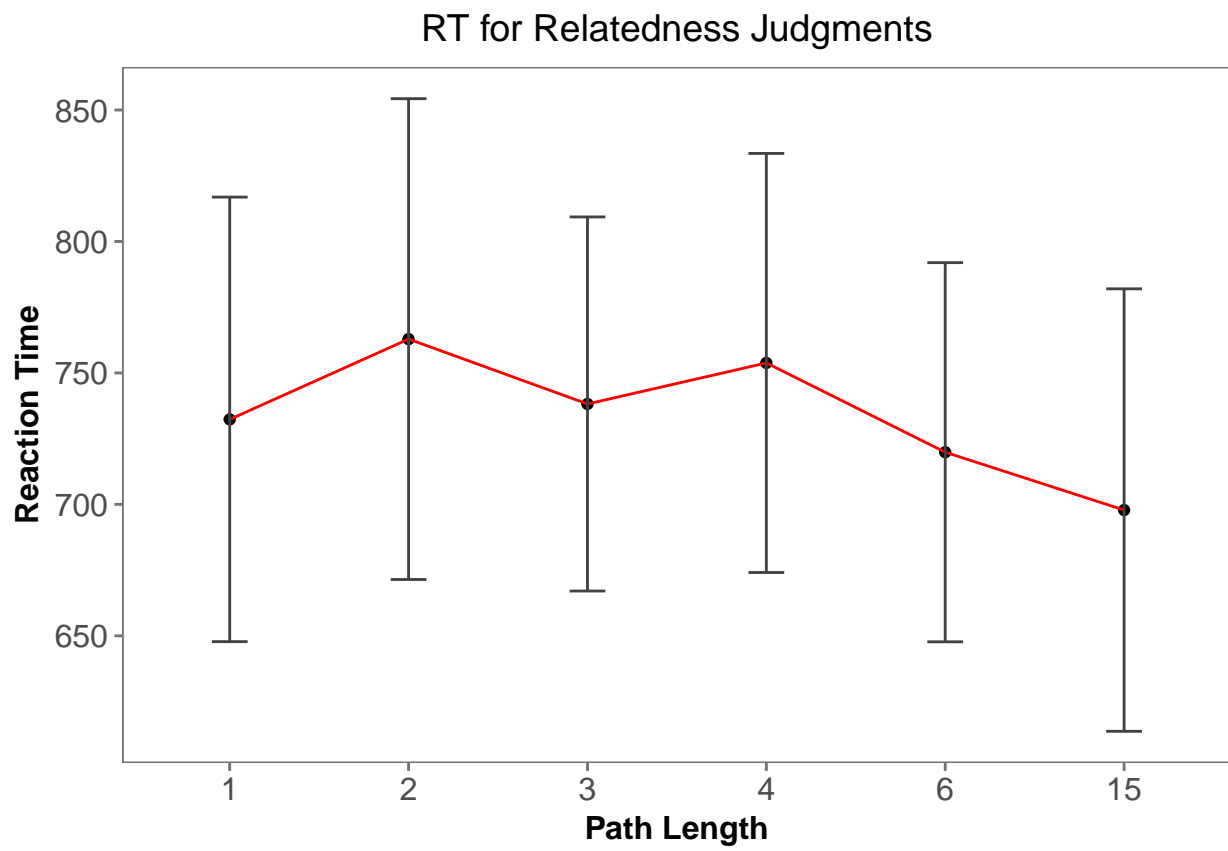
Raw Reaction Time

ANOVA

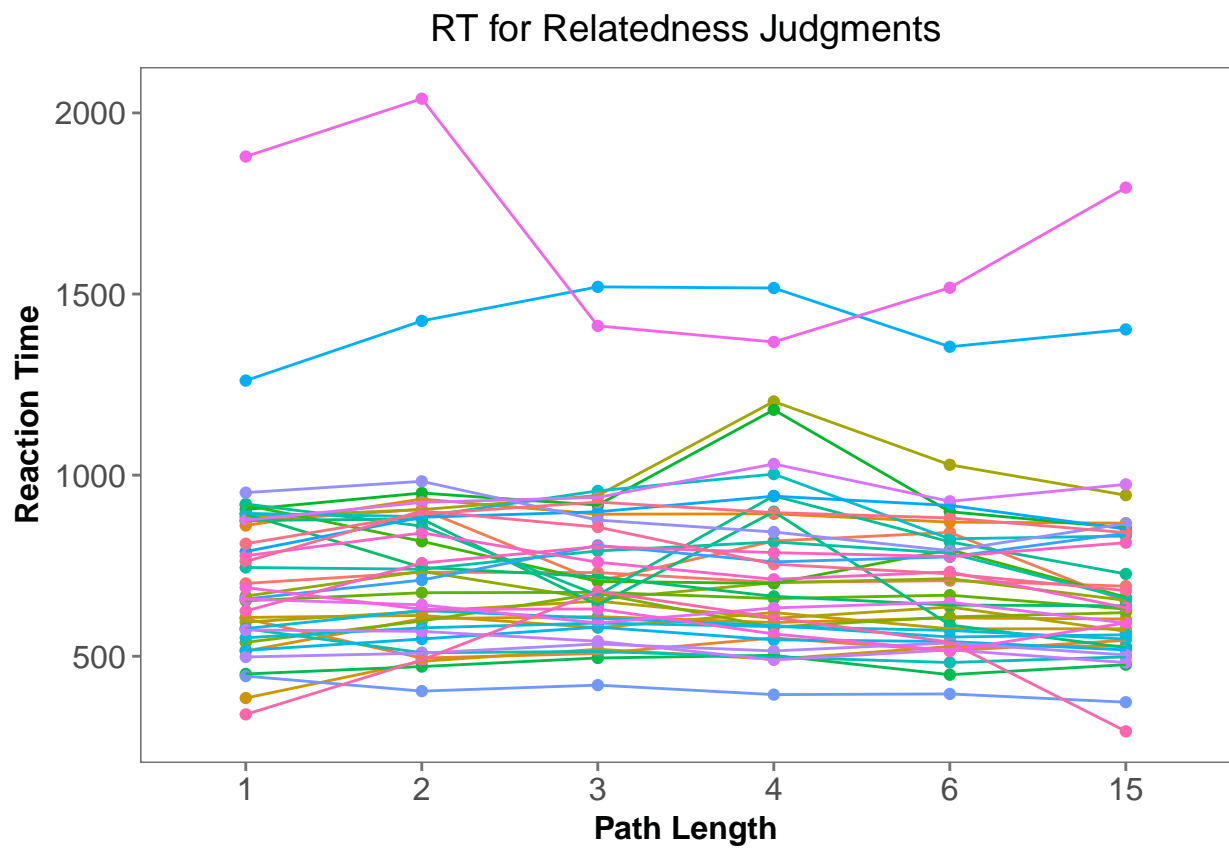
```
sem_rt$pathlengthfac = ordered(as.factor(as.character(sem_rt$pathlength)),
                               levels = c("1", "2", "3", "4", "6", "15"))
sem_rt$subject = as.factor(sem_rt$subject)
rt_aov = aov(data = sem_rt, rt ~ pathlengthfac +
              Error(subject/(pathlengthfac)))
summary(rt_aov)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

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

Plotting RTs



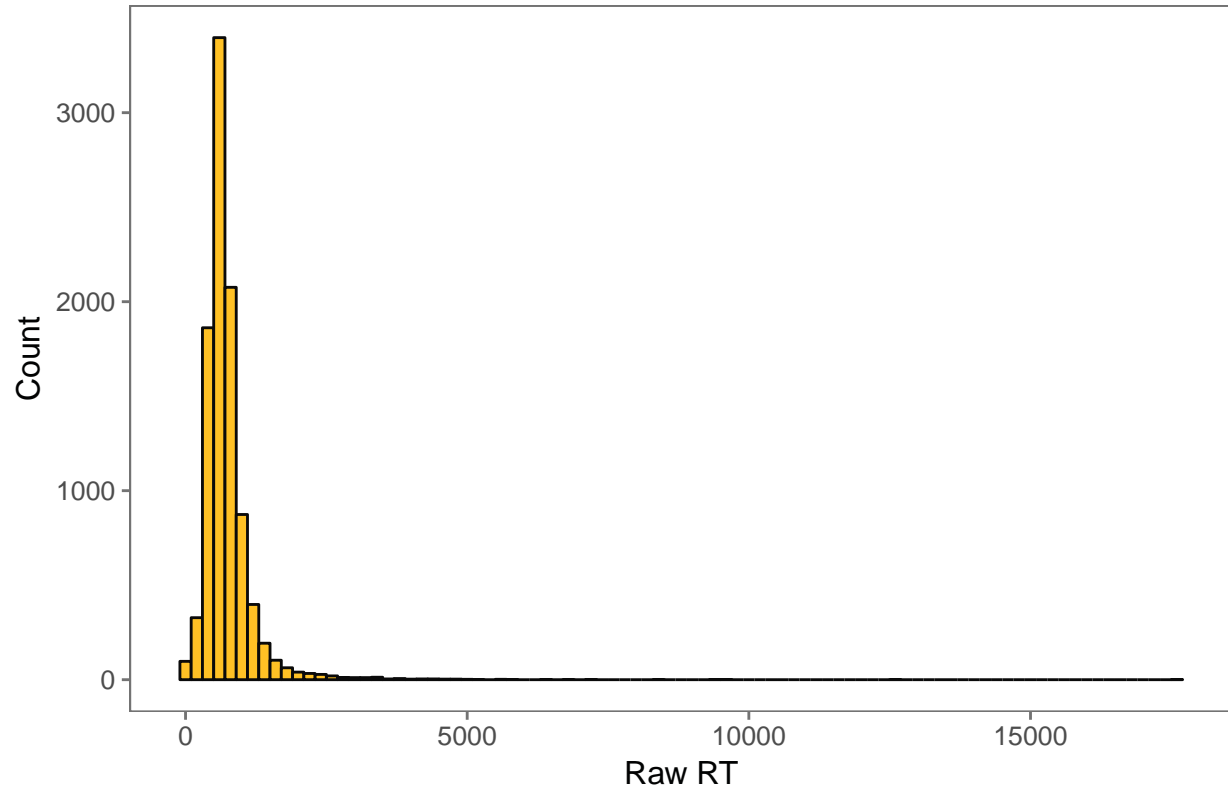
Subject-Wise



z-scored Reaction Time

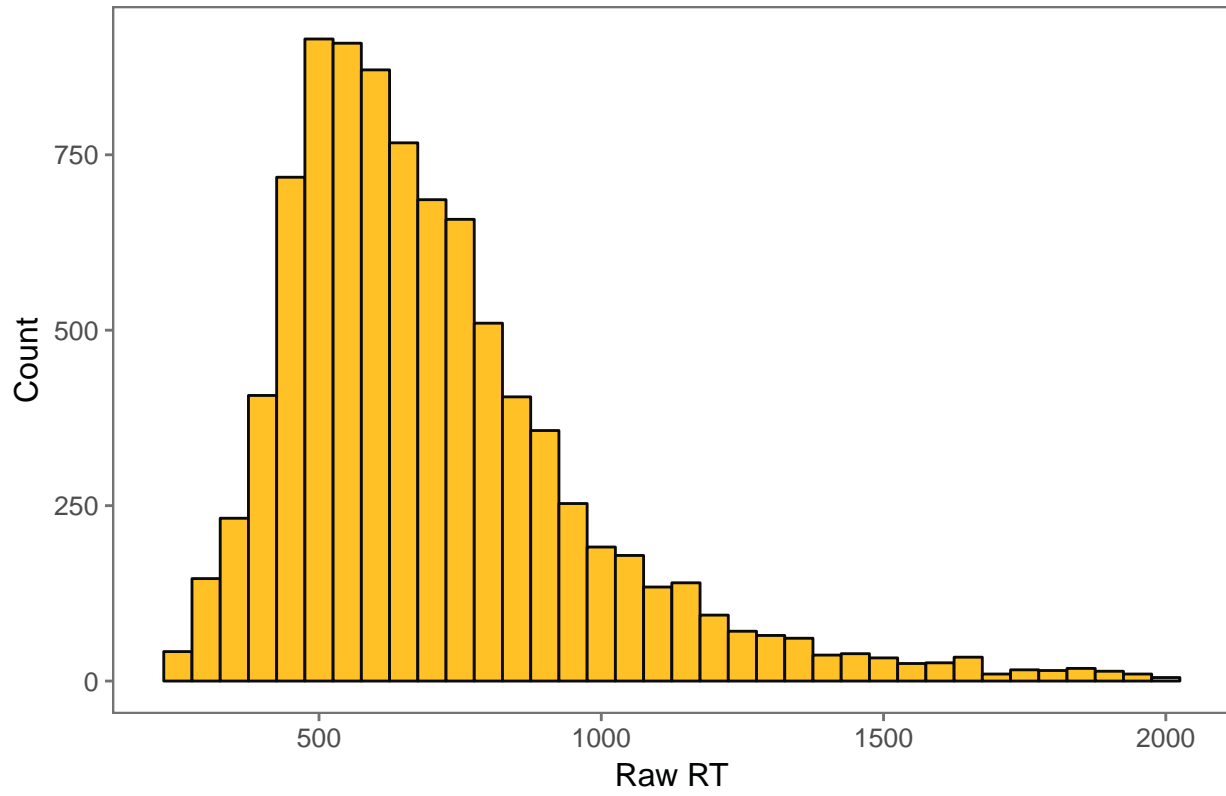
Histogram of RT

Raw RT Histogram for All Trials

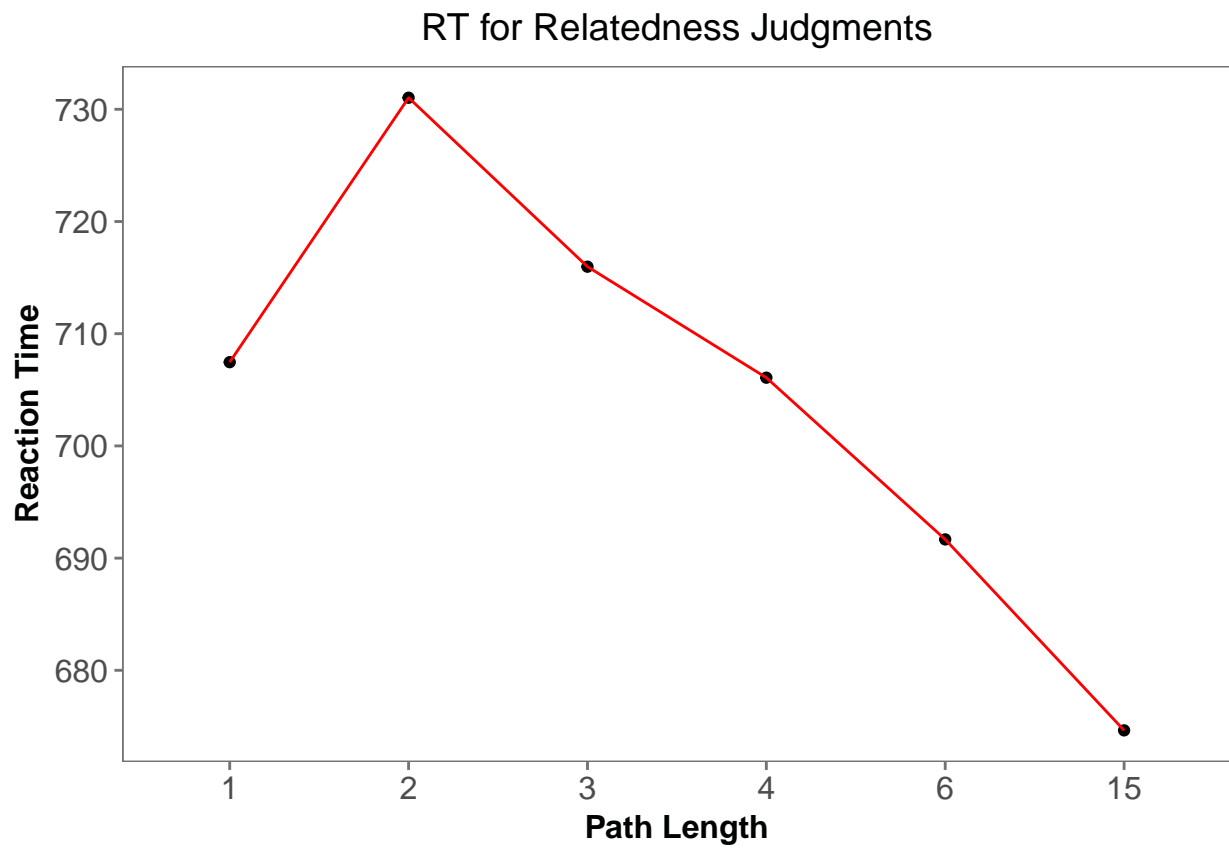


