432 Class 3 Slides

github.com/THOMASELOVE/2019-432

2019-01-29

Today

- More with SMART BRFSS 2017
- Analysis of Variance models with and without interaction
- Analysis of Covariance models

Recapping from Last Time

- We pulled in data from the SMART BRFSS (2017) (see the Data and Code - smart_2017 folder for details) from SAS XPT files.
- We cleaned up that data, which took a while, and saved it as an R data set.
- We pulled it back into R with `readRDS`, and selected 20 variables of interest for the six MMSAs which include Ohio. That gave us 6,277 subjects.
- We explored the data a bit, and used simple imputation to deal with NAs.
- That file (with imputations) is called `smart_a_imp` now.
- One new thing I did since last time was to save `smart_a_imp` as an R data set, and put it on our web site under Class 03 and the Data and Code folder.

Getting Where I Got Last Time

```
library(skimr); library(broom); library(janitor)
library(simputation); library(tidyverse)

smart_oh_2017 <- readRDS("data/smart_2017_oh.rds")

smart_a_raw <- smart_oh_2017 %>%
    select(subject, genhealth, physhealth, menthealth,
           bmi, bmigroup, weight_kg, height_m, exerany,
           numdocs2, flushot, smoke_100, educgroup,
           diagnoses, seatbelt_always, hx_diabetes,
           female, internet30, agegroup, mmsaname)

set.seed(20190124)
```

Getting Where I Got Last Time

```
smart_a_imp <- smart_a_raw %>%
    impute_pmm(smoke_100 ~ mmsaname) %>%
    impute_pmm(exerany ~ mmsaname) %>%
    impute_pmm(flushot ~ mmsaname) %>%
    impute_pmm(internet30 ~ mmsaname) %>%
    impute_cart(numdocs2 ~ mmsaname + flushot) %>%
    impute_cart(genhealth ~ mmsaname + smoke_100) %>%
    impute_cart(educgroup ~ mmsaname) %>%
    impute_cart(agegroup ~ mmsaname) %>%
    impute_cart(seatbelt_always ~ mmsaname) %>%
    impute_pmm(physhealth ~ mmsaname) %>%
    impute_pmm(menthealth ~ mmsaname) %>%
    impute_rlm(diagnoses ~ numdocs2) %>%
    impute_rlm(weight_kg ~ physhealth + exerany) %>%
    impute_rlm(height_m ~ physhealth + female) %>%
    impute_pmm(hx_diabetes ~ weight_kg + exerany)
```

Recalculating BMI and BMI group after imputation

```
smart_a_imp <- smart_a_imp %>%
    mutate(bmi = weight_kg / (height_m^2)) %>%
    mutate(bmigroup = factor(
        Hmisc::cut2(bmi, cuts = c(18.5, 25.0, 30.0))))
```

The New Step (if you want to skip the rest)

```
saveRDS(smart_a_imp, "data/smart_a_imp.rds")
```

Now, we could have started with . . .

```
smart_a_imp <- readRDS("data/smart_a_imp.rds")
```

and ignored everything except for the package loading.

Onward: Predicting bmi

We'll investigate the prediction of bmi using smart_a_imp.

- The outcome of interest is bmi, which is quantitative.
- Inputs/predictors in the models we build will include:
    - seatbelt_always = 1 if subject always wears seatbelt, else 0
    - hx_diabetes = 1 if the subject has a diabetes diagnosis, else 0
    - exerany = 1 if the subject exercises, and 0 otherwise
    - genhealth = five-category self-reported overall health
    - menthealth = days (in last 30) where mental health impeded activity
    - diagnoses = diagnoses (out of 10) that apply to the subject

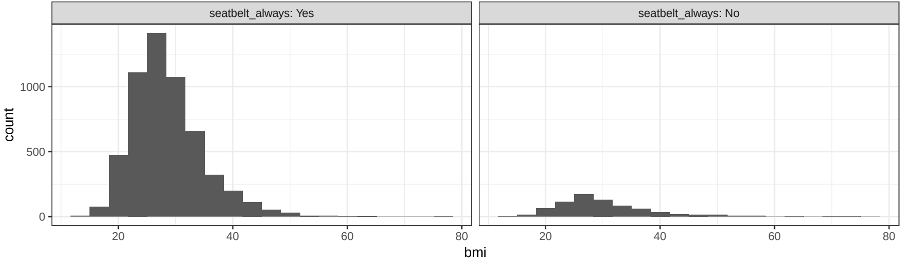
Predicting bmi using seatbelt_always

```
ggplot(smart_a_imp, aes(x = seatbelt_always, y = bmi)) +
    geom_point()
```

Not so helpful.

