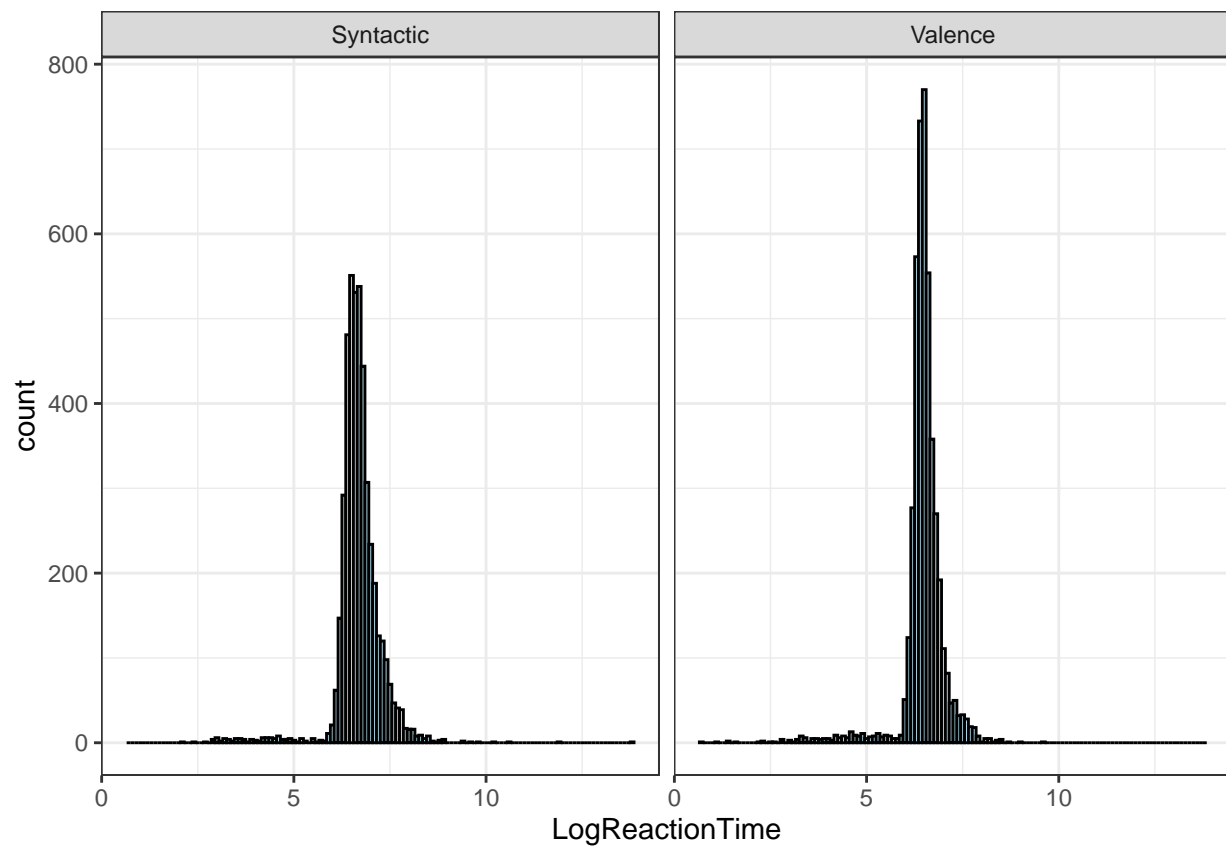


Syntactic: Analysis

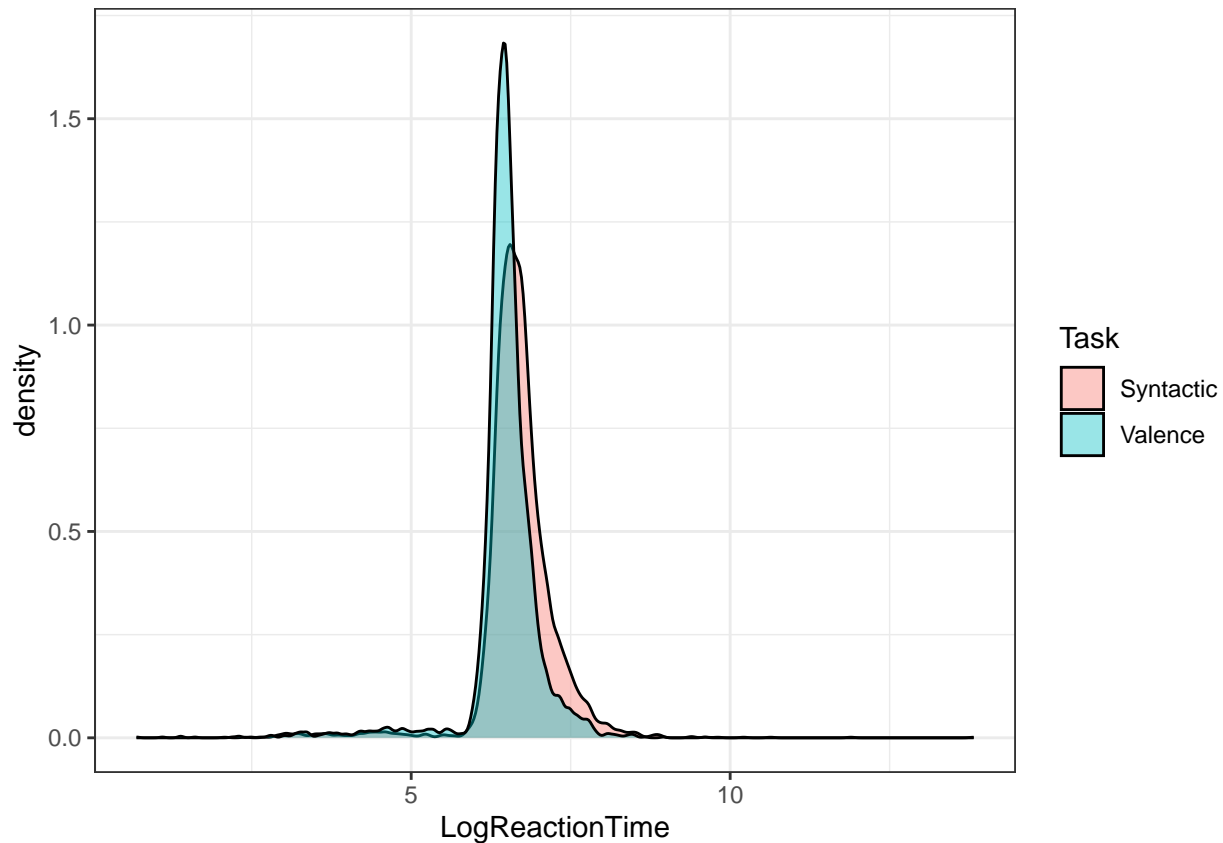
morgan moyer

2025-03-28

```
ggplot(d, aes(x=LogReactionTime)) +  
  geom_histogram(binwidth = .1, fill = "lightblue", color = "black") +  
  facet_wrap(~Task)
```



```
ggplot(d, aes(x=LogReactionTime, fill=Task)) +  
  geom_density(alpha = .4)
```



```
names(d)
```

```
## [1] "X" "ID.true" "Word" "Label"
## [5] "ConcValCombo" "Category" "Task" "BlockOrder"
## [9] "Group" "Response" "Accuracy" "EventTime"
## [13] "Value" "ReactionTime" "Key_value_F" "Key_value_J"
## [17] "Comments" "LogReactionTime" "TrialNumber"
```

```
dcen <- d %>%
```

```
  mutate(Word = as.factor(Word),
         ID.true = as.factor(ID.true),
         Task = as.factor(Task),
         cAccuracy = as.numeric(Accuracy) - mean(as.numeric(Accuracy)),
         cTask = as.numeric(Task) - mean(as.numeric(Task)))
```

```
m <- lmer(LogReactionTime ~ cAccuracy*cTask + (1+cTask|Word) + (1+cTask|ID.true), data = dcen)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00278333 (tol = 0.002, component 1)
```

```
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LogReactionTime ~ cAccuracy * cTask + (1 + cTask | Word) + (1 +
##       cTask | ID.true)
## Data: dcen
##
## REML criterion at convergence: 10278
```

