432 Class 2 Slides

github.com/THOMASELOVE/2019-432

2019-01-24

BRFSS and SMART

The Centers for Disease Control analyzes Behavioral Risk Factor Surveillance System (BRFSS) survey data for specific metropolitan and micropolitan statistical areas (MMSAs) in a program called the Selected Metropolitan/Micropolitan Area Risk Trends of BRFSS (SMART BRFSS.)

In this work, we will focus on data from the 2017 SMART, and in particular on data from the Cleveland-Elyria, OH, Metropolitan Statistical Area.

Note that the Course Notes concentrate instead on an earlier data set from the 2016 SMART BRFSS.

Setup

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

smart_cle_2017 <- readRDS("data/smart_2017_cle.rds")
smart_oh_2017 <- readRDS("data/smart_2017_oh.rds")
```

How Did I Build the Data?

Visit https://github.com/THOMASELOVE/2019-432/tree/master/data-and-code/smart_2017
on our Data and Code pages (smart_2017 folder) for all of the details.

Today's Variables

```
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)

dim(smart_a_raw)
```

```
[1] 6277   20
```

Variables by Type

```
head(smart_a_raw)
```

```
# A tibble: 6 x 20
  subject genhealth physhealth menthealth   bmi bmigroup
  <chr>   <fct>           <dbl>      <dbl> <dbl> <fct>
1 171402~ 2_VeryGo~           0          1  26.6 [25.0,3~
2 171402~ 2_VeryGo~           2         16  29.6 [25.0,3~
3 171402~ 2_VeryGo~           0          0  29.4 [25.0,3~
4 171402~ 3_Good              2          0  27.5 [25.0,3~
5 171402~ 2_VeryGo~           0          0  20.0 [18.5,2~
6 171402~ 4_Fair              0          0  20.4 [18.5,2~
# ... with 14 more variables: weight_kg <dbl>,
#   height_m <dbl>, exerany <dbl>, numdocs2 <fct>,
#   flushot <dbl>, smoke_100 <dbl>, educgroup <fct>,
#   diagnoses <dbl>, seatbelt_always <fct>,
#   hx_diabetes <dbl>, female <dbl>, internet30 <dbl>,
#   agegroup <fct>, mmsaname <chr>
```

Structure of the data frame with `str`

```
str(smart_a_raw)
```

```
Classes 'tbl_df', 'tbl' and 'data.frame':   6277 obs. of  20 v
 $ subject        : chr  "171402017000002" "171402017000003" "
 $ genhealth      : Factor w/ 5 levels "1_Excellent",..: 2 2 2
 $ physhealth     : num  0 2 0 2 0 0 30 2 0 0 ...
 $ menthealth     : num  1 16 0 0 0 0 15 15 2 0 ...
 $ bmi            : num  26.6 29.6 29.4 27.5 20 ...
 $ bmigroup       : Factor w/ 4 levels "[13.3,18.5)",..: 3 3 3
 $ weight_kg      : num  68 90.7 95.2 79.4 63.5 ...
 $ height_m       : num  1.6 1.75 1.8 1.7 1.78 1.52 1.75 1.73
 $ exerany        : num  1 1 1 0 1 1 0 1 1 0 ...
 $ numdocs2       : Factor w/ 3 levels "1_One","2_MoreThanOne"
 $ flushot        : num  1 NA 0 1 0 0 0 1 1 1 ...
 $ smoke_100      : num  0 0 0 1 1 0 1 1 1 1 ...
 $ educgroup      : Factor w/ 6 levels "Kindergarten",..: 4 4
 $ diagnoses      : num  0 1 0 1 0 2 2 0 3 3 ...
```

```
summary(smart_a_raw %>% select(bmi, diagnoses, exerany,
                               female, genhealth, seatbelt_always))
```

```
      bmi           diagnoses         exerany
 Min.   :13.30   Min.   :0.000   Min.   :0.0000
 1st Qu.:24.25   1st Qu.:0.000   1st Qu.:0.0000
 Median :27.48   Median :1.000   Median :1.0000
 Mean   :28.74   Mean   :1.279   Mean   :0.6893
 3rd Qu.:31.92   3rd Qu.:2.000   3rd Qu.:1.0000
 Max.   :75.52   Max.   :8.000   Max.   :1.0000
 NA's   :323     NA's   :238     NA's   :13
     female                genhealth    seatbelt_always
 Min.   :0.0000   1_Excellent: 872   Yes :5512
 1st Qu.:0.0000   2_VeryGood :2046   No  : 739
 Median :1.0000   3_Good     :1987   NA's:  26
 Mean   :0.5831   4_Fair     : 991
 3rd Qu.:1.0000   5_Poor     : 370
 Max.   :1.0000   NA's       :  11
```

More Detailed Summaries

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

```
  min    Q1 median    Q3   max     mean       sd    n
 13.3 24.25  27.48 31.92 75.52 28.74082 6.642863 5954
 missing
     323
```

```
mosaic::favstats(bmi ~ exerany, data = smart_a_raw)
```

```
  exerany   min    Q1 median    Q3   max     mean       sd
1       0 13.64 24.99  28.93 34.125 75.52 30.19068 7.681224
2       1 13.30 23.95  27.09 31.070 64.67 28.09344 6.011350
     n missing
1 1831     115
2 4111     207
```

Counting is Wonderful!

