

## 432 Class 3 Slides

[github.com/THOMASELOVE/2020-432](https://github.com/THOMASELOVE/2020-432)

2020-01-21

# Today's Agenda

- Creating the `smart1` and `smart1_sh` data sets
  - Working with factors
  - Working with simple imputation (`nanmiar` tools)
  - Creating a “shadow” to track what is imputed
- A few words on PPDAC and the combination of knowledge
- What is the effect of a diabetes diagnosis on BMI?
  - One-way analysis of variance (linear model)
- Does whether you have health insurance matter?
  - Two-way analysis of variance (linear model)
  - Thinking meaningfully about interaction
- Adjusting for a covariate: poor physical health days
  - Analysis of Covariance

# Setup

```
library(here); library(magrittr); library(janitor)
library(broom); library(simputation); library(patchwork)
library(naniar); library(visdat)
library(tidyverse)

theme_set(theme_bw())

smart0 <- read_csv(here("data/smart_ohio.csv"))
```

## BRFSS and SMART (Creating smart1)

```
smart1 <- smart0 %>%  
  mutate(SEQNO = as.character(SEQNO)) %>%  
  select(SEQNO, mmsa, mmsa_wt, landline,  
         age_imp, healthplan, dm_status,  
         fruit_day, drinks_wk, activity,  
         smoker, physhealth, bmi, genhealth)
```

## smart1 Variables, by Type

Variable	Type	Description
landline	Binary (1/0)	survey conducted by landline? (vs. cell)
healthplan	Binary (1/0)	subject has health insurance?
age_imp	Quantitative	age (imputed from groups - see Notes)
fruit_day	Quantitative	mean servings of fruit / day
drinks_wk	Quantitative	mean alcoholic drinks / week
bmi	Quantitative	body-mass index (in kg/m <sup>2</sup> )
physhealth	Count (0-30)	of last 30 days, # in poor physical health
dm_status	Categorical	diabetes status (4 levels, <i>we'll collapse to 2</i> )
activity	Categorical	physical activity level (4 levels, <i>we'll re-level</i> )
smoker	Categorical	smoking status (4 levels, <i>we'll collapse to 3</i> )
genhealth	Categorical	self-reported overall health (5 levels)

## Collapsing Two Factors, Re-leveling another

```
smart1 <- smart1 %>% type.convert() %>%  
  mutate(SEQNO = as.character(SEQNO)) %>%  
  mutate(dm_status =  
    fct_collapse(factor(dm_status),  
                  Yes = "Diabetes",  
                  No = c("No-Diabetes",  
                        "Pre-Diabetes",  
                        "Pregnancy-Induced")) %>%  
  mutate(smoker =  
    fct_collapse(factor(smoker),  
                  Current = c("Current_not_daily",  
                              "Current_daily")) %>%  
  mutate(activity =  
    fct_relevel(factor(activity),  
                "Highly_Active", "Active",  
                "Insufficiently_Active",  
                "Inactive"))
```

# The naniar and visdat packages

add functions to:

- display missing data, in many useful ways, often with `ggplot` approaches that you can modify as desired
- replace existing values with NA
- visualize imputed values
- numerically summarize imputed values
- model missingness

See Getting Started with `naniar` vignette linked at [our Class 3 README](#).

# How many missing values in smart1?

```
miss_var_table(smart1)
```

```
# A tibble: 11 x 3
```

	n_miss_in_var <int>	n_vars <int>	pct_vars <dbl>
1	0	4	28.6
2	14	1	7.14
3	15	1	7.14
4	20	1	7.14
5	68	1	7.14
6	138	1	7.14
7	242	1	7.14
8	392	1	7.14
9	493	1	7.14
10	557	1	7.14
11	723	1	7.14



# How many missing values in smart1?

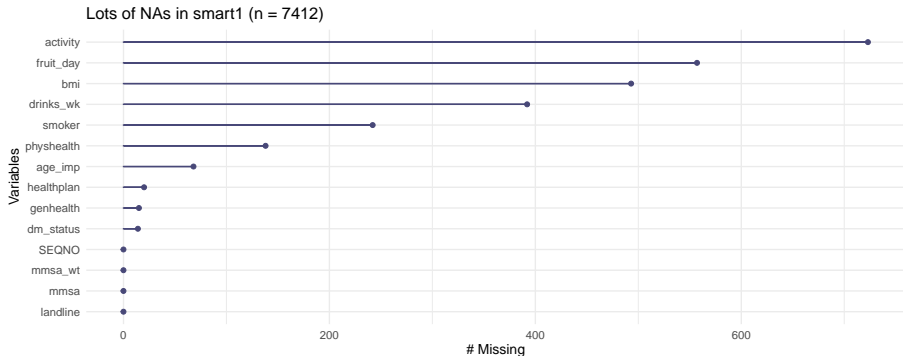
```
miss_var_summary(smart1)
```

```
# A tibble: 14 x 3
```