First Trim

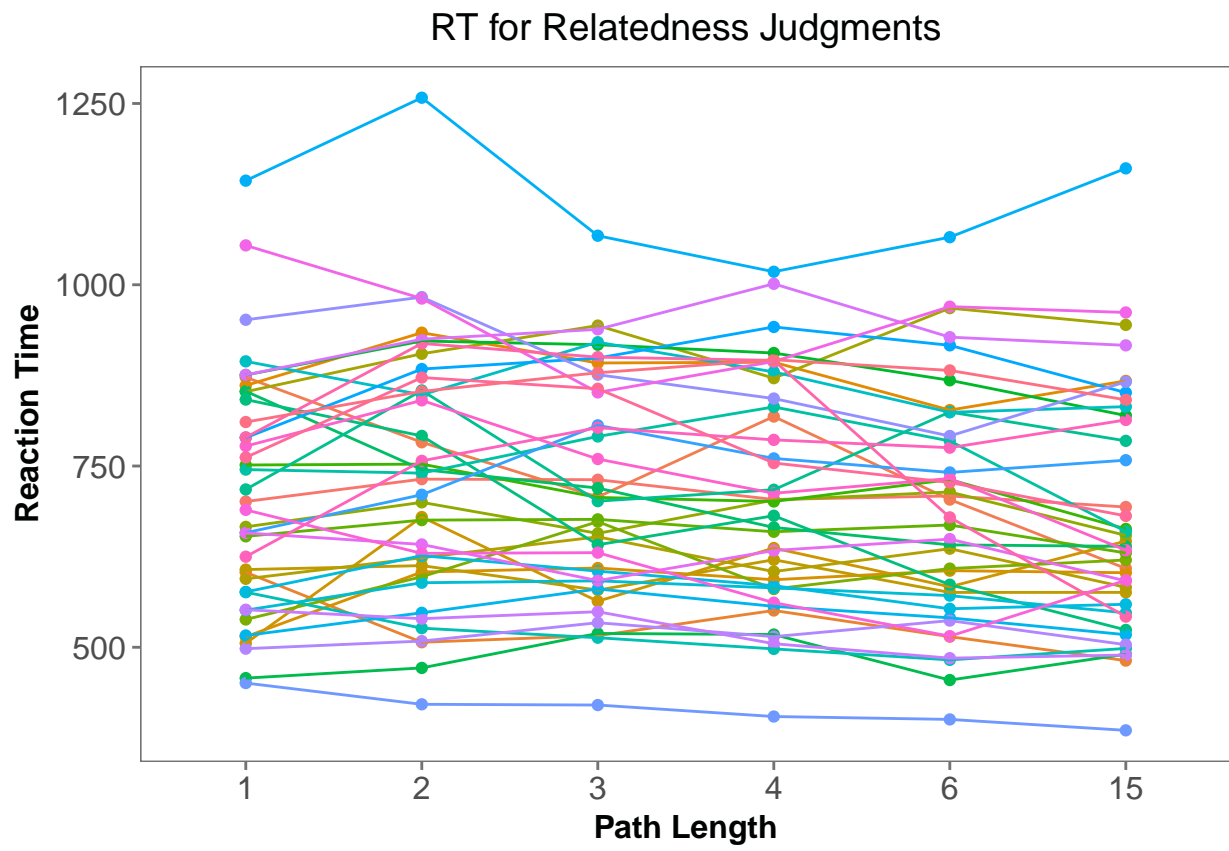
Raw RT Histogram for Trials Above 250 ms and below 2000 ms



Raw RT aggregates After Trimming



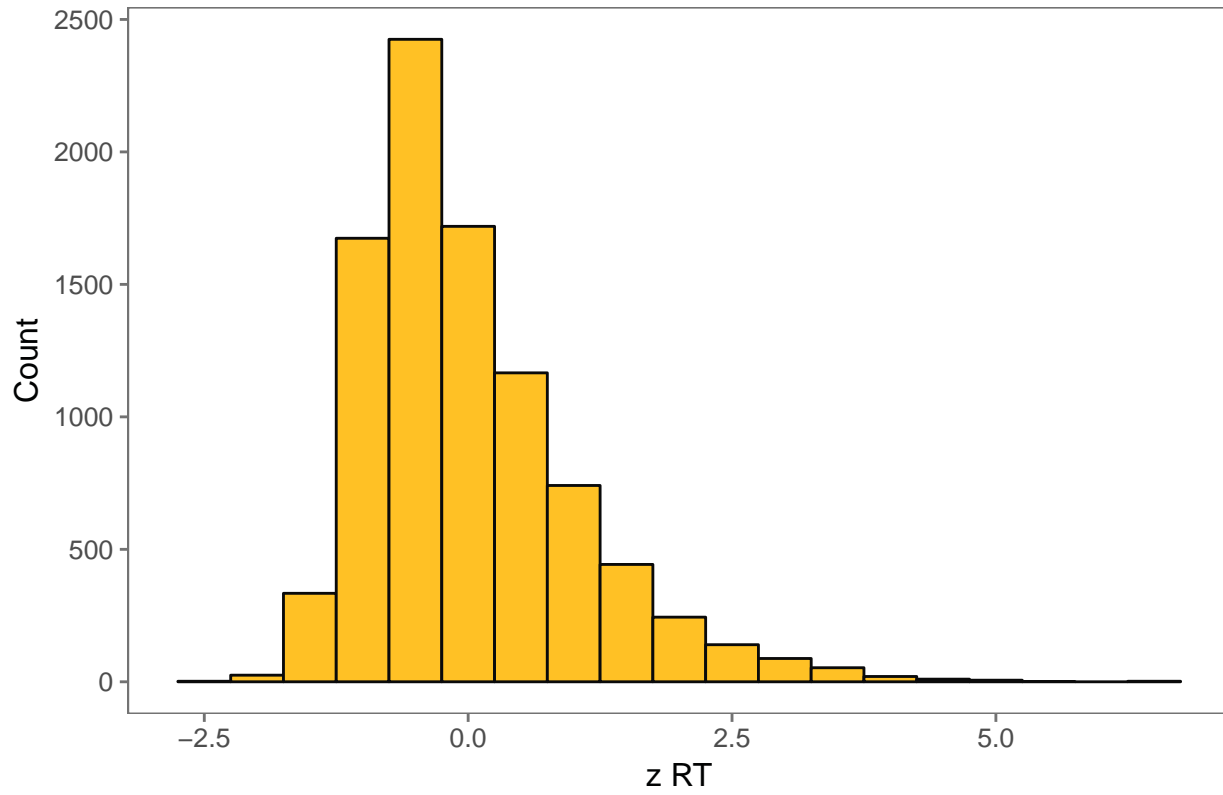
Subject Raw RT again



Making the z-scores

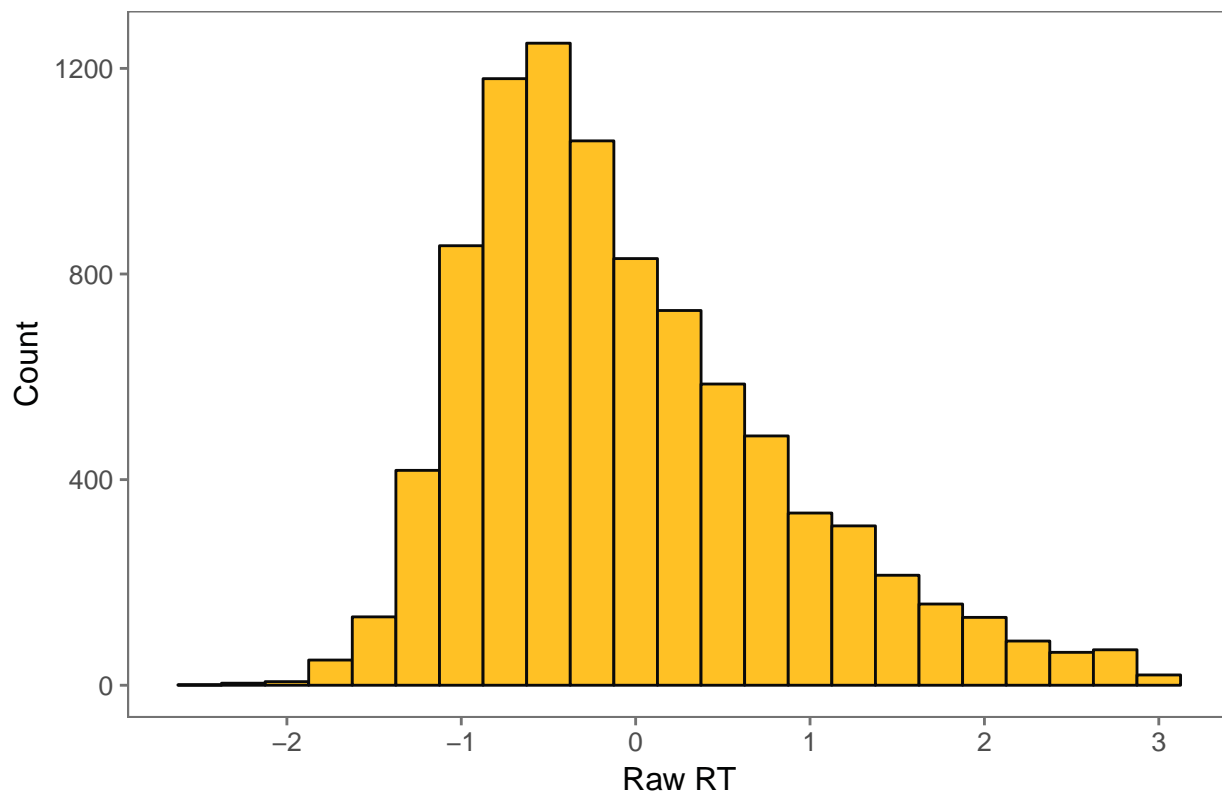
z-RT Distribution

z-RT Histogram for above 250 ms & <2s Trials



Trimming z-RT

Trimmed (3 SD) z-RT Histogram for above 250 ms & <2s Trials



Repeating z-scoring

Aggregating zRT

ANOVA

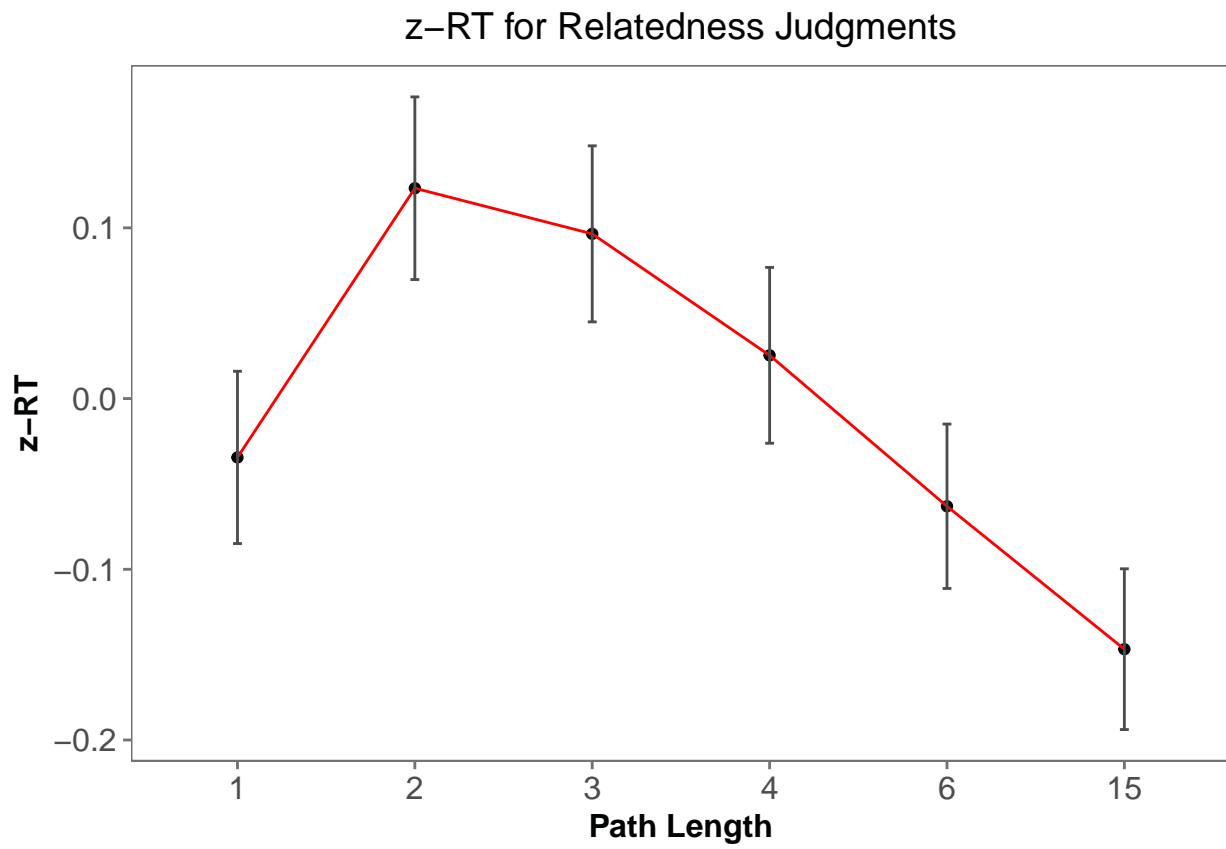
```
z_sem_rt$pathlengthfac = ordered(as.factor(as.character(z_sem_rt$pathlength)),
                                  levels = c("1", "2", "3", "4", "6", "15"))
z_sem_rt$subject = as.factor(z_sem_rt$subject)

