431 Class 15

thomaselove.github.io/431

2020-10-13

Today's Agenda

- Comparing Means using Regression Models
  - Comparing Two Groups
  - Comparing More Than Two Groups
  - What you'll be using in Project A
- Alternatives for Comparing Means using Two Independent Samples
  - Welch's t test (not assuming equal population variances)
  - Bootstrap methods for comparing means in 2 samples
  - Rank-based alternatives (Wilcoxon-Mann-Whitney)

Today's R Packages and Data

```
library(broom)
library(ggrepel)
library(janitor)
library(knitr)
library(magrittr)
library(patchwork)
library(tidyverse)

theme_set(theme_bw())

source("data/Love-boost.R") # new today!

dm431 <- readRDS("data/dm431_2020.Rds")
```

Comparing Means with `dm431`

Our population: ALL adults ages 31-70 seen for care this year and two years ago who live in Northeast Ohio with a diabetes diagnosis.

Our `dm431` sample: 431 of those people, drawn in a way we hope is representative (but certainly isn't random).

1. Can we estimate the difference in the population mean LDL cholesterol for those who have a statin prescription as compared to those who do not?
2. Can we estimate the difference between people with four types of `insurance` in terms of their population mean hemoglobin A1c? (or maybe their diastolic BP?)
3. Can we estimate the difference between females and males in terms of the population mean systolic blood pressure?

Comparing Population Means using Regression Models

dm431 Example 1 (Comparing LDL by Statin usage)

Estimate the difference in the population mean LDL cholesterol for those who have a statin prescription as compared to those who do not.

```
mosaic::favstats(ldl ~ statin, data = dm431) %>%
  kable(digits = 2)
```

statin	min	Q1	median	Q3	max	mean	sd	n	missing
0	31	76	98.0	114.5	177	97.42	29.22	72	14
1	36	70	88.5	113.0	227	96.41	35.33	322	23

- What is the outcome here?

dm431 Example 1 (Comparing LDL by Statin usage)

Estimate the difference in the population mean LDL cholesterol for those who have a statin prescription as compared to those who do not.

```
mosaic::favstats(ldl ~ statin, data = dm431) %>%
  kable(digits = 2)
```

statin	min	Q1	median	Q3	max	mean	sd	n	missing
0	31	76	98.0	114.5	177	97.42	29.22	72	14
1	36	70	88.5	113.0	227	96.41	35.33	322	23

- What is the outcome here?
- What are the two exposure groups we are comparing?

dm431 Example 1 (Comparing LDL by Statin usage)

Estimate the difference in the population mean LDL cholesterol for those who have a statin prescription as compared to those who do not.

```
mosaic::favstats(ldl ~ statin, data = dm431) %>%
  kable(digits = 2)
```

statin	min	Q1	median	Q3	max	mean	sd	n	missing
0	31	76	98.0	114.5	177	97.42	29.22	72	14
1	36	70	88.5	113.0	227	96.41	35.33	322	23

- What is the outcome here?
- What are the two exposure groups we are comparing?
- What are the sample means, $\bar{x}_{Statin}$ and $\bar{x}_{NoStatin}$?

dm431 Example 1 (Comparing LDL by Statin usage)

Estimate the difference in the population mean LDL cholesterol for those who have a statin prescription as compared to those who do not.

```
mosaic::favstats(ldl ~ statin, data = dm431) %>%
  kable(digits = 2)
```

statin	min	Q1	median	Q3	max	mean	sd	n	missing
0	31	76	98.0	114.5	177	97.42	29.22	72	14
1	36	70	88.5	113.0	227	96.41	35.33	322	23

- What is the outcome here?
- What are the two exposure groups we are comparing?
- What are the sample means, $\bar{x}_{Statin}$ and $\bar{x}_{NoStatin}$?
- How might we estimate the difference in population means, $\mu_S - \mu_N$?

dm431 Example 1 (Comparing LDL by Statin usage)

Estimate the difference in the population mean LDL cholesterol for those who have a statin prescription as compared to those who do not.

```
mosaic::favstats(ldl ~ statin, data = dm431) %>%
  kable(digits = 2)
```

statin	min	Q1	median	Q3	max	mean	sd	n	missing
0	31	76	98.0	114.5	177	97.42	29.22	72	14
1	36	70	88.5	113.0	227	96.41	35.33	322	23

- What is the outcome here?
- What are the two exposure groups we are comparing?
- What are the sample means, $\bar{x}_{Statin}$ and $\bar{x}_{NoStatin}$?
- How might we estimate the difference in population means, $\mu_S - \mu_N$?
- Is there a problem in these data we need to deal with?

How much missing data do we have?

Do we have missing values in both columns, or just one?

```
dm431 %>% summarize(across(c(statin, ldl), ~ sum(is.na(.x))))
```

```
# A tibble: 1 x 2
  statin   ldl
   <int> <int>
1      0    37
```

So what shall we do?

- Drop the 37 cases, or
- Something else?

On Missing Data

Drop the Missing = A "Complete Case" analysis

- We could drop these 37, and do a **complete case analysis** on the other 431-37 = 394 subjects.
- We'll also create a factor (`statin_f`) with the statin information.

```
dm431_cc <- dm431 %>% filter(complete.cases(ldl, statin)) %>%
  mutate(statin_f = fct_recode(factor(statin),
                               "Statin" = "1", "No" = "0"))

mosaic::favstats(ldl ~ statin_f, data = dm431_cc) %>%
  kable(dig = 2)
```

statin_f	min	Q1	median	Q3	max	mean	sd	n	missing
No	31	76	98.0	114.5	177	97.42	29.22	72	0
Statin	36	70	88.5	113.0	227	96.41	35.33	322	0

- HUGE assumption: The 37 missing `ldl` are MCAR.

Missing Completely at Random (MCAR)

Our complete case analysis requires the HUGE assumption that these 37 observations are what Donald Rubin called "missing completely at random."

Missing Completely at Random (MCAR) means that there is no relationship between whether a data point is missing and any values in the data set, missing or observed. Thus, the missing values are just a random subset of the data.

- That is the huge assumption that is both impossible to prove and that is also tacitly made in many settings, more or less by default.
- The alternative is to consider other possible mechanisms (besides MCAR) for why data might be missing.

Assuming data are Missing at Random (MAR)?

Missing at Random (MAR): the reason a data point is missing is related to some observed data, but unrelated to the actual missing values.

So we assume that we can predict the missing values effectively using other variables in the data, without causing any problems. That's a big assumption, but then we could *impute* (or fill in with predictions based on other variables) the missing data.

So to impute predicted `ldl` values for these 37 subjects, we'd need to:

- account for the fact that we're imputing in building estimates, and
- control for the variables which (together) predict why the data were missing, and
- remember that we are making a large and unverifiable assumption about why the data are missing.

If missing data aren't MCAR or MAR, then they are MNAR.