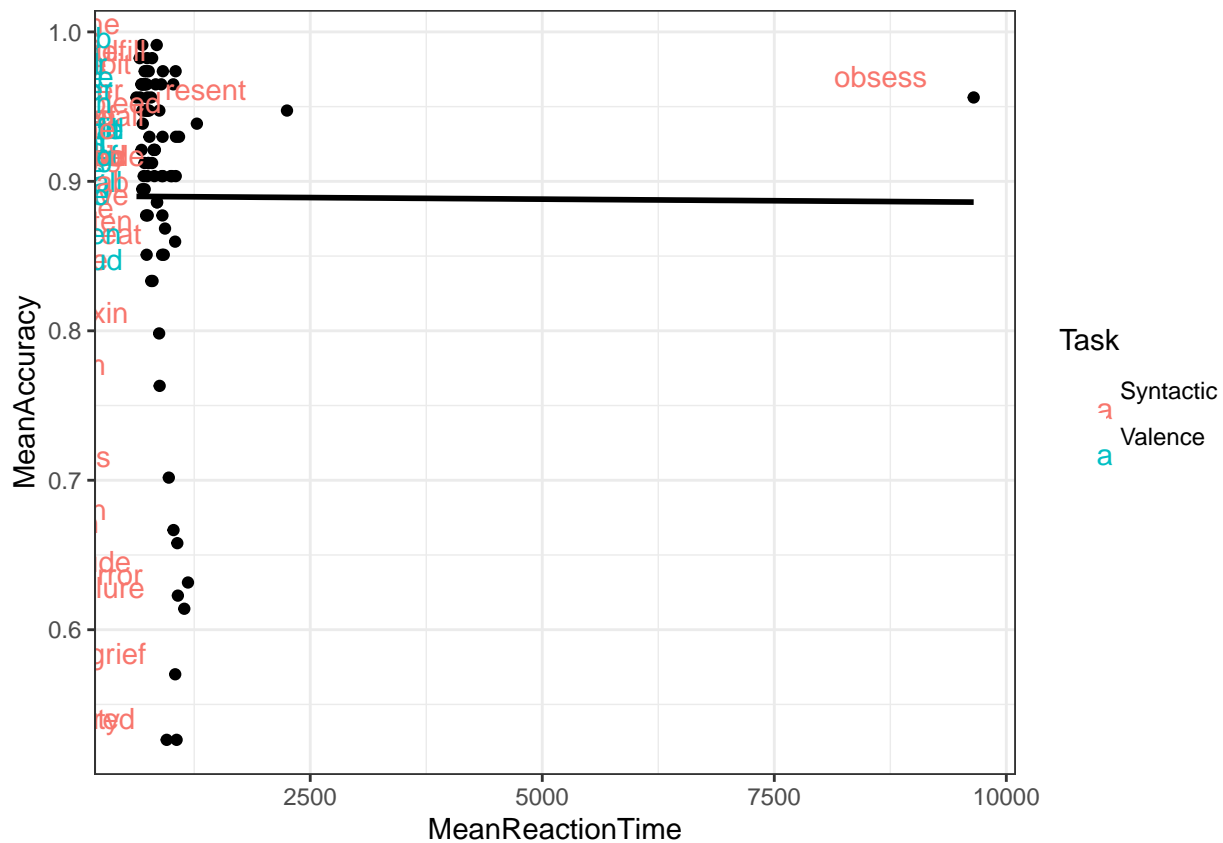
```
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -10.2915  -0.4795  -0.1272   0.3151  17.6875
##
## Random effects:
##   Groups   Name                Variance Std.Dev. Corr
##   Word     (Intercept)  0.003202  0.05659
##             cTask       0.007315  0.08553  -0.49
##   ID.true  (Intercept)  0.162938  0.40366
##             cTask       0.093025  0.30500   0.10
## Residual                   0.171784  0.41447
## Number of obs: 9120, groups: Word, 40; ID.true, 38
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)    6.58252    0.06624   38.34736  99.380 < 2e-16 ***
## cAccuracy      0.04687    0.01664  8842.73027   2.817  0.004854 **
## cTask          -0.21868    0.05204   42.16650  -4.202  0.000134 ***
## cAccuracy:cTask -0.02013    0.03327  8941.66521  -0.605  0.545126
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) cAccrc cTask
## cAccuracy   -0.003
## cTask        0.079 -0.026
## cAccrcy:cTs -0.010  0.301 -0.008
## optimizer (nloptwrap) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 0.00278333 (tol = 0.002, component 1)

agr <- d %>%
  group_by(Word,Task) %>%
  summarize(MeanAccuracy = mean(Accuracy),
            MeanReactionTime = mean(ReactionTime))

## `summarise()` has grouped output by 'Word'. You can override using the
## `.groups` argument.

ggplot(agr, aes(x = MeanReactionTime, y = MeanAccuracy)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "black") +
  geom_text(aes(label = Word, color = Task), vjust = -0.5, hjust = 1.5)

## `geom_smooth()` using formula = 'y ~ x'
```



```
# guides(legend = "none")
# theme(legend.position = "none") # Remove the legend
# ggsave("../graphs/exp1b_accXrt.pdf", width = 5, height = 3)
```

```
# Compute highest accuracy for Concrete
concrete_accuracy <- d %>%
  group_by(Word, Task) %>%
  summarize(MeanAccuracy = mean(Accuracy),
             MeanReactionTime = mean(ReactionTime)) %>%
  filter(Task == "Syntactic") %>%
  select(Word, MeanAccuracy) %>%
  rename(ConcreteAccuracy = MeanAccuracy) %>%
  arrange(desc(ConcreteAccuracy)) %>%
  head(10)
```

```
## `summarise()` has grouped output by 'Word'. You can override using the
## `.groups` argument.
```

```
agr <- d %>%
  filter(Word %in% concrete_accuracy$Word) %>%
  group_by(Word, Task) %>%
  summarize(MeanAccuracy = mean(Accuracy),
             MeanReactionTime = mean(ReactionTime))
```