Faceted Histograms?

```
ggplot(smart_a_imp, aes(x = bmi)) +
    geom_histogram(bins = 20) + theme_bw() +
    facet_wrap(~ seatbelt_always, labeller = "label_both")
```



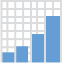
R Studio Cheat Sheets to the rescue?

- https://www.rstudio.com/resources/cheatsheets/ or
- just google, or
- Help ... Cheatsheets ... Data Visualization with ggplot2

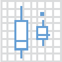
downloads a PDF.

From R Studio Cheat Sheet for ggplot2

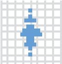
discrete x , continuous y
f <- ggplot(mpg, aes(class, hwy))



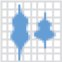
f + geom_col(), x, y, alpha, color, fill, group, linetype, size



f + geom_boxplot(), x, y, lower, middle, upper, ymax, ymin, alpha, color, fill, group, linetype, shape, size, weight



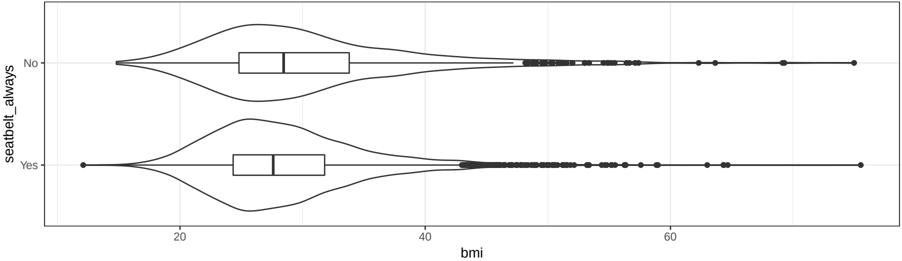
f + geom_dotplot(binaxis = "y", stackdir = "center"), x, y, alpha, color, fill, group



f + geom_violin(scale = "area"), x, y, alpha, color, fill, group, linetype, size, weight

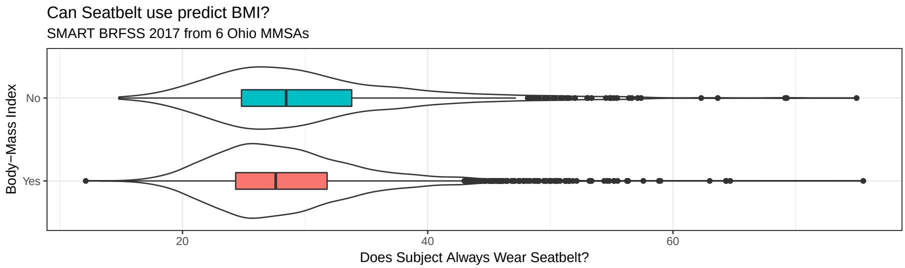
Predicting `bmi` using `seatbelt_always`

```
ggplot(smart_a_imp, aes(x = seatbelt_always, y = bmi)) +
    geom_violin() +
    geom_boxplot(width = 0.2) +
    coord_flip() + theme_bw()
```



Cleaning Up

```
ggplot(smart_a_imp, aes(x = seatbelt_always, y = bmi)) +
    geom_violin() +
    geom_boxplot(aes(fill = seatbelt_always), width = 0.2) +
    coord_flip() + theme_bw() + guides(fill = FALSE) +
    labs(x = "Body-Mass Index",
         y = "Does Subject Always Wear Seatbelt?",
         title = "Can Seatbelt use predict BMI?",
         subtitle = "SMART BRFSS 2017 from 6 Ohio MMSAs")
```



Numerical Summary of BMI by Seatbelt Status

```
mosaic::favstats(bmi ~ seatbelt_always, data = smart_a_imp)
```

```
  seatbelt_always      min       Q1   median       Q3
1             Yes 12.11097 24.33720 27.60355 31.79628
2              No 14.81143 24.81081 28.45451 33.80255
       max     mean       sd    n missing
1 75.52133 28.58543 6.227591 5538       0
2 74.97521 30.22454 8.316329  739       0
```

- How would you want to do this comparison?
- What would be a rational way to predict `bmi` with `seatbelt_always` alone, based on this summary?