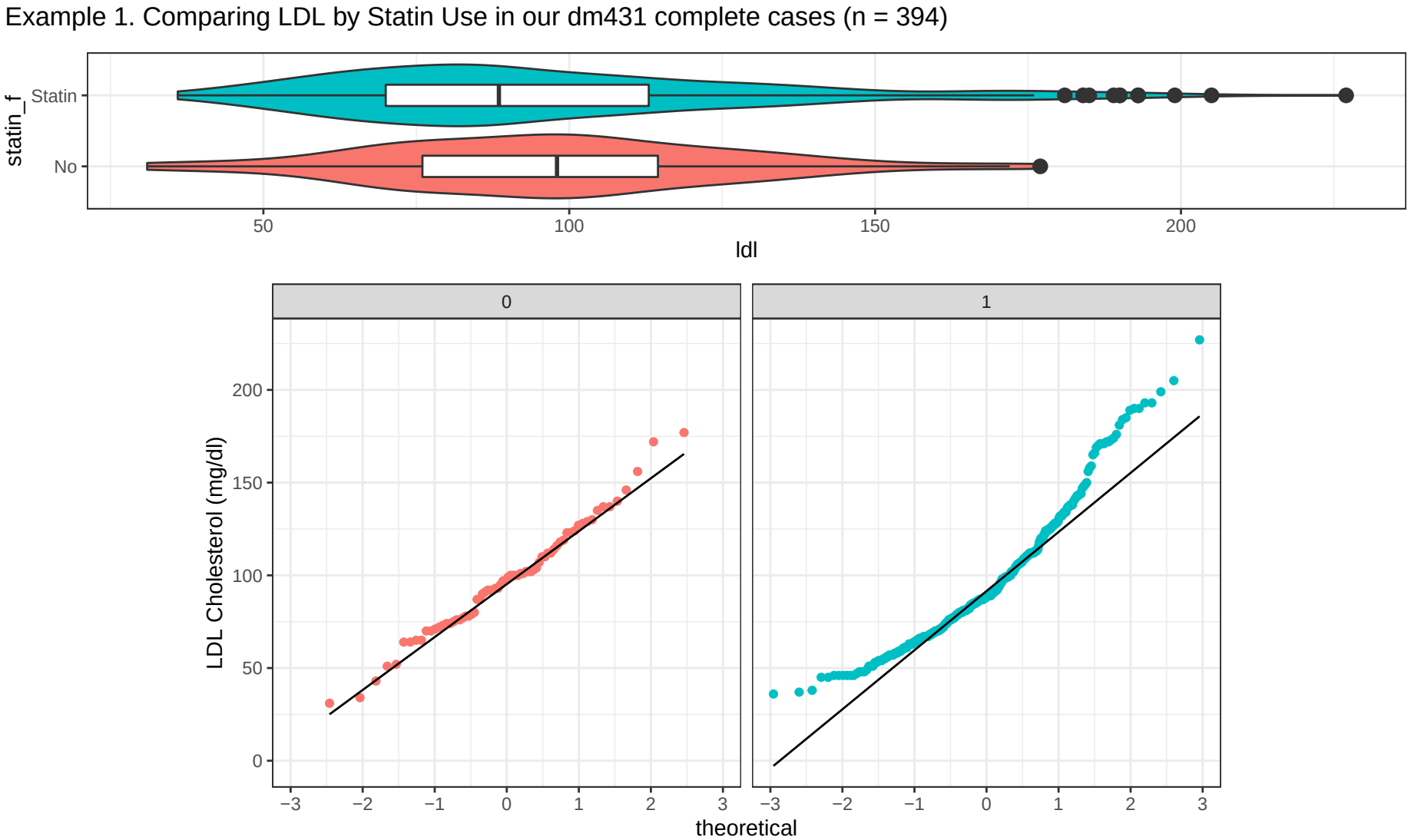
Three Types of Missingness

1. MCAR: Missing Completely At Random (ignorable nonresponse)
   - missing values are just a random subset of the data
   - unrealistically strong assumption in practice, although it's easy
   - makes a complete case analysis unbiased
2. MAR: Missing At Random
   - reason for missingness can be completely accounted for by variables where there is complete information
   - much more reasonable in many settings than MCAR, but impossible to verify statistically
   - imputing missing values here leads to a more robust conclusion
3. MNAR: Missing Not at Random (nonignorable nonresponse)
   - data are neither MCAR nor MAR
   - the reason the data is missing is related to its value, even after controlling for other variables.

These have different effects on the validity of the conclusions you build.

DTDP: Example 1. (Comparing LDL by Statin Use)

Assuming MCAR, we'll press on with a complete case analysis.



Linear Model for Example 1 (slide A)

Estimate the difference in population mean LDL cholesterol among people taking a statin as compared to those not taking a statin.

```
app1 <- lm(ldl ~ statin, data = dm431_cc)

tidy(app1, conf.int = T, conf.level = 0.90) %>% kable(dig = 2)
```

term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	97.42	4.04	24.09	0.00	90.75	104.08
statin	-1.01	4.47	-0.23	0.82	-8.38	6.36

- What can we learn from this output?
  - What is the sample mean `ldl` for those not on a statin?
  - What is the sample mean `ldl` for statin users?
  - The point estimate for $\mu_S - \mu_N$ is ...

Linear Model for Example 2 (slide B)

Estimate the difference in population mean LDL cholesterol among people taking a statin as compared to those not taking a statin.

```
app1 <- lm(ldl ~ statin, data = dm431_cc)

tidy(app1, conf.int = T, conf.level = 0.90) %>% kable(dig = 2)
```

term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	97.42	4.04	24.09	0.00	90.75	104.08
statin	-1.01	4.47	-0.23	0.82	-8.38	6.36

- What can we learn from this output?
  - The point estimate for $\mu_S - \mu_N$ is **-1.01**
  - The 90% confidence interval for $\mu_S - \mu_N$ is ...

Linear Model for Example 2 (slide C)

Estimate the difference in population mean LDL cholesterol among people taking a statin as compared to those not taking a statin.

```
app1 <- lm(ldl ~ statin, data = dm431_cc)

tidy(app1, conf.int = T, conf.level = 0.90) %>% kable(dig = 2)
```

term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	97.42	4.04	24.09	0.00	90.75	104.08
statin	-1.01	4.47	-0.23	0.82	-8.38	6.36

- What can we learn from this output?
  - The point estimate for $\mu_S - \mu_N$ is -1.01
  - The 90% confidence interval for $\mu_S - \mu_N$ is (**-8.38**, **6.36**)

Example 2 (Comparing Hemoglobin A1c by Insurance type)

Comparing A1c by Insurance Type in dm431

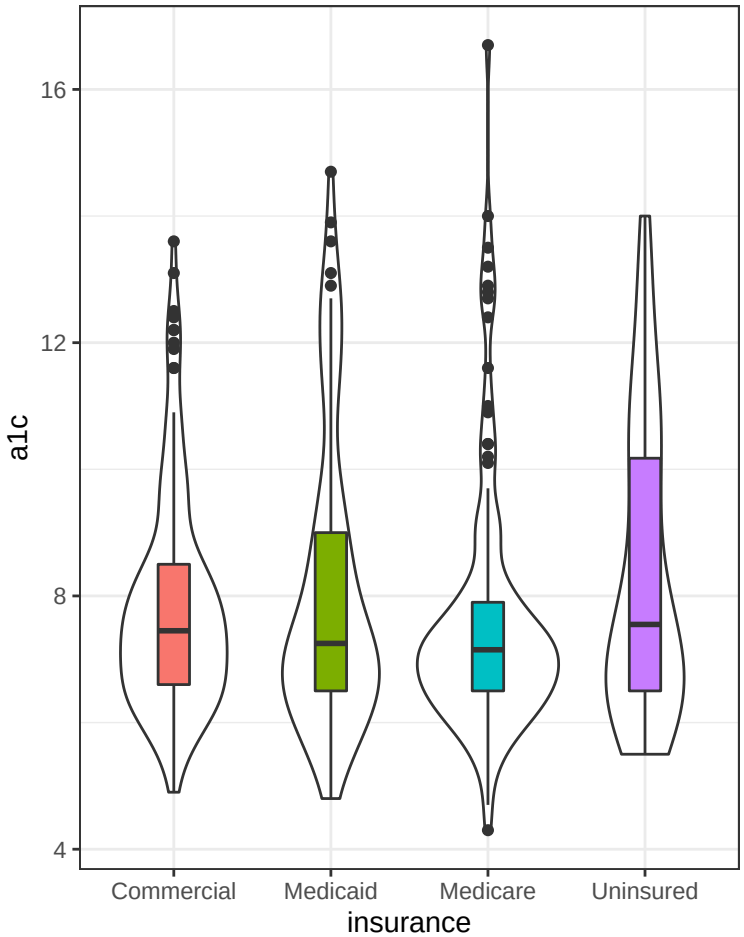
```
dm431 %>% select(insurance, a1c) %>% glimpse()
```

