STAT440 Project Appendix

Ian & Rosie

April 20, 2018

Visual Tests

Here, we'll perform some visual tests allowing the user to ensure that `stat440pkg` is returning sane results.

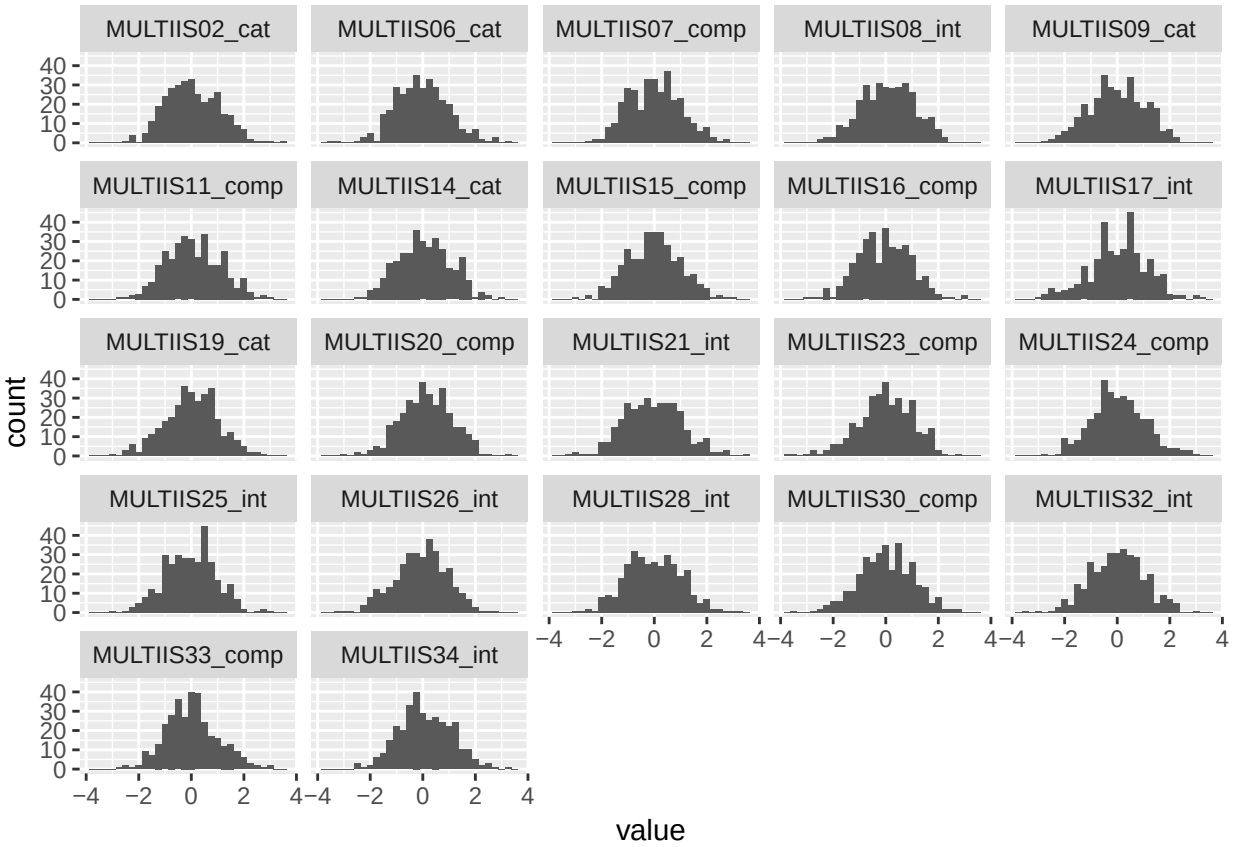
`gen.imp.resp`

Let's make sure `gen.imp.resp` is returning somewhat normally-distributed data.

```
library(stat440pkg)
library(tidyr)
library(ggplot2)

imp.resp <- gen.imp.resp(data = multiis, num.iter = 5)
gathered.data <- gather(imp.resp)
p <- ggplot(gathered.data) +
  geom_histogram(aes(x = value), binwidth = 0.25) +
  facet_wrap(~ key)

plot(p)
```



gen.latent.vars

Let's make sure `gen.latent.vars` is returning somewhat normally-distributed data. First we'll do so for Bartlett scores, then Thompson regression scores.