Building a t test

```
t.test(bmi ~ seatbelt_always,
       data = smart_a_imp, var.equal = TRUE)
```

```

    Two Sample t-test

data:  bmi by seatbelt_always
t = -6.431, df = 6275, p-value = 1.361e-10
alternative hypothesis: true difference in means is not equal
95 percent confidence interval:
 -2.138762 -1.139464
sample estimates:
mean in group Yes  mean in group No
         28.58543          30.22454
```

Building a t-test Model: model1

```
model1 <- lm(bmi ~ seatbelt_always, data = smart_a_imp)

model1
```

```
Call:
lm(formula = bmi ~ seatbelt_always, data = smart_a_imp)

Coefficients:
      (Intercept)  seatbelt_alwaysNo
           28.585              1.639
```

```
confint(model1, level = 0.90)
```

```
                        5 %      95 %
(Intercept)       28.441559 28.729299
seatbelt_alwaysNo  1.219813  2.058412
```

Summarizing `model1` with `tidy`

```
tidy(model1, conf.int = TRUE, conf.level = 0.90) %>%
    print.data.frame(digits = 2)
```

```
              term estimate std.error statistic p.value
1      (Intercept)     28.6     0.087     326.9 0.0e+00
2 seatbelt_alwaysNo     1.6     0.255       6.4 1.4e-10
  conf.low conf.high
1     28.4      28.7
2      1.2       2.1
```

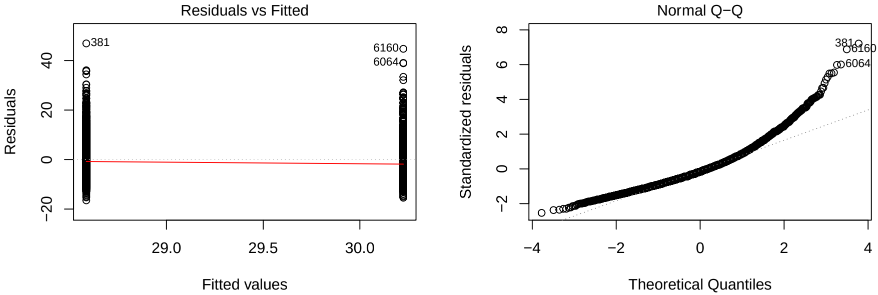
Summarizing `model1` with `glance`

```
glance(model1) %>%
    print.data.frame(digits = 2)
```

```
  r.squared adj.r.squared sigma statistic p.value df logLik
1    0.0065        0.0064   6.5        41 1.4e-10  2 -20663
    AIC   BIC deviance df.residual
1 41332 41352   265782        6275
```

Regression Diagnostics for `model1`

```
par(mfrow=c(1,2))
plot(model1, which = c(1,2))
```



What have we learned from `model1`?

Based on our sample of 6277 subjects, the model suggests that:

- the ordinary least squares prediction of BMI for people who always wear a seatbelt is 28.59 kg/m^2, and
- the OLS prediction of BMI for people who don't always wear a seatbelt is $28.585429 + 1.639113 = 30.22$ kg/m^2
- the mean difference between those who don't wear a seatbelt and those who do is 1.64 kg/m^2
- a 90% confidence (uncertainty) interval for that mean difference ranges from (1.22, 2.06) kg/m^2

What else have we learned from `model1`?

