```
# .define.fonts <- function () {
#     quartzFonts(avenir =   c('Avenir Book', 'Avenir Black', 'Avenir Book Oblique', 'Avenir Bla
#                 helvetica = c('Helvetica New Light', 'Helvetica New Bold', 'Helvetica New Light
#                 cmodern = c('CMU Serif', 'CMU Serif Extra','CMU Classical Serif','CMU Serif Ext
# }
# .define.fonts()
```

```
library(extrafont)
```

Registering fonts with R

```
loadfonts()
```

```
## .Keyboard already registered with pdfFonts().
## Andale Mono already registered with pdfFonts().
## More than one version of regular/bold/italic found for Apple Braille.  Skipping
setup for this font.
## AppleMyungjo already registered with pdfFonts().
## Arial Black already registered with pdfFonts().
## Arial already registered with pdfFonts().
## Arial Narrow already registered with pdfFonts().
## Arial Rounded MT Bold already registered with pdfFonts().
## Arial Unicode MS already registered with pdfFonts().
## Bodoni Ornaments already registered with pdfFonts().
## Bodoni 72 Smallcaps already registered with pdfFonts().
## No regular (non-bold, non-italic) version of .  Skipping setup for this font.
## No regular (non-bold, non-italic) version of Brush Script MT. Skipping setup for
this font.
## Comic Sans MS already registered with pdfFonts().
## Courier New already registered with pdfFonts().
## No regular (non-bold, non-italic) version of DIN Alternate.  Skipping setup for
this font.
## No regular (non-bold, non-italic) version of DIN Condensed.  Skipping setup for
this font.
## Georgia already registered with pdfFonts().
## Impact already registered with pdfFonts().
## Khmer Sangam MN already registered with pdfFonts().
## Lao Sangam MN already registered with pdfFonts().
## Luminari already registered with pdfFonts().
## Microsoft Sans Serif already registered with pdfFonts().
## Tahoma already registered with pdfFonts().
## Times New Roman already registered with pdfFonts().
## Trattatello already registered with pdfFonts().
## Trebuchet MS already registered with pdfFonts().
## Verdana already registered with pdfFonts().
## Webdings already registered with pdfFonts().
## Wingdings already registered with pdfFonts().
## Wingdings 2 already registered with pdfFonts().
```

```
## Wingdings 3 already registered with pdfFonts().
## More than one version of regular/bold/italic found for CMU Bright.  Skipping setup
for this font.
## No regular (non-bold, non-italic) version of CMU Classical Serif.  Skipping setup
for this font.
## CMU Concrete already registered with pdfFonts().
## CMU Sans Serif already registered with pdfFonts().
## CMU Sans Serif Demi Condensed already registered with pdfFonts().
## CMU Serif already registered with pdfFonts().
## No regular (non-bold, non-italic) version of CMU Serif Extra.  Skipping setup
for this font.
## No regular (non-bold, non-italic) version of CMU Serif Upright Italic.  Skipping
setup for this font.
## More than one version of regular/bold/italic found for CMU Typewriter Text.  Skipping
setup for this font.
## CMU Typewriter Text Variable Width already registered with pdfFonts().
```
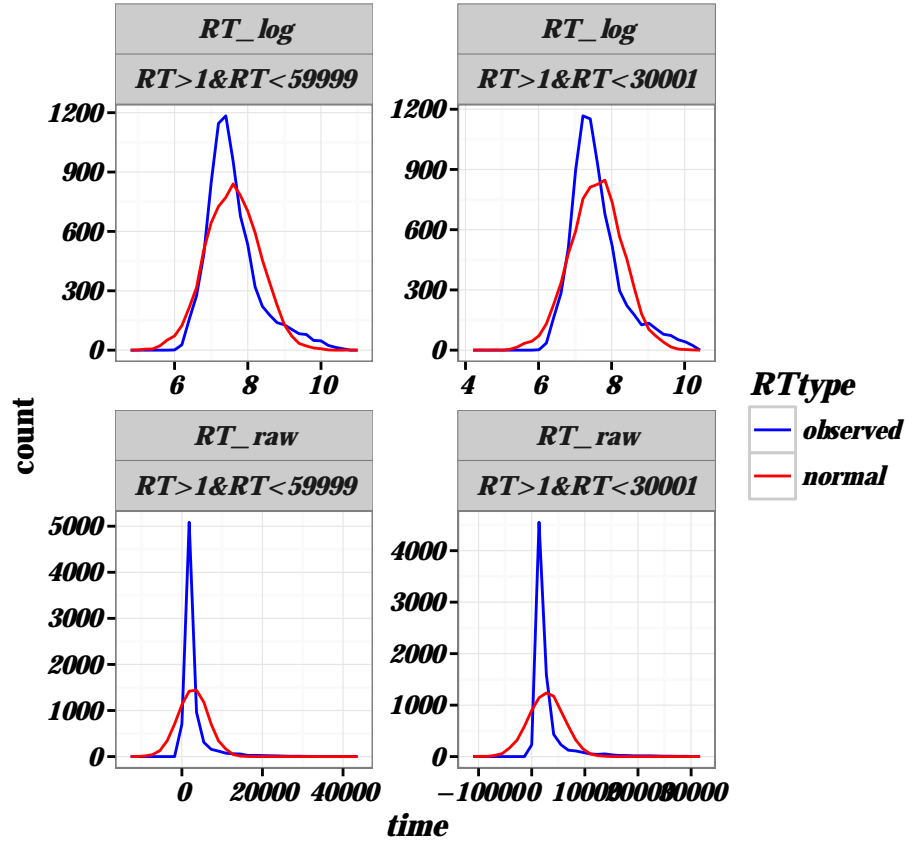
1 Get the data