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

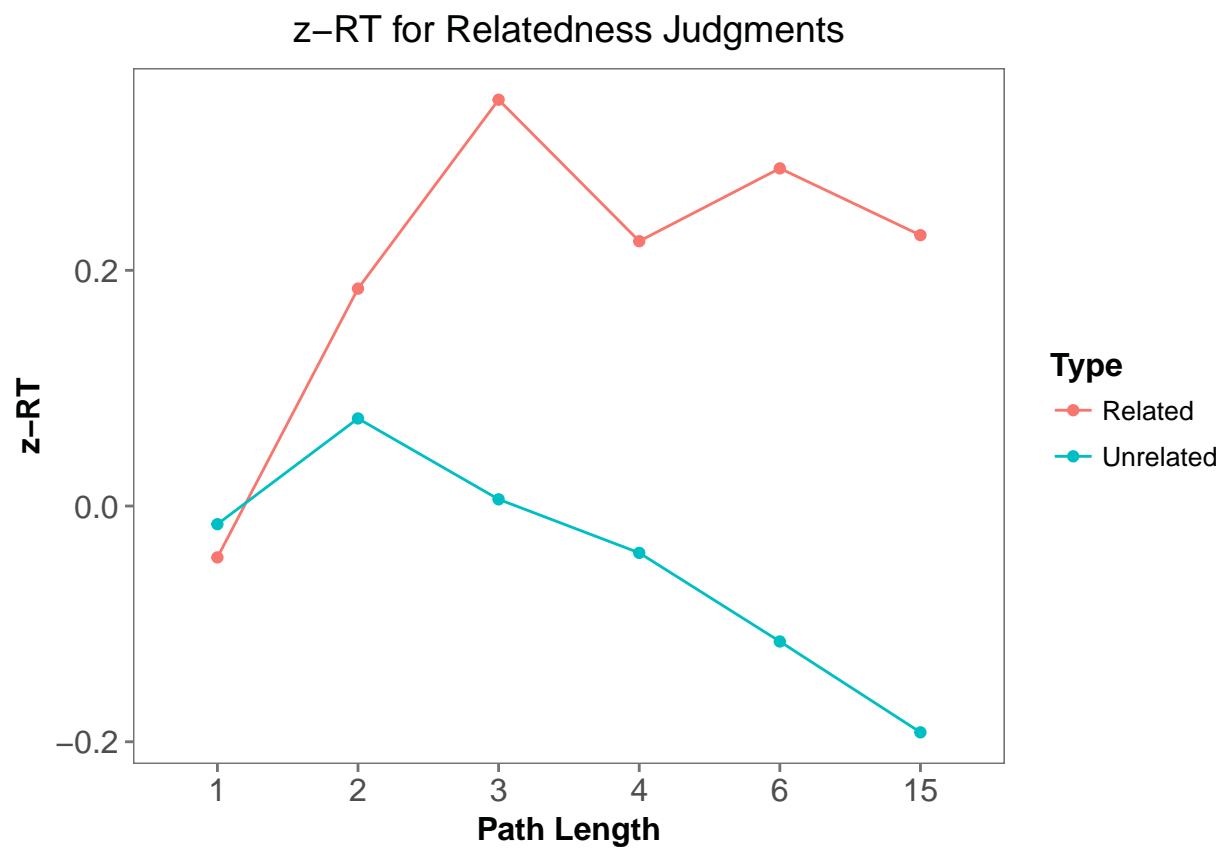
```
##
## Error: subject
##           Df Sum Sq Mean Sq F value Pr(>F)
## Residuals 39 0.005412 0.0001388
##
## Error: subject:pathlengthfac
##           Df Sum Sq Mean Sq F value Pr(>F)
## pathlengthfac 5 2.143 0.4287 9.764 2.43e-08 ***
## Residuals 195 8.561 0.0439
```