- `model1` accounts for 0.65% of the variation in `bmi`, so that knowing the subject's seatbelt status does very little to reduce the size of the prediction errors, as compared to an "intercept-only" model that just predicts the overall mean `bmi` for all subjects
- despite this, the model is highly "statistically significant" with a *p* value for seatbelt status that is on the order of 10^{-10}.
- the model makes some very large errors, since the standard deviation of those prediction errors (labeled as `sigma`, or σ) is 6.5, which is enormous on the scale of `bmi`...

```
mosaic::favstats(~ bmi, data = smart_a_imp)
```

```
      min      Q1   median       Q3      max    mean
 12.11097 24.3372 27.64314 31.89453 75.52133 28.7784
       sd    n missing
 6.529016 6277       0
```

OK. So `model1` isn't good enough.

- What about a two-factor model?

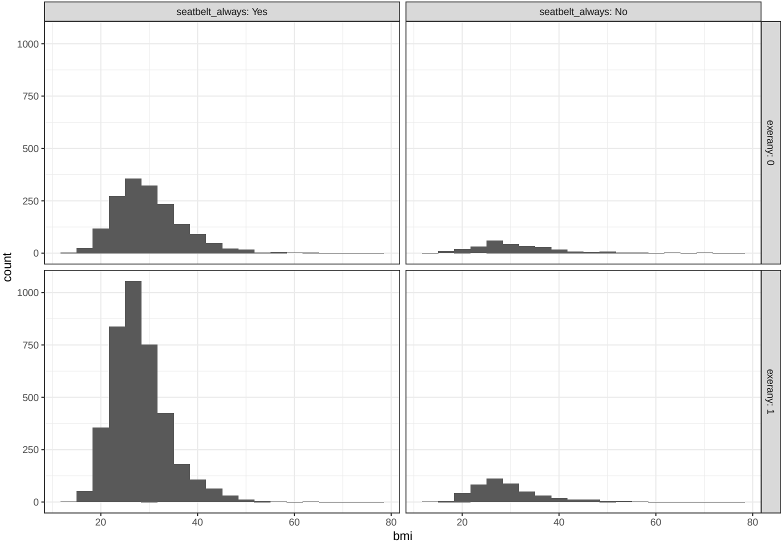
Suppose we decide to predict `bmi` using both `seatbelt_always` and also `exerany`.

- Can we draw a picture?

```
ggplot(smart_a_imp, aes(x = bmi)) +
    geom_histogram(bins = 20) + theme_bw() +
    facet_grid(exerany ~ seatbelt_always,
               labeller = "label_both")
```

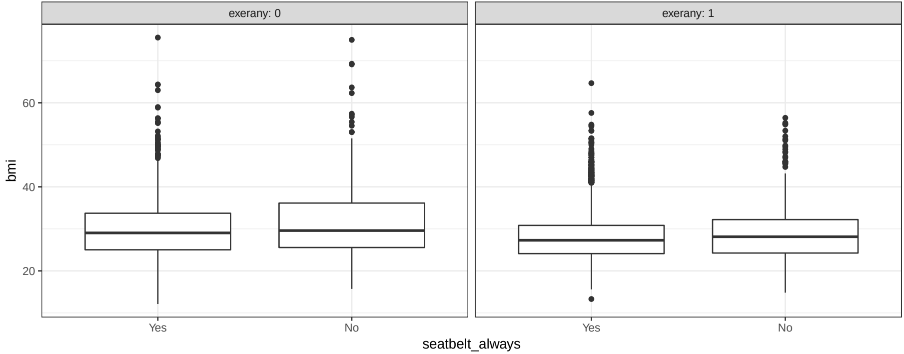
What will this do?

The resulting plot of faceted histograms



Would boxplots be better?

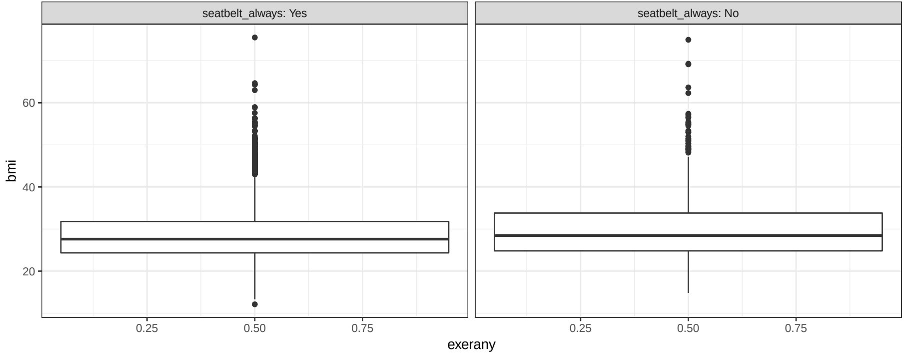
```
ggplot(smart_a_imp, aes(x = seatbelt_always, y = bmi)) +
    geom_boxplot() + theme_bw() +
    facet_wrap(~ exerany, labeller = "label_both")
```



Why doesn't this work?