	variable	n_miss	pct_miss
	<chr>	<int>	<dbl>
1	activity	723	9.75
2	fruit_day	557	7.51
3	bmi	493	6.65
4	drinks_wk	392	5.29
5	smoker	242	3.26
6	physhealth	138	1.86
7	age_imp	68	0.917
8	healthplan	20	0.270
9	genhealth	15	0.202
10	dm_status	14	0.189
11	SEQNO	0	0
12	mmsa	0	0
13	mmsa_wt	0	0

# Visualizing Missingness in Variables

```
gg_miss_var(smart1) +  
  labs(title = "Lots of NAs in smart1 (n = 7412)")
```



## prop\_miss\_case and pct\_miss\_case

```
prop_miss_case(smart1)
```

```
[1] 0.1891527
```

```
smart1 %>% select(genhealth) %>% pct_miss_case(.)
```

```
[1] 0.2023745
```

Obtain the proportion or percentage of missing values in the data frame, or any piece of it.

## prop\_miss\_var or pct\_miss\_var

```
prop_miss_var(smart1)
```

```
[1] 0.7142857
```

```
pct_miss_var(smart1)
```

```
[1] 71.42857
```

This is the proportion (or percentage) of variables in the data frame with missing values.

## miss\_case\_table

```
miss_case_table(smart1)
```

```
# A tibble: 7 x 3
```

	n_miss_in_case <int>	n_cases <int>	pct_cases <dbl>
1	0	6010	81.1
2	1	830	11.2
3	2	223	3.01
4	3	119	1.61
5	4	133	1.79
6	5	85	1.15
7	6	12	0.162

## miss\_case\_summary

```
miss_case_summary(smart1)
```

```
# A tibble: 7,412 x 3
  case n_miss pct_miss
  <int> <int>    <dbl>
1     336      6    42.9
2     786      6    42.9
3    1102      6    42.9
4    1389      6    42.9
5    2788      6    42.9
6    3094      6    42.9
7    3373      6    42.9
8    5524      6    42.9
9    5733      6    42.9
10   6422      6    42.9
# ... with 7,402 more rows
```

# Creating a “Shadow” to track what is imputed

```
smart1_sh <- smart1 %>% bind_shadow()
```

## smart1\_sh creates new variables, ending in \_NA

```
names(smart1_sh)
```

```
[1] "SEQNO"          "mmsa"           "mmsa_wt"
[4] "landline"       "age_imp"        "healthplan"
[7] "dm_status"      "fruit_day"      "drinks_wk"
[10] "activity"       "smoker"         "physhealth"
[13] "bmi"            "genhealth"      "SEQNO_NA"
[16] "mmsa_NA"        "mmsa_wt_NA"     "landline_NA"
[19] "age_imp_NA"     "healthplan_NA"  "dm_status_NA"
[22] "fruit_day_NA"   "drinks_wk_NA"   "activity_NA"
[25] "smoker_NA"      "physhealth_NA"  "bmi_NA"
[28] "genhealth_NA"
```



# What are the new variables tracking?

```
smart1_sh %>% count(smoker, smoker_NA)
```

Warning: Factor `smoker` contains implicit NA, consider using `forcats::fct\_explicit\_na`

```
# A tibble: 4 x 3
  smoker  smoker_NA      n
  <fct>   <fct>    <int>
1 Current !NA        1290
2 Former  !NA        1999
3 Never   !NA        3881
4 <NA>    NA         242
```

## The fct\_explicit\_na warning: A pain point

My general preference is to not use `fct_explicit_na` in general, and I typically suppress this warning from printing by labeling the code chunk with `{r, warning = FALSE}`

## What do new variables track? (with warning = FALSE)

```
smart1_sh %>% count(genhealth, genhealth_NA)
```

```
# A tibble: 6 x 3
  genhealth  genhealth_NA      n
  <fct>      <fct>      <int>
1 1_Excellent !NA          1057
2 2_VeryGood  !NA          2406
3 3_Good      !NA          2367
4 4_Fair      !NA          1139
5 5_Poor      !NA           428
6 <NA>        NA           15
```