2 Plots

Figure 1: Compare distributions of the various transformations of RT against random samples from normal distributions with the same mean and sd to see which transformations best approximate normal distributions
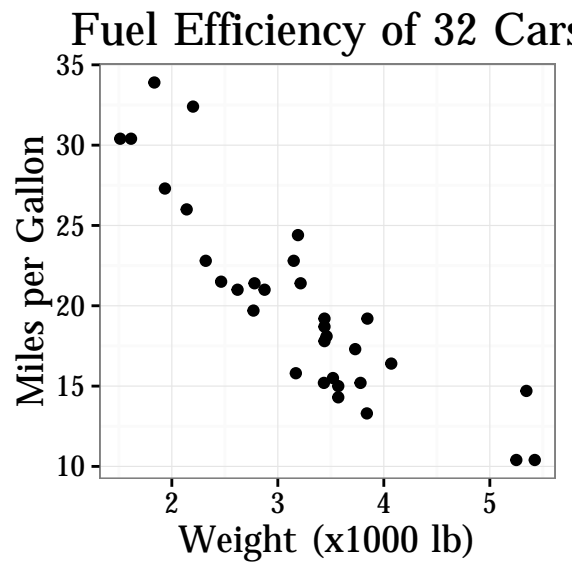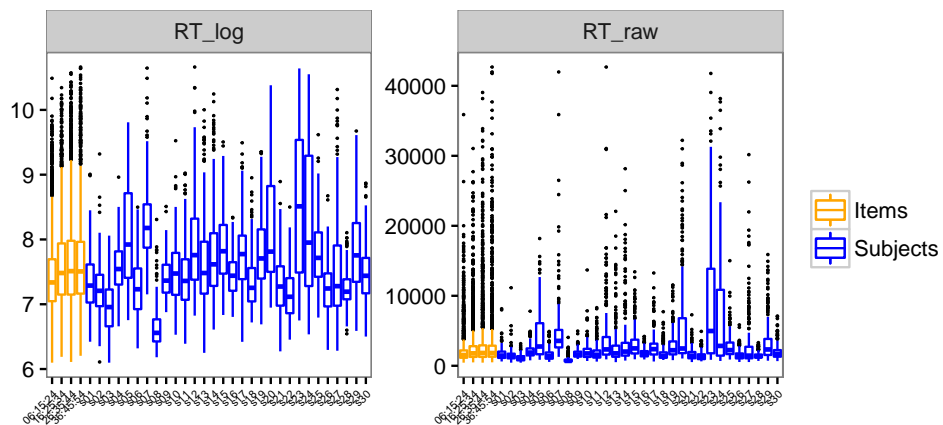
Figure 2: font example

Figure 3: Show how transformations of RT affect distribution of times, and how they affect which times are outliers.

How mean times for individual subjects and items vary with respect to the grand mean Log RT
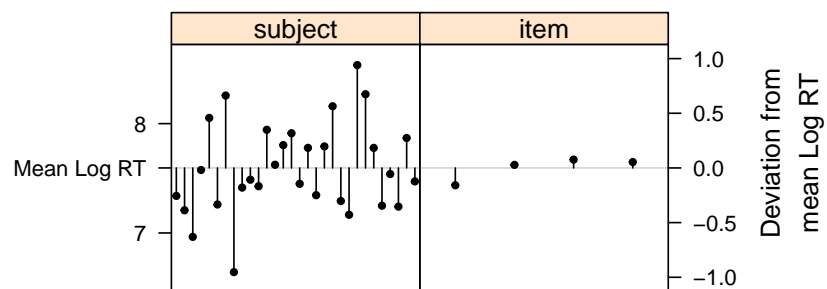