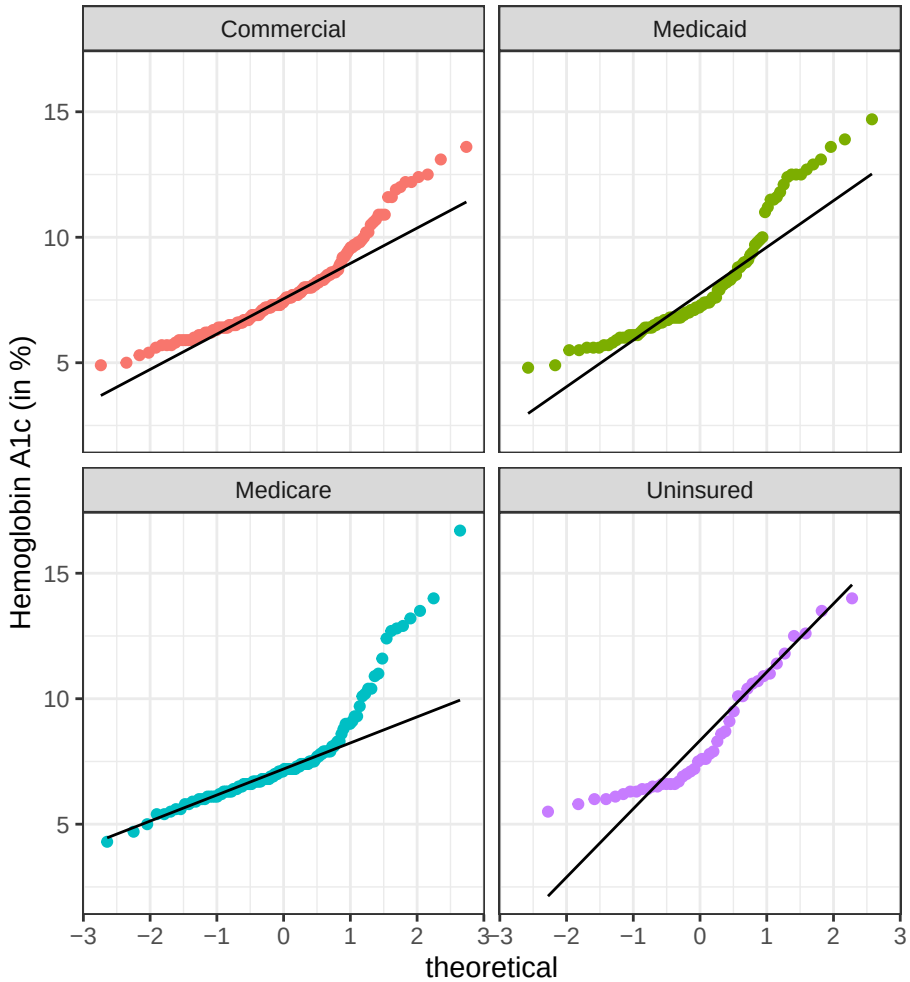
```
Rows: 431
Columns: 2
$ insurance <fct> Commercial, Uninsured, Uninsured...
$ a1c       <dbl> 6.3, 11.0, 8.7, 6.5, 6.7, 5.8, 9...
```

```
dm431 %$% mosaic::favstats(a1c ~ insurance) %>%
  rename(na = missing) %>% kable(dig = 2)
```

insurance	min	Q1	median	Q3	max	mean	sd	n	na
Commercial	4.9	6.6	7.45	8.50	13.6	7.83	1.72	162	2
Medicaid	4.8	6.5	7.25	9.00	14.7	8.07	2.30	100	0
Medicare	4.3	6.5	7.15	7.90	16.7	7.64	2.04	122	1
Uninsured	5.5	6.5	7.55	10.17	14.0	8.35	2.33	44	0

Distribution of A1c in insurance groups





Code for previous slide

```
dm_comp <- dm431 %>%
  filter(complete.cases(a1c, insurance))

p1 <- ggplot(dm_comp, aes(x = insurance, y = a1c)) +
  geom_violin() +
  geom_boxplot(aes(fill = insurance), width = 0.2) +
  guides(fill = FALSE)

p2 <- ggplot(dm_comp, aes(sample = a1c, col = insurance)) +
  geom_qq() + geom_qq_line(col = "black") +
  guides(col = FALSE) +
  theme(aspect.ratio = 1) +
  labs(y = "Hemoglobin A1c (in %)") +
  facet_wrap(~ insurance)

p1 + p2 + plot_layout(widths = c(2,3))
```

We'll assume MCAR and run a model

```
dm_comp <- dm431 %>%
  filter(complete.cases(a1c, insurance))

modA <- lm(a1c ~ insurance, data = dm_comp)
modA
```

```
Call:
lm(formula = a1c ~ insurance, data = dm_comp)

Coefficients:
       (Intercept)   insuranceMedicaid
            7.8272              0.2468
 insuranceMedicare  insuranceUninsured
           -0.1919              0.5251
```

- It was very helpful that `insurance` was a factor already.

Model A Fit Summary

```
glance(modA) %>%
  select(r.squared, statistic, df, df.residual,
         p.value, sigma, nobs) %>%
  kable(dig = c(3, 2, 0, 0, 4, 2, 0))
```

r.squared	statistic	df	df.residual	p.value	sigma	nobs
0.012	1.73	3	424	0.1598	2.03	428

What can we conclude about whether `insurance` is an effective predictor of `a1c` in these data?

Model A Coefficients

```
tidy(modA, conf.int = TRUE, conf.level = 0.90) %>%
  select(term, estimate, std.error, conf.low, conf.high) %>%
  kable(dig = 2)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	7.83	0.16	7.56	8.09
insuranceMedicaid	0.25	0.26	-0.18	0.67
insuranceMedicare	-0.19	0.24	-0.59	0.21
insuranceUninsured	0.53	0.34	-0.04	1.09

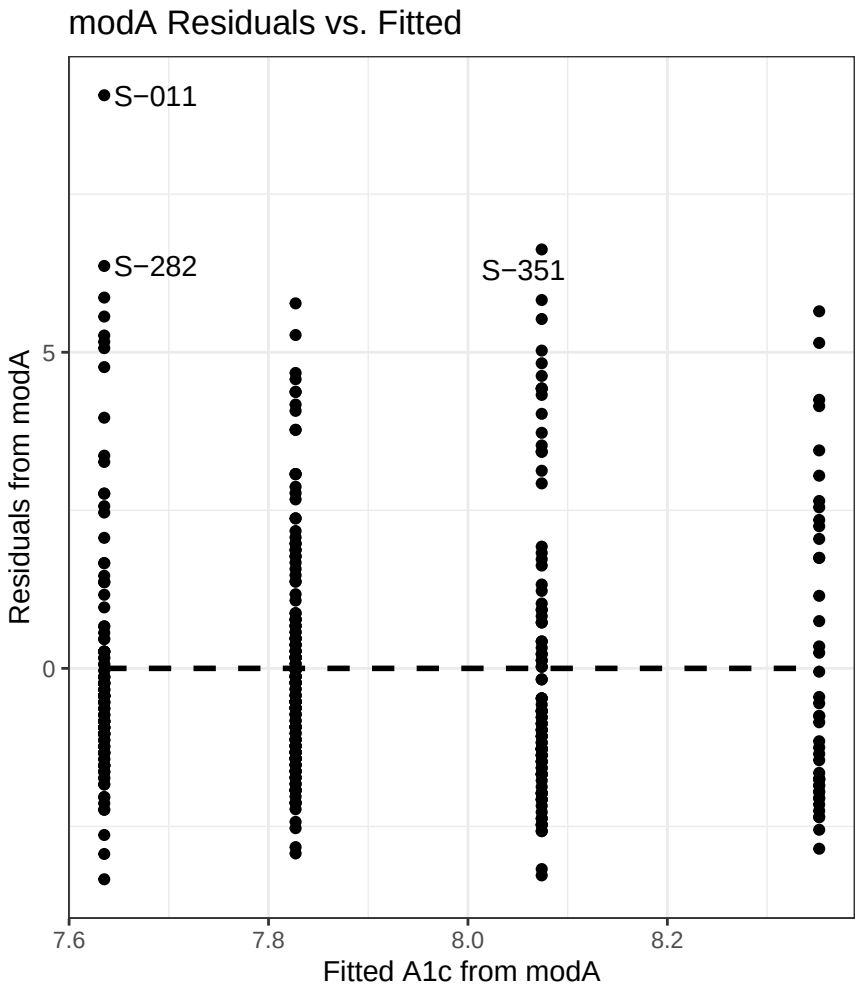
- Which insurance type is associated with the highest (worst) A1c?
- Which has the lowest predicted A1c? Are these results surprising?

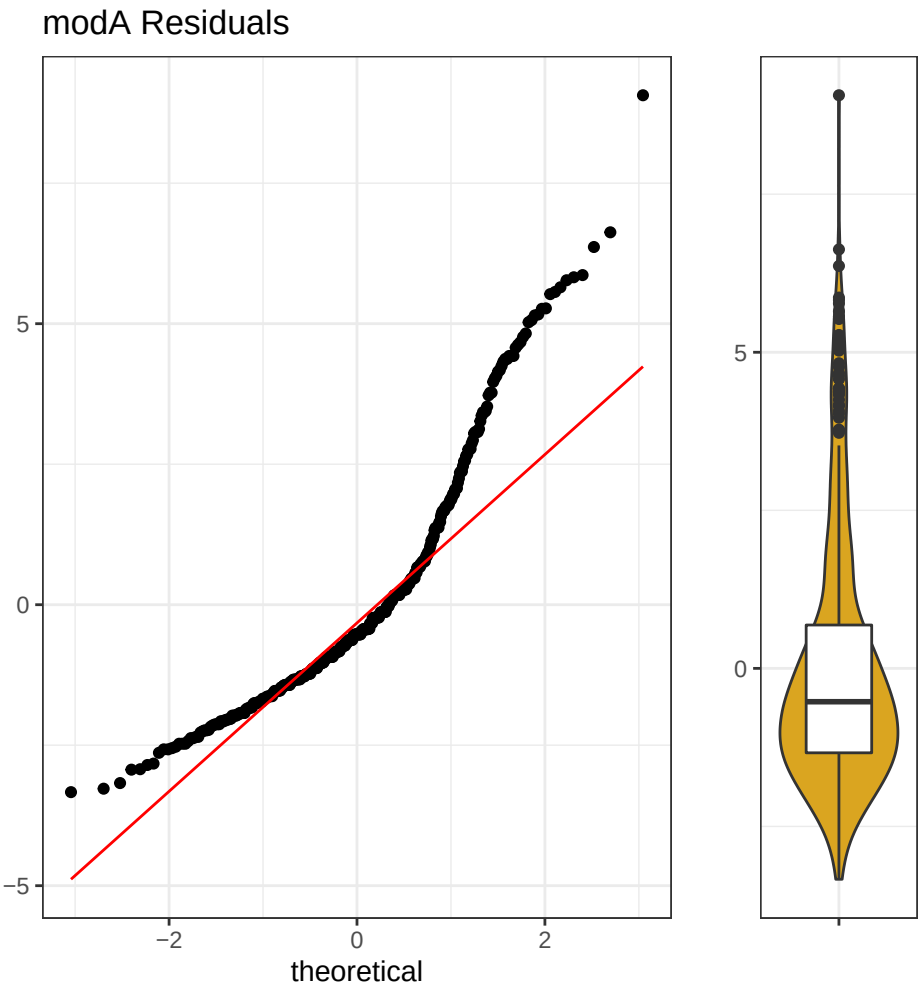
Making Predictions with augment