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

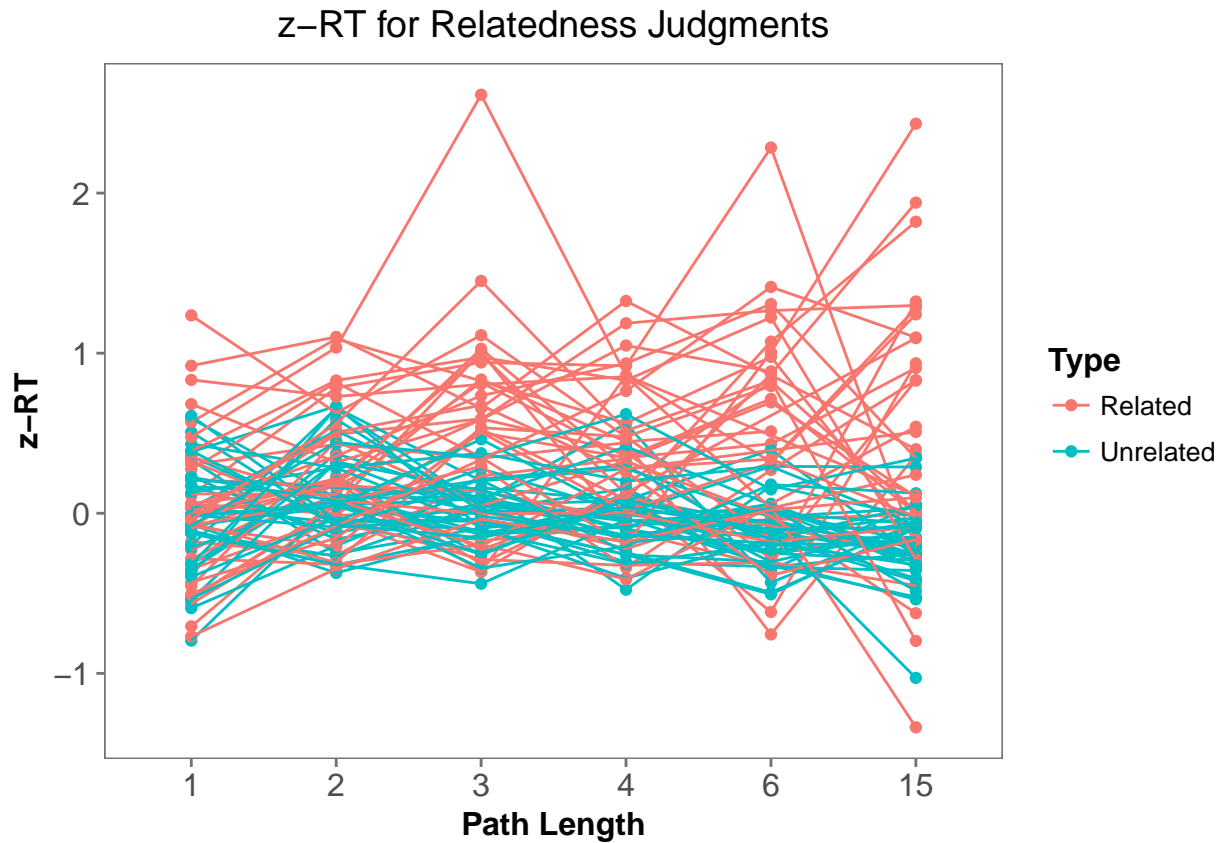
Plotting RTs: collapsed



Plotting RTs: Rel-Unrel

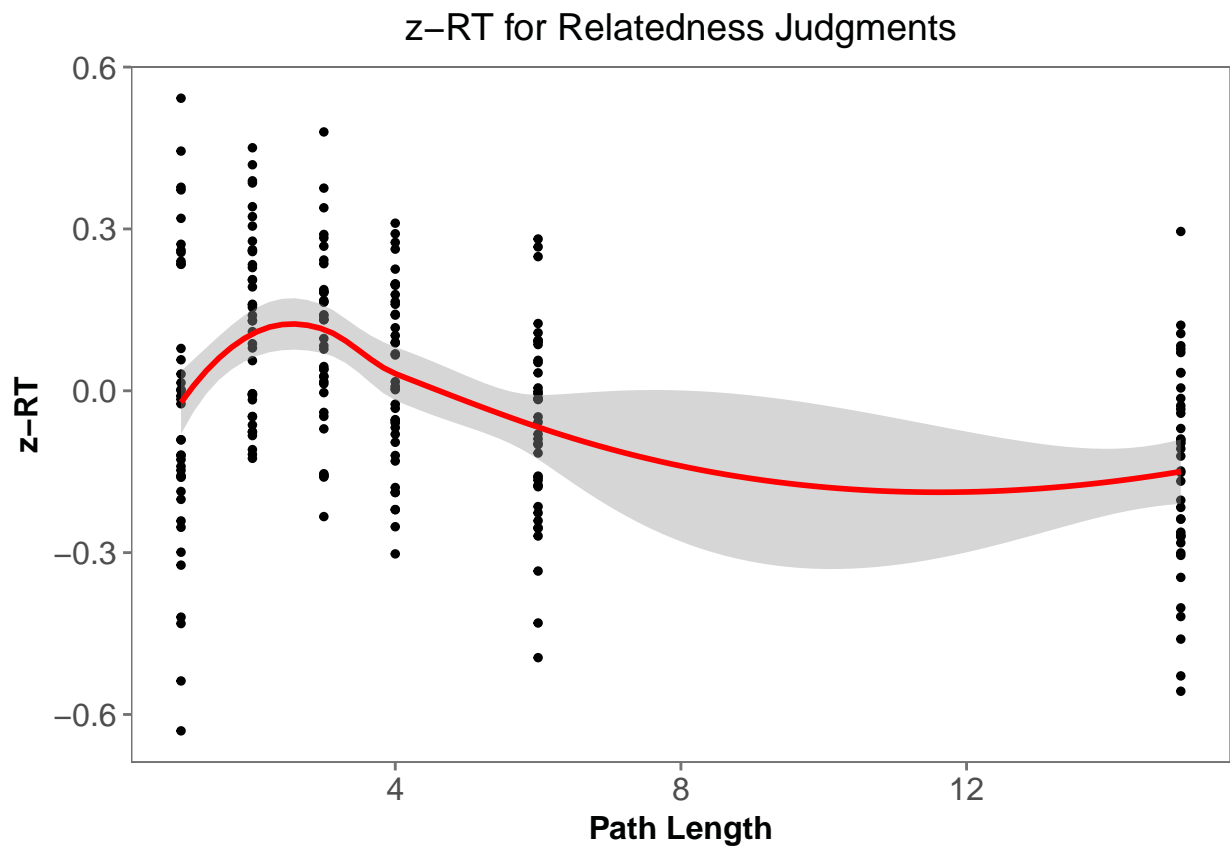


Plotting RTs: Subject Wise Rel-Unrel

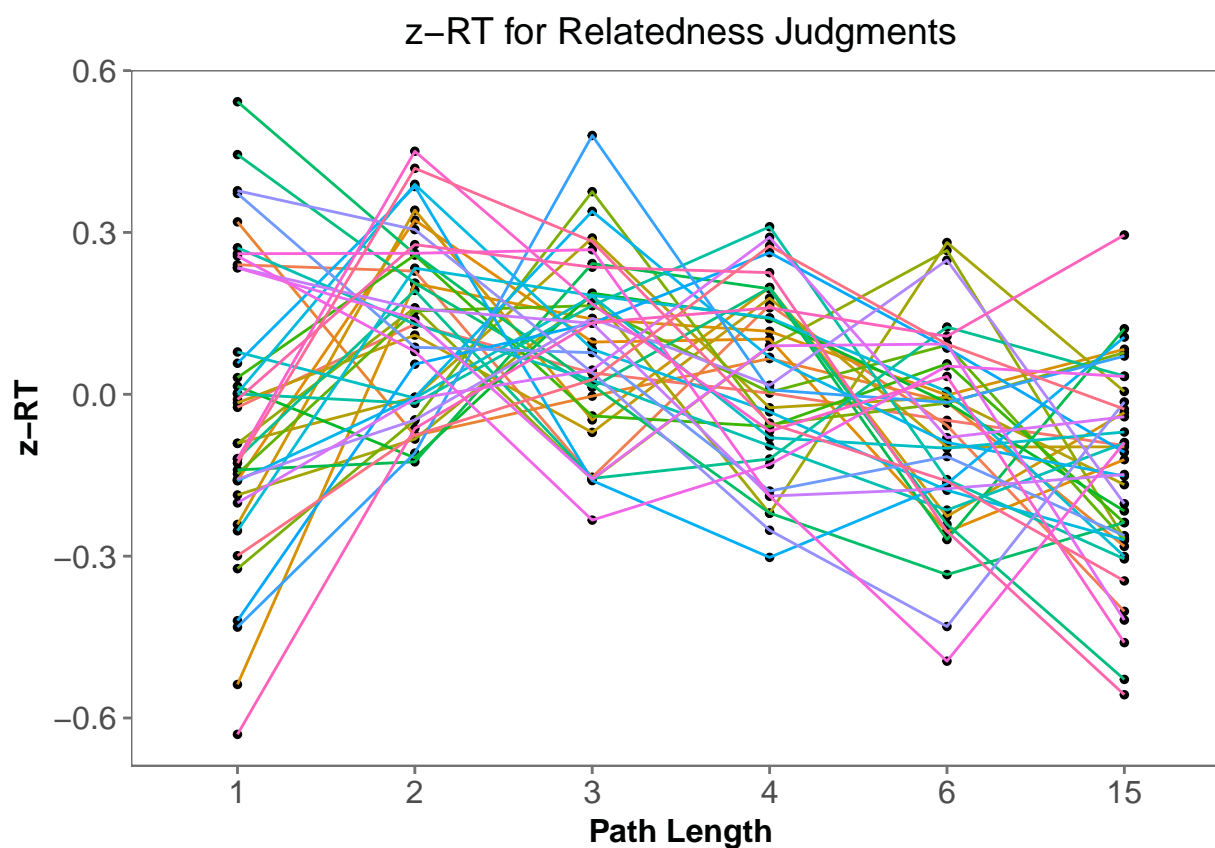


Quadratic Trend

```
z_sem_rt %>%
  ggplot(aes(x = pathlength, y = zRT_trim))+
  geom_point(color = "black", size = 1)+
  geom_smooth(method = "loess", color = "red")+
  #geom_errorbar(aes(ymin=Trials - ci, ymax=Trials + ci),
  #              width=.2, color = "gray26",
  #              position = position_dodge(0.7))+
  theme_few()+
  # scale_x_continuous(breaks = c(1,2,3,4,5,6,10,15,20))+
  xlab("Path Length") + ylab("z-RT") +
  ggtitle("z-RT for Relatedness Judgments") +
  # facet_wrap(~subject)+
  theme(axis.text = element_text(size = rel(1)),
        axis.title = element_text(face = "bold", size = rel(1)),
        legend.title = element_text(face = "bold", size = rel(1)),
        plot.title = element_text(hjust = .5),
        strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

Subject-Wise



Regressions

```
## Loading required package: Matrix

## Linear mixed model fit by REML ['lmerMod']
## Formula: rt ~ 1 + (1 | subject) + (1 | trial_index) + (1 | prime_word) +
##          (1 | target_word)
## Data: new_sem_z
##
## REML criterion at convergence: 121468.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.9975 -0.5992 -0.1583  0.4368  5.8497
##
## Random effects:
## Groups      Name                Variance Std.Dev.
## target_word (Intercept)    587.4    24.24
## prime_word  (Intercept)    914.1    30.23
## trial_index (Intercept)    124.9    11.18
## subject     (Intercept)   23886.8   154.55
## Residual                    41893.2   204.68
## Number of obs: 8973, groups:
```

```

## target_word, 1918; prime_word, 1918; trial_index, 240; subject, 40
##
## Fixed effects:
##           Estimate Std. Error t value
## (Intercept)   697.88     24.57   28.41
## [1] 0.3784984
## Linear mixed model fit by REML ['lmerMod']
## Formula: zRT_trim ~ pathlengthfac + (1 | subject)
##   Data: new_sem_z
##
## REML criterion at convergence: 25378.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.5287 -0.7314 -0.2061  0.5651  3.9283
##
## Random effects:
##   Groups   Name      Variance Std.Dev.
## subject (Intercept) 3.626e-32 1.904e-16
## Residual          9.875e-01 9.938e-01
## Number of obs: 8973, groups: subject, 40
##
## Fixed effects:
##           Estimate Std. Error t value
## (Intercept)    0.12320    0.02579   4.777
## pathlengthfac1 -0.15767    0.03645  -4.326
## pathlengthfac3 -0.02672    0.03632  -0.736
## pathlengthfac4 -0.09787    0.03643  -2.687
## pathlengthfac5 -0.18629    0.03641  -5.117
## pathlengthfac6 -0.27000    0.03636  -7.426
##
## Correlation of Fixed Effects:
##           (Intr) pthln1 pthln3 pthln4 pthln5
## pthlnghfc1 -0.708
## pthlnghfc3 -0.710  0.502
## pthlnghfc4 -0.708  0.501  0.503
## pthlnghfc5 -0.708  0.501  0.503  0.501
## pthlnghfc6 -0.709  0.502  0.504  0.502  0.502
## Linear mixed model fit by REML ['lmerMod']
## Formula: rt ~ pathlengthfac + (pathlengthfac | subject)
##   Data: sem
##
## REML criterion at convergence: 144701.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.951 -0.351 -0.115  0.184 34.778
##
## Random effects:
##   Groups   Name      Variance Std.Dev. Corr
## subject (Intercept)  66078    257.06
##           pathlengthfac2    1530    39.11    0.56

```

```
##          pathlengthfac3 12139   110.18   -0.57  0.34
##          pathlengthfac4 13916   117.96   -0.37  0.41  0.92
##          pathlengthfac5  7966    89.25   -0.56  0.36  1.00  0.92
##          pathlengthfac6  5580    74.70   -0.17  0.71  0.84  0.70  0.84
## Residual                202173   449.64
## Number of obs: 9600, groups:  subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)    732.331    42.170  17.366
## pathlengthfac2   30.535    17.058   1.790
## pathlengthfac3    5.857    23.584   0.248
## pathlengthfac4   21.454    24.507   0.875
## pathlengthfac5  -12.491    21.257  -0.588
## pathlengthfac6  -34.499    19.805  -1.742
##
## Correlation of Fixed Effects:
##              (Intr) pthln2 pthln3 pthln4 pthln5
## pthlngthfc2  0.019
## pthlngthfc3 -0.531  0.406
## pthlngthfc4 -0.394  0.416  0.736
## pthlngthfc5 -0.496  0.435  0.742  0.709
## pthlngthfc6 -0.249  0.528  0.640  0.580  0.635
```

ELP Model

```
## Adding ELP covariates

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

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

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

## Linear mixed model fit by REML ['lmerMod']
## Formula: .resid ~ pathlengthfac + (1 | subject) + (1 | trial_index)
## Data: fit_from_elp
##
## REML criterion at convergence: 121039.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.0866 -0.6125 -0.1655  0.4418  6.1215
##
## Random effects:
## Groups      Name                Variance Std.Dev.
## trial_index (Intercept) 8.186e-10 2.861e-05
## subject      (Intercept) 0.000e+00 0.000e+00
```

```

## Residual                      4.268e+04 2.066e+02
## Number of obs: 8969, groups:  trial_index, 240; subject, 40
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)    25.228      5.361    4.706
## pathlengthfac1 -29.957      7.577   -3.954
## pathlengthfac3 -11.585      7.552   -1.534
## pathlengthfac4 -20.829      7.578   -2.749
## pathlengthfac5 -37.177      7.569   -4.912
## pathlengthfac6 -51.635      7.559   -6.831
##
## Correlation of Fixed Effects:
##               (Intr) pthln1 pthln3 pthln4 pthln5
## pthlngthfc1 -0.708
## pthlngthfc3 -0.710  0.502
## pthlngthfc4 -0.707  0.501  0.502
## pthlngthfc5 -0.708  0.501  0.503  0.501
## pthlngthfc6 -0.709  0.502  0.504  0.502  0.502

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

## Linear mixed model fit by REML ['lmerMod']
## Formula: rt ~ pathlengthfac + mean_len + mean_logf + mean_ldtz + (1 |
##          subject) + (1 | trial_index)
## Data: new_sem_z
##
## REML criterion at convergence: 121309.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.0679 -0.6095 -0.1639  0.4401  6.0992
##
## Random effects:
## Groups      Name             Variance Std.Dev.
## trial_index (Intercept)    178.6    13.36
## subject      (Intercept) 23884.6   154.55
## Residual                        42987.9  207.34
## Number of obs: 8969, groups:  trial_index, 240; subject, 40
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)    690.45896    31.66869   21.803
## pathlengthfac1 -30.18356     7.62546   -3.958
## pathlengthfac3 -11.60839     7.60125   -1.527
## pathlengthfac4 -20.96165     7.62511   -2.749
## pathlengthfac5 -37.44405     7.61759   -4.915
## pathlengthfac6 -51.99523     7.61151   -6.831
## mean_len         5.74846     1.74779    3.289
## mean_logf         0.09196     1.99780    0.046
## mean_ldtz         5.32761    16.65093    0.320
##