Figure 4: Show how mean times for individual subjects and items vary with respect to the grand mean Log RT.
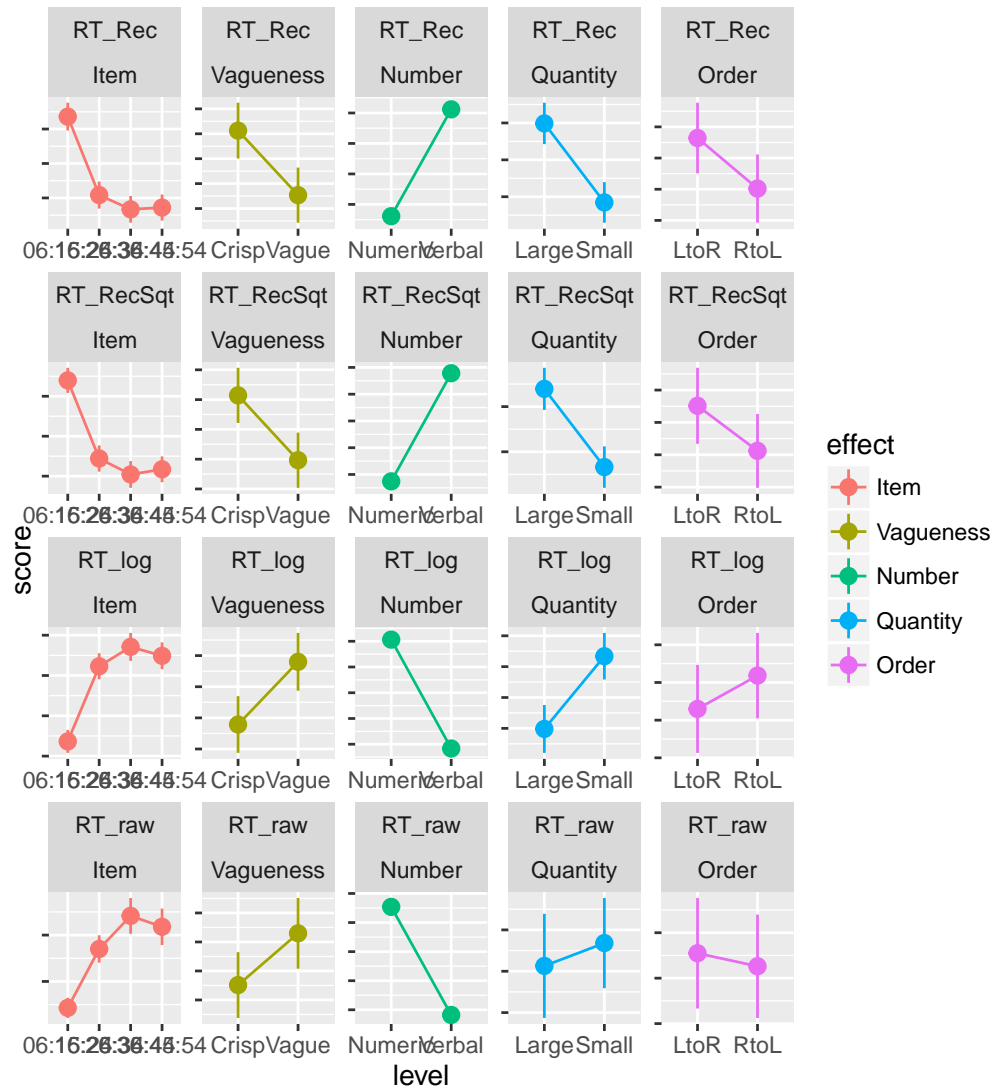
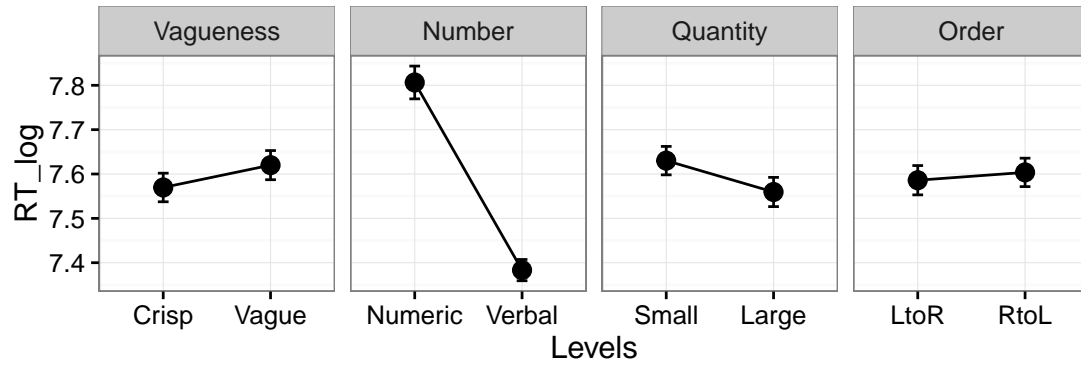Figure 5: Plot main effects on several transformations