```
augA <- augment(modA, dm_comp)

augA %>% select(subject, insurance, a1c, .fitted, .resid) %>%
  head() %>% kable(dig = 2)
```

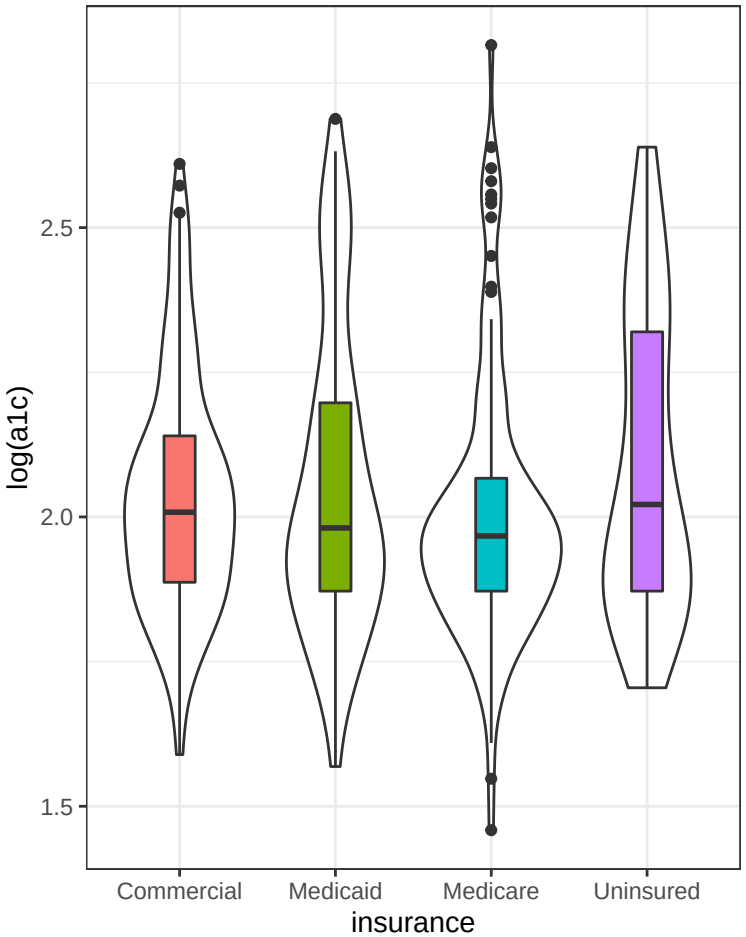
subject	insurance	a1c	.fitted	.resid
S-001	Commercial	6.3	7.83	-1.53
S-002	Uninsured	11.0	8.35	2.65
S-003	Uninsured	8.7	8.35	0.35
S-004	Commercial	6.5	7.83	-1.33
S-005	Commercial	6.7	7.83	-1.13
S-006	Medicare	5.8	7.64	-1.84

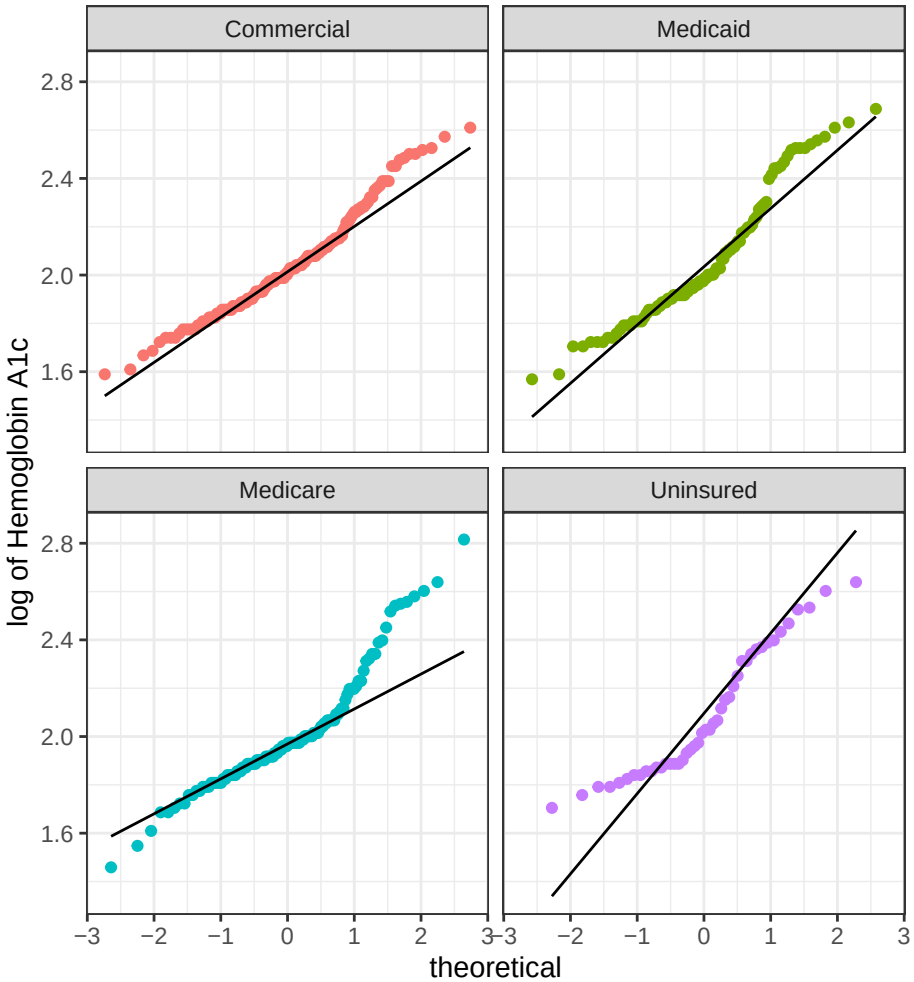
Residual Plots for `modA`





Try `log(a1c)` as our outcome instead?





log(A1c) by Insurance Type in dm431

```
dm431 %$% mosaic::favstats(log(a1c) ~ insurance) %>%
  rename(na = missing) %>% kable(dig = 3)
```

insurance	min	Q1	median	Q3	max	mean	sd	n
Commercial	1.589	1.887	2.008	2.140	2.610	2.036	0.205	162
Medicaid	1.569	1.872	1.981	2.197	2.688	2.053	0.261	100
Medicare	1.459	1.872	1.967	2.067	2.815	2.004	0.232	122
Uninsured	1.705	1.872	2.022	2.320	2.639	2.087	0.263	44

We'll assume MCAR and run the logged A1c model

```
dm_comp <- dm431 %>%
  filter(complete.cases(a1c, insurance))

modB <- lm(log(a1c) ~ insurance, data = dm_comp)
modB
```

```
Call:
lm(formula = log(a1c) ~ insurance, data = dm_comp)

Coefficients:
       (Intercept)   insuranceMedicaid
           2.03576             0.01718
 insuranceMedicare  insuranceUninsured
          -0.03202             0.05171
```

Model B Fit Summary