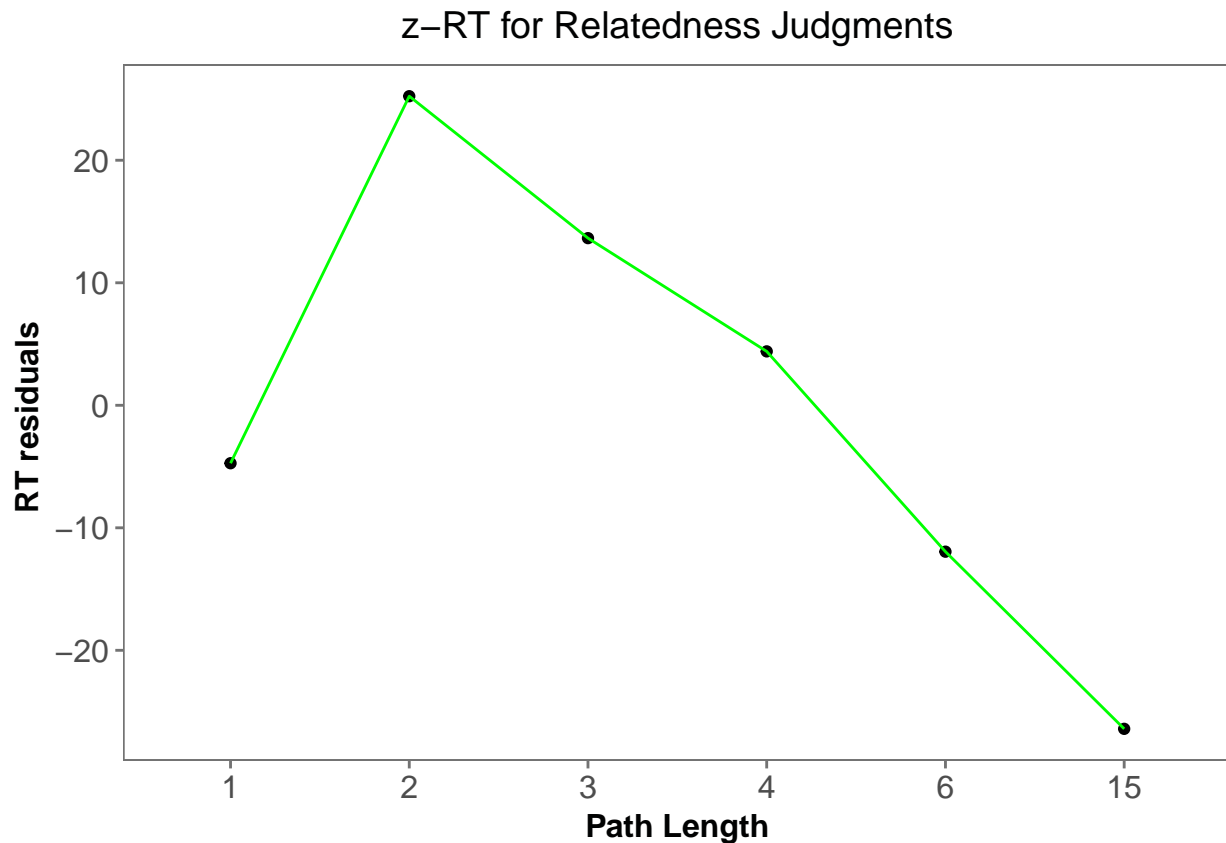
```

```
## Correlation of Fixed Effects:
##          (Intr) pthln1 pthln3 pthln4 pthln5 pthln6 men_ln mn_lgf
## pthlngthfc1 -0.133
## pthlngthfc3 -0.143  0.502
## pthlngthfc4 -0.139  0.501  0.502
## pthlngthfc5 -0.142  0.502  0.503  0.501
## pthlngthfc6 -0.152  0.502  0.503  0.502  0.503
## mean_len    -0.388 -0.003  0.003  0.011  0.019  0.037
## mean_logf    -0.391  0.039  0.034  0.030  0.033  0.036 -0.109
## mean_ldtz     0.108  0.032 -0.008  0.005  0.013  0.005 -0.464  0.527
```

Plot from ELP Model

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

fixed.frame %>%
  mutate(Pathlength = factor(pathlengthfac,
                             levels = unique(pathlengthfac),
                             labels = c("1","2", "3",
                                           "4", "6", "15"))) %>%
  ggplot(aes(x = Pathlength, y = pred, group = 1))+
  geom_point()+
  # geom_smooth(method = "loess")+
  geom_line(color = "green")+
  theme_few()+
  xlab("Path Length") + ylab("RT residuals ") +
  ggtitle("z-RT for Relatedness Judgments") +
  theme(axis.text = element_text(size = rel(1)),
        axis.title = element_text(face = "bold", size = rel(1)),
        legend.title = element_text(face = "bold", size = rel(1)),
        plot.title = element_text(hjust = .5),
        strip.text.x = element_text(face = "bold", size = rel(1.4)))
```



Concreteness Norms

```

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

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

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

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

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

```

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

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

Concreteness Model

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

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

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

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

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: .resid ~ pathlengthfac + (1 | subject) + (1 | trial_index)
## Data: fit_from_elp
##
## REML criterion at convergence: 22999.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.5403 -0.7231 -0.2069  0.5713  4.1219
##
## Random effects:
## Groups      Name                Variance Std.Dev.
## trial_index (Intercept) 9.132e-15 9.556e-08
## subject      (Intercept) 0.000e+00 0.000e+00
## Residual                    9.744e-01 9.871e-01
## Number of obs: 8170, groups: trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)    0.12752    0.02710   4.706
## pathlengthfac1 -0.16409    0.03803  -4.315
## pathlengthfac3 -0.02534    0.03813  -0.665
## pathlengthfac4 -0.10481    0.03781  -2.772
## pathlengthfac5 -0.17793    0.03835  -4.640
## pathlengthfac6 -0.28701    0.03784  -7.584
##
## Correlation of Fixed Effects:
```



```

##          (Intr) pthln1 pthln3 pthln4 pthln5
## pthlnghfc1 -0.713
## pthlnghfc3 -0.711  0.506
## pthlnghfc4 -0.717  0.511  0.509
## pthlnghfc5 -0.707  0.503  0.502  0.506
## pthlnghfc6 -0.716  0.510  0.509  0.513  0.506

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

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

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## zRT_trim ~ pathlengthfac * Type + mean_len + mean_logf + mean_ldtz +
## mean_conc + (1 | subject) + (1 | trial_index) + +(1 | target_word)
## Data: final_sem
##
## REML criterion at convergence: 23050.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6093 -0.7039 -0.1884  0.5536  4.0610
##
## Random effects:
## Groups      Name                Variance Std.Dev.
## target_word (Intercept) 0.032611 0.18059
## trial_index (Intercept) 0.007258 0.08519
## subject      (Intercept) 0.000000 0.00000
## Residual                0.939300 0.96918
## Number of obs: 8170, groups:
## target_word, 1741; trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)      0.513120   0.144977   3.539
## pathlengthfac1    -0.216878   0.053472  -4.056
## pathlengthfac3     0.158169   0.066902   2.364
## pathlengthfac4    -0.006934   0.068349  -0.101
## pathlengthfac5     0.150777   0.087102   1.731
## pathlengthfac6    -0.061106   0.094001  -0.650
## TypeUnrelated    -0.065704   0.055128  -1.192
## mean_len          0.018830   0.009514   1.979
## mean_logf         -0.018096   0.011248  -1.609
## mean_ldtz         -0.012876   0.089039  -0.145
## mean_conc         -0.083593   0.014986  -5.578
## pathlengthfac1:TypeUnrelated 0.118195   0.079853   1.480
## pathlengthfac3:TypeUnrelated -0.240884   0.082009  -2.937
## pathlengthfac4:TypeUnrelated -0.118733   0.082758  -1.435
## pathlengthfac5:TypeUnrelated -0.356285   0.098469  -3.618
## pathlengthfac6:TypeUnrelated -0.231131   0.104158  -2.219

```

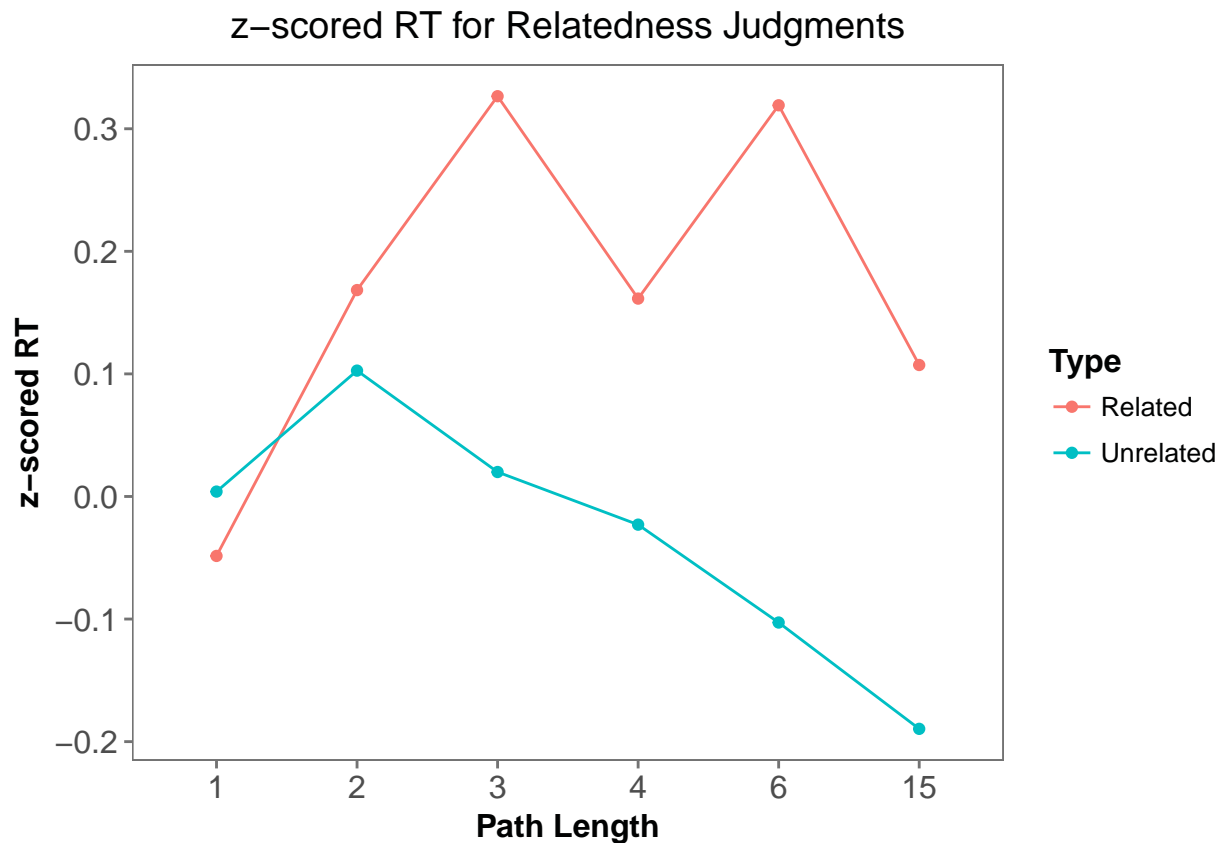
```
##
## Correlation matrix not shown by default, as p = 16 > 12.
## Use print(x, correlation=TRUE) or
##   vcov(x)       if you need it
```

Plot

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

fixed.frame <-
  data.frame(expand.grid( pathlengthfac = c("1","2", "3",
                                           "4", "6", "15"),
                          Type= c("Related", "Unrelated"),
                          mean_len = mean_length,
                          mean_logf = mean_logfreq,
                          mean_ldtz = mean_lexdec,
                          mean_conc = mean_concreteness)) %>%
  mutate(pred = predict(m1_all_elp, newdata = ., re.form = NA))

fixed.frame %>%
  mutate(Pathlength = factor(pathlengthfac,
                             levels = unique(pathlengthfac),
                             labels = c("1","2", "3",
                                         "4", "6", "15")))%>%
  ggplot(aes(x = Pathlength, y = pred, group = Type, color = Type))+
  geom_point()+
  geom_line()+
  # geom_smooth(method = "loess")+
  #geom_line(color = "green")+
  theme_few()+
  xlab("Path Length") + ylab("z-scored RT") +
  ggtitle("z-scored RT for Relatedness Judgments") +
  theme(axis.text = element_text(size = rel(1)),
        axis.title = element_text(face = "bold", size = rel(1)),
        legend.title = element_text(face = "bold", size = rel(1)),
        plot.title = element_text(hjust = .5),
        strip.text.x = element_text(face = "bold", size = rel(1.4)))
```



Quadratic Model

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

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## zRT_trim ~ pathlength + mean_len + mean_logf + mean_ldtz + mean_conc +
## (1 | subject) + (1 | trial_index) + (1 | target_word)
## Data: final_sem
##
## REML criterion at convergence: 23115
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.7061 -0.7092 -0.1986  0.5474  3.9650
##
## Random effects:
## Groups      Name                Variance Std.Dev.
## target_word (Intercept) 0.041142 0.20283
## trial_index (Intercept) 0.007316 0.08554
```