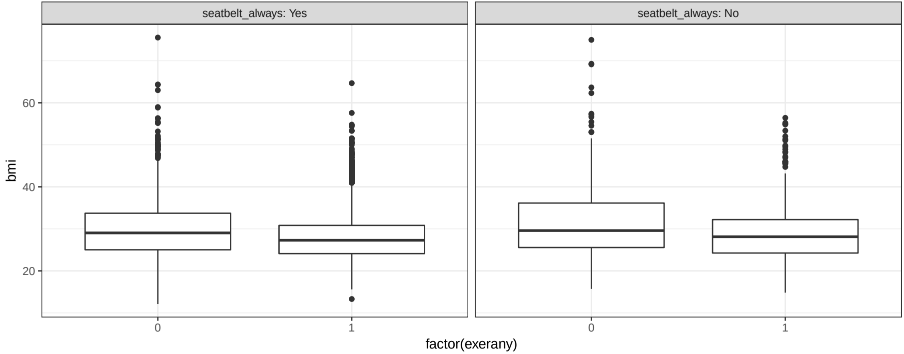
```
ggplot(smart_a_imp, aes(x = exerany, y = bmi)) +
    geom_boxplot() + theme_bw() +
    facet_wrap(~ seatbelt_always, labeller = "label_both")
```

```
Warning: Continuous x aesthetic -- did you forget
aes(group=...)?
```



Make exerany a factor!

```
ggplot(smart_a_imp, aes(x = factor(exerany), y = bmi)) +
    geom_boxplot() + theme_bw() +
    facet_wrap(~ seatbelt_always, labeller = "label_both")
```



Maybe we should just concentrate on the means?

```
summaries1 <- smart_a_imp %>%
    group_by(seatbelt_always, exerany) %>%
    summarize(n = n(), mean = mean(bmi), stdev = sd(bmi))
summaries1
```

```
# A tibble: 4 x 5
# Groups:   seatbelt_always [?]
  seatbelt_always exerany     n  mean stdev
  <fct>             <dbl> <int> <dbl> <dbl>
1 Yes                   0  1668  30.0  7.09
2 Yes                   1  3870  28.0  5.72
3 No                    0   278  31.7  9.67
4 No                    1   461  29.3  7.24
```

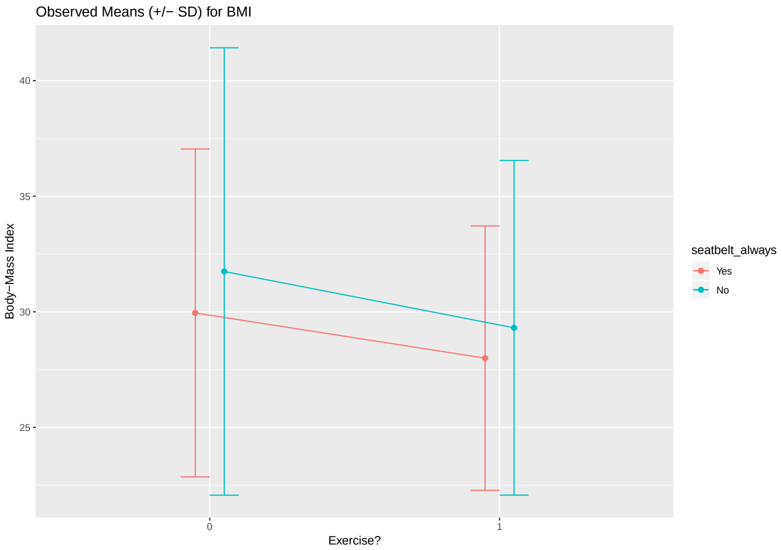
We could use `favstats` from `mosaic` for more detail if needed.