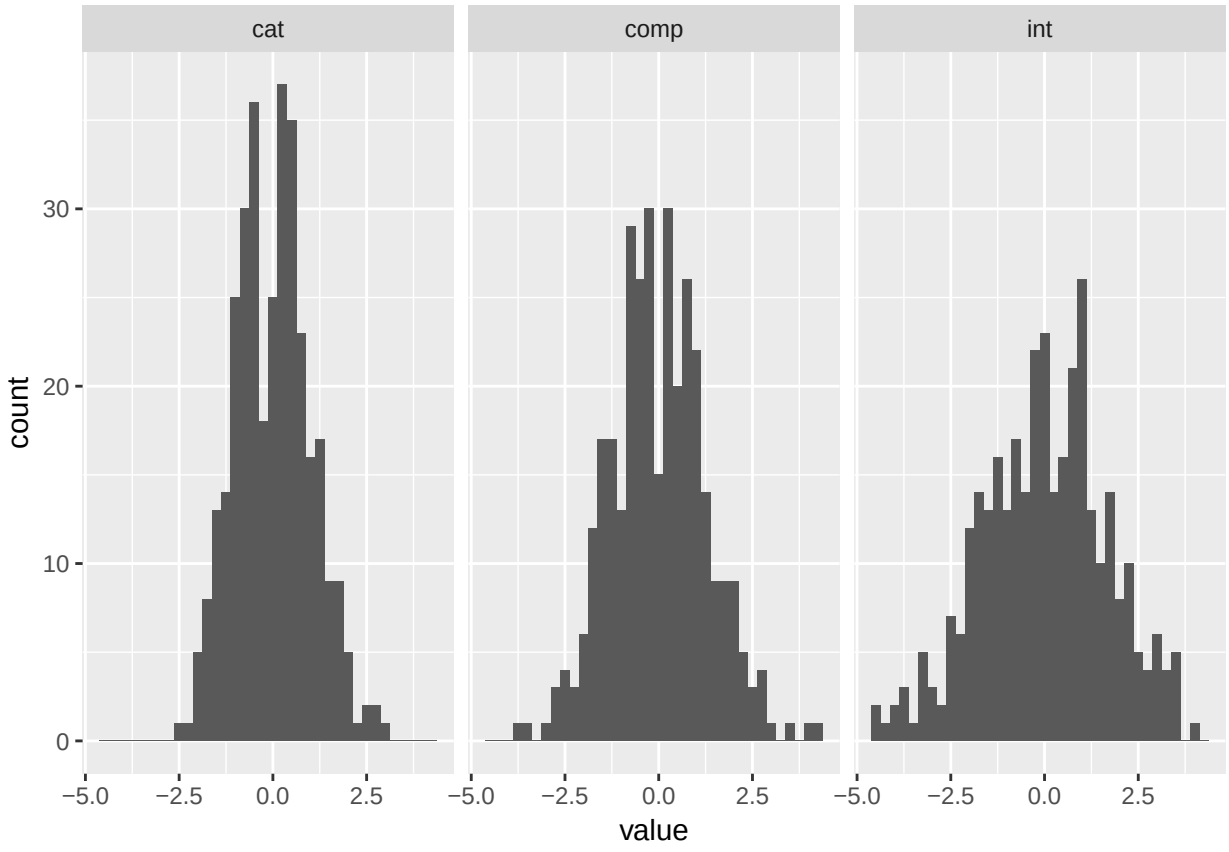
Bartlett factor scores

```
grp.indicator <- sapply(names(multiis), FUN =
                            function(x){strsplit(x, split = "_")[[1]][2]})

lv <- gen.latent.vars(data = multiis, grp.indicator = grp.indicator, num.iter = 5, scores = "Bartlett")

gathered.data <- gather(lv)
p <- ggplot(gathered.data) +
  geom_histogram(aes(x = value), binwidth = 0.25) +
  facet_wrap(~ key)

plot(p)
```