```
smart_a_raw %>% count(mmsaname)
```

```
# A tibble: 6 x 2
  mmsaname                                                        n
  <chr>                                                       <int>
1 Cincinnati, OH-KY-IN, Metropolitan Statistical Area          1460
2 Cleveland-Elyria, OH, Metropolitan Statistical Area           966
3 Columbus, OH, Metropolitan Statistical Area                  1681
4 Dayton, OH, Metropolitan Statistical Area                     497
5 Huntington-Ashland, WV-KY-OH, Metropolitan Statisti~         1020
6 Toledo, OH, Metropolitan Statistical Area                     653
```

Counting is Marvelous!

```
smart_a_raw %>% count(educgroup)
```

```
# A tibble: 7 x 2
  educgroup        n
  <fct>        <int>
1 Kindergarten     3
2 Elementary      95
3 Some_HS        265
4 HS_Grad       1873
5 Some_College  1800
6 College_Grad  2226
7 <NA>            15
```

Tabyls (from janitor) are great, too...

```
smart_a_raw %>% tabyl(genhealth)
```

```
   genhealth    n    percent valid_percent
 1_Excellent  872 0.13891987    0.13916374
  2_VeryGood 2046 0.32595189    0.32652410
      3_Good 1987 0.31655249    0.31710820
      4_Fair  991 0.15787797    0.15815512
      5_Poor  370 0.05894536    0.05904883
        <NA>   11 0.00175243            NA
```

Counting: The Best Thing to Do

```
smart_a_raw %>% count(female, seatbelt_always)
```

```
# A tibble: 6 x 3
  female seatbelt_always     n
   <dbl> <fct>           <int>
1      0 Yes              2205
2      0 No                399
3      0 <NA>               13
4      1 Yes              3307
5      1 No                340
6      1 <NA>               13
```

`tabyl` for Quick Cross-Tabs

```
smart_a_raw %>% tabyl(genhealth, numdocs2)
```

genhealth	1_One	2_MoreThanOne	3_Zero	NA_
1_Excellent	668	59	142	3
2_VeryGood	1664	120	260	2
3_Good	1599	160	224	4
4_Fair	786	124	79	2
5_Poor	274	69	25	2
<NA>	7	2	2	0

Or for fancier cross-tabulations. . .

```
smart_a_raw %>%
    tabyl(smoke_100, seatbelt_always) %>%
    adorn_totals() %>%
    adorn_percentages("row") %>%
    adorn_pct_formatting(digits = 1) %>%
    adorn_ns(position = "front") %>%
    adorn_title()
```

```
           seatbelt_always
 smoke_100                Yes                No               NA_
         0    2984 (89.3%) 350 (10.5%)   9 (0.3%)
         1    2505 (86.1%) 388 (13.3%)  17 (0.6%)
      <NA>      23 (95.8%)   1  (4.2%)   0 (0.0%)
     Total    5512 (87.8%) 739 (11.8%)  26 (0.4%)
```

Using describe from Hmisc

```
Hmisc::describe(smart_a_raw %>% select(bmi))
```

```
smart_a_raw %>% select(bmi)

 1  Variables      6277  Observations
--------------------------------------------------------------------------------
bmi
       n  missing distinct     Info     Mean      Gmd
    5954      323     1283        1    28.74    7.094
     .05      .10      .25      .50      .75      .90
   20.32    21.66    24.25    27.48    31.92    37.41
     .95
   41.47

lowest : 13.30 13.64 14.18 14.72 14.81
highest: 64.67 69.14 69.29 74.98 75.52
--------------------------------------------------------------------------------
```

```
Hmisc::describe(smart_a_raw %>% select(genhealth))
```

```
smart_a_raw %>% select(genhealth)

 1  Variables      6277  Observations
--------------------------------------------------------
genhealth
       n  missing distinct
    6266       11        5

Value      1_Excellent  2_VeryGood      3_Good      4_Fair
Frequency          872        2046        1987         991
Proportion       0.139       0.327       0.317       0.158

Value           5_Poor
Frequency          370
Proportion       0.059
--------------------------------------------------------
```

```
Hmisc::describe(smart_a_raw %>% select(female))
```

```
smart_a_raw %>% select(female)

 1  Variables      6277  Observations
---------------------------------------------------------------
female
       n  missing distinct     Info      Sum     Mean
    6277        0        2    0.729     3660   0.5831
     Gmd
  0.4863

---------------------------------------------------------------
```

Using `skim` to summarize the `smart_a_raw` data

```
skim(smart_a_raw)
```

```
Skim summary statistics
 n obs: 6277
 n variables: 20

-- Variable type:character -----------------------------------------------------
 variable missing complete    n min max empty n_unique
 mmsaname       0     6277 6277  41  59     0        6
  subject       0     6277 6277  15  15     0     6277

-- Variable type:factor --------------------------------------------------------
        variable missing complete    n n_unique                                top_counts
        agegroup      43     6234 6277       13    60-: 743, 65-: 731, 55-: 694, 70-: 612
        bmigroup     323     5954 6277        4  [25: 2104, [30: 2046, [18: 1701, NA: 323
       educgroup      15     6262 6277        6 Col: 2226, HS_: 1873, Som: 1800, Som: 265
       genhealth      11     6266 6277        5  2_V: 2046, 3_G: 1987, 4_F: 991, 1_E: 872
        numdocs2      13     6264 6277        3     1_O: 4998, 3_Z: 732, 2_M: 534, NA: 13
 seatbelt_always      26     6251 6277        2                 Yes: 5512, No: 739, NA: 26
```

Using `skim` to summarize the `smart_a_raw` data