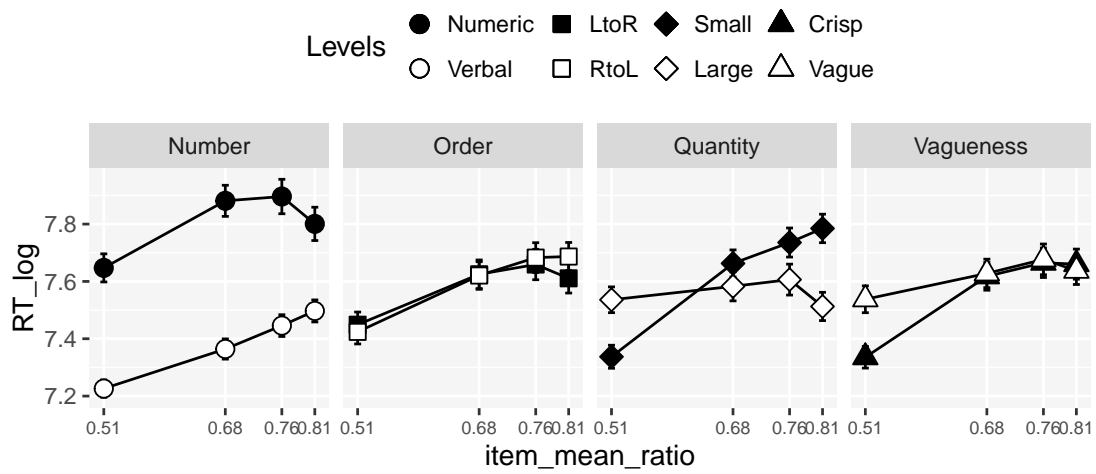Figure 6: Just the main effects

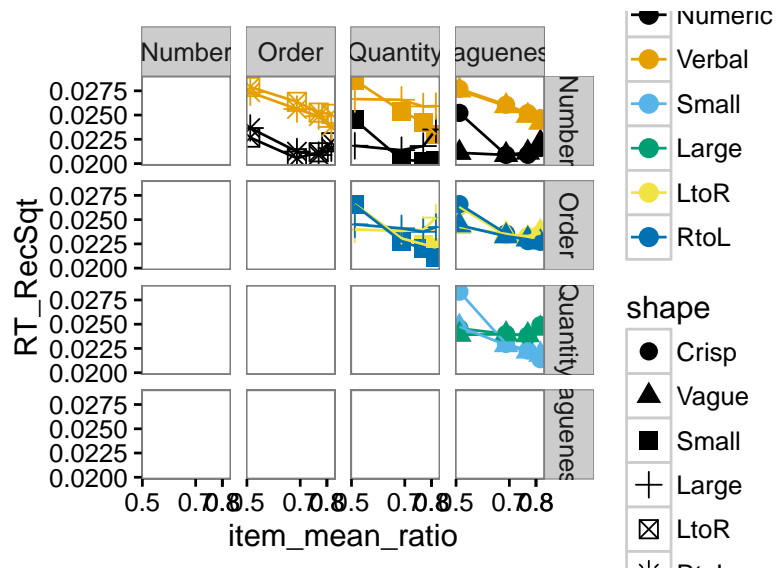Figure 7: Main effects over item ratios

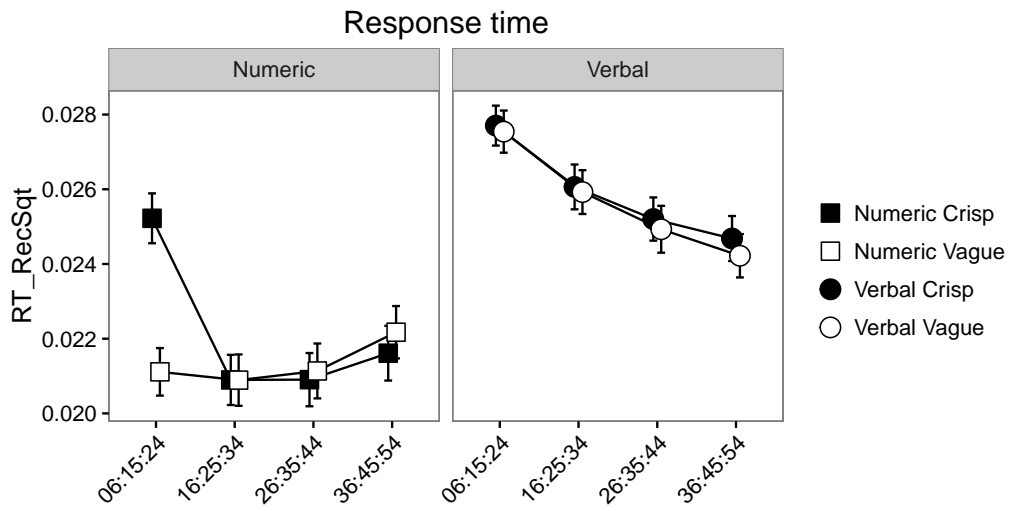Figure 8: 2-way interactions over items

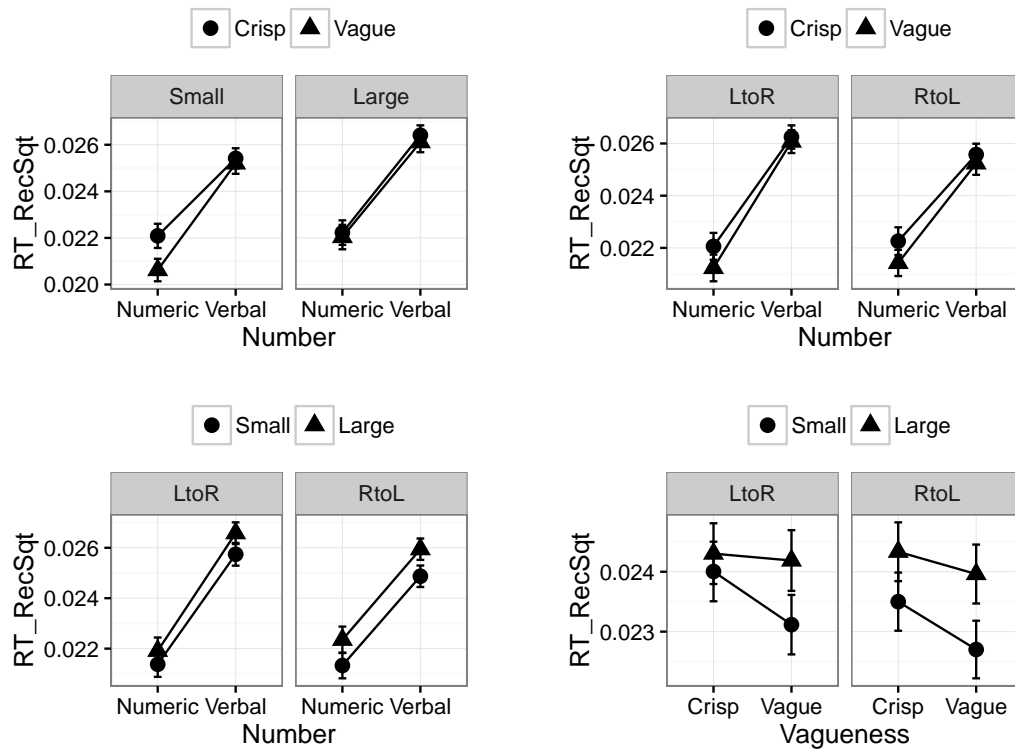Figure 9: Plot vagueness by number interaction over items