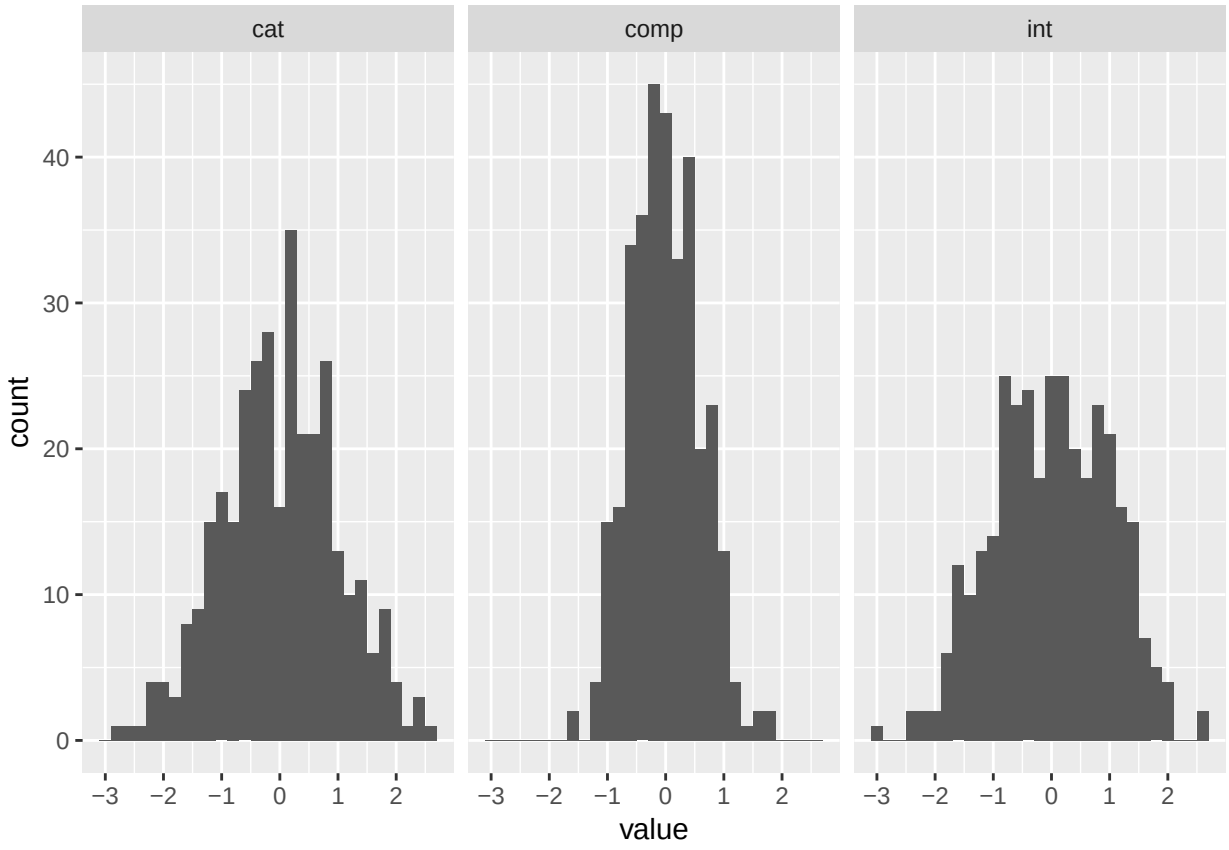
Thompson factor scores

```
grp.indicator <- sapply(names(multiis), FUN =
                            function(x){strsplit(x, split = "_")[[1]][2]})
```

```
lv <- gen.latent.vars(data = multiis, grp.indicator = grp.indicator, num.iter = 5, scores = "regression"

gathered.data <- gather(lv)
p <- ggplot(gathered.data) +
  geom_histogram(aes(x = value), binwidth = 0.2) +
  facet_wrap(~ key)

plot(p)
```



Results using Thompson scores for latent variables

Here, we'll create the similar plots to those that appear in the Results section of the report, but using Thompson scores.

```
M <- 50
latent.datasets <- gen.latent.datasets(M, multiis, grp.indicator = grp.indicator, num.iter = 5, scores =

pooled.add1 <- pool.analyses(latent.datasets, cat~comp + int, lm)
```

```
##                 Estimate  Std. Error   p value
## (Intercept)  4.206121e-19 0.006251122 1.0000000
## comp         1.727816e-02 0.017694082 0.3288198
## int         -1.453957e-02 0.016254854 0.3710676
```

```
pooled.add2 <- pool.analyses(latent.datasets, comp~cat + int, lm)
```

```
##                  Estimate Std. Error     p value
## (Intercept) -3.378923e-18 0.00569294 1.000000000
## cat          1.459822e-02 0.01501253 0.330850476
## int         -6.702097e-02 0.01928956 0.000511867
```

```
pooled.add3 <- pool.analyses(latent.datasets, int~comp + cat, lm)
```

```
##                  Estimate  Std. Error      p value
## (Intercept)  4.443759e-18 0.005975258 1.0000000000
## comp        -6.054266e-02 0.016651567 0.0002770617
## cat         -1.130570e-02 0.016675224 0.4977753971
```

```
signif(pooled.add1$hypothesis.test, digits = 3)
```

```
##              Estimate Std. Error p value
## (Intercept)  4.21e-19    0.00625   1.000
## comp         1.73e-02    0.01770   0.329
## int         -1.45e-02    0.01630   0.371
```

```
signif(pooled.add2$hypothesis.test, digits = 3)
```