```
-- Variable type:numeric -------------------------------------------------------
    variable missing complete    n  mean    sd    p0   p25   p50   p75   p100     hist
         bmi     323     5954 6277 28.74  6.64 13.3  24.25 27.48 31.92  75.52 ▁▇▃▁▁▁▁▁
   diagnoses     238     6039 6277  1.28  1.37  0     0     1     2      8    ▇▂▁▁▁▁▁▁
     exerany      13     6264 6277  0.69  0.46  0     0     1     1      1    ▃▁▁▁▁▁▁▇
      female       0     6277 6277  0.58  0.49  0     0     1     1      1    ▅▁▁▁▁▁▁▇
     flushot      28     6249 6277  0.51  0.5   0     0     1     1      1    ▇▁▁▁▁▁▁▇
    height_m      84     6193 6277  1.69  0.1   1.35  1.63  1.68  1.78   2.06 ▁▃▇▇▃▁▁▁
 hx_diabetes      13     6264 6277  0.16  0.36  0     0     0     0      1    ▇▁▁▁▁▁▁▂
  internet30      14     6263 6277  0.82  0.39  0     1     1     1      1    ▂▁▁▁▁▁▁▇
  menthealth      96     6181 6277  4.04  8.4   0     0     0     3     30    ▇▁▁▁▁▁▁▁
  physhealth     118     6159 6277  5.07  9.48  0     0     0     5     30    ▇▁▁▁▁▁▁▁
   smoke_100      24     6253 6277  0.47  0.5   0     0     0     1      1    ▇▁▁▁▁▁▁▇
   weight_kg     296     5981 6277 82.96 21.72 31.75 68.04 79.38 95.25 208.65 ▁▇▅▂▁▁▁▁
```

Counts of Missing Data

```
smart_a_raw %>% summarise_all(funs(sum(is.na(.))))
```

```
# A tibble: 1 x 20
  subject genhealth physhealth menthealth   bmi bmigroup
    <int>     <int>      <int>      <int> <int>    <int>
1       0        11        118         96   323      323
# ... with 14 more variables: weight_kg <int>,
#   height_m <int>, exerany <int>, numdocs2 <int>,
#   flushot <int>, smoke_100 <int>, educgroup <int>,
#   diagnoses <int>, seatbelt_always <int>,
#   hx_diabetes <int>, female <int>, internet30 <int>,
#   agegroup <int>, mmsaname <int>
```

I need to override the usual tibble printing behavior.

Use print.data.frame?

```
smart_a_raw %>% summarise_all(funs(sum(is.na(.)))) %>%
    print.data.frame
```

```
  subject genhealth physhealth menthealth bmi bmigroup
1       0        11        118          96 323      323
  weight_kg height_m exerany numdocs2 flushot smoke_100
1       296       84      13       13      28         24
  educgroup diagnoses seatbelt_always hx_diabetes female
1        15       238               26          13      0
  internet30 agegroup mmsaname
1         14       43        0
```

- Which variables have the most missing data? The least?
- How many rows have at least one missing element?

What does this code tell us?

```
smart_a_raw %>% dim
```

```
[1] 6277   20
```

```
smart_a_raw %>% filter(!complete.cases(.)) %>% nrow
```

```
[1] 815
```

```
smart_a_raw %>% filter(complete.cases(.))
```

```
# A tibble: 5,462 x 20
   subject genhealth physhealth menthealth   bmi bmigroup
   <chr>   <fct>           <dbl>      <dbl> <dbl> <fct>
 1 171402~ 2_VeryGo~           0          1  26.6 [25.0,3~
 2 171402~ 2_VeryGo~           0          0  29.4 [25.0,3~
 3 171402~ 3_Good              2          0  27.5 [25.0,3~
 4 171402~ 2_VeryGo~           0          0  20.0 [18.5,2~
```

Simple Imputation

- I'll use `impute_pmm` on most numeric variables to predict them using the MSA (and maybe some other things.)
- I'll use `impute_cart` similarly on most of the character/factor variables.
- I'll use robust linear models to impute a few things via `impute_rlm`, and
- I won't impute `bmi` or `bmigroup` directly, but instead recalculate them using imputed `weight_kg` and `height_m` values.

All of these (but the last) are essentially arbitrary decisions here.