```

## subject      (Intercept) 0.000000 0.00000
## Residual              0.942842 0.97100
## Number of obs: 8170, groups:
## target_word, 1741; trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  0.426239   0.141600   3.010
## pathlength  -0.016265   0.002474  -6.575
## mean_len     0.021271   0.009662   2.202
## mean_logf    -0.016994   0.011427  -1.487
## mean_ldtz    -0.023441   0.090454  -0.259
## mean_conc    -0.090186   0.015203  -5.932
##
## Correlation of Fixed Effects:
##              (Intr) pthlng men_ln mn_lgf mn_ldt
## pathlength  -0.117
## mean_len     -0.599  0.045
## mean_logf    -0.659  0.014 -0.022
## mean_ldtz     0.038 -0.008 -0.403  0.541
## mean_conc    -0.665 -0.011  0.209  0.314  0.112

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

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## zRT_trim ~ pathlength + I((pathlength)^2) + mean_len + mean_logf +
## mean_ldtz + mean_conc + (1 | subject) + (1 | trial_index) +
## +(1 | target_word)
## Data: final_sem
##
## REML criterion at convergence: 23127
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6941 -0.7088 -0.1992  0.5477  3.9580
##
## Random effects:
## Groups      Name             Variance Std.Dev.
## target_word (Intercept) 0.041064 0.20264
## trial_index (Intercept) 0.007354 0.08575
## subject      (Intercept) 0.000000 0.00000
## Residual              0.942921 0.97104
## Number of obs: 8170, groups:
## target_word, 1741; trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)    0.3981975  0.1458140   2.731
## pathlength     -0.0066325  0.0122529  -0.541
## I((pathlength)^2) -0.0005749  0.0007162  -0.803

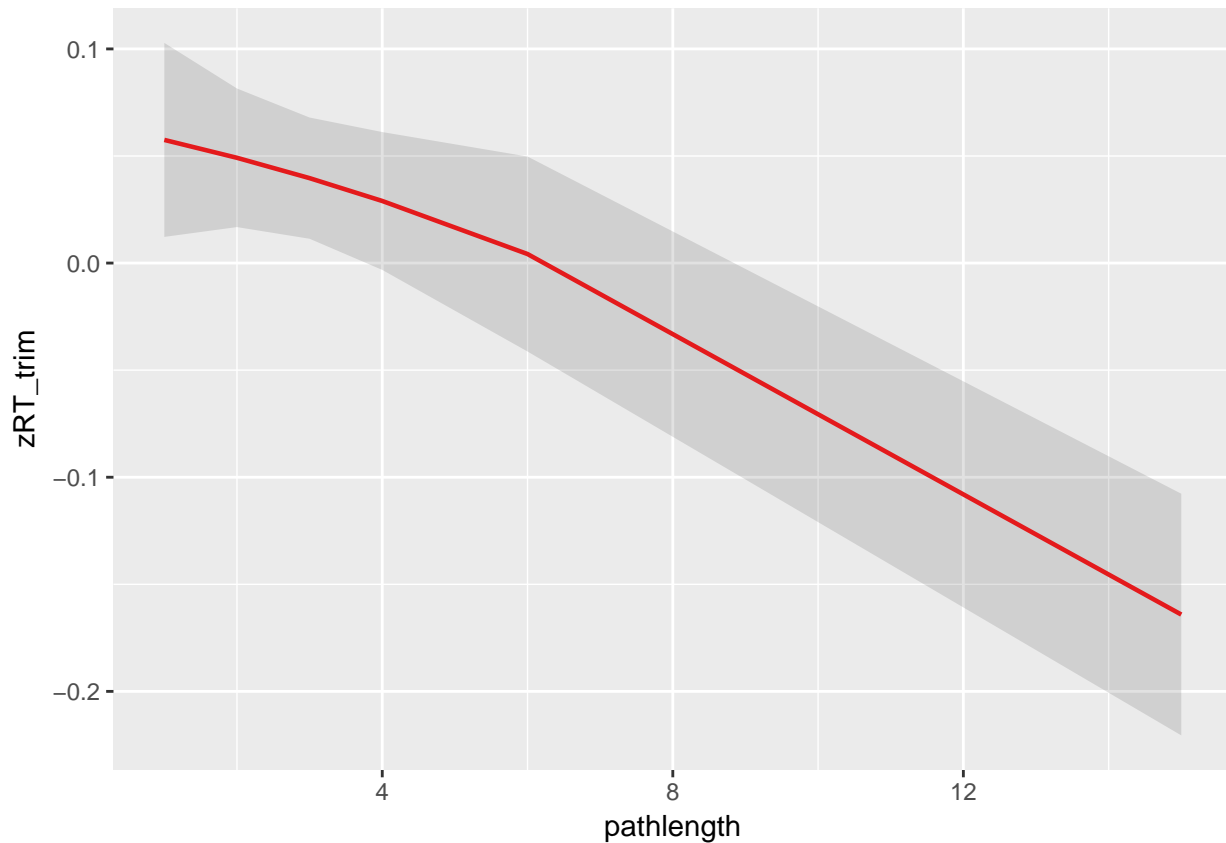
```

```

## mean_len          0.0215773  0.0096681  2.232
## mean_logf        -0.0167243  0.0114309 -1.463
## mean_ldtz        -0.0238290  0.0904468 -0.263
## mean_conc        -0.0896771  0.0152139 -5.894
##
## Correlation of Fixed Effects:
##          (Intr) pthlng I(())^2 men_ln mn_lgf mn_ldt
## pathlength -0.257
## I((pthl)^2)  0.239 -0.979
## mean_len    -0.590  0.047 -0.039
## mean_logf   -0.647  0.031 -0.029 -0.020
## mean_ldtz    0.038 -0.007  0.005 -0.403  0.540
## mean_conc   -0.655  0.038 -0.041  0.210  0.315  0.111
anova(m3_linear, m3_quad)

## refitting model(s) with ML (instead of REML)
## Data: final_sem
## Models:
## m3_linear: zRT_trim ~ pathlength + mean_len + mean_logf + mean_ldtz + mean_conc +
## m3_linear:      (1 | subject) + (1 | trial_index) + +(1 | target_word)
## m3_quad: zRT_trim ~ pathlength + I((pathlength)^2) + mean_len + mean_logf +
## m3_quad:      mean_ldtz + mean_conc + (1 | subject) + (1 | trial_index) +
## m3_quad:      +(1 | target_word)
##          Df   AIC   BIC logLik deviance  Chisq Chi Df Pr(>Chisq)
## m3_linear 10 23093 23163 -11537    23073
## m3_quad   11 23094 23172 -11536    23072 0.6467     1    0.4213
sjPlot::sjp.lm(m3_quad, type = "poly", poly.term = "pathlength")

```

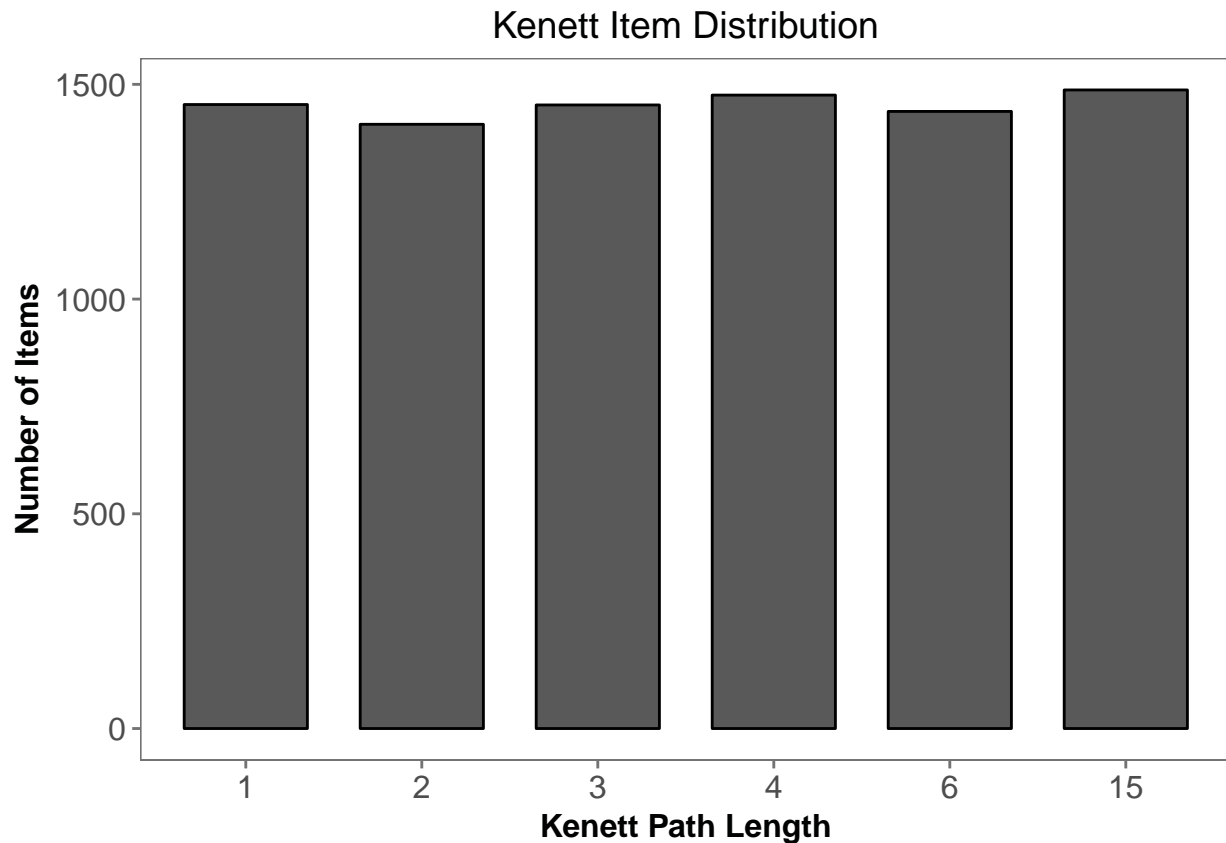


Other Networks

Kenett Path Length

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

items_kenett$pathlength = as.factor(items_kenett$pathlength)
ggplot(items_kenett, aes(x = pathlength, y = items))+
  geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
  theme_few()+
  xlab("Kenett Path Length") + ylab("Number of Items") +
  ggtitle("Kenett Item Distribution") +
  theme(axis.text = element_text(size = rel(1)),
        axis.title = element_text(face = "bold", size = rel(1)),
        legend.title = element_text(face = "bold", size = rel(1)),
        plot.title = element_text(hjust = .5),
        strip.text.x = element_text(face = "bold", size = rel(1.4)))
```



Undirected

```

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

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

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

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

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

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

```

```

## zRT_trim ~ undirectedfac * Type + mean_len + mean_logf + mean_ldtz +
##      mean_conc + (1 | subject) + (1 | trial_index) + +(1 | target_word)
##      Data: final_sem
##
## REML criterion at convergence: 23023.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.8245 -0.7128 -0.1948  0.5497  4.1351
##
## Random effects:
##      Groups      Name      Variance Std.Dev.
## target_word (Intercept) 0.031985 0.17884
## trial_index (Intercept) 0.006751 0.08217
## subject      (Intercept) 0.000000 0.00000
## Residual                0.938698 0.96886
## Number of obs: 8170, groups:
## target_word, 1741; trial_index, 240; subject, 40
##
## Fixed effects:
##
##              Estimate Std. Error t value
## (Intercept)      0.576026   0.141109   4.082
## undirectedfac1    -0.433002   0.051167  -8.462
## undirectedfac3     0.033574   0.051731   0.649
## undirectedfac4    -0.268627   0.185574  -1.448
## TypeUnrelated    -0.173571   0.034809  -4.986
## mean_len          0.019052   0.009492   2.007
## mean_logf        -0.013732   0.011234  -1.222
## mean_ldtz         0.021990   0.089317   0.246
## mean_conc        -0.096772   0.014951  -6.473
## undirectedfac1:TypeUnrelated 0.395010  0.093843   4.209
## undirectedfac3:TypeUnrelated -0.144225  0.059005  -2.444
## undirectedfac4:TypeUnrelated 0.031456  0.191528   0.164
##
## Correlation of Fixed Effects:
##      (Intr) undrc1 undrc3 undrc4 TypUnr men_ln mn_lgf mn_ldt mn_cnc
## undirctdfc1 -0.105
## undirctdfc3 -0.111  0.268
## undirctdfc4 -0.038  0.074  0.075
## TypeUnreltd -0.169  0.380  0.375  0.104
## mean_len    -0.595  0.012 -0.001 -0.002  0.059
## mean_logf   -0.647 -0.028  0.000 -0.006  0.007 -0.022
## mean_ldtz    0.043  0.016  0.006 -0.034 -0.003 -0.403  0.538
## mean_conc   -0.658  0.070  0.045  0.016  0.002  0.207  0.309  0.111
## undrctd1:TU  0.074 -0.510 -0.140 -0.039 -0.371 -0.015 -0.026 -0.016 -0.007
## undrctd3:TU  0.091 -0.228 -0.863 -0.063 -0.587 -0.009  0.001 -0.012 -0.028
## undrctd4:TU  0.030 -0.070 -0.070 -0.962 -0.181  0.005 -0.001  0.006 -0.010
##      un1:TU un3:TU
## undirctdfc1
## undirctdfc3
## undirctdfc4
## TypeUnreltd
## mean_len
## mean_logf