```
glance(modB) %>%
  select(r.squared, statistic, df, df.residual,
         p.value, sigma, nobs) %>%
  kable(dig = c(3, 2, 0, 0, 4, 2, 0))
```

r.squared	statistic	df	df.residual	p.value	sigma	nobs
0.012	1.67	3	424	0.1727	0.23	428

What can we conclude about whether `insurance` is an effective predictor of `log(a1c)` in these data?

Model B Coefficients

```
tidy(modB, conf.int = TRUE, conf.level = 0.90) %>%
  select(term, estimate, std.error, conf.low, conf.high) %>%
  kable(dig = 3)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	2.036	0.018	2.006	2.066
insuranceMedicaid	0.017	0.030	-0.032	0.066
insuranceMedicare	-0.032	0.028	-0.078	0.014
insuranceUninsured	0.052	0.040	-0.014	0.117

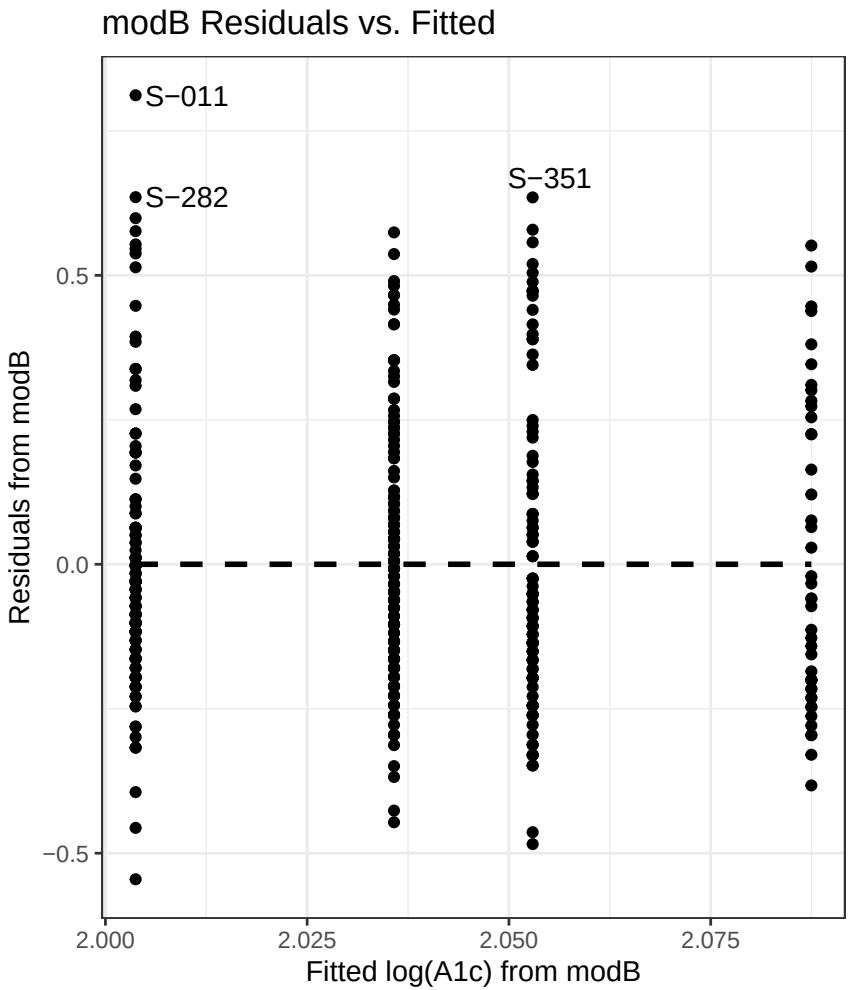
- Which insurance type is associated with the highest (worst) A1c?
- Which has the lowest predicted A1c? Are these results surprising?

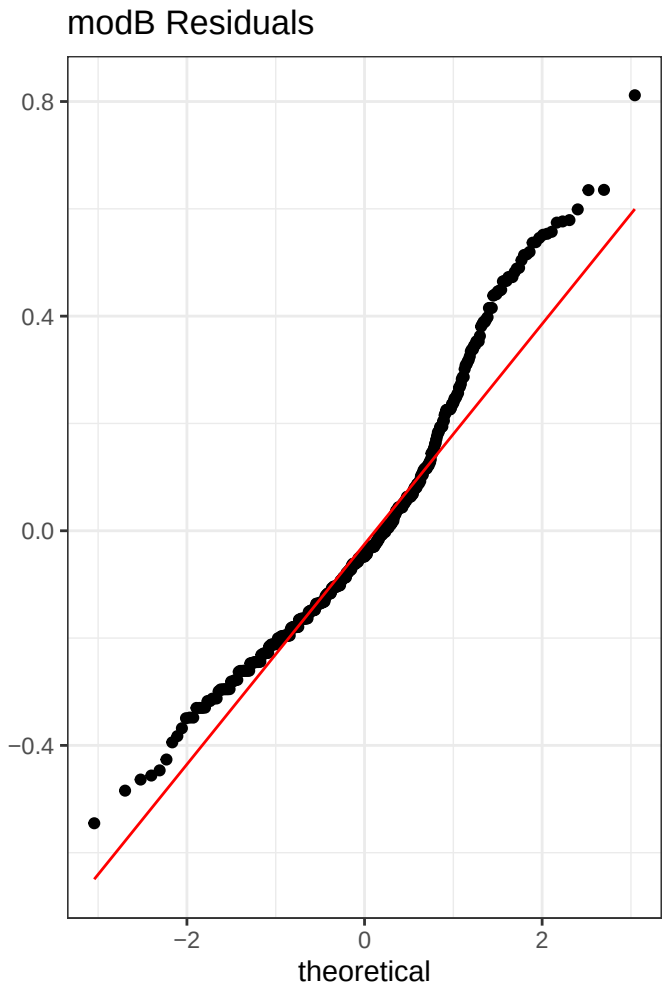
Making Predictions with augment

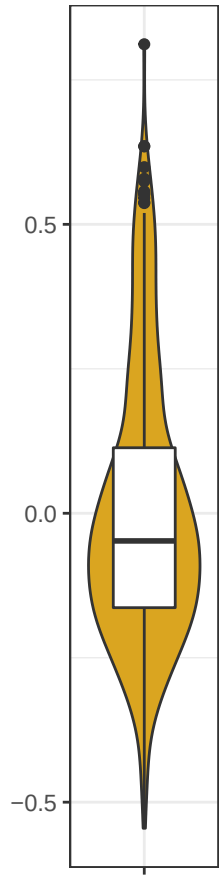
```
augB <- augment(modB, dm_comp) %>%
  mutate(log_a1c = log(a1c))

augB %>% select(subject, insurance, a1c,
                log_a1c, .fitted, .resid) %>%
  head() %>% kable(dig = 3)
```

subject	insurance	a1c	log_a1c	.fitted	.resid
S-001	Commercial	6.3	1.841	2.036	-0.195
S-002	Uninsured	11.0	2.398	2.087	0.310
S-003	Uninsured	8.7	2.163	2.087	0.076
S-004	Commercial	6.5	1.872	2.036	-0.164
S-005	Commercial	6.7	1.902	2.036	-0.134
S-006	Medicare	5.8	1.758	2.004	-0.246

Residual Plots for `modB`







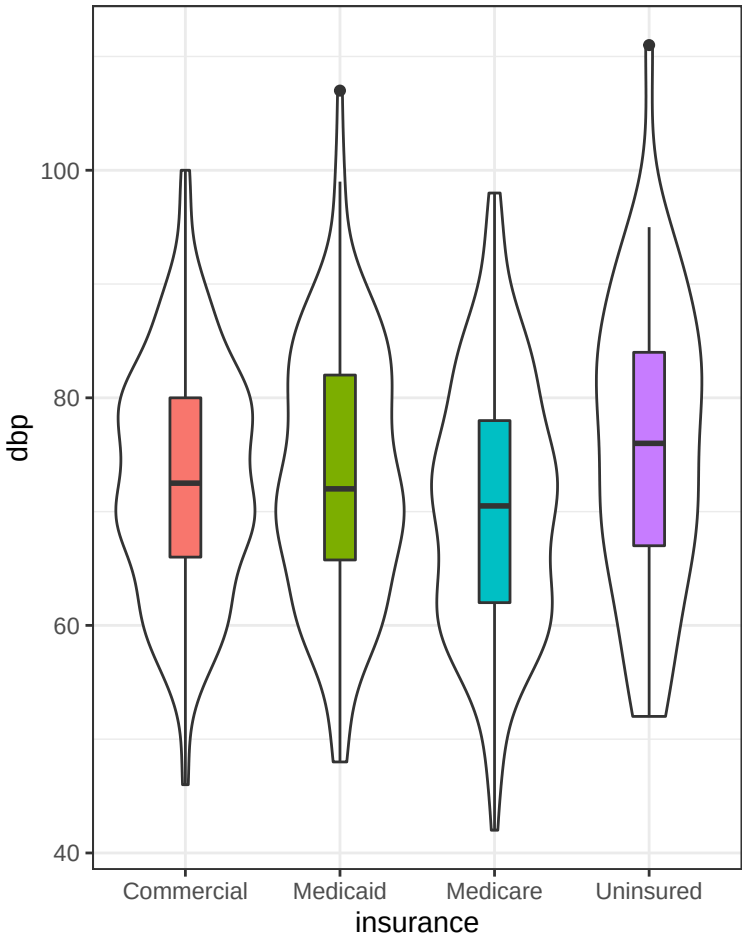
Try `dbp` as our outcome instead?