```
set.seed(20190124)

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)
```

Sanity Check on exerany imputation

```
smart_a_raw %>% count(exerany) %>% mutate(prop = n / sum(n))
```

```
# A tibble: 3 x 3
  exerany     n    prop
    <dbl> <int>   <dbl>
1       0  1946 0.310
2       1  4318 0.688
3      NA    13 0.00207
```

```
smart_a_imp %>% count(exerany) %>% mutate(prop = n / sum(n))
```

```
# A tibble: 2 x 3
  exerany     n  prop
    <dbl> <int> <dbl>
1       0  1946 0.310
2       1  4331 0.690
```

Sanity Check on genhealth imputation

The original, unimputed data:

```
smart_a_raw %>% count(genhealth) %>%
    mutate(pct = round(100*n / sum(n), 1))
```

```
# A tibble: 6 x 3
  genhealth       n   pct
  <fct>       <int> <dbl>
1 1_Excellent   872  13.9
2 2_VeryGood   2046  32.6
3 3_Good       1987  31.7
4 4_Fair        991  15.8
5 5_Poor        370   5.9
6 <NA>           11   0.2
```

Sanity Check on genhealth imputation

The data after simple imputation:

```
smart_a_imp %>% count(genhealth) %>%
    mutate(pct = round(100*n / sum(n), 1))
```

```
# A tibble: 5 x 3
  genhealth       n   pct
  <fct>       <int> <dbl>
1 1_Excellent   872  13.9
2 2_VeryGood   2052  32.7
3 3_Good       1992  31.7
4 4_Fair        991  15.8
5 5_Poor        370   5.9
```

Calculating BMI and BMI group

```
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))))
```

Sanity Check on BMI imputations/calculations

```
mosaic::favstats(~ bmi, data = smart_a_raw) %>%
    round(digits = 1)
```

```
  min   Q1 median   Q3  max mean  sd    n missing
 13.3 24.2   27.5 31.9 75.5 28.7 6.6 5954     323
```

```
mosaic::favstats(~ bmi, data = smart_a_imp) %>%
    round(digits = 1)
```

```
  min   Q1 median   Q3  max mean  sd    n missing
 12.1 24.3   27.6 31.9 75.5 28.8 6.5 6277       0
```

Did I impute away all missing values?

```
smart_a_imp %>% summarise_all(funs(sum(is.na(.)))) %>%
    print.data.frame
```

```
  subject genhealth physhealth menthealth bmi bmigroup
1       0         0          0          0   0         0
  weight_kg height_m exerany numdocs2 flushot smoke_100
1         0        0       0        0       0         0
  educgroup diagnoses seatbelt_always hx_diabetes female
1         0         0                0           0      0
  internet30 agegroup mmsaname
1          0        0        0
```

```
smart_a_imp %>% filter(!complete.cases(.)) %>% nrow
```

```
[1] 0
```

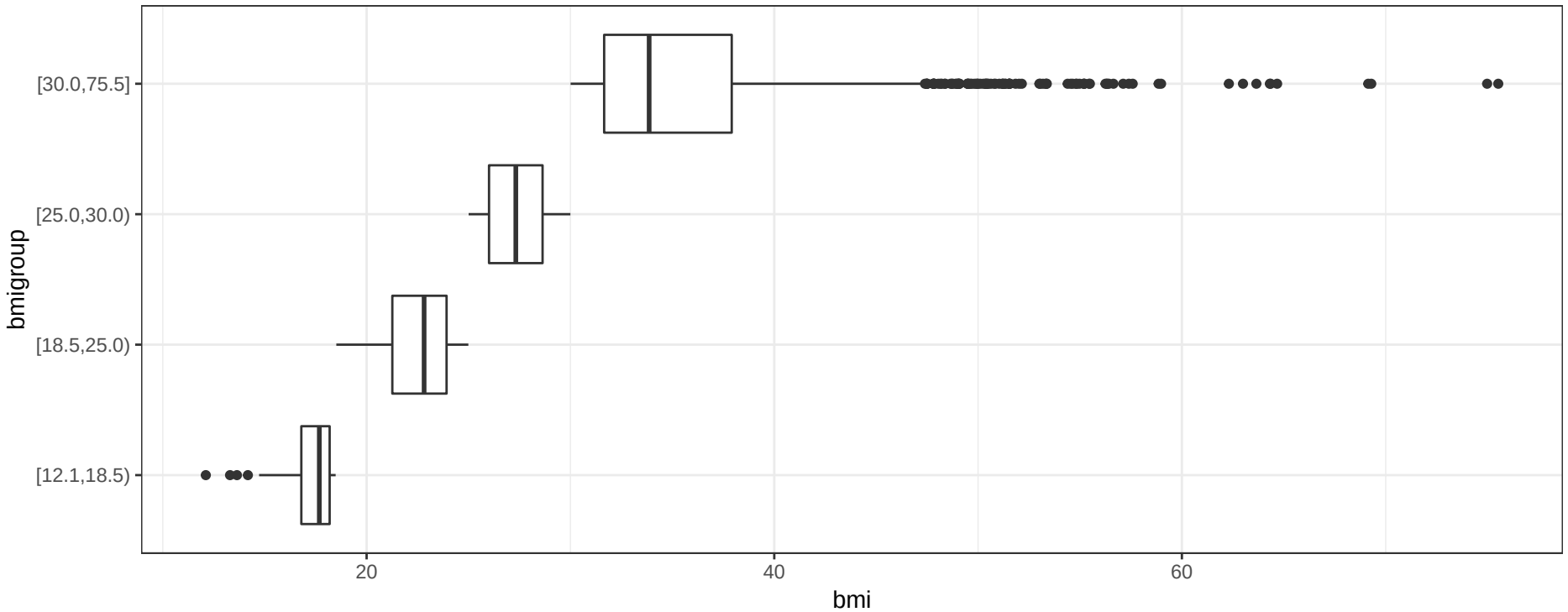
BMI Groups - do they make sense?

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

```
     bmigroup      min       Q1   median       Q3      max
1 [12.1,18.5) 12.11097 16.79763 17.67999 18.18637 18.48073
2 [18.5,25.0) 18.51429 21.26172 22.82099 23.92389 24.99174
3 [25.0,30.0) 25.00585 26.00555 27.32072 28.63275 29.99716
4 [30.0,75.5] 30.00768 31.65473 33.86187 37.91403 75.52133
      mean       sd    n missing
1 17.28723 1.202602  108       0
2 22.57005 1.649085 1723       0
3 27.37454 1.464309 2248       0
4 35.64552 5.609182 2198       0
```

Wouldn't a Picture help?