Figure 10: 3 way interactions
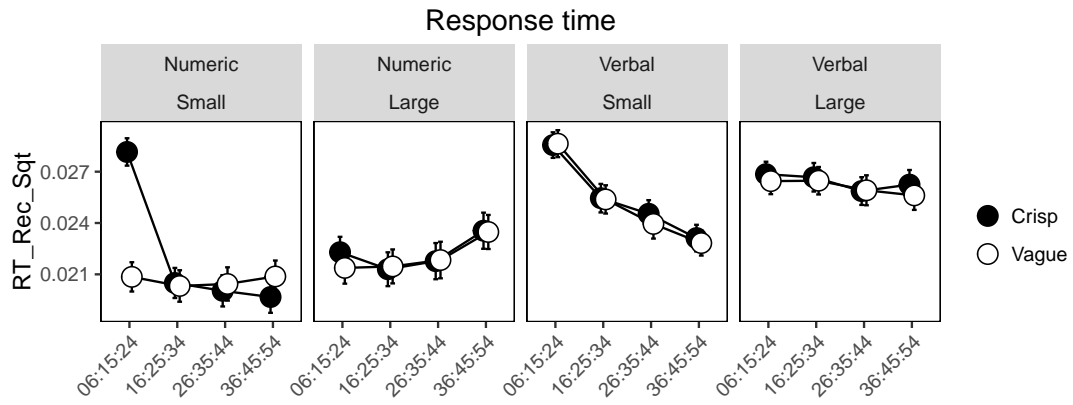
Figure 11: Plot vagueness by number by quantity over items.
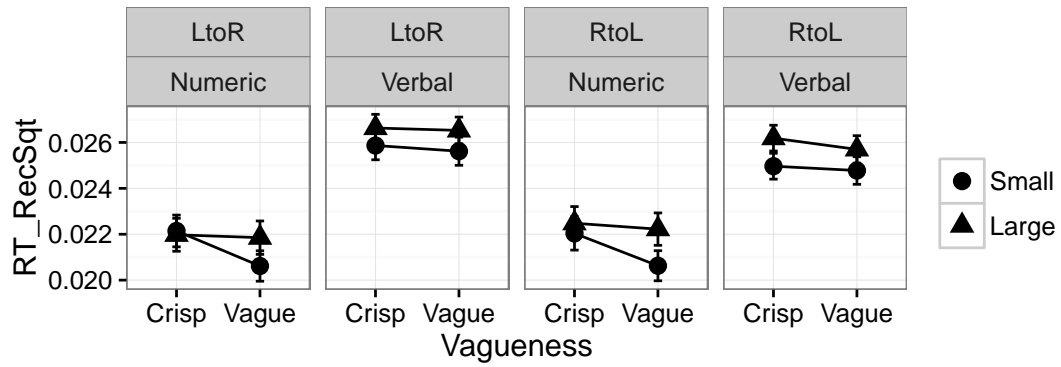
Figure 12: 4 way interaction