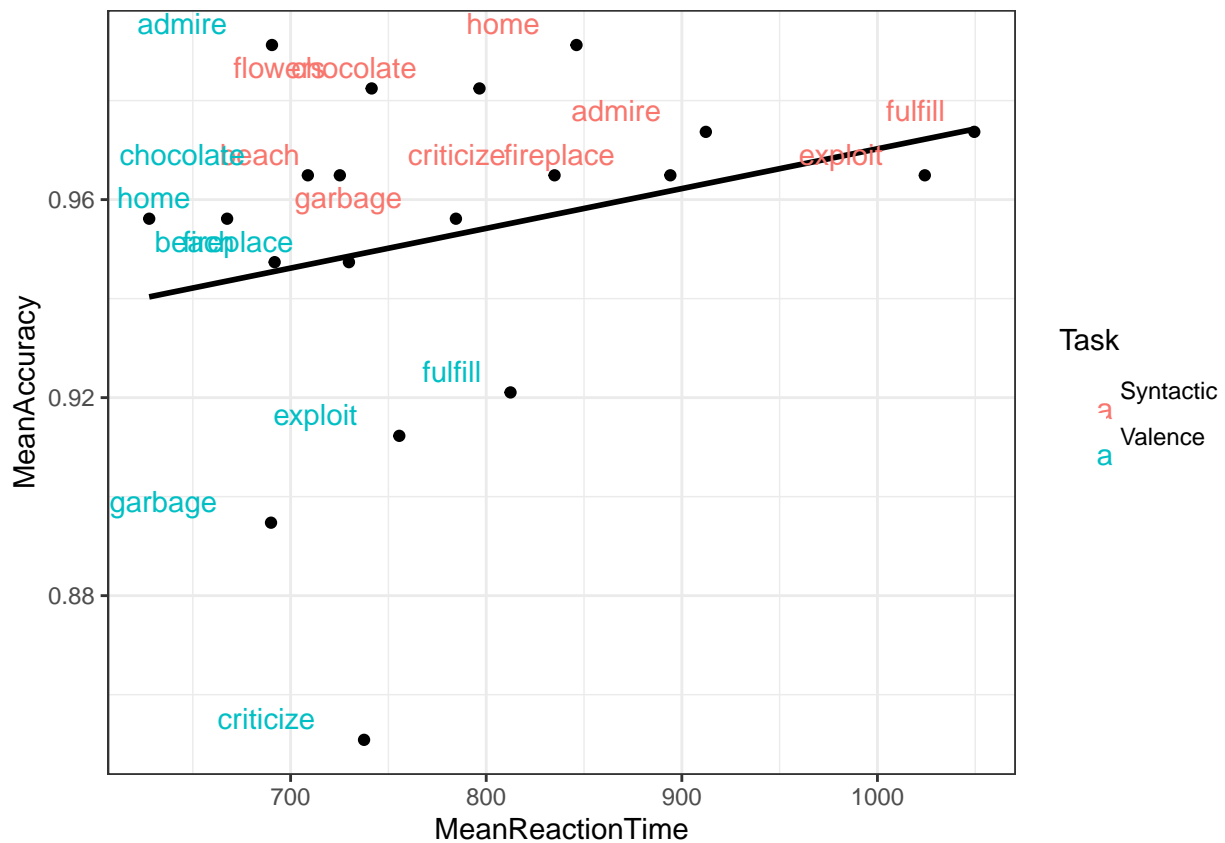
```
## `summarise()` has grouped output by 'Word'. You can override using the
## `.groups` argument.
```

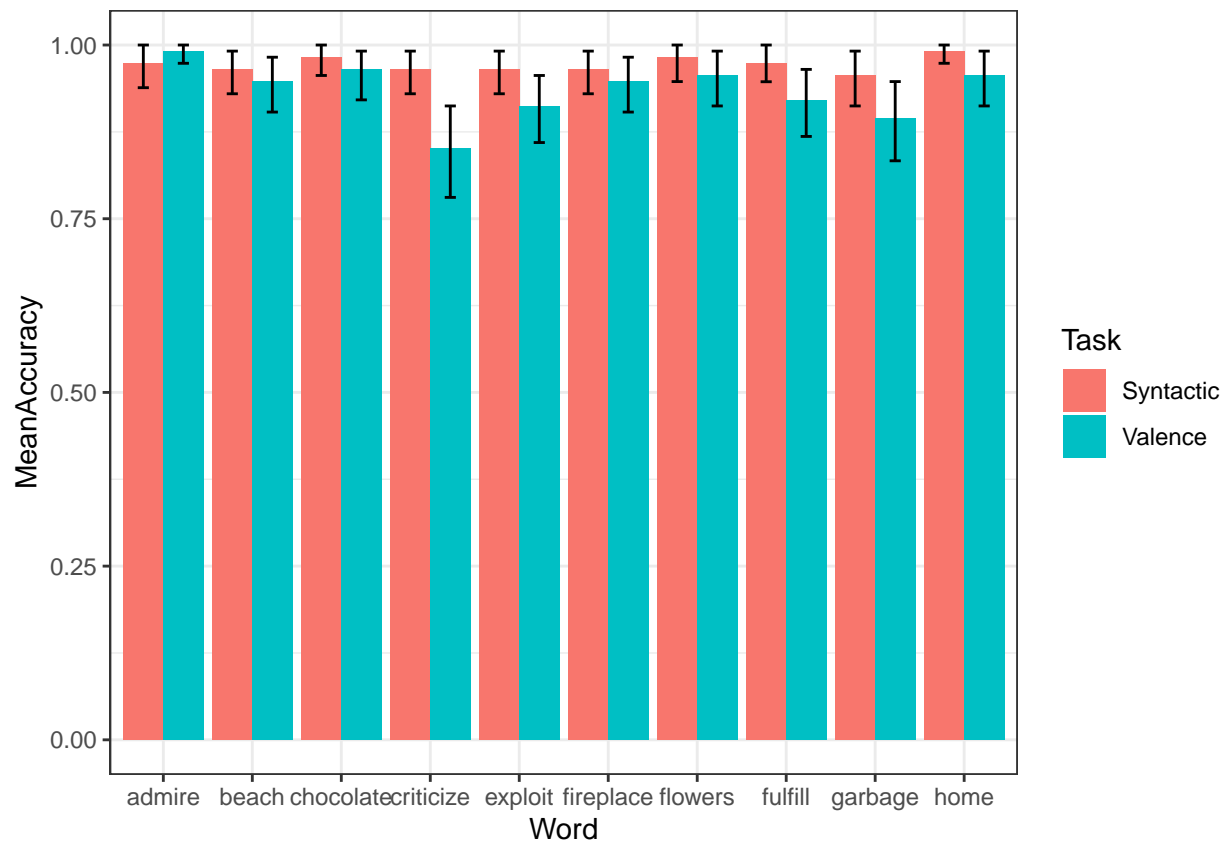
```
print(agr)
```

```
## # A tibble: 20 x 4
## # Groups:   Word [10]
##   Word      Task MeanAccuracy MeanReactionTime
##   <chr>    <chr>         <dbl>         <dbl>
## 1 admire  Syntactic      0.974          912.
## 2 admire  Valence         0.991          690.
## 3 beach   Syntactic      0.965          725.
## 4 beach   Valence         0.947          692.
## 5 chocolate Syntactic    0.982          797.
## 6 chocolate Valence       0.965          709.
## 7 criticize Syntactic    0.965          835.
## 8 criticize Valence       0.851          738.
## 9 exploit  Syntactic      0.965         1024.
## 10 exploit  Valence         0.912          756.
## 11 fireplace Syntactic    0.965          894.
## 12 fireplace Valence       0.947          730.
## 13 flowers  Syntactic      0.982          741.
## 14 flowers  Valence         0.956          628.
## 15 fulfill  Syntactic      0.974         1050.
## 16 fulfill  Valence         0.921          812.
## 17 garbage  Syntactic      0.956          785.
## 18 garbage  Valence         0.895          690.
## 19 home     Syntactic      0.991          846.
## 20 home     Valence         0.956          668.
```

```
ggplot(agr, aes(x = MeanReactionTime, y = MeanAccuracy)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "black") +
  geom_text(aes(label = Word, color = Task), vjust = -0.5, hjust = 1.5)
```