Plot the Means

```
pd <- position_dodge(0.2)

ggplot(summaries1, aes(x = factor(exerany), y = mean,
                       col = seatbelt_always)) +
  geom_errorbar(aes(ymin = mean - stdev,
                    ymax = mean + stdev),
                width = 0.2, position = pd) +
  geom_point(size = 2, position = pd) +
  geom_line(aes(group = seatbelt_always), position = pd) +
  labs(y = "Body-Mass Index",
       x = "Exercise?",
       title = "Observed Means (+/- SD) for BMI")
```

Means Plot (result)



Running the Two-Way ANOVA model

We can run a model to predict a quantitative outcome using two categorical factors, either with or without an interaction between the two factors.

In our case, we can run either:

```
model2_noint <- lm(bmi ~ seatbelt_always + exerany,
              data = smart_a_imp)
```

or

```
model2_int <- lm(bmi ~ seatbelt_always * exerany,
                data = smart_a_imp)
```

ANOVA "No-Interaction" Model (Main Effects Model)

```
anova(model2_noint)
```

```
Analysis of Variance Table

Response: bmi
                  Df Sum Sq Mean Sq F value    Pr(>F)
seatbelt_always    1   1752  1751.7  42.216 8.802e-11 ***
exerany            1   5446  5446.2 131.251 < 2.2e-16 ***
Residuals       6274 260336    41.5
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpreting the Main Effects Model

```
tidy(model2_noint, conf.int = TRUE, conf.level = 0.90) %>%
    print.data.frame(digits = 2)
```

```
              term estimate std.error statistic p.value
1      (Intercept)     30.0      0.15     199.4 0.0e+00
2 seatbelt_alwaysNo     1.5      0.25       5.9 4.1e-09
3          exerany     -2.0      0.18     -11.5 4.3e-30
  conf.low conf.high
1     29.7      30.2
2      1.1       1.9
3     -2.3      -1.7
```

ANOVA Model with Interaction

```
anova(model2_int)
```

```
Analysis of Variance Table

Response: bmi
                          Df Sum Sq Mean Sq F value
seatbelt_always            1   1752  1751.7  42.215
exerany                    1   5446  5446.2 131.248
seatbelt_always:exerany    1     35    35.0   0.843
Residuals               6273 260301    41.5
                           Pr(>F)
seatbelt_always         8.807e-11 ***
exerany                 < 2.2e-16 ***
seatbelt_always:exerany    0.3586
Residuals
---
Signif. codes:
```

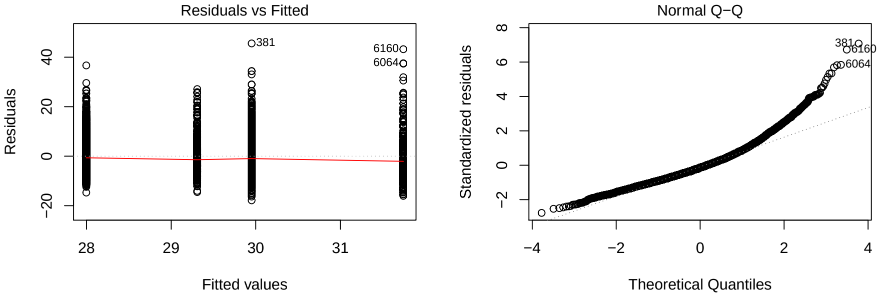
Interpreting the Model with Interaction

```
tidy(model2_int, conf.int = TRUE, conf.level = 0.90) %>%
    print.data.frame(digits = 2)
```

```
                         term estimate std.error statistic
1                 (Intercept)    29.95      0.16    189.89
2            seatbelt_alwaysNo    1.79      0.42      4.30
3                      exerany   -1.95      0.19    -10.36
4 seatbelt_alwaysNo:exerany   -0.48      0.52     -0.92
  p.value conf.low conf.high
1 0.0e+00     29.7     30.21
2 1.8e-05      1.1      2.48
3 6.1e-25     -2.3     -1.64
4 3.6e-01     -1.3      0.38
```

Regression Diagnostics for `model2_int`