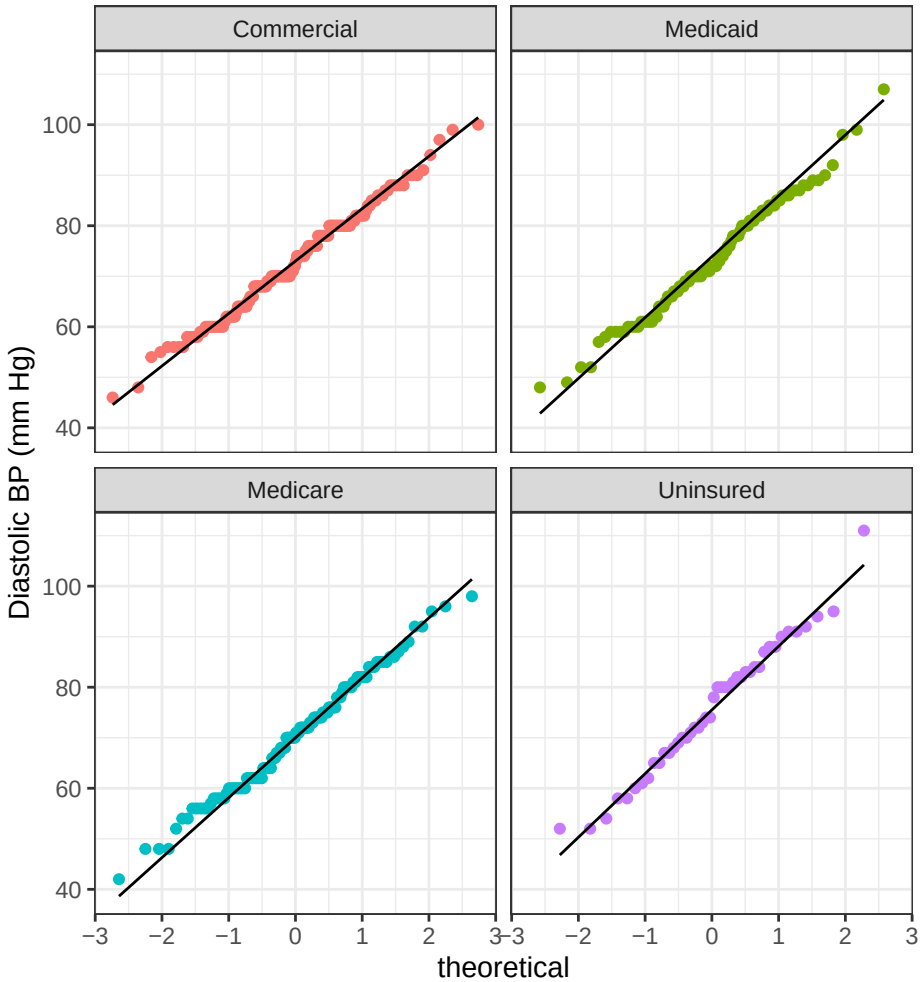
Diastolic BP by Insurance Type in dm431

```
dm431 %$% mosaic::favstats(dbp ~ insurance) %>%
  rename(na = missing) %>% kable(dig = 1)
```

insurance	min	Q1	median	Q3	max	mean	sd	n	na
Commercial	46	66.0	73.5	80	100	73.1	10.2	164	0
Medicaid	48	65.8	72.0	82	107	73.2	11.3	100	0
Medicare	42	62.0	70.0	78	98	70.3	11.1	123	0
Uninsured	52	67.0	76.0	84	111	75.9	13.0	44	0

Compare dbp across insurance types?





We'll assume MCAR and try to predict dbp

```
modD <- lm(dbp ~ insurance, data = dm431)
modD
```

```
Call:
lm(formula = dbp ~ insurance, data = dm431)

Coefficients:
       (Intercept)   insuranceMedicaid
           73.0854              0.1546
 insuranceMedicare  insuranceUninsured
           -2.7358              2.7783
```

Model D Fit Summary

```
glance(modD) %>%
  select(r.squared, statistic, df, df.residual,
         p.value, sigma, nobs) %>%
  kable(dig = c(3, 2, 0, 0, 4, 2, 0))
```

r.squared	statistic	df	df.residual	p.value	sigma	nobs
0.022	3.2	3	427	0.0234	11.05	431

What can we conclude about whether `insurance` is an effective predictor of dbp in these data?

Model D Coefficients

```
tidy(modD, conf.int = TRUE, conf.level = 0.90) %>%
  select(term, estimate, std.error, conf.low, conf.high) %>%
  kable(dig = 1)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	73.1	0.9	71.7	74.5
insuranceMedicaid	0.2	1.4	-2.2	2.5
insuranceMedicare	-2.7	1.3	-4.9	-0.6
insuranceUninsured	2.8	1.9	-0.3	5.9

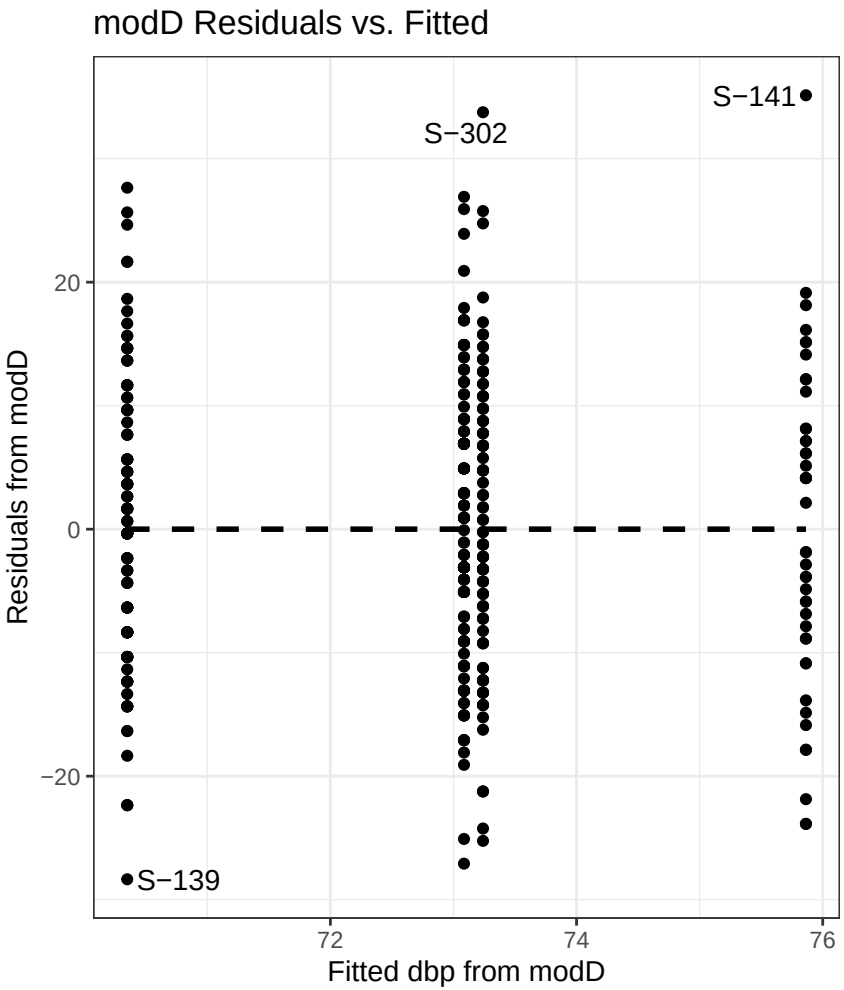
- Which insurance type is associated with the highest (worst) dbp?
- Which has the lowest predicted dbp? Are these results surprising?

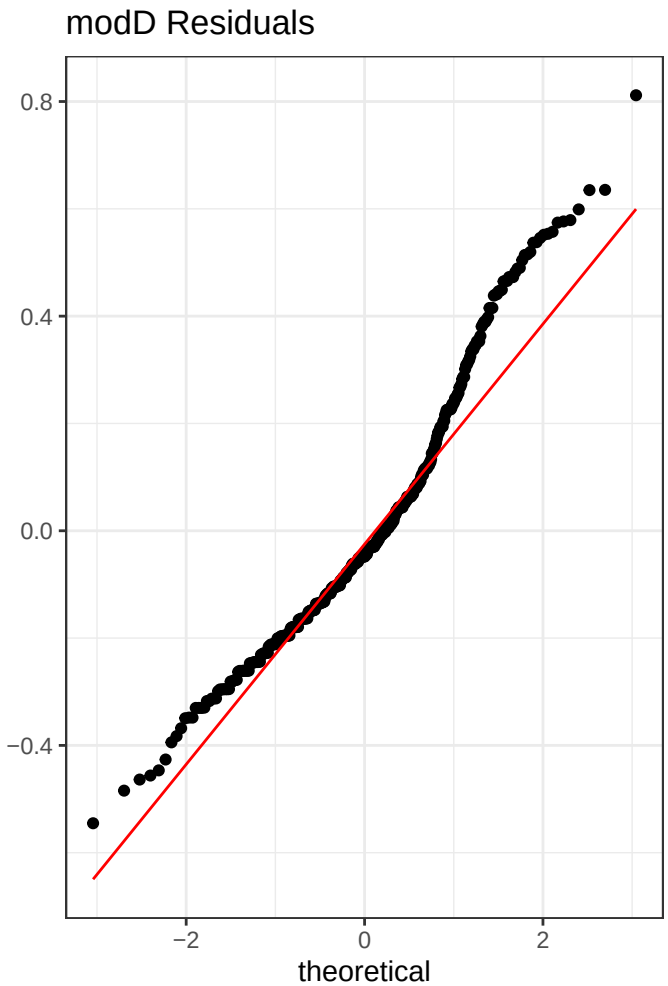
Making Predictions with augment

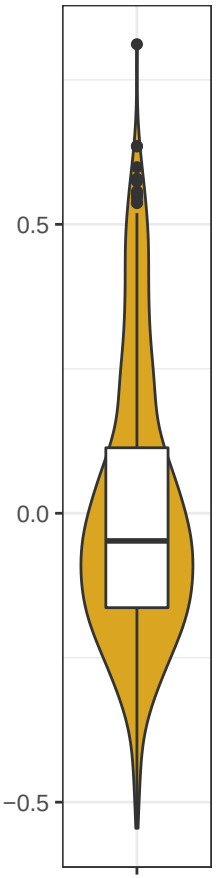
```
augD <- augment(modD, dm431)