```
ggplot(smart_a_imp, aes(x = bmigroup, y = bmi)) +
    geom_boxplot() + coord_flip() + theme_bw()
```



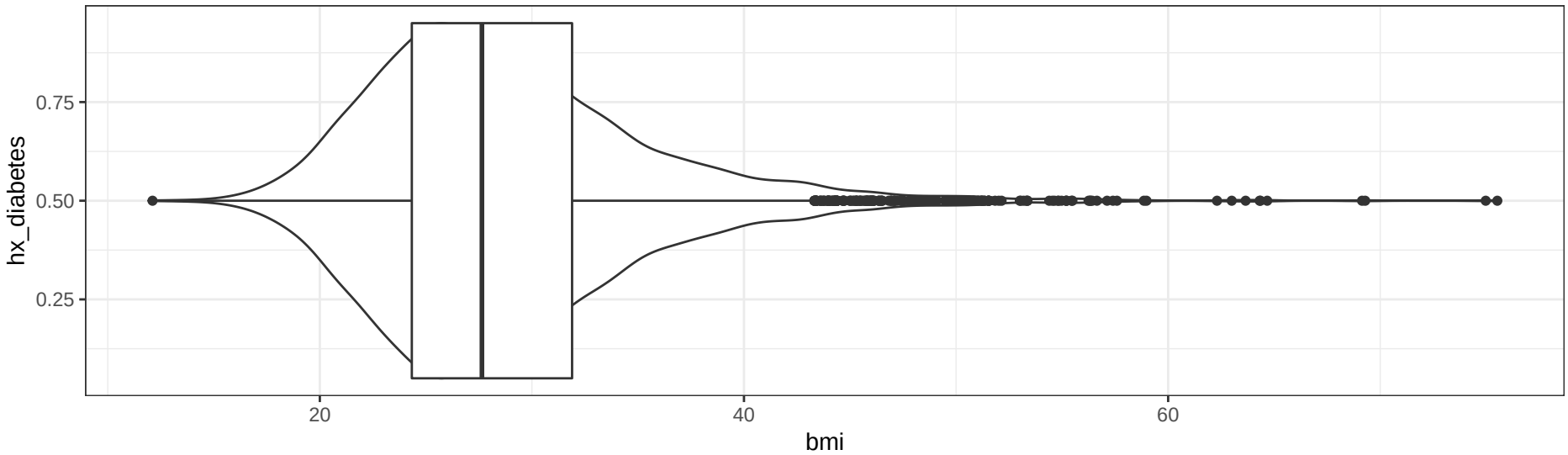
OK. Let's ask a question. . .

1. Did people with a history of diabetes show meaningfully different BMI values than people without such a history?
2. Does the answer to the question change if you take into account the subject's sex?
3. Does the answer to question 2 change if you also take into account the number of chronic diagnoses the person has?

Is hx_diabetes associated with bmi?

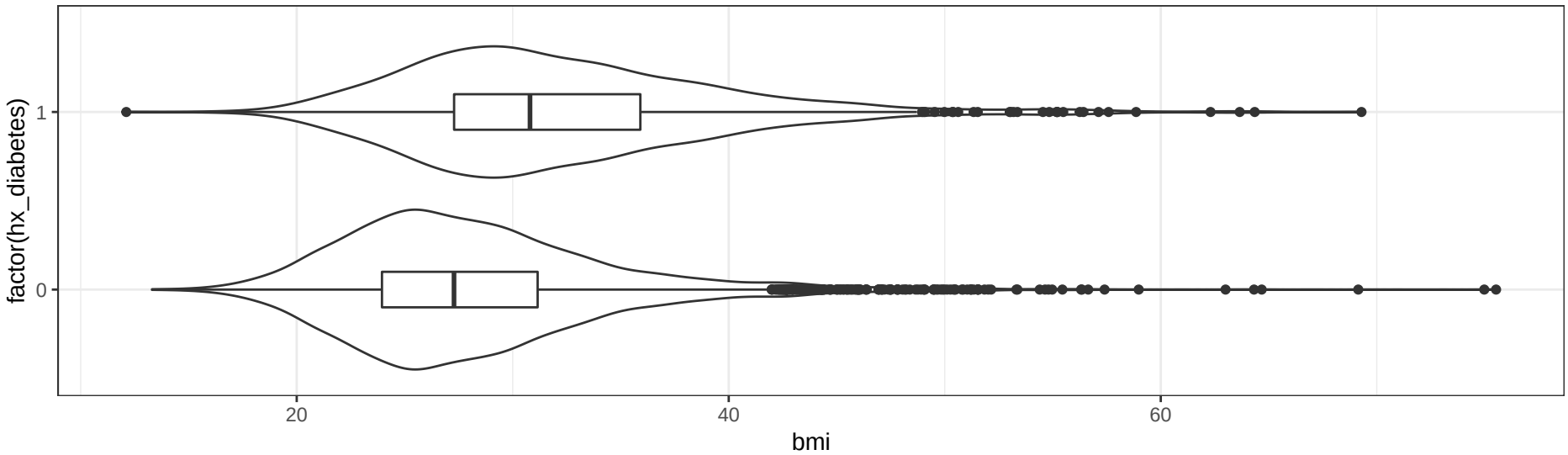
```
ggplot(smart_a_imp, aes(x = hx_diabetes, y = bmi)) +
    geom_violin() +
    geom_boxplot(width = 0.2) +
    coord_flip() + theme_bw()
```

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



Is hx_diabetes associated with bmi? (Redone)

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



Numerical Summary?

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