```
par(mfrow=c(1,2))
plot(model2_int, which = c(1,2))
```



Assessing these Two-Factor ANOVA models

Check the interaction first!

- Does the means plot (interaction plot) show a meaningful interaction between the factors?
- Does the interaction term account for a substantial amount of the variation in the outcome?
- Does the interaction term significantly improve the model?

If all three of these are YES, or all three are NO, the choice is obvious.

- If all three are YES, we certainly will use the model including the interaction.
- If all three are NO, then a main-effects model (without interaction) is likely to work out well.

What do we do otherwise? It depends.

In our case . . .

- The means plot showed essentially parallel lines. There's no evidence there of a strong or meaningful interaction.
- The interaction term sum of squares is 35, out of a total sum of squares of 267,534. That's an incredibly small fraction, so there's no sign of substantial interaction.
- The interaction term doesn't significantly improve the model - its p value is 0.3586

So, would the main-effect model in this case be a reasonable approach?

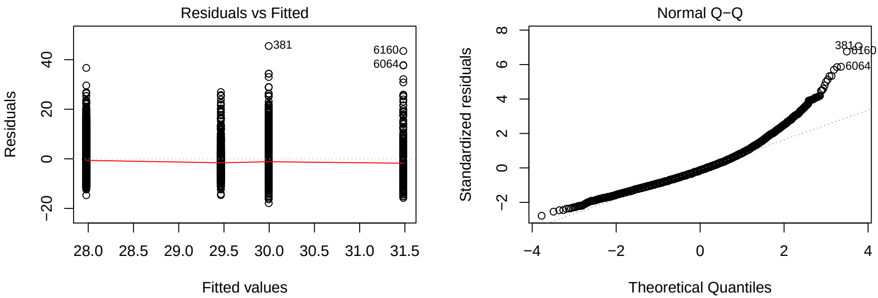
Main Effects Model, again

```
tidy(model2_noint, conf.int = TRUE, conf.level = 0.90) %>%
    print.data.frame(digits = 2)
```

```
             term estimate std.error statistic p.value
1     (Intercept)     30.0      0.15     199.4 0.0e+00
2 seatbelt_alwaysNo    1.5      0.25       5.9 4.1e-09
3         exerany     -2.0      0.18     -11.5 4.3e-30
  conf.low conf.high
1     29.7      30.2
2      1.1       1.9
3     -2.3      -1.7
```

Regression Diagnostics for `model2_noint`

```
par(mfrow=c(1,2))
plot(model2_noint, which = c(1,2))
```



Two-Factor Analysis of Variance

1. Check interaction first.
    - Is there evidence of substantial interaction in a plot?
    - Is the interaction effect a large part of the model?
    - Is the interaction term statistically significant?
2. If interaction is deemed to be meaningful, then "it depends" is the right conclusion, and we cannot easily separate the effect of one factor from another.
3. If interaction is not deemed to be meaningful, we might consider fitting the model without the interaction (the "main effects" model) and separately interpreting the impact of each of the factors.

What if we add menthealth to the model?

```
model3_noint <- lm(bmi ~ menthealth +
                     seatbelt_always + exerany,
                   data = smart_a_imp)
anova(model3_noint)
```

```
Analysis of Variance Table

Response: bmi
                  Df Sum Sq Mean Sq F value    Pr(>F)
menthealth         1   2007  2007.3  48.583 3.491e-12 ***
seatbelt_always    1   1587  1587.3  38.416 6.081e-10 ***
exerany            1   4750  4750.2 114.966 < 2.2e-16 ***
Residuals       6273 259189    41.3
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Comparing Main Effect Models with `anova`

```r
anova(model3_noint, model2_noint)
```

```
Analysis of Variance Table

Model 1: bmi ~ menthealth + seatbelt_always + exerany
Model 2: bmi ~ seatbelt_always + exerany
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1   6273 259189
2   6274 260336 -1   -1146.9 27.757 1.421e-07 ***
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Other Comparison Strategies