augD %>% select(subject, insurance, dbp, .fitted, .resid) %>%
  head() %>% kable(dig = 2)
```

subject	insurance	dbp	.fitted	.resid
S-001	Commercial	64	73.09	-9.09
S-002	Uninsured	84	75.86	8.14
S-003	Uninsured	95	75.86	19.14
S-004	Commercial	87	73.09	13.91
S-005	Commercial	58	73.09	-15.09
S-006	Medicare	60	70.35	-10.35

Residual Plots for `modD`







That's the end of the material I expect you to use in Project A. Now, let's go back to Example 1. (Comparing LDL by Statin use)

Building Confidence Intervals for $\mu_1 - \mu_2$

The hypotheses we are testing are (Δ_0 is usually zero):

- H_0: $\mu_1 = \mu_2 +$ hypothesized difference Δ_0 vs.
- H_A: $\mu_1 \neq \mu_2 +$ hypothesized difference Δ_0.

Four Approaches

1. Indicator Variable Regression Model ("Pooled" t approach, or "t test" assuming equal population variances)
2. Welch t CI (t approach without assuming equal population variances)
3. Wilcoxon-Mann-Whitney Rank Sum Test (non-parametric test not assuming Normality but needing symmetry to be related to means)
4. Bootstrap confidence interval for the difference in population means (fewest assumptions of these options)

The Pooled t procedure (same as indicator variable regression)

Building a Pooled t CI

1. Best approach: use indicator variable regression
2. Also: direct call to t test with pooled variance estimate

```
t.test(ldl ~ statin, data = dm431_cc, alt = "two.sided", mu =
       var.equal = TRUE, conf.level = 0.90)
```

```
    Two Sample t-test

data:  ldl by statin
t = 0.22579, df = 392, p-value = 0.8215
alternative hypothesis: true difference in means is not equal
90 percent confidence interval:
 -6.363975  8.383644
sample estimates:
mean in group 0 mean in group 1
       97.41667        96.40683
```

t test can be tidied

```
t1 <- tidy(t.test(ldl ~ statin, data = dm431_cc,
                  var.equal = TRUE, conf.level = 0.90))
```

- `conf.level` must be specified to `t.test`. Otherwise, it uses 0.95.

Elements of `t1` are printed below (after rearrangement)

method	alternative	estimate1	estimate2
Two Sample t-test	two.sided	97.42	96.41

estimate	conf.low	conf.high	statistic	parameter	p.value
1.01	-6.36	8.38	0.23	392	0.82

- This estimates $\mu_{NoStatin} - \mu_{Statin}$. Invert the signs of the estimate and the endpoints of the CI to estimate $\mu_{Statin} - \mu_{NoStatin}$.

Assumptions of the Pooled T test

The standard method for comparing population means based on two independent samples is based on the t distribution, and requires the following assumptions:

1. [Independence] The samples for the two groups are drawn independently.
2. [Random Samples] The samples for each of the groups are drawn at random from the populations of interest.
3. [Normal Population] The two populations are each Normally distributed
4. [Equal Variances or Balanced Design] We must assume:

- Either the population variances in the two groups are the same, so a pooled estimate of their joint variance makes sense,
- OR the two samples are the same size (a balanced design.)

The Welch t procedure (t approach, not assuming equal population variances)

Assumptions of the Welch t approach

The Welch test still requires:

1. [Independence] The samples for the two groups are drawn independently.
2. [Random Samples] The samples for each of the groups are drawn at random from the populations of interest.
3. [Normal Population] The two populations are each Normally distributed

But it doesn't require:

4. [Equal Variances] The population variances in the two groups being compared are the same. (for instance, Welch's test still works if the larger variance σ_1^2 is more than 1.5 times as large as σ_2^2).

- If the design is balanced ($n_1 = n_2$) or nearly so, the impact of assuming equal variances is minimal.

Welch's t test is the default `t.test` in R.

Building the Welch t CI

- Sensible approach when assuming Normal populations is OK, but we don't want to assume the two populations have the same variance (as pooled t requires)

```
t.test(ldl ~ statin, data = dm431_cc, alt = "two.sided", mu =
       conf.level = 0.90)
```

```
    Welch Two Sample t-test

data:  ldl by statin
t = 0.25455, df = 122.11, p-value = 0.7995
alternative hypothesis: true difference in means is not equal
90 percent confidence interval:
 -5.565462  7.585131
sample estimates:
mean in group 0 mean in group 1
       97.41667        96.40683
```

Welch `t` test can also be tidied

```
t2 <- tidy(t.test(ldl ~ statin, data = dm431_cc,
                  conf.level = 0.90))
```

- We must specify `conf.level` in the `t.test` unless we want 0.95.

Elements of `t2` are printed below (after rearrangement)

method	alternative	estimate1	estimate2
Welch Two Sample t-test	two.sided	97.42	96.41

estimate	conf.low	conf.high	statistic	parameter	p.value
1.01	-5.57	7.59	0.25	122.11	0.8

- Invert signs of estimate and CI limits to get $\mu_{Statin} - \mu_{No}$.

The Wilcoxon-Mann-Whitney Rank Sum procedure

Wilcoxon-Mann-Whitney Rank Sum Approach

The Wilcoxon-Mann-Whitney Rank Sum procedure requires:

1. [Independence] The samples for the two groups are drawn independently.
2. [Random Samples] The samples for each of the groups are drawn at random from the populations of interest.
3. [Symmetry] The two populations are each symmetrically distributed, and as a result, we're comfortable estimating the shift in location (measured by the pseudo-medians) rather than a shift in means.

But it doesn't require:

3. [Normal Population] The two populations are each Normally distributed
4. [Equal Variances] The population variances in the two groups being compared are the same.

As mentioned, it doesn't really compare population means, but instead pseudo-medians.

Wilcoxon-Mann-Whitney Rank Sum Approach