# “Simple” Imputation of Missing Factor Values

Let's impute some of the factors by random draws from their distributions...

```
set.seed(2020432)
smart1_sh <- smart1_sh %>%
  data.frame() %>%
  impute_rhd(.,
             dm_status + smoker + activity ~ 1) %>%
  tbl_df()
```

## Did this work? (Code Chunk has warning = FALSE)

```
smart1 %>% count(dm_status)
```

```
# A tibble: 3 x 2
  dm_status      n
  <fct>      <int>
1 Yes        1098
2 No         6300
3 <NA>         14
```

```
smart1_sh %>% count(dm_status)
```

```
# A tibble: 2 x 2
  dm_status      n
  <fct>      <int>
1 Yes        1102
2 No         6310
```

# What happens if you impute a 1/0 variable this way?

```
set.seed(2020432)
smart1_sh <- smart1_sh %>%
  data.frame() %>%
  impute_rhd(.,
             healthplan ~ 1) %>%
tbl_df()
```

## Look at whether this worked...

```
smart1 %>% tabyl(healthplan)
```

healthplan	n	percent	valid_percent
0	398	0.053696708	0.05384199
1	6994	0.943604965	0.94615801
NA	20	0.002698327	NA

```
smart1_sh %>% tabyl(healthplan)
```

healthplan	n	percent
0	399	0.05383162
1	7013	0.94616838

Looks OK

```
smart1_sh %$% n_distinct(healthplan)
```

```
[1] 2
```

## Another Sanity Check

```
smart1 %>%  
  select(healthplan, dm_status, smoker, activity) %>%  
  summarize_each(list(n_miss))
```

```
# A tibble: 1 x 4  
  healthplan dm_status smoker activity  
    <int>      <int> <int>    <int>  
1         20        14   242      723
```

```
smart1_sh %>%  
  select(healthplan, dm_status, smoker, activity) %>%  
  summarize_each(list(n_miss))
```

```
# A tibble: 1 x 4  
  healthplan dm_status smoker activity  
    <int>      <int> <int>    <int>  
1         0         0     0        0
```

# “Simple” Imputation with Robust Linear Models

```
set.seed(2020432)
smart1_sh <- smart1_sh %>%
  data.frame() %>%
  impute_rlm(.,
             age_imp + fruit_day +
             drinks_wk + bmi ~
             mmsa + landline + healthplan) %>%
  tbl_df()
```



# “Simple” Imputation with Other Methods

```
set.seed(2020432)
smart1_sh <- smart1_sh %>%
  data.frame() %>%
  impute_knn(., physhealth ~ bmi) %>%
  impute_cart(.,
              genhealth ~ activity +
                physhealth +
                mmsa + healthplan) %>%
tbl_df()
```

## Sanity Check 2

Before imputation...

```
pct_miss_var(smart1)
```

```
[1] 71.42857
```

After imputation ...

```
pct_miss_var(smart1_sh)
```

```
[1] 0
```

## Resulting smart1 and smart1\_sh tibbles saved to .Rds

```
saveRDS(smart1, "data/smart1.Rds")  
saveRDS(smart1_sh, "data/smart1_sh.Rds")
```

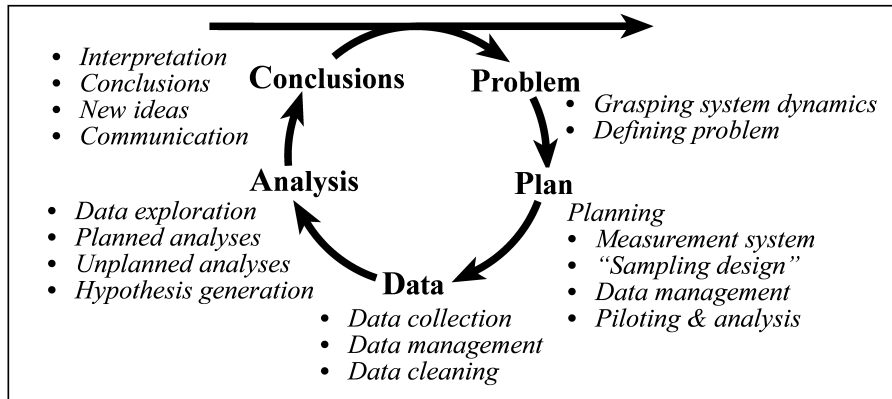
# *The Art of Statistics: How to Learn From Data*

## **Introduction:** Why We Need Statistics / Turning the World into Data

- Turning experiences into data is not straightforward, and data is inevitably limited in its capacity to describe the world.
- Statistical science has a long and successful history, but is now changing in the light of increased availability of data.
- The PPDAC cycle provides a convenient framework. . .
  - Problem - Plan - Data - Analysis - Conclusion and communication.