```



```
## mean_ldtz
## mean_conc
## undrctd1:TU
## undrctd3:TU 0.219
## undrctd4:TU 0.068 0.107
```

Plot Item Distribution

```
ggplot(undirected_rmisc, aes(x = undirected, y = items))+
  geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
  theme_few()+
  xlab("Non-directed Path Length") + ylab("Number of Items") +
  ggtitle("Non-Directed Item Distribution") +
  theme(axis.text = element_text(size = rel(1)),
        axis.title = element_text(face = "bold", size = rel(1)),
        legend.title = element_text(face = "bold", size = rel(1)),
        plot.title = element_text(hjust = .5),
        strip.text.x = element_text(face = "bold", size = rel(1.4)))
```



Plot Undirected

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

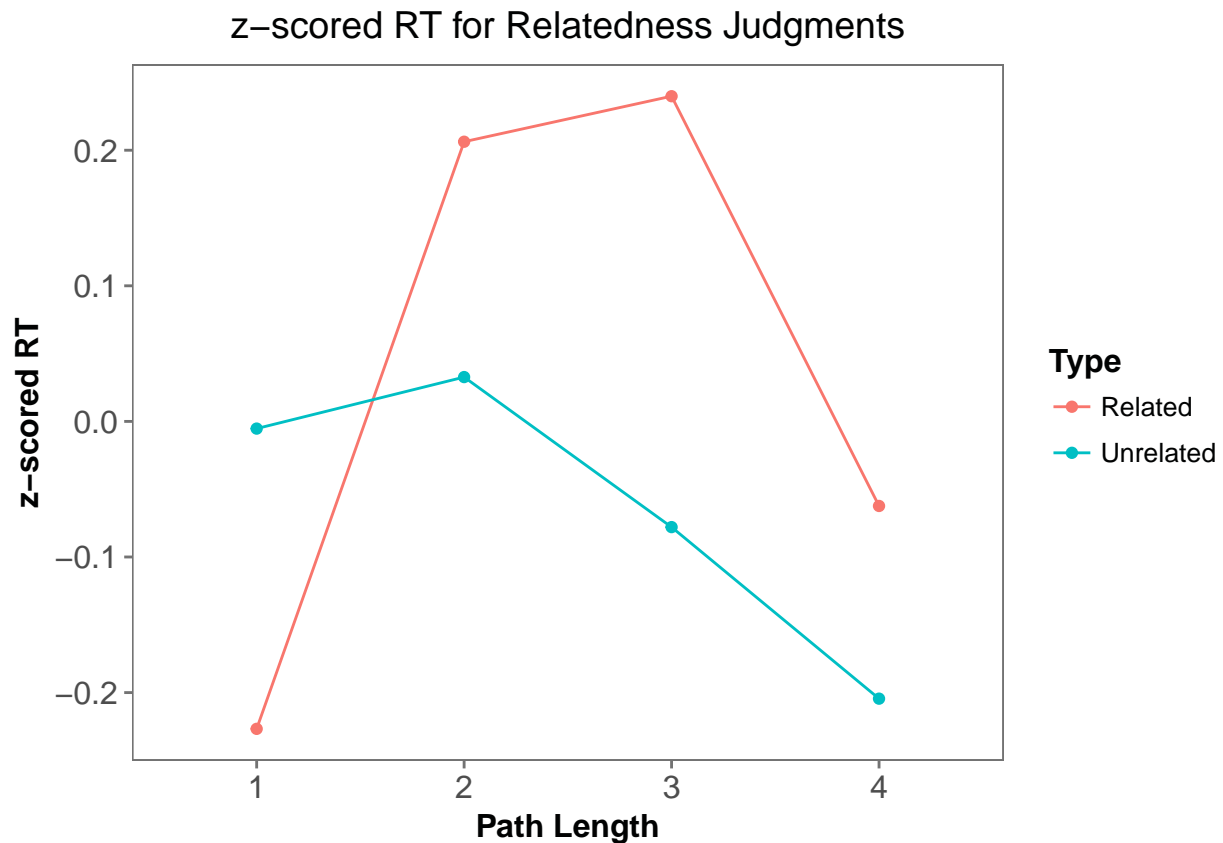
```

mean_concreteness = mean(final_sem$mean_conc, na.rm = TRUE)

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

fixed.frame %>%
  mutate(Pathlength = factor(undirectedfac,
                             levels = unique(undirectedfac),
                             labels = c("1", "2", "3", "4")))) %>%
  ggplot(aes(x = Pathlength, y = pred, group = Type, color = Type)) +
  geom_point() +
  # geom_smooth(method = "loess") +
  geom_line() +
  theme_few() +
  xlab("Path Length") + ylab("z-scored RT") +
  ggtitle("z-scored RT for Relatedness Judgments") +
  theme(axis.text = element_text(size = rel(1)),
        axis.title = element_text(face = "bold", size = rel(1)),
        legend.title = element_text(face = "bold", size = rel(1)),
        plot.title = element_text(hjust = .5),
        strip.text.x = element_text(face = "bold", size = rel(1.4)))

```



Directed

```
final_sem$newdirected = ifelse(final_sem$directed == "Inf" |
                              final_sem$directed == "NA", NA,
                              final_sem$directed)

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

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

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

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

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

```

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

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

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

```

```

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## zRT_trim ~ collapsedfac * Type + mean_len + mean_logf + mean_ldtz +
## mean_conc + (1 | subject) + (1 | trial_index) + (1 | target_word)
## Data: final_sem
##
## REML criterion at convergence: 22180
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.7216 -0.7076 -0.1930  0.5531  4.0088
##
## Random effects:
##   Groups      Name      Variance Std.Dev.
## target_word (Intercept) 0.035101 0.18735
## trial_index (Intercept) 0.007171 0.08468
## subject      (Intercept) 0.000000 0.00000
## Residual                0.936141 0.96754
## Number of obs: 7866, groups:
## target_word, 1673; trial_index, 240; subject, 40
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)      0.506507   0.150197   3.372
## collapsedfac1    -0.354059   0.065641  -5.394
## collapsedfac3      0.114629   0.058779   1.950
## collapsedfac4      0.098296   0.058606   1.677
## collapsedfac5      0.152836   0.065456   2.335
## TypeUnrelated    -0.052784   0.057540  -0.917
## mean_len          0.015727   0.009738   1.615
## mean_logf        -0.014750   0.011563  -1.276
## mean_ldtz         0.030201   0.091877   0.329
## mean_conc        -0.098539   0.015394  -6.401
## collapsedfac1:TypeUnrelated 0.363319   0.120918   3.005
## collapsedfac3:TypeUnrelated -0.111086   0.078230  -1.420
## collapsedfac4:TypeUnrelated -0.200311   0.075843  -2.641
## collapsedfac5:TypeUnrelated -0.334690   0.081757  -4.094
##

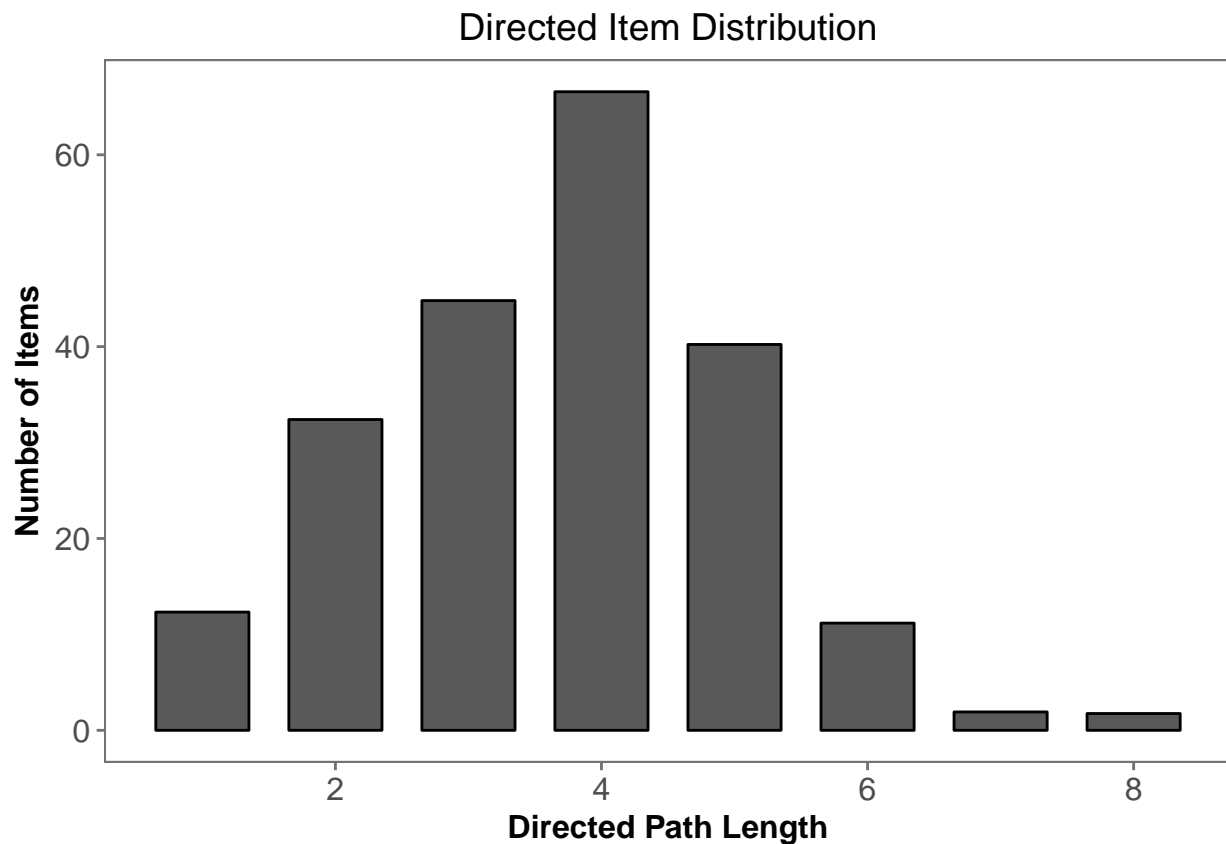
```

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

Plot Item Distribution

```
items_directed$newdirected = as.factor(items_directed$newdirected)
ggplot(directed_rmisc, aes(x = newdirected, y = items))+
  geom_bar(stat = "identity", position = "dodge", width = 0.7, color= "black")+
  theme_few()+
  xlab("Directed Path Length") + ylab("Number of Items") +
  ggtitle("Directed Item Distribution") +
  theme(axis.text = element_text(size = rel(1)),
        axis.title = element_text(face = "bold", size = rel(1)),
        legend.title = element_text(face = "bold", size = rel(1)),
        plot.title = element_text(hjust = .5),
        strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

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



```
### Plot Directed
```