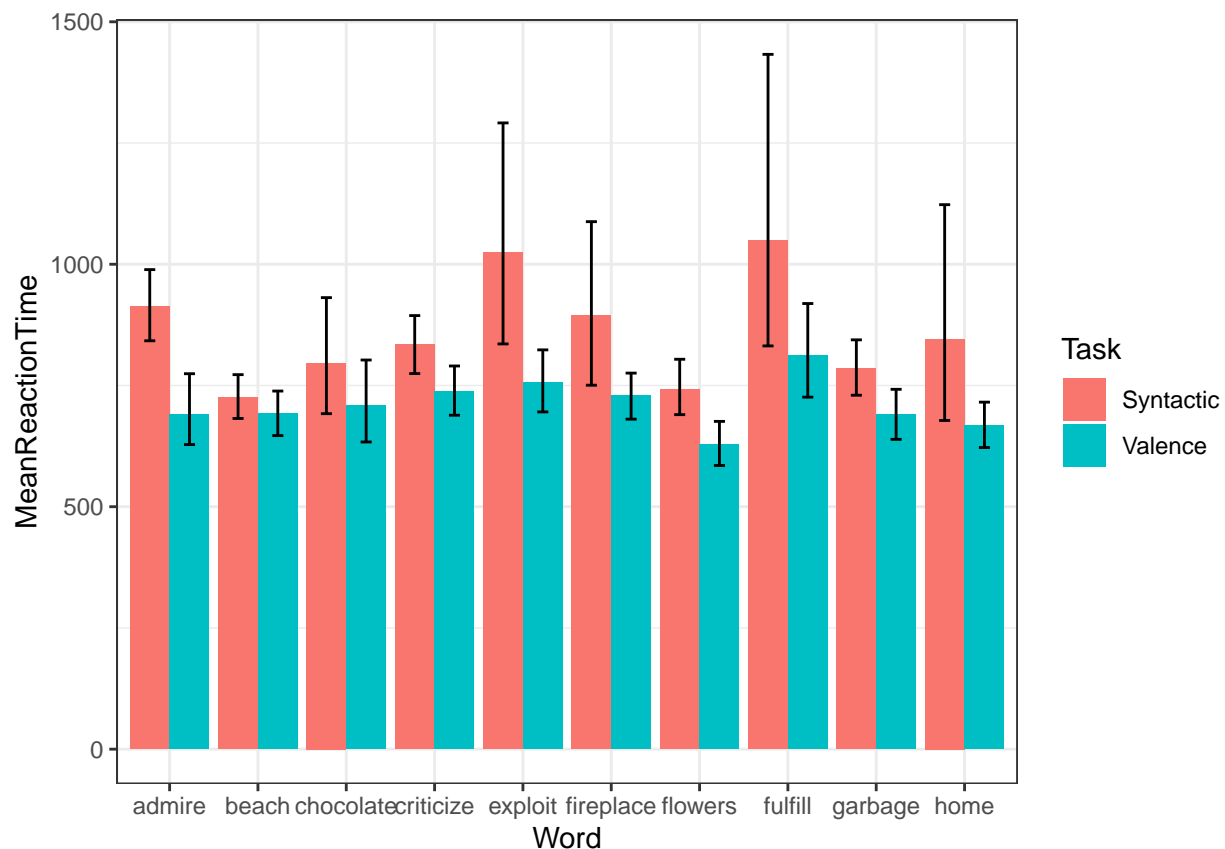
```
## `geom_smooth()` using formula = 'y ~ x'
```





```
agr <- d %>%
  filter(Word %in% concrete_accuracy$Word) %>%
  group_by(Word, Task) %>%
  reframe(MeanReactionTime = mean(ReactionTime),
           CILow = ci.low(ReactionTime),
           CIHigh = ci.high(ReactionTime)) %>%
  mutate(YMin = MeanReactionTime - CILow,
         YMax = MeanReactionTime + CIHigh)
# View(agr)

dodge = position_dodge(.9)
ggplot(data=agr, aes(x=Word, y=MeanReactionTime, fill = Task)) +
  geom_bar(position=dodge, stat="identity") +
  geom_errorbar(aes(ymin=YMin, ymax=YMax), width=.25, position=position_dodge(0.9))
```



First Remove participants who aren't super , aggregating over Task

```
length(unique(d$ID.true))
```

```
## [1] 38
```

```
inacc.parts <- d %>%
  group_by(ID.true, Task) %>%
  summarise(MeanAccuracy = mean(Accuracy)) %>%
  filter(MeanAccuracy < .75)
```

```
## `summarise()` has grouped output by 'ID.true'. You can override using the
## `.groups` argument.
```

```
# How many participants have Accuracy < .75?
length(unique(inacc.parts$ID.true))
```

```
## [1] 12
```

```
d.inaccurate.removed <- d %>%
  anti_join(inacc.parts, by = "ID.true")

# Sanity check
length(unique(d.inaccurate.removed$ID.true))
```

```
## [1] 26
```



```
# remove all inaccurate trials

orig <- nrow(d.inaccurate.removed)
d.inaccurate.removed <- d.inaccurate.removed %>%
  filter(Accuracy == 1)
nrow(d.inaccurate.removed)/orig*100

## [1] 94.29487

# Remove subjects with ReactionTime higher than 3x IQR
summary(d.inaccurate.removed$LogReactionTime)

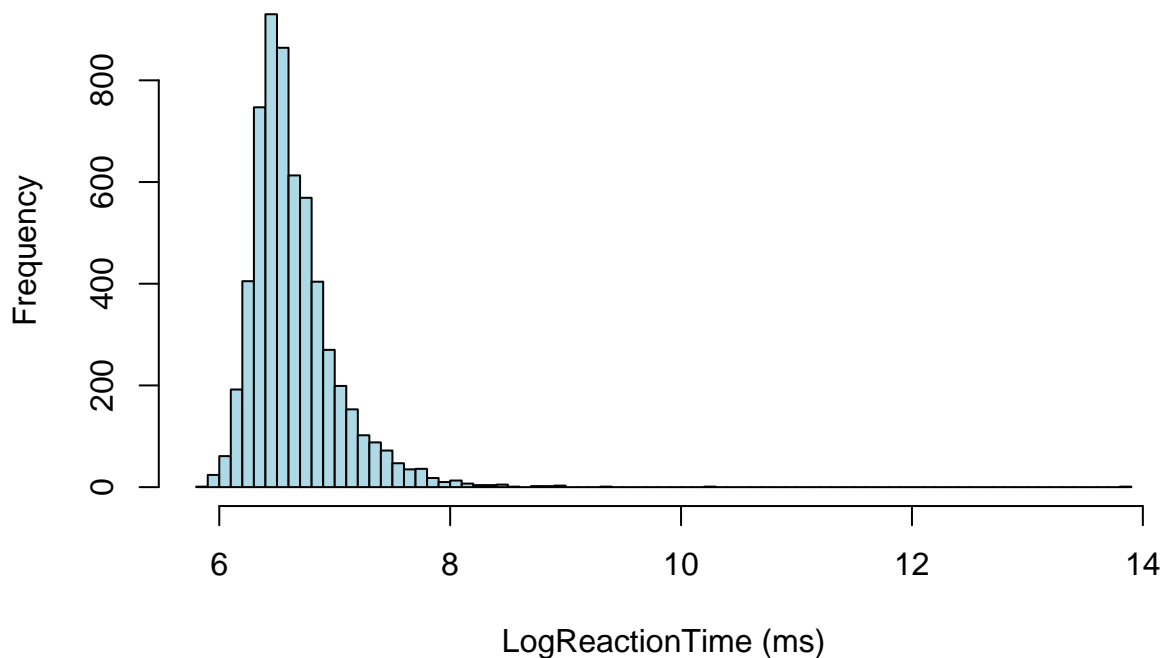
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      5.855   6.404   6.567   6.647   6.801  13.818

# Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
# 6.924   7.328   7.436   7.479   7.579  10.008
range(d.inaccurate.removed$LogReactionTime)

## [1] 5.855072 13.817820

hist(d.inaccurate.removed$LogReactionTime, breaks=100, col="lightblue", xlab="LogReactionTime (ms)",
      main="Histogram with Normal Curve")
```