```
  hx_diabetes      min       Q1   median       Q3      max
1           0 13.29938 23.94667 27.28123 31.15351 75.52133
2           1 12.11097 27.28516 30.80014 35.90775 69.29032
      mean       sd    n missing
1 28.15276 6.189911 5298       0
2 32.16417 7.241287  979       0
```

Can we model this?

```
model_01 <- lm(bmi ~ hx_diabetes, data = smart_a_imp)
tidy(model_01)
```

```
# A tibble: 2 x 5
  term         estimate std.error statistic  p.value
  <chr>           <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)    28.2     0.0874      322.  0.
2 hx_diabetes     4.01    0.221        18.1 1.55e-71
```

Is this what we want?

Effect Sizes and 95% Confidence Intervals

```
tidy(model_01, conf.int = TRUE, conf.level = 0.95) %>%
    select(term, estimate, conf.low, conf.high, std.error)
```

```
# A tibble: 2 x 5
  term         estimate conf.low conf.high std.error
  <chr>           <dbl>    <dbl>     <dbl>     <dbl>
1 (Intercept)     28.2     28.0      28.3     0.0874
2 hx_diabetes      4.01     3.58      4.45    0.221
```

And this is just a two-sample t test

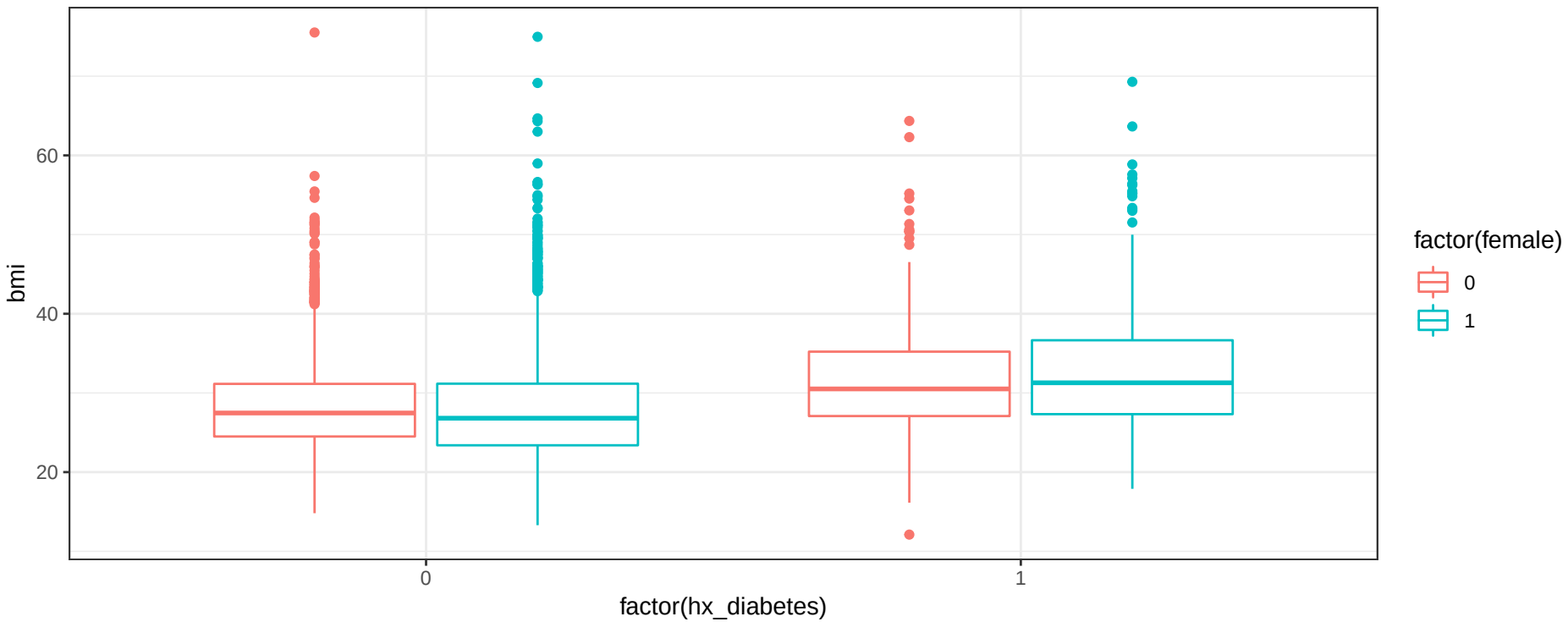
```
t.test(bmi ~ hx_diabetes, data = smart_a_imp,
       var.equal = TRUE)
```

```
    Two Sample t-test

data:  bmi by hx_diabetes
t = -18.116, df = 6275, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal
95 percent confidence interval:
 -4.445491 -3.577324
sample estimates:
mean in group 0 mean in group 1
       28.15276        32.16417
```

Does the bmi to hx_diabetes relationship depend on sex?