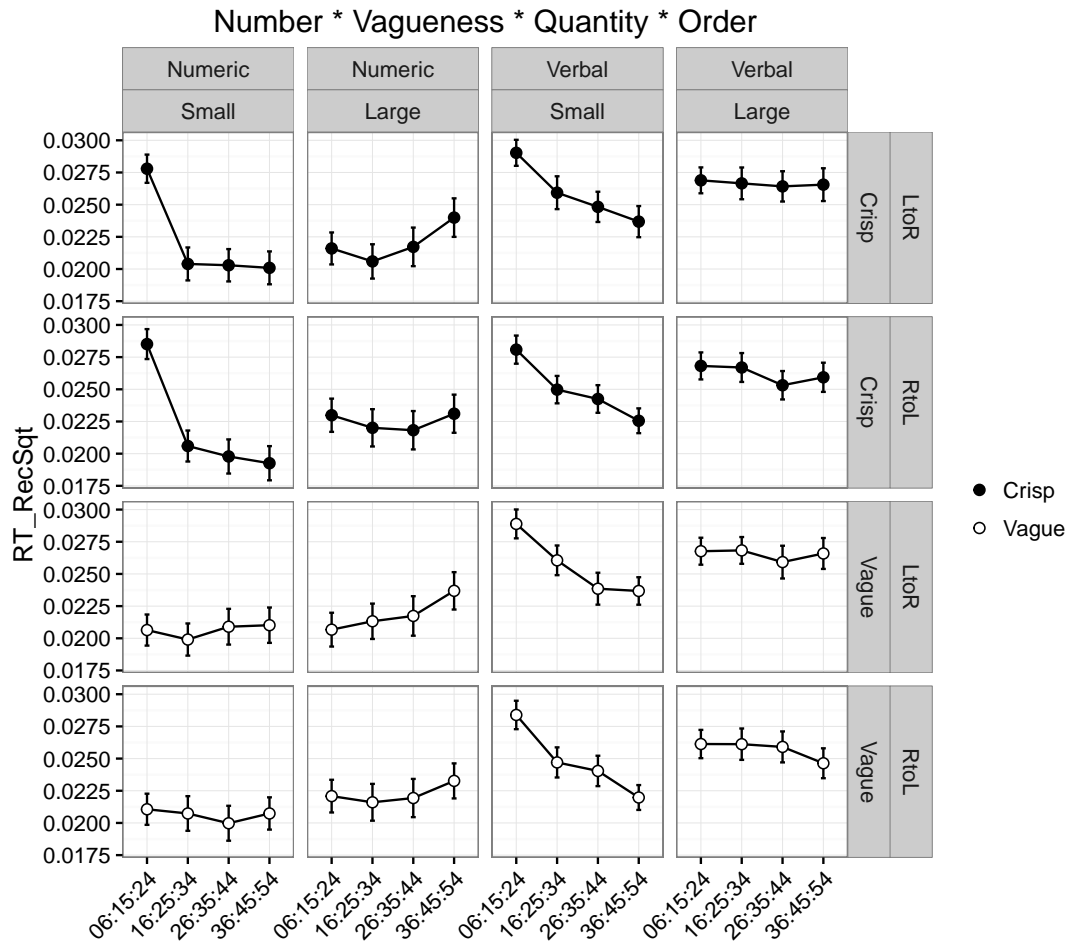
Figure 13: 4 way interaction split over items
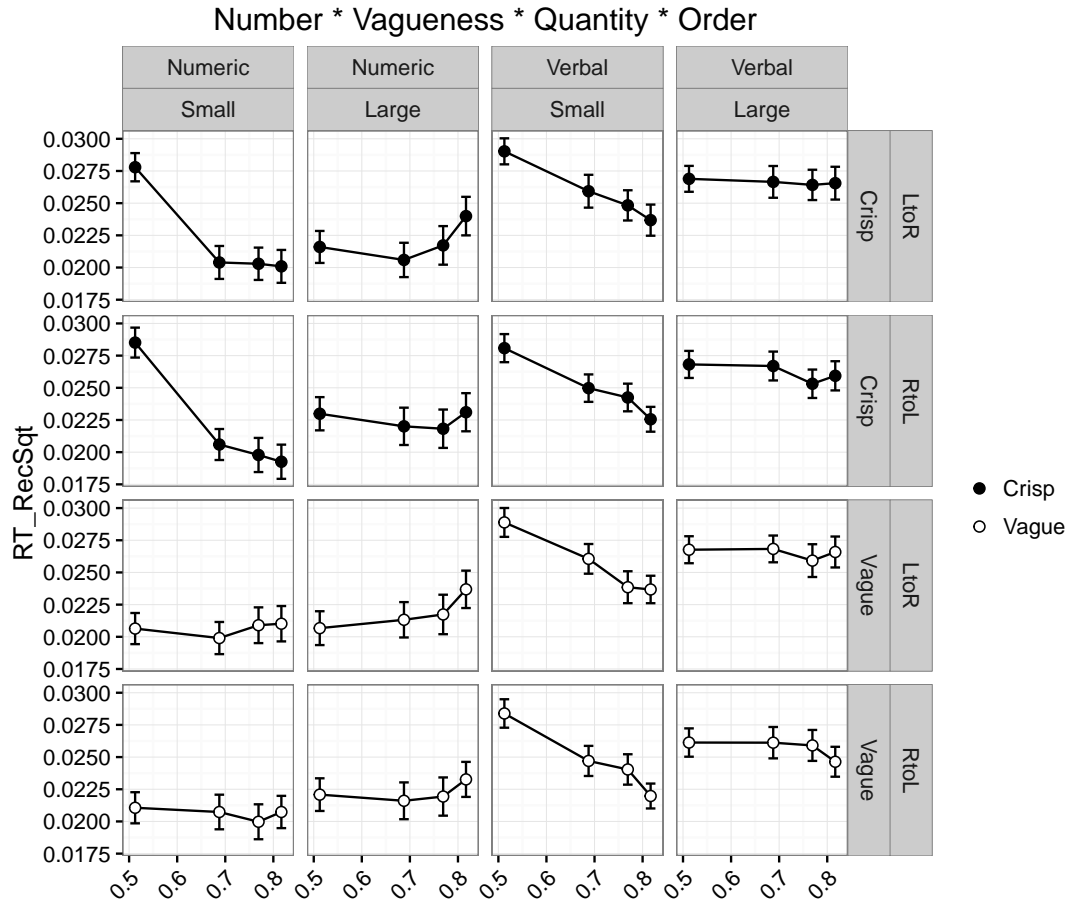
Figure 14: 4 way interaction split over items showing ratio of item dots.