Histogram with Normal Curve



```
quantile(d.inaccurate.removed$LogReactionTime)

##      0%      25%      50%      75%     100%
## 5.855072 6.403574 6.566672 6.801283 13.817820

IQR(d.inaccurate.removed$LogReactionTime)*3 # 0.7526289

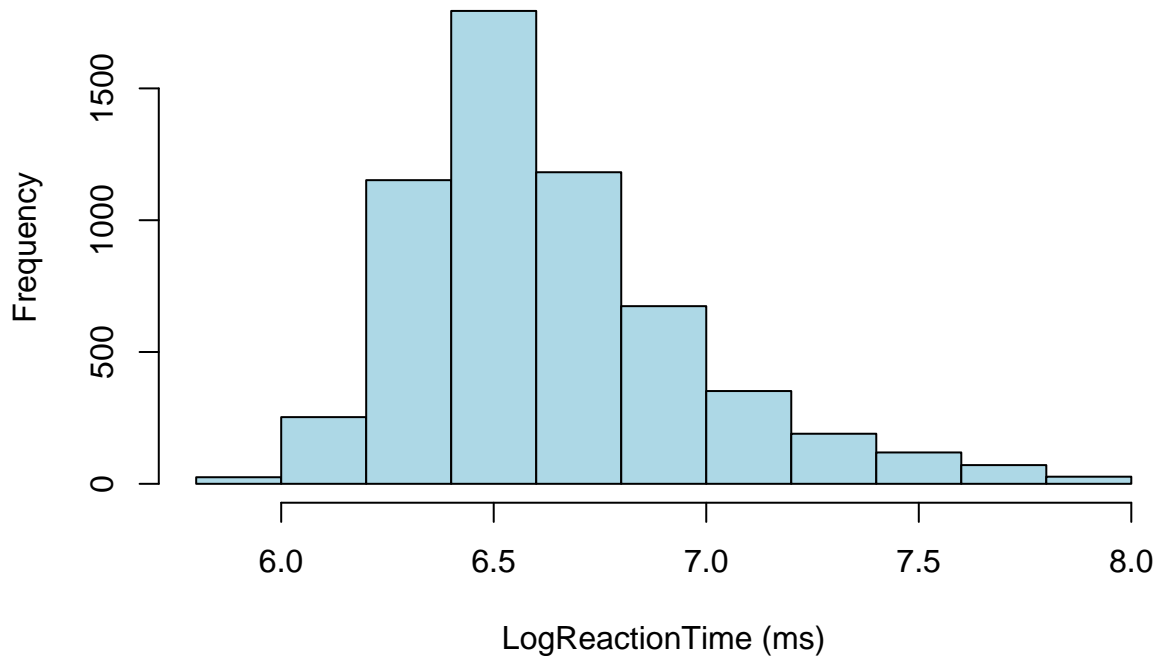
## [1] 1.193127
```

```
cutoff.high <- quantile(d.inaccurate.removed$LogReactionTime)[4] + IQR(d.inaccurate.removed$LogReactionTime)
cutoff.low <- quantile(d.inaccurate.removed$LogReactionTime)[2] - IQR(d.inaccurate.removed$LogReactionTime)

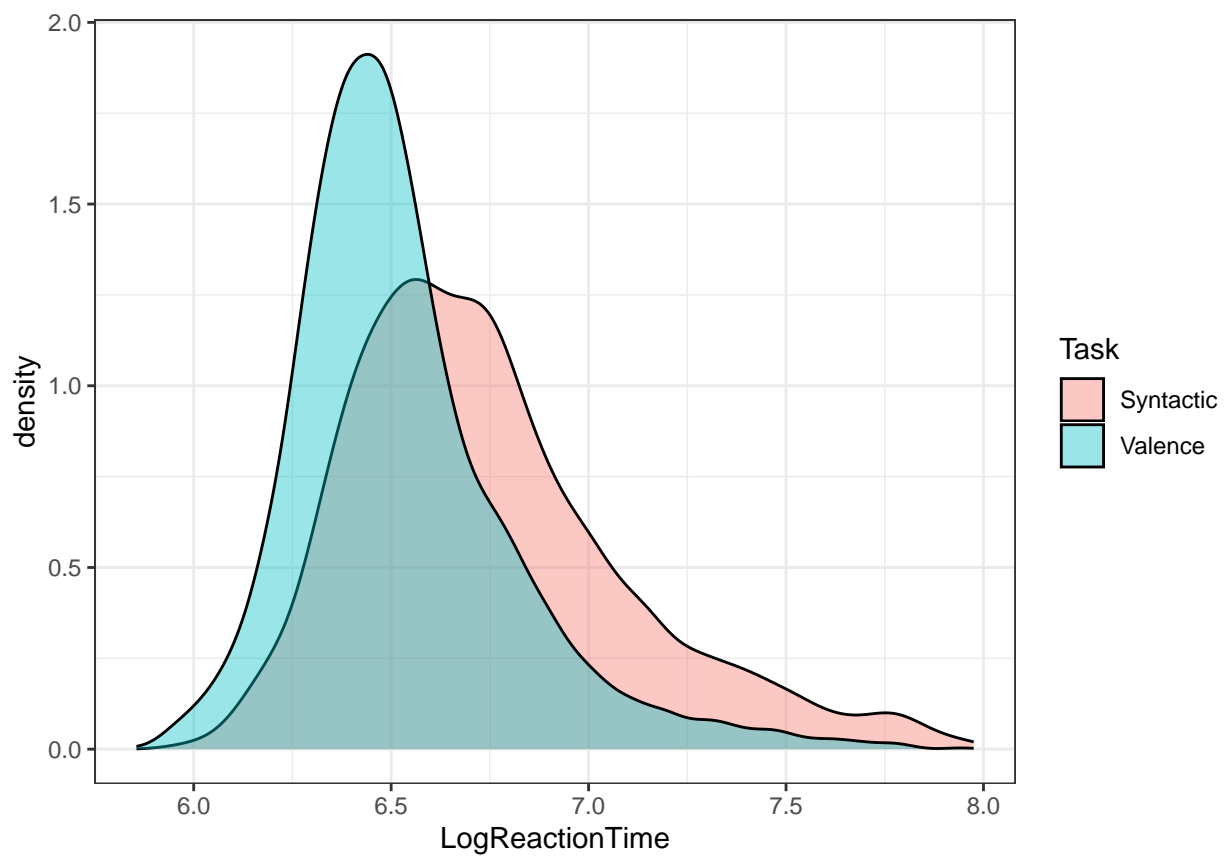
# remove subjects with ReactionTime higher than 3 x IQR
df.outliers.removed <- subset(d.inaccurate.removed, (d.inaccurate.removed$LogReactionTime > cutoff.low))

hist(df.outliers.removed$LogReactionTime, col="lightblue", xlab="LogReactionTime (ms)",
      main="Histogram with Normal Curve")
```