```
ggplot(smart_a_imp, aes(x = factor(hx_diabetes), y = bmi,
                        col = factor(female))) +
    geom_boxplot() + theme_bw()
```



Does the bmi to hx_diabetes relationship depend on sex?

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

```
  hx_diabetes.female      min       Q1   median       Q3      max     mean
1                0.0 14.81143 24.50139 27.46713 31.14390 75.52133 28.35483
2                1.0 12.11097 27.08157 30.50893 35.21455 64.34948 31.68363
3                0.1 13.29938 23.38816 26.80957 31.17103 74.97521 28.00960
4                1.1 17.89453 27.32072 31.28322 36.65381 69.29032 32.52522
         sd    n missing
1  5.623548 2197       0
2  6.833382  420       0
3  6.558775 3101       0
4  7.519096  559       0
```

Model bmi with hx_diabetes and female?

First, with no interaction term

```
model_02_no <- lm(bmi ~ hx_diabetes + female,
                  data = smart_a_imp)
anova(model_02_no)
```

```
Analysis of Variance Table

Response: bmi
              Df Sum Sq Mean Sq  F value Pr(>F)
hx_diabetes    1  13296 13296.5 328.1763 <2e-16 ***
female         1     38    38.4   0.9482 0.3302
Residuals   6274 254199    40.5
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model with Interaction Term

```
model_02_yes <- lm(bmi ~ hx_diabetes * female,
                   data = smart_a_imp)
anova(model_02_yes)
```

```
Analysis of Variance Table

Response: bmi
                     Df Sum Sq Mean Sq  F value  Pr(>F)
hx_diabetes           1  13296 13296.5 328.4918 < 2e-16 ***
female                1     38    38.4   0.9492 0.32997
hx_diabetes:female    1    285   284.7   7.0335 0.00802 **
Residuals          6273 253914    40.5
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

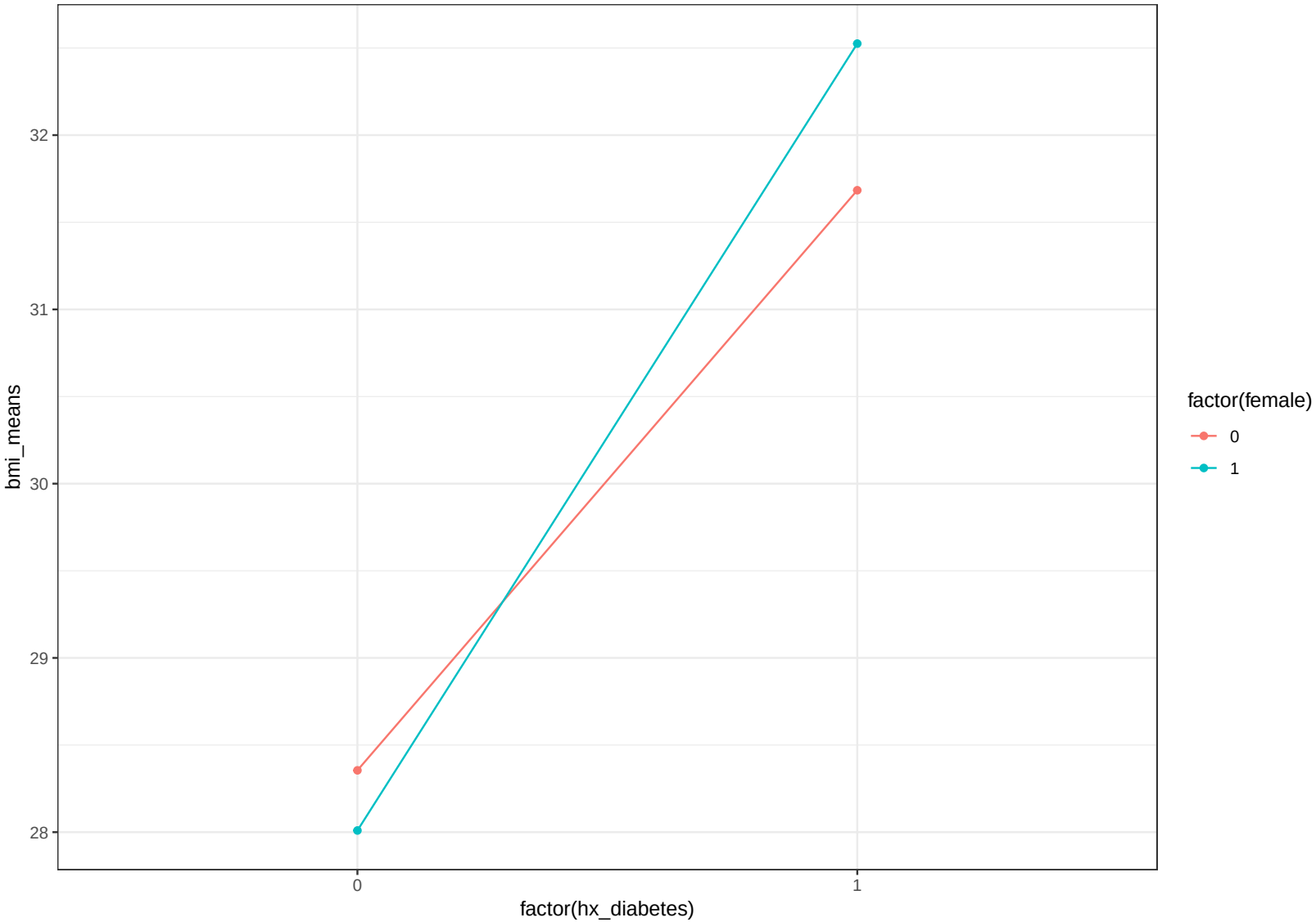
Two-Factor Analysis of Variance

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.

Interaction Plot for BMI Means

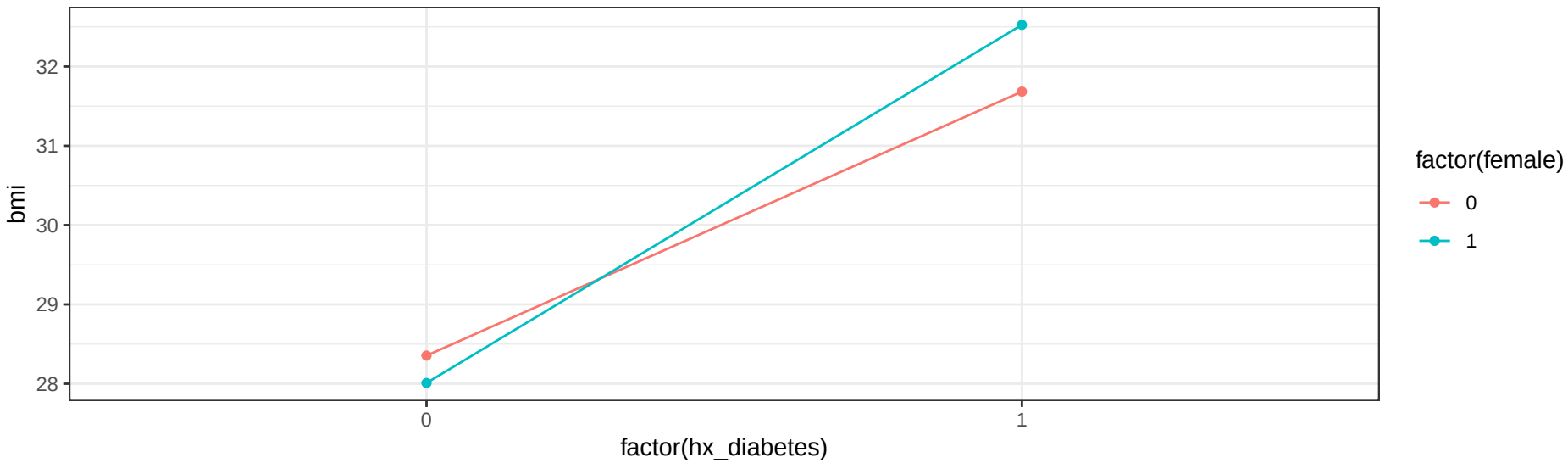


Code for Previous Slide