## (a) DIMENSION 1 : THE INVESTIGATIVE CYCLE

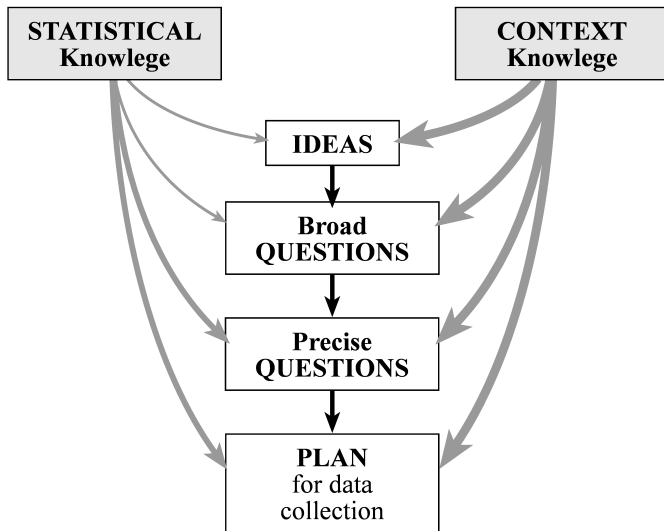
(PPDAC)



Chris Wild

- Chris Wild, <https://www.stat.auckland.ac.nz/~wild/StatThink/>

## From inkling to plan



*Chris Wild*

# Using the Analysis of Variance (ANOVA) and the Analysis of Covariance (ANCOVA) to model Categorical Predictors in Linear Models

# Answering Questions

- ① What is the effect of having a diagnosis of diabetes on body mass index (BMI)?
- ② Does whether you have health insurance affect how we think about the BMI-diabetes association?
- ③ Does adjusting for physical health (as measured by the number of poor physical health days in the past 30) affect our Question 2 assessment?

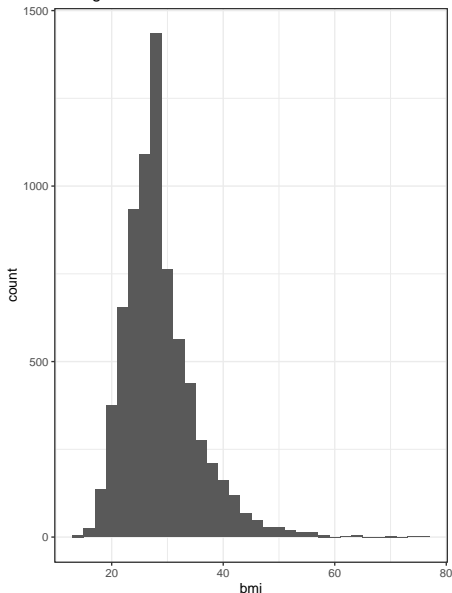


## Distribution of BMI? (code)

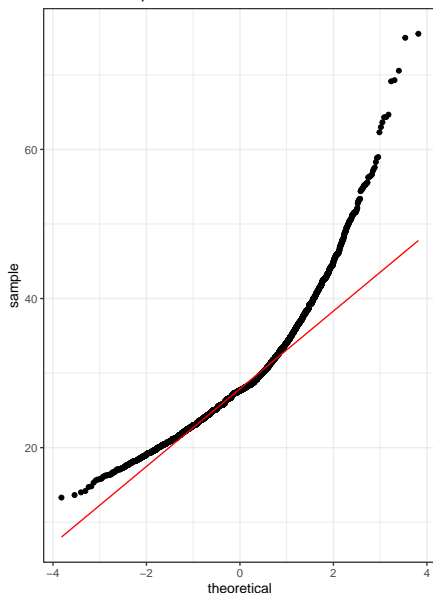
```
p1 <- ggplot(smart1_sh, aes(x = bmi)) +  
  geom_histogram(binwidth = 2) +  
  labs(title = "Histogram of BMI")  
  
p2 <- ggplot(smart1_sh, aes(sample = bmi)) +  
  geom_qq() + geom_qq_line(col = "red") +  
  labs(title = "Normal Q-Q plot of BMI")  
  
p1 + p2
```

# Distribution of BMI? (results)

Histogram of BMI



Normal Q-Q plot of BMI



# Answering Questions

- 1 What is the effect of having a diagnosis of diabetes on body mass index?

```
smart1_sh %$% mosaic::favstats(bmi ~ dm_status)
```

Registered S3 method overwritten by 'mosaic':

```
method          from  
fortify.SpatialPolygonsDataFrame ggplot2
```

	dm_status	min	Q1	median	Q3	max	mean
1	Yes	16.07	27.37061	30.295	35.7875	70.56	31.98108
2	No	13.30	24.11000	27.320	30.6100	75.52	28.01261

	sd	n	missing
1	7.301795	1102	0
2	6.033544	6310	0

How can we repair this?