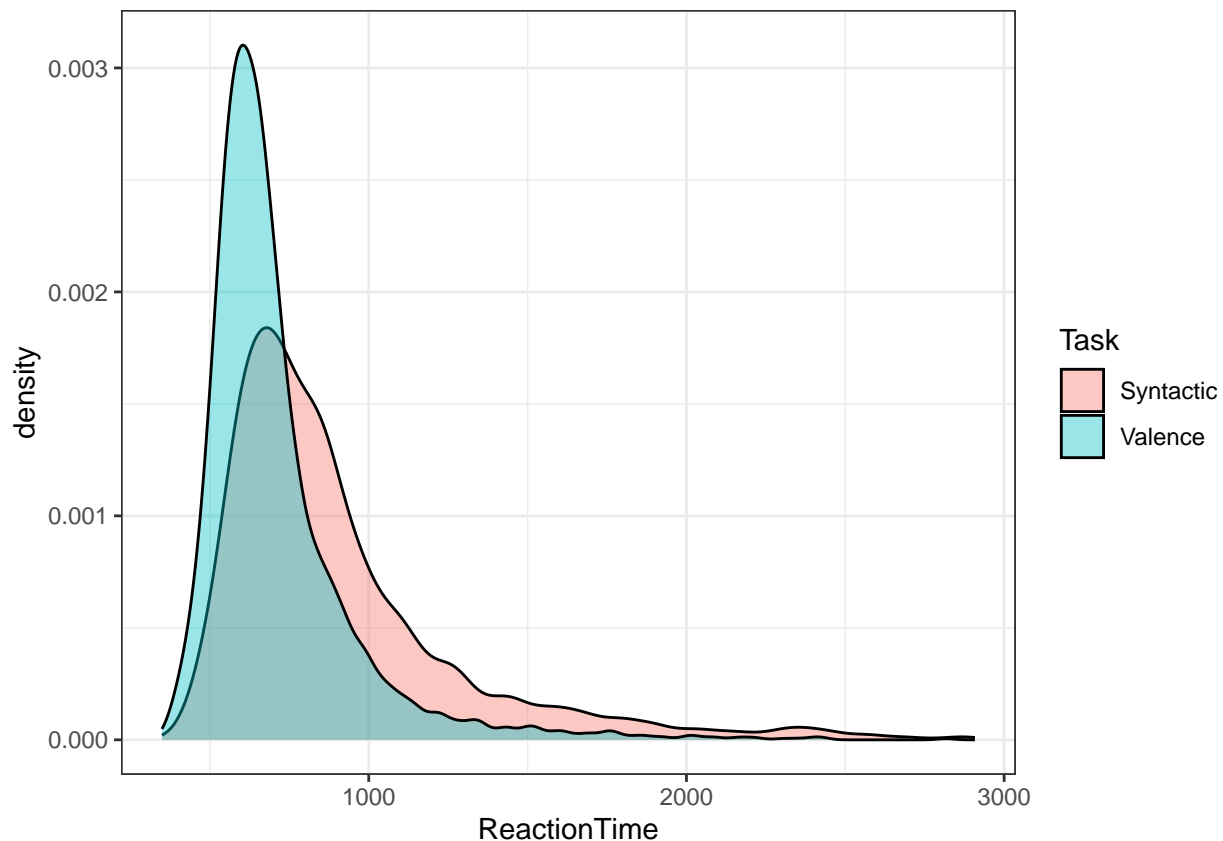
Histogram with Normal Curve



```
ggplot(df.outliers.removed, aes(x=LogReactionTime, fill=Task)) +
  # facet_wrap(~BlockOrder) +
  geom_density(alpha = .4)
```



```
ggplot(df.outliers.removed, aes(x=ReactionTime, fill=Task)) +  
  # facet_wrap(~BlockOrder) +  
  geom_density(alpha = .4)
```



convert everything to factors

Is there a difference between Semantic and Valence Tasks?

Yes

```
m = lmer(LogReactionTime ~ cTask + (1+cTask|ID.true) + (1+cTask|Word), data=center)
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LogReactionTime ~ cTask + (1 + cTask | ID.true) + (1 + cTask |
##      Word)
##      Data: center
##
## REML criterion at convergence: 1466.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.9018 -0.6370 -0.1656  0.4215  5.6813
##
## Random effects:
##      Groups      Name              Variance Std.Dev. Corr
##      Word       (Intercept) 0.003538 0.05949
##              cTask         0.011500 0.10724 -0.61
##      ID.true    (Intercept) 0.020943 0.14472
##              cTask         0.012020 0.10964 -0.29
```

```
## Residual          0.071234 0.26690
## Number of obs: 5839, groups: Word, 40; ID.true, 26
##
## Fixed effects:
##           Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  6.63720    0.03010 30.39139 220.474 < 2e-16 ***
## cTask        -0.22815    0.02827 47.52549  -8.071 1.84e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##      (Intr)
## cTask -0.325
```

Is there an Interaction between Task and WordType (ConcVal-Combo)?

Y.

```
m = lmer(LogReactionTime ~ cTask*ConcValCombo + (1+ConcValCombo+cTask|ID.true) + (1+cTask|Word), data=c
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00550469 (tol = 0.002, component 1)
```

```
# saveRDS(m, "../models/model-Task-ConcValCombo_outlier_excl_ReactionTime.rds")
```

```
# m <- readRDS("../models/model-Task-ConcValCombo_outlier_excl_ReactionTime.rds")
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LogReactionTime ~ cTask * ConcValCombo + (1 + ConcValCombo +
##      cTask | ID.true) + (1 + cTask | Word)
##      Data: center
##
## REML criterion at convergence: 1455.5
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.8808 -0.6407 -0.1621  0.4250  5.7333
##
## Random effects:
##      Groups      Name                Variance Std.Dev. Corr
##      Word      (Intercept)           0.0025060 0.05006
##              cTask                 0.0071576 0.08460  -0.42
##      ID.true  (Intercept)           0.0230068 0.15168
##              ConcValComboabstract-positive 0.0003307 0.01819   0.11
##              ConcValComboconcrete-negative 0.0018524 0.04304  -0.26 -0.39
##              ConcValComboconcrete-positive 0.0015111 0.03887  -0.58  0.17  0.71
##              cTask                 0.0120585 0.10981  -0.29  0.22  0.02
##      Residual                    0.0706719 0.26584
##
##
##
```

```

##
##
##
##
## 0.00
##
## Number of obs: 5839, groups: Word, 40; ID.true, 26
##
## Fixed effects:
##
## Estimate Std. Error df t value
## (Intercept) 6.68137 0.03445 37.50765 193.954
## cTask -0.26735 0.03722 56.27721 -7.184
## ConcValComboabstract-positive -0.02101 0.02476 36.98153 -0.849
## ConcValComboconcrete-negative -0.06874 0.02591 41.22225 -2.653
## ConcValComboconcrete-positive -0.08690 0.02566 39.78034 -3.386
## cTask:ConcValComboabstract-positive -0.05437 0.04277 35.65521 -1.271
## cTask:ConcValComboconcrete-negative 0.09214 0.04274 35.55021 2.156
## cTask:ConcValComboconcrete-positive 0.11992 0.04277 35.62404 2.804
## Pr(>|t|)
## (Intercept) < 2e-16 ***
## cTask 1.68e-09 ***
## ConcValComboabstract-positive 0.40154
## ConcValComboconcrete-negative 0.01127 *
## ConcValComboconcrete-positive 0.00161 **
## cTask:ConcValComboabstract-positive 0.21191
## cTask:ConcValComboconcrete-negative 0.03797 *
## cTask:ConcValComboconcrete-positive 0.00811 **
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
## (Intr) cTask CncVlCmbb- CncVlCmbcnrt-n CncVlCmbcnrt-p
## cTask -0.289
## CncVlCmbb- -0.340 0.218
## CncVlCmbcnrt-n -0.411 0.194 0.452
## CncVlCmbcnrt-p -0.490 0.192 0.482 0.523
## cTsk:CncVlCmbb- 0.125 -0.578 -0.345 -0.166 -0.168
## cTsk:CncVlCmbcnrt-n 0.125 -0.579 -0.174 -0.327 -0.168
## cTsk:CncVlCmbcnrt-p 0.125 -0.579 -0.174 -0.166 -0.330
## cTsk:CncVlCmbb- cTsk:CncVlCmbcnrt-n
## cTask
## CncVlCmbb-
## CncVlCmbcnrt-n
## CncVlCmbcnrt-p
## cTsk:CncVlCmbb-
## cTsk:CncVlCmbcnrt-n 0.504
## cTsk:CncVlCmbcnrt-p 0.503 0.504
## optimizer (nloptwrap) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 0.00550469 (tol = 0.002, component 1)

```

Is there an Interaction between Task and Syntactic Category?