**Discriminability of each item,
expressed as ratios between array magnitudes**

ratio min/max   0.25   0.47   0.59 0.67

mean (ratio min/med,
      ratio med/max)   0.51   0.69 0.70.82

ratio: small is easily discriminable

Figure 15: Show compression of item ratios

3 Lmer model

```
load("data_processed.Rda")
```

```
v5 <- lmerTest::lmer(data=dd,
RT_log ~
c_Vag + c_Num + c_Qty + c_Ord +
c_Num:c_Vag:c_Qty +
item_mean_ratio +
s_Trl +
RTprev_log +
nchar_instr +
(1+c_Vag + c_Num + c_Qty + c_Ord|Subject))
v5b <- lme4::lmer(data=dd,
RT_log ~
c_Vag + c_Num + c_Qty + c_Ord +
c_Num:c_Vag:c_Qty +
item_mean_ratio +
s_Trl +
RTprev_log +
nchar_instr +
(1+c_Vag + c_Num + c_Qty + c_Ord|Subject))
```
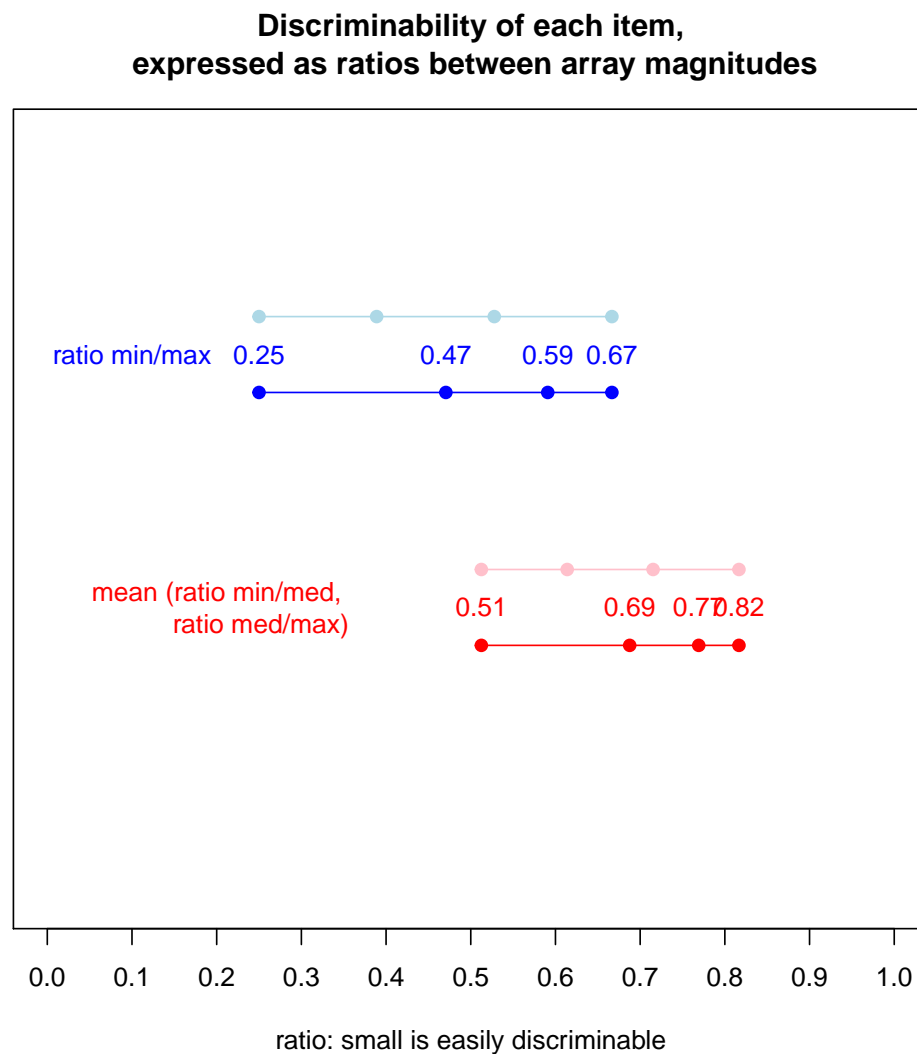
```
summary(v5)

## Linear mixed model fit by REML t-tests use Satterthwaite approximations
##   to degrees of freedom [lmerMod]
## Formula: RT_log ~ c_Vag + c_Num + c_Qty + c_Ord + c_Num:c_Vag:c_Qty +
##     item_mean_ratio + s_Trl + RTprev_log + nchar_instr + (1 +
##     c_Vag + c_Num + c_Qty + c_Ord | Subject)
##    Data: dd
##
## REML criterion at convergence: 11474.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.5470 -0.6351 -0.0955  0.5372  5.0914
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  Subject  (Intercept) 0.153949 0.39236
##           c_Vag       0.001546 0.03932   0.69
##           c_Num       0.165314 0.40659  -0.67 -0.64
##           c_Qty       0.008148 0.09027   0.16  0.26 -0.34
##           c_Ord       0.001559 0.03949  -0.13  0.02 -0.39 -0.52
##  Residual             0.249734 0.49973
```

14

```
## Number of obs: 7677, groups:  Subject, 30
##
## Fixed effects:
##                     Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)        6.400e+00  1.252e-01  2.600e+02  51.112  < 2e-16 ***
## c_Vag              6.094e-02  1.387e-02  3.300e+01   4.393 0.000112 ***
## c_Num             -4.336e-01  7.515e-02  2.900e+01  -5.770 2.97e-06 ***
## c_Qty             -6.907e-02  2.005e-02  2.900e+01  -3.445 0.001743 **
## c_Ord              1.796e-02  1.350e-02  5.100e+01   1.331 0.189164
## item_mean_ratio    7.714e-01  4.927e-02  7.551e+03  15.658  < 2e-16 ***
## s_Trl             -1.070e-01  5.807e-03  7.558e+03 -18.421  < 2e-16 ***
## RTprev_log         6.047e-02  9.692e-03  7.594e+03   6.239 4.63e-10 ***
## nchar_instr        6.086e-03  1.944e-03  7.551e+03   3.131 0.001749 **
## c_Vag:c_Num:c_Qty  1.049e-01  4.607e-02  7.551e+03   2.278 0.022742 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##            (Intr) c_Vag  c_Num  c_Qty  c_Ord  itm_m_ s_Trl  RTprv_ nchr_n
## c_Vag       0.083
## c_Num      -0.360 -0.337
## c_Qty       0.067  0.117 -0.278
## c_Ord      -0.051  0.007 -0.206 -0.228
## item_men_rt -0.257 -0.004  0.001  0.000 -0.001
## s_Trl      -0.104  0.003 -0.001 -0.002  0.006 -0.017
## RTprev_log -0.587  0.006 -0.004 -0.003  0.019 -0.015  0.185
## nchar_instr -0.505  0.237 -0.034  0.018  0.000 -0.017  0.000  0.006
## c_Vg:c_N:_Q  0.070 -0.033  0.005 -0.003  0.000  0.002 -0.001 -0.002 -0.138
```

```
print(xtable(coef(summary(v5b))))
```