- `r, message = FALSE` in chunk name
- show only a single decimal place?

# Answering Questions

- 1 What is the effect of having a diagnosis of diabetes on body mass index?

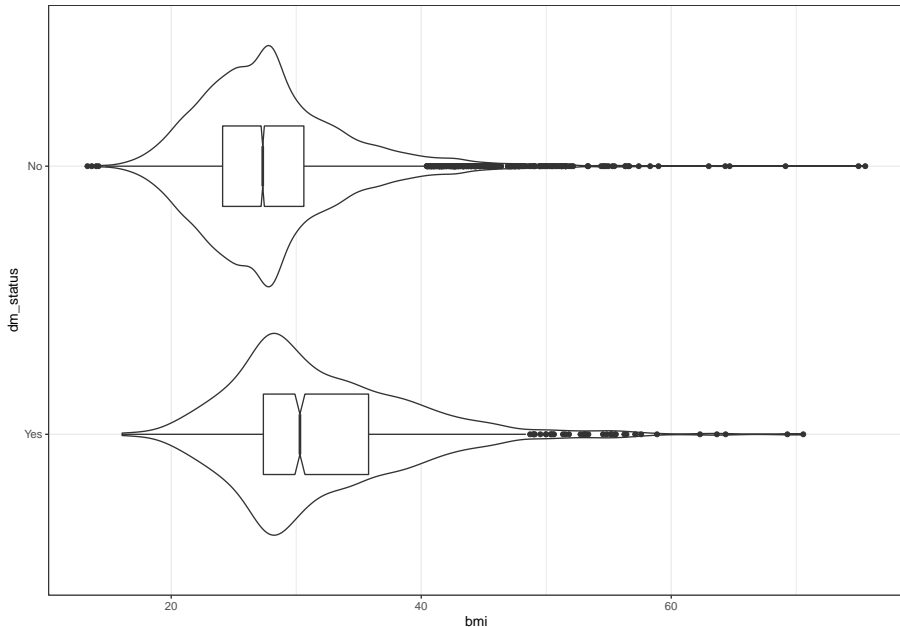
```
smart1_sh %>% mosaic::favstats(bmi ~ dm_status) %>%  
  rename(dm = dm_status) %>%  
  knitr::kable(digits = 1)
```

dm	min	Q1	median	Q3	max	mean	sd	n	missing
Yes	16.1	27.4	30.3	35.8	70.6	32	7.3	1102	0
No	13.3	24.1	27.3	30.6	75.5	28	6.0	6310	0

Plot the data!

```
ggplot(smart1_sh, aes(x = dm_status, y = bmi)) +  
  geom_violin() + geom_boxplot(width = 0.3, notch = TRUE) +  
  coord_flip()
```

# Visualizing the Data in Boxplots (with Violins)



# Analysis of Variance

- 1 What is the effect of having a diagnosis of diabetes on body mass index?

```
a1 <- smart1_sh %$% lm(bmi ~ dm_status)
anova(a1)
```

## Analysis of Variance Table

Response: bmi

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
dm_status	1	14775	14774.8	379.65	< 2.2e-16 ***
Residuals	7410	288372	38.9		

---

Signif. codes:

0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## Estimate effect of dm\_status on bmi...

```
tidy(a1, conf.int = TRUE, conf.level = 0.90) %>%  
  select(term, estimate, std.error, conf.low, conf.high) %>%  
  knitr::kable(digits = 3)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	31.981	0.188	31.672	32.290
dm_statusNo	-3.968	0.204	-4.304	-3.633

Is this easy to interpret?

## Re-level the dm\_status variable...