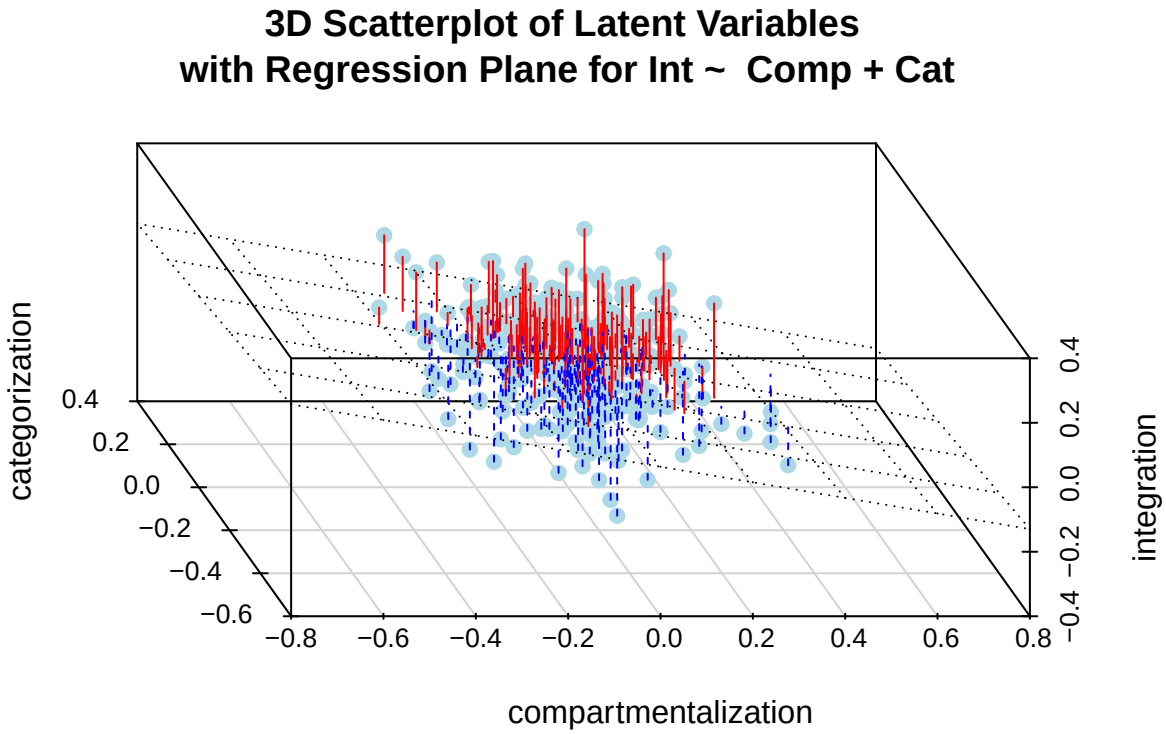
```
##              Estimate Std. Error  p value
## (Intercept) -3.38e-18    0.00569 1.000000
## cat          1.46e-02    0.01500 0.331000
## int         -6.70e-02    0.01930 0.000512
```

```
signif(pooled.add3$hypothesis.test, digits = 3)
```

```
##              Estimate Std. Error  p value
## (Intercept)  4.44e-18    0.00598 1.000000
## comp        -6.05e-02    0.01670 0.000277
## cat         -1.13e-02    0.01670 0.498000
```