	Estimate	Std. Error	t value
(Intercept)	6.40	0.13	51.11
c_Vag	0.06	0.01	4.39
c_Num	-0.43	0.08	-5.77
c_Qty	-0.07	0.02	-3.45
c_Ord	0.02	0.01	1.33
item_mean_ratio	0.77	0.05	15.66
s_Trl	-0.11	0.01	-18.42
RTprev_log	0.06	0.01	6.24
nchar_instr	0.01	0.00	3.13
c_Vag:c_Num:c_Qty	0.10	0.05	2.28

```
cat("R^2")
```

```
## R^2
```

15

```
cor(fitted(v5), dd$RT_RecSqt)^2

## [1] 0.5095834
```

```
par(mfrow=c(3,4), family='cmodern')
plotLMER.fnc(v5)

## effect size (range) for  c_Vag is  0.03470056
## effect size (range) for  c_Num is  0.4073765
## effect size (range) for  c_Qty is  0.09530422
## effect size (range) for  c_Ord is  0.0179595
## effect size (range) for  item_mean_ratio is  0.2346348
## effect size (range) for  s_Trl is  0.369093
## effect size (range) for  RTprev_log is  0.2759498
## effect size (range) for  nchar_instr is  0.05477539

## Error in plot.new():  figure margins too large
```

Plot model coefficients and ci's

```
e=data.frame(coef=summary(v5)$coef[-1,1]) # estimates, without intercept
q=as.data.frame(confint(v5, method='Wald')[18:26,1:2])
eq=cbind(e,q)
y=eq[rev(rownames(eq)),]
par(mar=c(4,10,1,1))
plot(y=1:nrow(y), x=y$coef, xlim=range(y$"2.5 %", y$"97.5 %"), type='n', axes=F, xlab="", ylab=""
segments(y0=1:nrow(y), y1=1:nrow(y), x0=y$"2.5 %", x1=y$"97.5 %", lwd=.55)
points(y=1:nrow(y), x=y$coef, pch=20, cex=.75)
abline(v=0, lty=3, col='grey')
axis(2, labels=row.names(y), at=1:nrow(y), las=1)
axis(1, las=1)
mtext("estimate", side=1, line=2.5, cex.lab=1, las=1)
box(col='grey')
```

Model criticism baayen plots

```
# Baayen 4-plot model criticism
par(mfrow=c(1,4), pty='s')
# create scaled residuals
dd$rstand = as.vector(scale(resid(v5)))
# plot scaled residuals density
plot(density(dd$rstand))
# plot sample quantiles versus theoretical quantiles
qqnorm(dd$rstand, cex=.5)
qqline(dd$rstand)
# plot standardised residuals versus fitted values
plot(dd$rstand ~ fitted(v5), pch='.')
# absolute standardised residuals greater than 2.5 are candidates for being outliers, the abline
abline(h=c(-2.5,2.5))
```
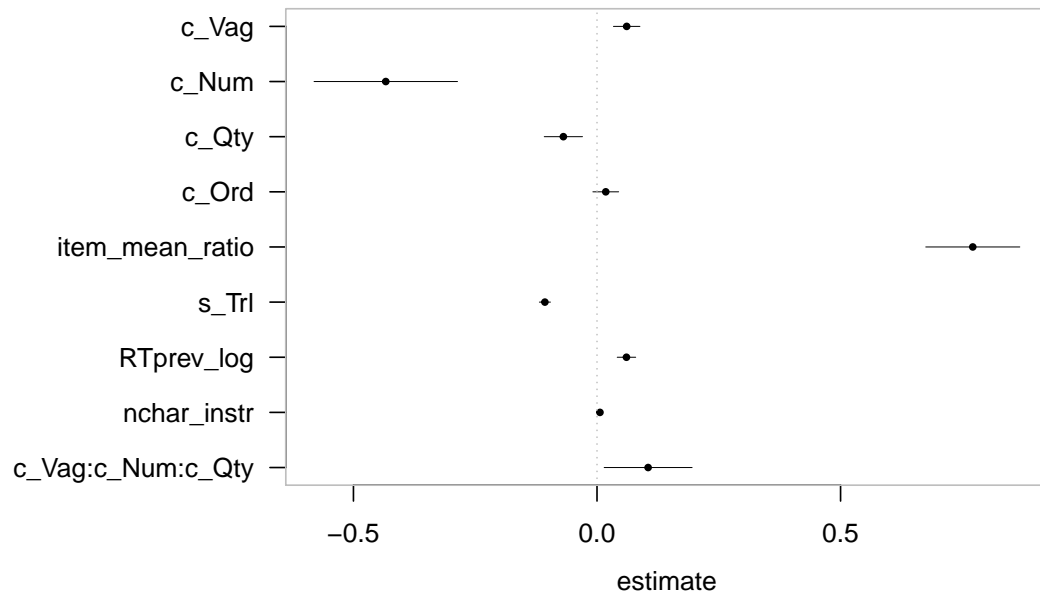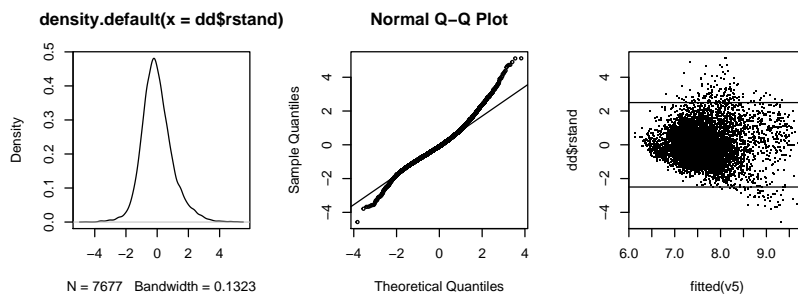
16

Figure 16: Estimates plot

Figure 17: caption