No.

```

m = lmer(LogReactionTime ~ cTask*cCategory + (1+cCategory+cTask|ID.true) + (1+cTask|Word), data=center)

summary(m)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LogReactionTime ~ cTask * cCategory + (1 + cCategory + cTask |
##      ID.true) + (1 + cTask | Word)
##      Data: center
##
## REML criterion at convergence: 1443.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.9318 -0.6360 -0.1612  0.4190  5.6991
##
## Random effects:
##   Groups      Name                Variance Std.Dev. Corr
##   Word       (Intercept)  0.003307  0.05751
##             cTask        0.011707  0.10820  -0.71
##   ID.true    (Intercept)  0.021017  0.14497
##             cCategory    0.002805  0.05296  -0.02
##             cTask        0.012234  0.11061  -0.30  0.13
## Residual                0.070556  0.26562
## Number of obs: 5839, groups:  Word, 40; ID.true, 26
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)    6.63766    0.03005 30.01717 220.861 < 2e-16 ***
## cTask          -0.22884    0.02850 47.39362  -8.031 2.16e-10 ***
## cCategory       0.03682    0.02208 49.37532   1.668  0.102
## cTask:cCategory 0.03440    0.03696 37.73229   0.931  0.358
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) cTask  cCtgry
## cTask        -0.345
## cCategory    -0.006  0.046
## cTsk:cCtgry  0.000  0.002 -0.539

```

Three-way interaction?Y

Yes.

```

m = lmer(LogReactionTime ~ cTask*cCategory*ConcValCombo + (1+cCategory+ConcValCombo+cTask|ID.true) + (1+cTask|Word), data=center)

## boundary (singular) fit: see help('isSingular')
## Warning: Model failed to converge with 1 negative eigenvalue: -9.1e+00
# saveRDS(m, "../models/model-Task-ConcValCombo-Category.rds")
# m <- readRDS("../models/model-Task-ConcValComb-Category.rds")
summary(m)

```

```

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LogReactionTime ~ cTask * cCategory * ConcValCombo + (1 + cCategory +
##      ConcValCombo + cTask | ID.true) + (1 + cTask | Word)
##      Data: center
##
## REML criterion at convergence: 1400.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.9567 -0.6380 -0.1610  0.4214  5.7809
##
## Random effects:
##      Groups   Name                Variance Std.Dev. Corr
##      Word     (Intercept)          0.0005903 0.02430
##              cTask                0.0023847 0.04883  0.39
##      ID.true  (Intercept)          0.0236851 0.15390
##              cCategory            0.0028782 0.05365 -0.11
##              ConcValComboabstract-positive 0.0005086 0.02255 -0.05  1.00
##              ConcValComboconcrete-negative 0.0022091 0.04700 -0.31  0.12  0.10
##              ConcValComboconcrete-positive 0.0016637 0.04079 -0.61  0.41  0.38
##              cTask                0.0123633 0.11119 -0.30  0.15  0.13
##      Residual                    0.0699865 0.26455
##
##
##
##
##
##
##      0.79
##      0.02  0.01
##
## Number of obs: 5839, groups:  Word, 40; ID.true, 26
##
## Fixed effects:
##
##              Estimate Std. Error      df
## (Intercept)      6.68344    0.03195 27.90341
## cTask            -0.27035    0.03030 49.88012
## cCategory        -0.05952    0.02347 45.12167
## ConcValComboabstract-positive -0.02331    0.01536 33.79721
## ConcValComboconcrete-negative -0.07089    0.01735 35.34690
## ConcValComboconcrete-positive -0.08801    0.01675 34.89474
## cTask:cCategory    0.25548    0.04209 33.26507
## cTask:ConcValComboabstract-positive -0.05104    0.02952 32.19514
## cTask:ConcValComboconcrete-negative  0.09485    0.02948 32.03700
## cTask:ConcValComboconcrete-positive  0.12106    0.02952 32.19447
## cCategory:ConcValComboabstract-positive  0.08048    0.02943 33.04817
## cCategory:ConcValComboconcrete-negative  0.08605    0.02940 32.91535
## cCategory:ConcValComboconcrete-positive  0.21714    0.02943 33.06682
## cTask:cCategory:ConcValComboabstract-positive -0.22912    0.05905 32.21196
## cTask:cCategory:ConcValComboconcrete-negative -0.28343    0.05897 32.04771
## cTask:cCategory:ConcValComboconcrete-positive -0.36723    0.05905 32.21498
##
##              t value Pr(>|t|)

```



```
## (Intercept)                209.173 < 2e-16 ***
## cTask                      -8.922 6.59e-12 ***
## cCategory                  -2.536 0.014733 *
## ConcValComboabstract-positive -1.517 0.138447
## ConcValComboconcrete-negative -4.086 0.000240 ***
## ConcValComboconcrete-positive -5.255 7.48e-06 ***
## cTask:cCategory             6.070 7.61e-07 ***
## cTask:ConcValComboabstract-positive -1.729 0.093390 .
## cTask:ConcValComboconcrete-negative 3.217 0.002958 **
## cTask:ConcValComboconcrete-positive 4.101 0.000261 ***
## cCategory:ConcValComboabstract-positive 2.735 0.009960 **
## cCategory:ConcValComboconcrete-negative 2.927 0.006162 **
## cCategory:ConcValComboconcrete-positive 7.378 1.77e-08 ***
## cTask:cCategory:ConcValComboabstract-positive -3.880 0.000486 ***
## cTask:cCategory:ConcValComboconcrete-negative -4.806 3.48e-05 ***
## cTask:cCategory:ConcValComboconcrete-positive -6.219 5.61e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

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

## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

Main Effect of Block Order

On ReactionTime

- No.

```
m = lmer(LogReactionTime ~ cBlockOrder + (1|ID.true) + (1+cBlockOrder|Word), data=center)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LogReactionTime ~ cBlockOrder + (1 | ID.true) + (1 + cBlockOrder |
##      Word)
##      Data: center
##
## REML criterion at convergence: 2666.1
##
## Scaled residuals:
##      Min      1Q  Median      3Q      Max
## -2.4695 -0.6781 -0.2095  0.4707  4.8111
##
## Random effects:
##      Groups      Name      Variance Std.Dev. Corr
##      Word      (Intercept) 2.922e-03 0.054053
##              cBlockOrder 1.133e-05 0.003367 -1.00
##      ID.true  (Intercept) 2.122e-02 0.145682
```

```
## Residual 8.964e-02 0.299398
## Number of obs: 5839, groups: Word, 40; ID.true, 26
##
## Fixed effects:
## Estimate Std. Error df t value Pr(>|t|)
## (Intercept) 6.634217 0.030078 28.195294 220.563 <2e-16 ***
## cBlockOrder 0.004662 0.057679 23.996868 0.081 0.936
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
## (Intr)
## cBlockOrder -0.002
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

Simple effect of ConcValCombo on ReactionTime?

sorta

```
m = lmer(LogReactionTime ~ ConcValCombo + (1+ConcValCombo|ID.true) + (1|Word), data=center)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LogReactionTime ~ ConcValCombo + (1 + ConcValCombo | ID.true) +
## (1 | Word)
## Data: center
##
## REML criterion at convergence: 2656.1
##
## Scaled residuals:
## Min 1Q Median 3Q Max
## -2.5585 -0.6756 -0.2047 0.4760 4.8257
##
## Random effects:
## Groups Name Variance Std.Dev. Corr
## Word (Intercept) 0.0021659 0.04654
## ID.true (Intercept) 0.0210858 0.14521
## ConcValComboabstract-positive 0.0001667 0.01291 0.60
## ConcValComboconcrete-negative 0.0018568 0.04309 -0.12 -0.62
## ConcValComboconcrete-positive 0.0009527 0.03087 -0.52 -0.53 0.83
## Residual 0.0891650 0.29861
## Number of obs: 5839, groups: Word, 40; ID.true, 26
##
## Fixed effects:
## Estimate Std. Error df t value Pr(>|t|)
## (Intercept) 6.67061 0.03304 36.41789 201.906 < 2e-16 ***
## ConcValComboabstract-positive -0.01397 0.02375 36.68686 -0.588 0.56011
## ConcValComboconcrete-negative -0.05644 0.02507 40.41824 -2.252 0.02985 *
## ConcValComboconcrete-positive -0.07542 0.02438 36.99626 -3.094 0.00375 **
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) CncVlCmbb- CncVlCmbcncrt-n
## CncVlCmbbs-      -0.303
## CncVlCmbcncrt-n -0.373  0.449
## CncVlCmbcncrt-p -0.460  0.470      0.529
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

In the Concreteness task, is there a difference between concreteness and abstractness on ReactionTime?

- Nope