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

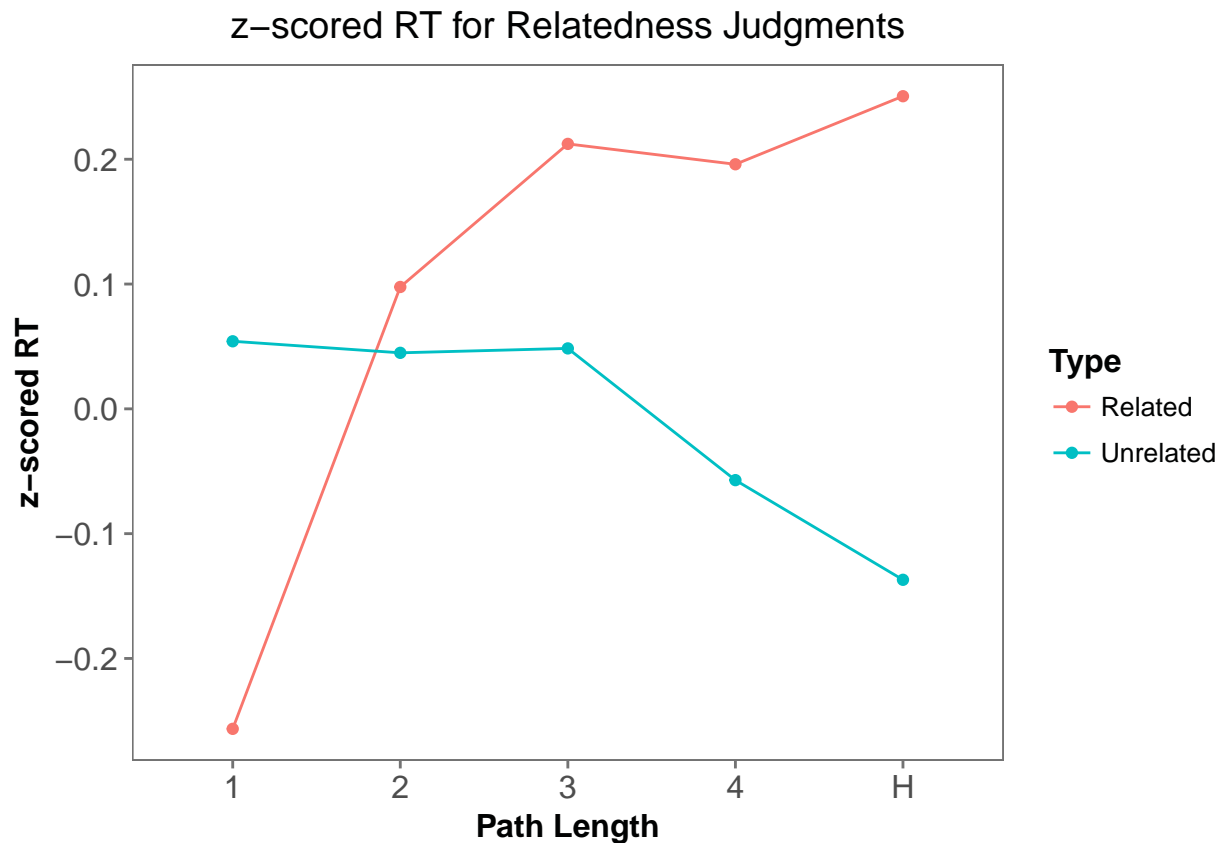
```

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

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

fixed.frame %>%
  mutate(Pathlength = factor(collapsedfac,
    levels = unique(collapsedfac),
    labels = c("1","2", "3","4", "H")))%>%
  ggplot(aes(x = collapsedfac, y = pred, group = Type, color = Type))+
    geom_point()+
    # geom_smooth(method = "loess")+
    geom_line()+
    theme_few()+
    xlab("Path Length") + ylab("z-scored RT") +
    ggtitle("z-scored RT for Relatedness Judgments") +
    theme(axis.text = element_text(size = rel(1)),
      axis.title = element_text(face = "bold", size = rel(1)),
      legend.title = element_text(face = "bold", size = rel(1)),
      plot.title = element_text(hjust = .5),
      strip.text.x = element_text(face = "bold", size = rel(1.4)))

```



```
## Log P
final_sem$logp = round(final_sem$logp, 2)

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

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

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

logp_rmisc = Rmisc::summarySE(items_logp_subject,
                              measurevar = "items",
                              groupvars = c("roundedlogp"))

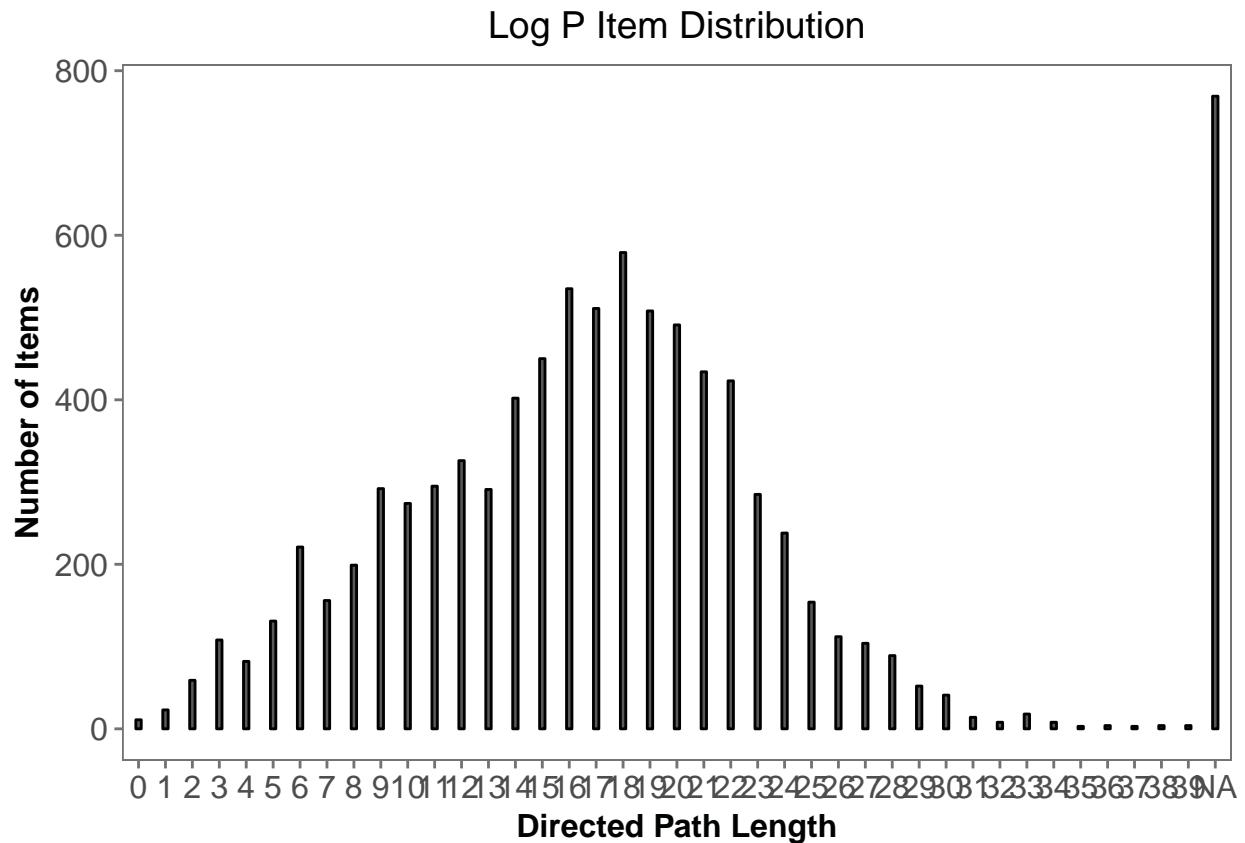
m_logp = lmer(data = final_sem, zRT_trim ~ newlogp +
              mean_len + mean_logf + mean_ldtz + mean_conc +
              (1|subject) + (1|trial_index) +
              (1|target_word))
summary(m_logp)

## Linear mixed model fit by REML ['lmerMod']
## Formula:
```

```
## zRT_trim ~ newlogp + mean_len + mean_logf + mean_ldtz + mean_conc +
## (1 | subject) + (1 | trial_index) + +(1 | target_word)
## Data: final_sem
##
## REML criterion at convergence: 21104.5
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6759 -0.7064 -0.2079  0.5514  3.9516
##
## Random effects:
##   Groups             Name             Variance Std.Dev.
## target_word (Intercept) 0.04587   0.21417
## trial_index (Intercept) 0.00858   0.09263
## subject      (Intercept) 0.00000   0.00000
## Residual                0.94299   0.97107
## Number of obs: 7444, groups:
## target_word, 1673; trial_index, 240; subject, 38
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  0.371358   0.158249   2.347
## newlogp      -0.001984   0.002003  -0.991
## mean_len      0.020905   0.010226   2.044
## mean_logf    -0.013852   0.012126  -1.142
## mean_ldtz     0.024448   0.096560   0.253
## mean_conc    -0.089674   0.016153  -5.551
##
## Correlation of Fixed Effects:
##              (Intr) newlgp men_ln mn_lgf mn_ldt
## newlogp      -0.300
## mean_len     -0.579 -0.005
## mean_logf    -0.654  0.060 -0.006
## mean_ldtz     0.068 -0.096 -0.403  0.525
## mean_conc    -0.666  0.079  0.219  0.332  0.109
```

Plot Item Distribution

```
items_logp$roundedlogp = as.factor(items_logp$roundedlogp)
ggplot(items_logp, aes(x = roundedlogp, y = items))+
  geom_bar(stat = "identity", position = "dodge", width = 0.2, color= "black")+
  theme_few()+
  xlab("Directed Path Length") + ylab("Number of Items") +
  ggtitle("Log P Item Distribution") +
  theme(axis.text = element_text(size = rel(1)),
        axis.title = element_text(face = "bold", size = rel(1)),
        legend.title = element_text(face = "bold", size = rel(1)),
        plot.title = element_text(hjust = .5),
        strip.text.x = element_text(face = "bold", size = rel(1.4)))
```

Plot LogP

```

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

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

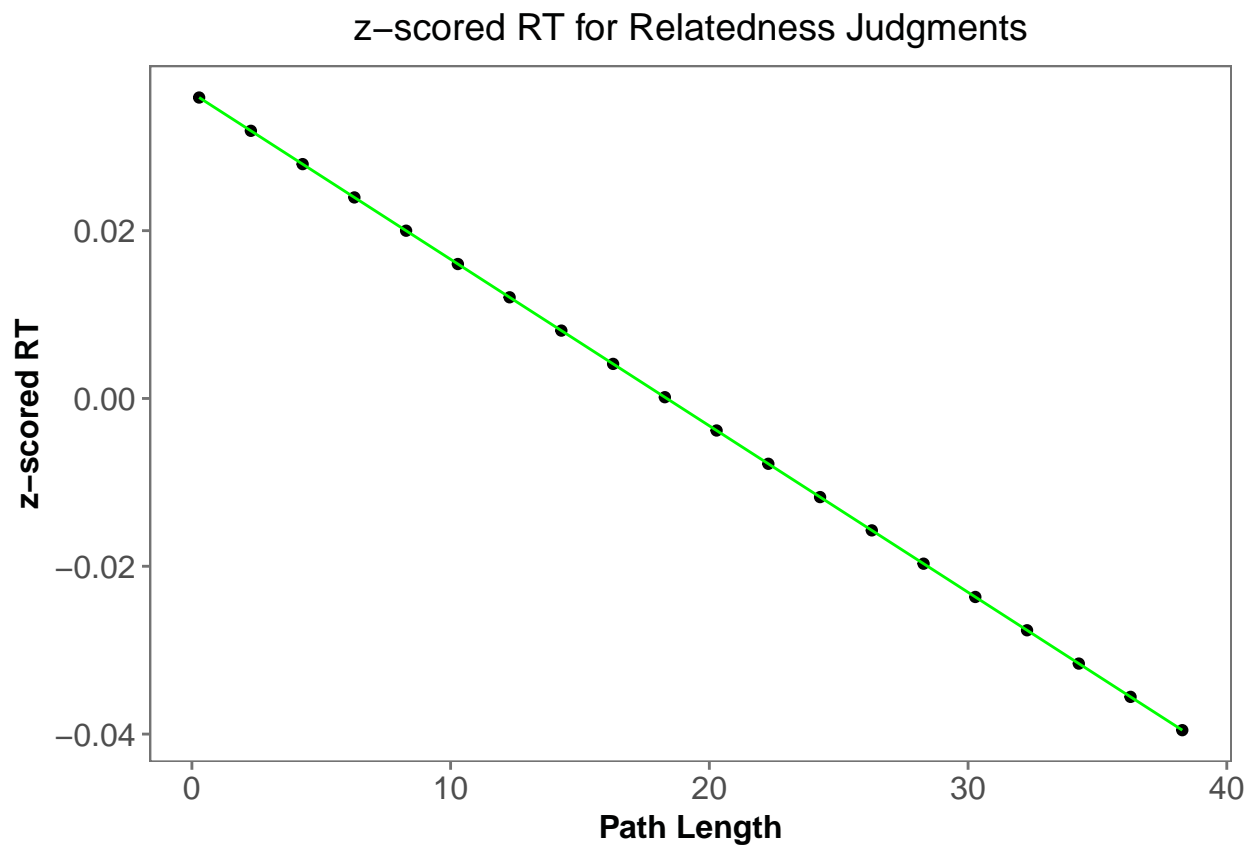
fixed.frame %>%
  # mutate(Pathlength = factor(directedfac,
  #   levels = unique(directedfac),
  #   labels = c("1", "2", "3", "4",
  #             "5", "6", "7", "8"))) %>%
  ggplot(aes(x = newlogp, y = pred, group = 1)) +

```

```

geom_point()+
# geom_smooth(method = "loess")+
geom_line(color = "green")+
theme_few()+
xlab("Path Length") + ylab("z-scored RT") +
ggtitle("z-scored RT for Relatedness Judgments") +
theme(axis.text = element_text(size = rel(1)),
      axis.title = element_text(face = "bold", size = rel(1)),
      legend.title = element_text(face = "bold", size = rel(1)),
      plot.title = element_text(hjust = .5),
      strip.text.x = element_text(face = "bold", size = rel(1.4)))

```



z-scores for Path Lengths

```

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

rawRT_p1 = sem %>% filter(pathlength == "1")

```

Creating separate z-scores

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

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

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

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

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

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

## checking: subject level means should be zero

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

““