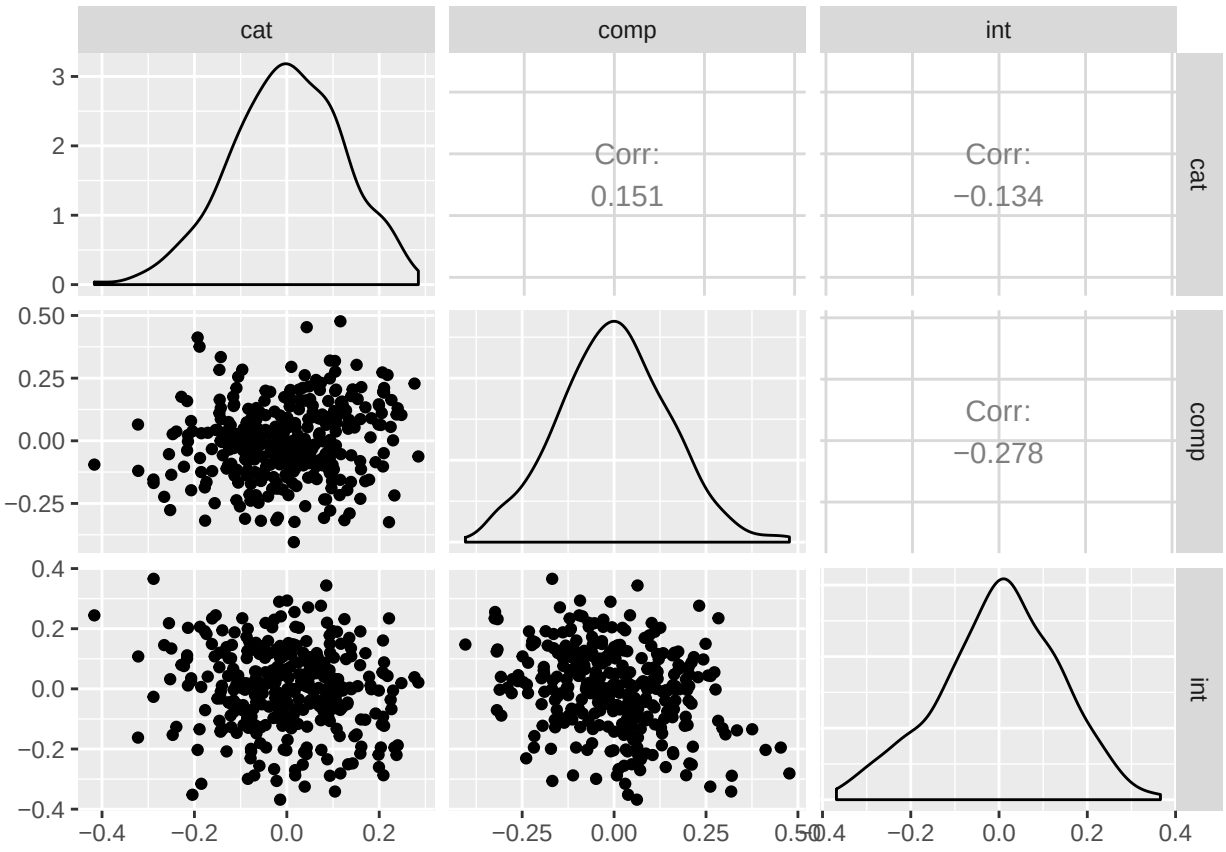
```
library(scatterplot3d)
add <- function(x) Reduce("+", x)
averaged <- add(latent.datasets)/M
fit <- lm(int~comp + cat, data = averaged)
scplot <- scatterplot3d(averaged$comp, averaged$cat, averaged$int,
              main="3D Scatterplot of Latent Variables\n with Regression Plane for Int ~  Comp + Cat", a
              xlab = "compartmentalization", ylab = "categorization", zlab = "integration",
              col.grid = "lightgrey", pch = 19, color = "lightblue")

scplot$plane3d(fit, lty = "dotted")
orig <- scplot$xyz.convert(averaged$comp, averaged$cat, averaged$int)
plane <- scplot$xyz.convert(averaged$comp, averaged$cat, fitted(fit))
i.negpos <- 1 + (resid(fit) > 0)
segments(orig$x, orig$y, plane$x, plane$y,
         col = c("blue", "red")[i.negpos], lty = (2:1)[i.negpos])
```




```
# ggplot2 pairs plot
library(ggplot2)
library(GGally)

ggpairs(averaged)
```


cat
comp
int
Corr:
0.151
Corr:
−0.134
Corr:
−0.278
cat
comp
int
3
2
1
0
0.50
0.25
0.00
−0.25
0.4
0.2
0.0
−0.2
−0.4
−0.4
−0.2
0.0
0.2
−0.25
0.00
0.25
0.50
−0.4
−0.2
0.0
0.2
0.4