```
str(df_factors)
```

```
## 'data.frame':   5839 obs. of  19 variables:
## $ X              : Factor w/ 5839 levels "1","2","3","4",...: 1 2 3 4 5 6 7 8 9 10 ...
## $ ID.true        : Factor w/ 26 levels "55a2fa33fdf99b073cd94947",...: 1 1 1 1 1 1 1 1 1 1 ...
## $ Word           : Factor w/ 40 levels "admire","anxiety",...: 1 35 5 25 27 38 30 11 29 24 ...
## $ Label          : Factor w/ 2 levels "test_syn","test_val": 1 1 1 1 1 1 1 1 1 1 ...
## $ ConcValCombo    : Factor w/ 4 levels "abstract-negative",...: 2 4 3 2 1 1 2 3 2 4 ...
## $ Category        : Factor w/ 2 levels "Nouns","Verb": 2 2 2 1 1 1 1 1 2 2 ...
## $ Task            : Factor w/ 2 levels "Syntactic","Valence": 1 1 1 1 1 1 1 1 1 1 ...
## $ BlockOrder      : Factor w/ 2 levels "SV","VS": 1 1 1 1 1 1 1 1 1 1 ...
## $ Group           : Factor w/ 4 levels "negative;positive",...: 4 4 4 4 4 4 4 4 4 4 ...
## $ Response        : Factor w/ 4 levels "negative","noun",...: 4 4 4 2 2 2 2 2 4 4 ...
## $ Accuracy        : Factor w/ 1 level "1": 1 1 1 1 1 1 1 1 1 1 ...
## $ EventTime       : Factor w/ 5838 levels "1739885303396",...: 2093 2095 2098 2101 2103 2107 2109 211...
## $ Value           : Factor w/ 4 levels "negative","noun",...: 4 4 4 2 2 2 2 2 4 4 ...
## $ ReactionTime    : int   1114 518 605 981 643 709 500 629 606 493 ...
## $ Key_value_F     : Factor w/ 4 levels "negative","noun",...: 2 2 2 2 2 2 2 2 2 2 ...
## $ Key_value_J     : Factor w/ 2 levels "A","B": 2 2 2 2 2 2 2 2 2 2 ...
## $ Comments        : Factor w/ 0 levels: NA NA NA NA NA NA NA NA NA ...
## $ LogReactionTime : num   7.02 6.25 6.41 6.89 6.47 ...
## $ TrialNumber      : Factor w/ 240 levels "1","2","3","4",...: 1 2 3 4 5 6 7 8 9 10 ...
```

```
sem <- df_factors %>%
  filter(Task == "Syntactic") %>%
  mutate(
    Syntactic = ifelse(grepl("verb", Category), "verb",
                      ifelse(grepl("noun", Category), "noun", NA)),
    Valence = ifelse(grepl("positive", ConcValCombo), "positive",
                    ifelse(grepl("negative", ConcValCombo), "negative", NA)),
    cConcValCombo = as.numeric(ConcValCombo) - mean(as.numeric(ConcValCombo)),
    cCategory = as.numeric(Category) - mean(as.numeric(Category)),
    # cSyntactic = as.numeric(factor(Syntactic)) - mean(as.numeric(factor(Syntactic)))
  )
```

```
m = lmer(LogReactionTime ~ cConcValCombo + (1+cConcValCombo|ID.true) + (1+cConcValCombo|Word), data=sem)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LogReactionTime ~ cConcValCombo + (1 + cConcValCombo | ID.true) +
##      (1 + cConcValCombo | Word)
##      Data: sem
##
## REML criterion at convergence: 1326.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6251 -0.6735 -0.1694  0.4546  4.5573
##
## Random effects:
##      Groups   Name                Variance Std.Dev. Corr
##      Word      (Intercept)    0.0001408 0.01187
##              cConcValCombo 0.0057251 0.07566 -1.00
##      ID.true   (Intercept)    0.0289161 0.17005
##              cConcValCombo 0.0006708 0.02590 -0.62
## Residual                0.0878382 0.29638
## Number of obs: 2833, groups:  Word, 40; ID.true, 26
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   6.75128    0.03494 28.38306 193.208 < 2e-16 ***
## cConcValCombo -0.07640    0.01518 37.35367  -5.033 1.25e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr)
## cConcValCmb -0.270
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

```
m = lmer(LogReactionTime ~ cCategory + (1+cCategory|ID.true) + (1|Word), data=sem)
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LogReactionTime ~ cCategory + (1 + cCategory | ID.true) + (1 |
##      Word)
##      Data: sem
##
## REML criterion at convergence: 1312.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6202 -0.6779 -0.1783  0.4833  4.4069
##
## Random effects:
##      Groups   Name                Variance Std.Dev. Corr
##      Word      (Intercept) 0.01085 0.1041
```

```
## ID.true (Intercept) 0.02948 0.1717
##          cCategory 0.01261 0.1123 -0.15
## Residual          0.08575 0.2928
## Number of obs: 2833, groups: Word, 40; ID.true, 26
##
## Fixed effects:
##          Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  6.75638    0.03790 36.29671 178.289 <2e-16 ***
## cCategory    0.01883    0.04115 52.55352   0.458   0.649
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr)
## cCategory -0.067
```

In the Valence task , is there a difference between positive and negative on ReactionTime?

- Nope.

```
val <- df_factors %>%
  filter(Task == "Valence") %>%
  mutate(
    Valence = ifelse(grepl("positive", ConcValCombo), "positive",
                     ifelse(grepl("negative", ConcValCombo), "negative", NA)),
    cConcValCombo = as.numeric(ConcValCombo) - mean(as.numeric(ConcValCombo)),
    cValence = as.numeric(factor(Valence)) - mean(as.numeric(factor(Valence)))
  )

m = lmer(LogReactionTime ~ cConcValCombo + (1+cConcValCombo|ID.true) + (1+cConcValCombo|Word), data=val,
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LogReactionTime ~ cConcValCombo + (1 + cConcValCombo | ID.true) +
##          (1 + cConcValCombo | Word)
## Data: val
##
## REML criterion at convergence: -35.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.9587 -0.6141 -0.1569  0.3910  6.4520
##
## Random effects:
## Groups   Name                Variance Std.Dev. Corr
## Word     (Intercept)    4.024e-04 0.020061
##           cConcValCombo 1.983e-03 0.044530 -0.12
## ID.true  (Intercept)    1.937e-02 0.139174
##           cConcValCombo 6.859e-05 0.008282 0.02
## Residual                    5.473e-02 0.233939
## Number of obs: 3006, groups: Word, 40; ID.true, 26
##
```

```

## Fixed effects:
##           Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  6.5211332  0.0284207 27.8023526 229.450  <2e-16 ***
## cConcValCombo -0.0002506  0.0096167 28.5084344  -0.026    0.979
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##           (Intr)
## cConcValCmb -0.011

```

```

m = lmer(LogReactionTime ~ cValence + (1+cValence|ID.true) + (1|Word), data=val)
summary(m)

```

```

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: LogReactionTime ~ cValence + (1 + cValence | ID.true) + (1 |
##      Word)
##      Data: val
##
## REML criterion at convergence: -36.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.0964 -0.6215 -0.1630  0.3871  6.4860
##
## Random effects:
##  Groups   Name                Variance Std.Dev. Corr
##  Word      (Intercept)  0.002605  0.05104
##  ID.true   (Intercept)  0.019393  0.13926
##           cValence      0.001115  0.03339  -0.19
## Residual                    0.054536  0.23353
## Number of obs: 3006, groups: Word, 40; ID.true, 26
##
## Fixed effects:
##           Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  6.52660    0.02880 29.21571 226.653  <2e-16 ***
## cValence     -0.02630    0.01939 40.69341  -1.356    0.183
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##           (Intr)
## cValence -0.061

```