```
smart1_sh <- smart1_sh %>%  
  mutate(dm_status = fct_relevel(dm_status, "No", "Yes"))  
  
a1 <- smart1_sh %$% lm(bmi ~ dm_status)  
  
anova(a1)
```

### Analysis of Variance Table

Response: bmi

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
dm_status	1	14775	14774.8	379.65	< 2.2e-16 ***
Residuals	7410	288372	38.9		

---

Signif. codes:

0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1



## Estimate effect of re-leveled dm\_status on bmi...

```
tidy(a1, conf.int = TRUE, conf.level = 0.90) %>%  
  select(term, estimate, std.error, conf.low, conf.high) %>%  
  knitr::kable(digits = 3)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	28.013	0.079	27.883	28.142
dm_statusYes	3.968	0.204	3.633	4.304

# Answering Questions

- ② Does whether you have health insurance affect this association?

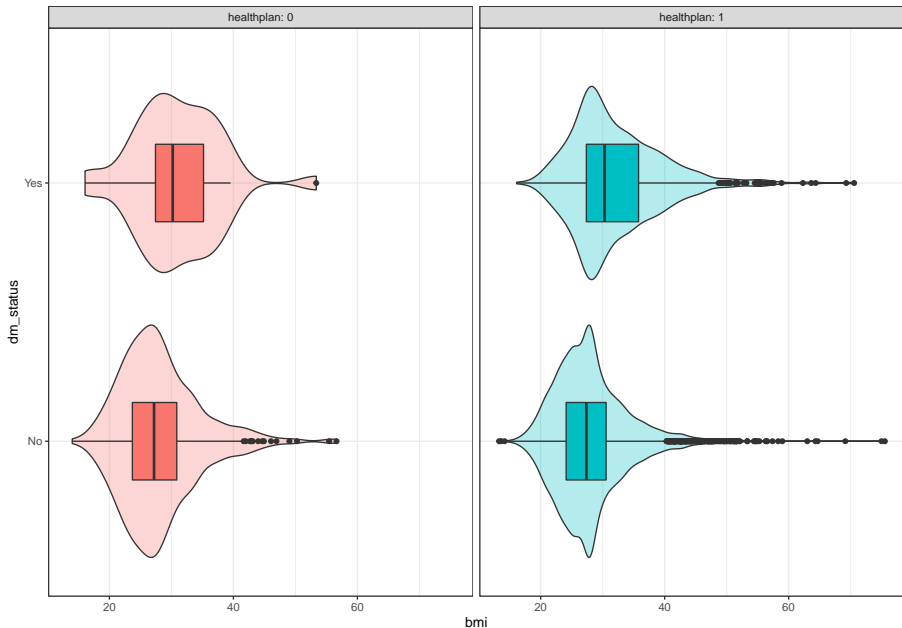
```
smart1_sh %$%  
  mosaic::favstats(bmi ~ dm_status + healthplan) %>%  
  rename(dm_hp = dm_status.healthplan) %>%  
  knitr::kable(digits = 1)
```

dm_hp	min	Q1	median	Q3	max	mean	sd	n	missing
No.0	14.0	23.7	27.2	30.9	56.6	28	6.4	364	0
Yes.0	16.1	27.4	30.2	35.2	53.4	31	6.9	35	0
No.1	13.3	24.1	27.4	30.6	75.5	28	6.0	5946	0
Yes.1	16.1	27.4	30.3	35.8	70.6	32	7.3	1067	0

# Visualize Three Variables (Code)

```
ggplot(smart1_sh, aes(x = dm_status, y = bmi,  
                      fill = factor(healthplan))) +  
  geom_violin(alpha = 0.3) +  
  geom_boxplot(width = 0.3, notch = TRUE) +  
  facet_wrap(~ healthplan, labeller = label_both) +  
  coord_flip() +  
  guides(fill = FALSE)
```

# Visualize Three Variables



## Direct Approach: An Interaction Plot

We'll plot the means of the `bmi` in the four combinations:

- two levels of `dm_status` combined with
- two levels of `healthplan`

```
summaries1 <- smart1_sh %>%  
  group_by(dm_status, healthplan) %>%  
  summarize(n = n(), mean = mean(bmi), stdev = sd(bmi))  
  
summaries1 %>% knitr::kable(digits = 2)
```