```
smart_a_imp %>%
    group_by(hx_diabetes, female) %>%
    summarise(bmi_means = mean(bmi)) %>%
    ggplot(., aes(x = factor(hx_diabetes), y = bmi_means,
                  color = factor(female))) +
    geom_line(aes(group = factor(female))) +
    geom_point() +
    theme_bw()
```

Alternative Coding for Visualizing Interaction

```
ggplot(smart_a_imp, aes(x = factor(hx_diabetes), y = bmi,
           group = factor(female), color = factor(female))) +
    stat_summary(fun.y = mean, geom = "point") +
    stat_summary(fun.y = mean, geom = "line") +
    theme_bw()
```



What Should We Conclude Here?

```
anova(model_02_yes)
```

```
Analysis of Variance Table

Response: bmi
                     Df Sum Sq Mean Sq  F value  Pr(>F)
hx_diabetes           1  13296 13296.5 328.4918 < 2e-16 ***
female                1     38    38.4   0.9492 0.32997
hx_diabetes:female    1    285   284.7   7.0335 0.00802 **
Residuals          6273 253914    40.5
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Making Predictions . . .

```
coef(model_02_yes) %>% round(digits = 2)
```

```
       (Intercept)       hx_diabetes            female
             28.35              3.33             -0.35
hx_diabetes:female
              1.19
```

```
coef(model_02_no) %>% round(digits = 2)
```

```
(Intercept) hx_diabetes      female
      28.25        4.01       -0.16
```

How well do these models work?

```
glance(model_02_yes) %>% round(digits = 2) %>%
    print.data.frame
```

```
  r.squared adj.r.squared sigma statistic p.value df
1      0.05          0.05  6.36    112.16       0  4
     logLik      AIC      BIC deviance df.residual
1 -20519.45 41048.91 41082.63 253914.1        6273
```

```
glance(model_02_no) %>% round(digits = 2) %>%
    print.data.frame
```

```
  r.squared adj.r.squared sigma statistic p.value df
1      0.05          0.05  6.37    164.56       0  3
     logLik      AIC      BIC deviance df.residual
1 -20522.97 41053.94 41080.92 254198.8        6274
```