```
glance(model3_noint) %>% print.data.frame(digits = 2)
```

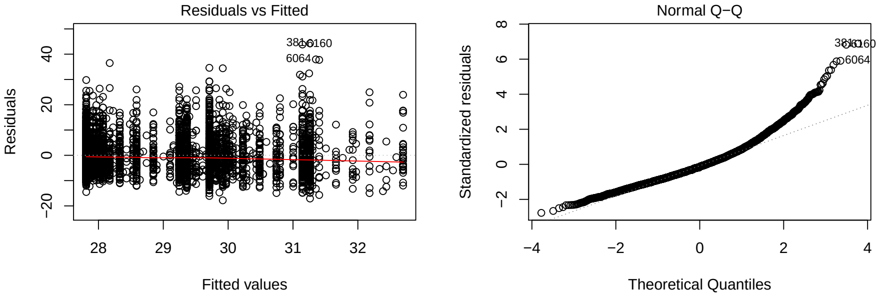
```
  r.squared adj.r.squared sigma statistic p.value df logLik
1     0.031         0.031   6.4        67 7.7e-43  4 -20584
    AIC   BIC deviance df.residual
1 41178 41212   259189        6273
```

```
glance(model2_noint) %>% print.data.frame(digits = 2)
```

```
  r.squared adj.r.squared sigma statistic p.value df logLik
1     0.027         0.027   6.4        87   7e-38  3 -20598
    AIC   BIC deviance df.residual
1 41204 41231   260336        6274
```

Regression Diagnostics for `model3_noint`

```
par(mfrow=c(1,2))
plot(model3_noint, which = c(1,2))
```



What if we consider the interaction again?

```
model3_int <- lm(bmi ~ menthealth +
                      seatbelt_always * exerany,
                 data = smart_a_imp)
anova(model3_int)
```

```
Analysis of Variance Table

Response: bmi
                          Df Sum Sq Mean Sq  F value
menthealth                 1   2007  2007.3  48.5814
seatbelt_always            1   1587  1587.3  38.4145
exerany                    1   4750  4750.2 114.9631
seatbelt_always:exerany    1     35    34.6   0.8376
Residuals               6272 259154    41.3
                           Pr(>F)
menthealth              3.493e-12 ***
seatbelt_always         6.084e-10 ***
```

Comparing Interaction Models with anova

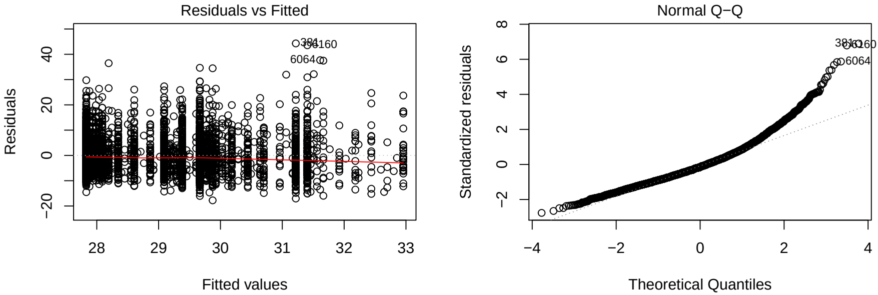
```
anova(model3_int, model2_int, model2_noint)
```

```
Analysis of Variance Table

Model 1: bmi ~ menthealth + seatbelt_always * exerany
Model 2: bmi ~ seatbelt_always * exerany
Model 3: bmi ~ seatbelt_always + exerany
  Res.Df    RSS Df Sum of Sq       F    Pr(>F)
1   6272 259154
2   6273 260301 -1  -1146.51 27.7477 1.428e-07 ***
3   6274 260336 -1    -34.98  0.8466    0.3576
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Regression Diagnostics for `model3_int`

```
par(mfrow=c(1,2))
plot(model3_int, which = c(1,2))
```



Coming up . . .

- Using factors with more than two levels as predictors in ANOVA/ANCOVA
- Linear regression using both quantitative and categorical predictors
- Improving on stepwise regression for model selection with "best subsets"
- Improving on cross-validation of linear regression models

Upcoming Deliverables

- Minute Paper after Class 3 is due tomorrow (Wednesday) at 2 PM.
- Homework 1 is due Friday at 2 PM, via Canvas.