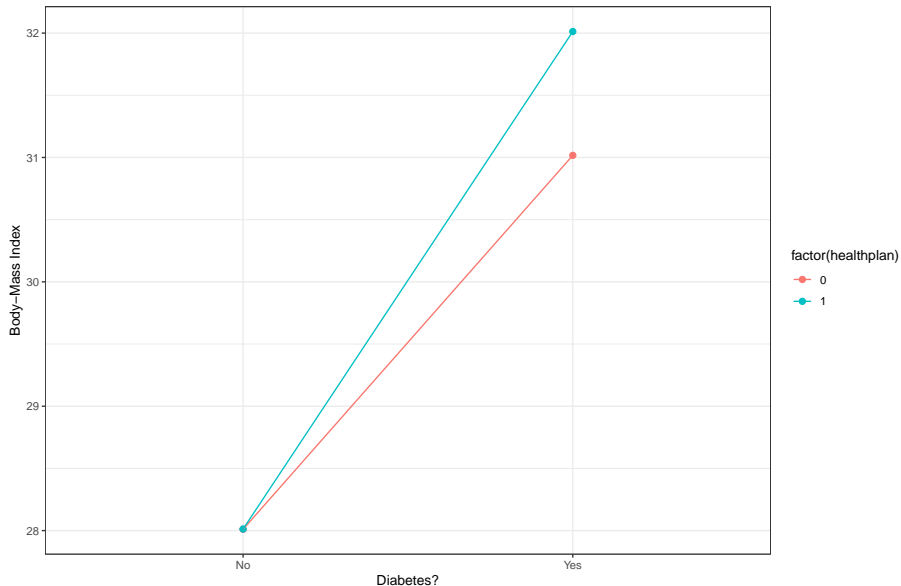
dm_status	healthplan	n	mean	stdev
No	0	364	28.01	6.41
No	1	5946	28.01	6.01
Yes	0	35	31.02	6.93
Yes	1	1067	32.01	7.31

# Simplest Interaction Plot (code)

```
ggplot(summaries1, aes(x = dm_status, y = mean,  
                        col = factor(healthplan))) +  
  geom_point(size = 2) +  
  geom_line(aes(group = healthplan)) +  
  labs(y = "Body-Mass Index",  
       x = "Diabetes?",  
       title = "Observed Means for BMI",  
       subtitle = "by Diabetes Status and Insurance")
```

# Simplest Interaction Plot (Results)

Observed Means for BMI  
by Diabetes Status and Insurance



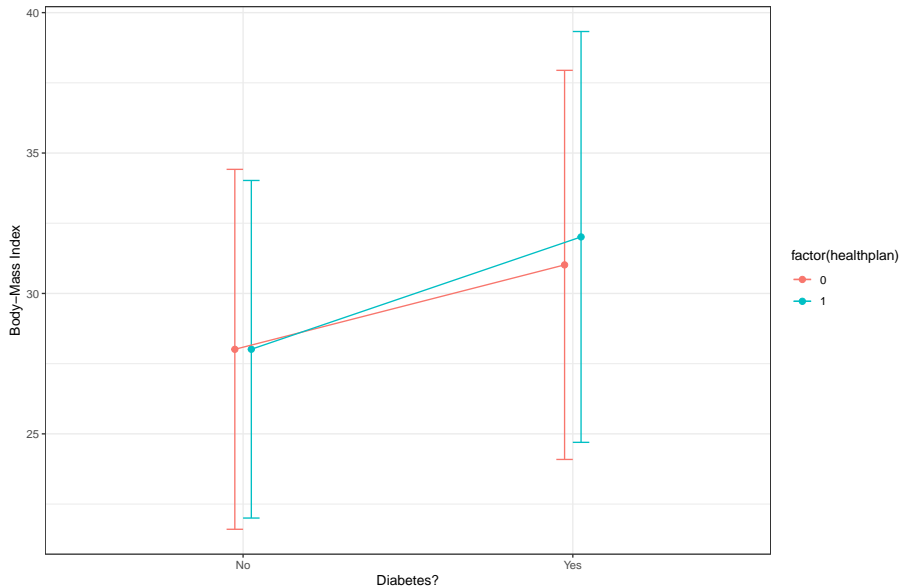
## Interaction Plot for Two-Way ANOVA (code)

```
pd <- position_dodge(0.1)
ggplot(summaries1, aes(x = dm_status, y = mean,
                        col = factor(healthplan))) +
  geom_errorbar(aes(ymin = mean - stdev,
                    ymax = mean + stdev),
                width = 0.1, position = pd) +
  geom_point(size = 2, position = pd) +
  geom_line(aes(group = healthplan), position = pd) +
  labs(y = "Body-Mass Index",
       x = "Diabetes?",
       title = "Observed Means (+/- SD) for BMI",
       subtitle = "by Diabetes Status and Insurance")
```



# Interaction Plot for Two-Way ANOVA

Observed Means ( $\pm$  SD) for BMI  
by Diabetes Status and Insurance



## Two-Way (Two Factor) Analysis of Variance

```
a2 <- smart1_sh %$% lm(bmi ~ dm_status * healthplan)

anova(a2) %>% knitr::kable(digits = 3)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
dm_status	1	14774.816	14774.816	379.595	0.000
healthplan	1	3.148	3.148	0.081	0.776
dm_status:healthplan	1	30.444	30.444	0.782	0.377
Residuals	7408	288338.239	38.923	NA	NA

Why am I using \* rather than + to connect dm\_status and healthplan?

# Two-Way (Two Factor) Analysis of Variance

Model without an interaction term:

```
a2_noint <- smart1_sh %$% lm(bmi ~ dm_status + healthplan)

anova(a2_noint) %>% knitr::kable(digits = 3)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
dm_status	1	14774.816	14774.816	379.606	0.000
healthplan	1	3.148	3.148	0.081	0.776
Residuals	7409	288368.683	38.921	NA	NA

## Model including an interaction term:

```
a2_switch <- smart1_sh %$% lm(bmi ~ healthplan * dm_status)

anova(a2_switch) %>% knitr::kable(digits = 3)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
healthplan	1	45.436	45.436	1.167	0.280
dm_status	1	14732.528	14732.528	378.509	0.000
healthplan:dm_status	1	30.444	30.444	0.782	0.377
Residuals	7408	288338.239	38.923	NA	NA

I switched the order of the two factors here. Does order matter?

## Model a2 tidied coefficients

```
tidy(a2, conf.int = TRUE, conf.level = 0.90) %>%  
  select(term, estimate, std.error, conf.low, conf.high) %>%  
  knitr::kable(digits = 3)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	28.011	0.327	27.473	28.549
dm_statusYes	3.006	1.104	1.190	4.823
healthplan	0.002	0.337	-0.552	0.556
dm_statusYes:healthplan	0.994	1.123	-0.855	2.842

## Model a2\_switch coefficients

```
tidy(a2_switch, conf.int = TRUE, conf.level = 0.90) %>%  
  select(term, estimate, std.error, conf.low, conf.high) %>%  
  knitr::kable(digits = 3)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	28.011	0.327	27.473	28.549
healthplan	0.002	0.337	-0.552	0.556
dm_statusYes	3.006	1.104	1.190	4.823
healthplan:dm_statusYes	0.994	1.123	-0.855	2.842

We can use this model to make predictions for each of four types of people:

- Those with diabetes, but not a health plan
- Those with diabetes and a health plan
- Those without diabetes, but who have a health plan
- Those without diabetes, and also without a health plan

# The Resulting Equations

The model with the interaction term is

$$\begin{aligned}\text{BMI} = & 28.011 + 3.006 (\text{dm\_status} = \text{Yes}) \\ & + 0.002 (\text{healthplan} = 1) \\ & + 0.994 (\text{dm\_status} = \text{Yes})(\text{healthplan} = 1)\end{aligned}$$

dm_status	healthplan	Predicted BMI
Yes	1 (Yes)	$28.011 + 3.006 + 0.002 + 0.994 = 32.013$
Yes	0 (No)	$28.011 + 3.006 = 31.017$
No	1 (Yes)	$28.011 + 0.002 = 28.013$
No	0 (No)	28.011

These are the original means (except for rounding error) of the four groups.

## Interpreting the Model with Interaction

```
tidy(a2, conf.int = TRUE, conf.level = 0.90) %>%  
  select(term, estimate, std.error, conf.low, conf.high) %>%  
  knitr::kable(digits = 3)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	28.011	0.327	27.473	28.549
dm_statusYes	3.006	1.104	1.190	4.823
healthplan	0.002	0.337	-0.552	0.556
dm_statusYes:healthplan	0.994	1.123	-0.855	2.842

- Our interpretation here would involve specifying that the interaction between `dm_status` and `healthplan` is important, and focusing on what that means, perhaps by specifying what happens to the four types of people we could see (Yes/Yes, Yes/No, No/Yes and No/No) in terms of our two factors.
- Do we need the interaction term here, or could we simplify the model?



# What's next?

- 1 Determining the importance of an interaction term
  - through an interaction plot
  - using sums of squares
  - using statistical significance testing
- 2 Building a two-factor ANOVA model with multi-categorical factors
  - again, focus on interpreting the interaction
  - add covariates, as desired
- 3 Building similar models for a binary outcome using linear probability models and then generalized linear models (specifically logistic regression).