```
wilcox.test(ldl ~ statin, data = dm431_cc,
            conf.int = TRUE, conf.level = 0.90)
```

```
    Wilcoxon rank sum test with continuity
    correction

data:  ldl by statin
W = 12560, p-value = 0.2683
alternative hypothesis: true location shift is not equal to 0
90 percent confidence interval:
 -2.000009 11.000062
sample estimates:
difference in location
              4.629345
```

Rank Sum test can also be tidied

```
w3 <- tidy(wilcox.test(ldl ~ statin, data = dm431_cc,
                       conf.int = TRUE, conf.level = 0.90))
```

- Specify `conf.int` and `conf.level` in the `wilcox.test`.

Elements of `w3` are printed below (after rearrangement)

method	alternative	statistic
Wilcoxon rank sum test with continuity correction	two.sided	12559.5

estimate	conf.low	conf.high	p.value
4.63	-2	11	0.27

- Invert signs of estimate and CI to describe shift from No to Statin.

The Bootstrap

This bootstrap approach to comparing population means using two independent samples still requires:

1. [Independence] The samples for the two groups are drawn independently.
2. [Random Samples] The samples for each of the groups are drawn at random from the populations of interest.

but does not require either of the other two assumptions:

3. [Normal Population] The two populations are each Normally distributed
4. [Equal Variances] The population variances in the two groups being compared are the same.

The bootstrap procedure I use in R was adapted from Frank Harrell and colleagues. http://biostat.mc.vanderbilt.edu/twiki/bin/view/Main/BootstrapMeansSoftware

The `bootdif` function

The procedure requires the definition of a function, which I have adapted a bit, called `bootdif`, which is part of the `Love-boost.R` script we loaded earlier.

As in our previous bootstrap procedures, we are sampling (with replacement) a series of many data sets (default: 2000).

- Here, we are building bootstrap samples based on the LDL levels in the two independent samples (statin users vs. non-users.)
- For each bootstrap sample, we are calculating a mean difference between the two groups (statin vs. no statin.)
- We then determine the 2.5th and 97.5th percentile of the resulting distribution of mean differences (for a 95% confidence interval).

Using `bootdif` to compare mean(LDL) by statin

So, to compare LDL (our outcome) across the two levels of statin (our grouping factor) for the adult patients with diabetes in NE Ohio, run. . .

```
set.seed(20201008)
boot4 <- dm431_cc %$% bootdif(ldl, statin, conf.level = 0.90)
boot4
```

```
Mean Difference              0.05              0.95
      -1.009834        -7.580275         5.326272
```

- The two columns must be separated here with a comma rather than a tilde (~), and are specified using $ notation.
- This CI estimates $\mu_{Statin} - \mu_{NoStatin}$. Observe the listed sample mean difference for the necessary context.
- If we change the `set.seed`, we'll get different endpoints for our CI.
- Note that we can infer the p value is above 0.10 from the CI. Why?

Results for the LDL and Statin Study

Procedure	p for $H_0 : \mu_S = \mu_N$	90% CI for $\mu_S - \mu_N$
Pooled t	0.82	(-8.4, 6.4)
Welch t	0.8	(-7.6, 5.6)
Bootstrap	$p > 0.100$	(-7.6, 5.3)

Procedure	p for $H_0 : psmed_S = psmed_N$	90% CI for S - N shift
Rank Sum	0.27	(-11, 2)

Which method should we use?

Which Method Should We Use?

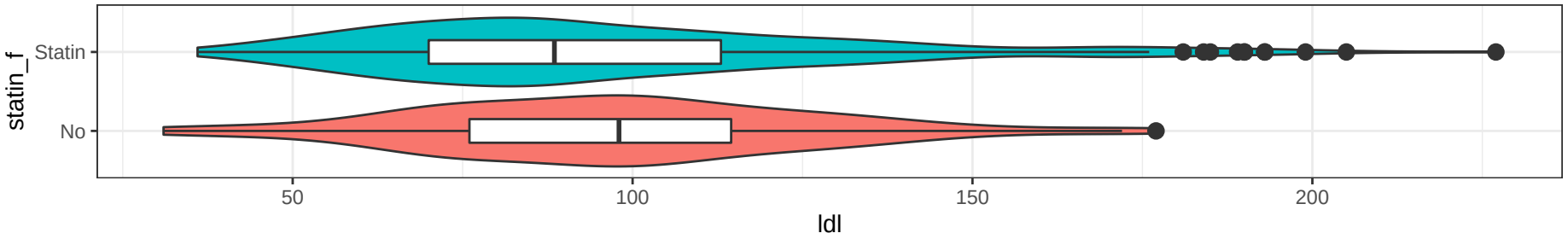
1. Plot the distributions of the two independent samples.
2. Does it seem reasonable to assume that **each** distribution (here, both `ldl` in statin users and `ldl` in non-users) follows an approximately Normal distribution?

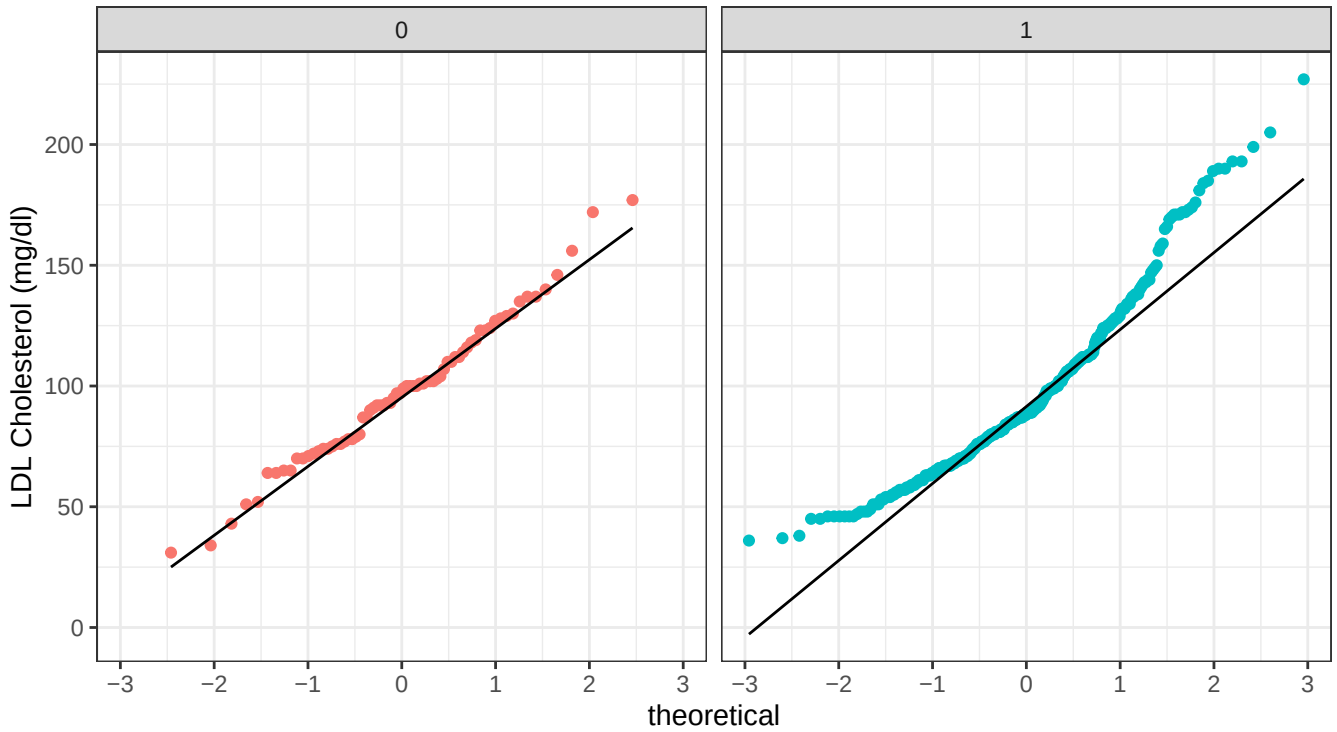
- If Yes, Normal models seem fairly appropriate, then
  - use the indicator variable regression (pooled t test) if the sample sizes are nearly the same, or if the sample variances are reasonably similar
  - use the Welch's t test, otherwise (default `t.test` in R)
- If No, Normal models don't seem appropriate at all, then
  - compare means using the bootstrap via `bootdif`, or
  - compare pseudo-medians using the WMW rank sum test

What did we see in our `ldl` data?

LDL, within groups defined by `statin`

Example 1. Comparing LDL by Statin Use in our dm431 complete cases (n = 394)





Results for the LDL and Statin Study

Procedure	p for $H_0 : \mu_S = \mu_N$	90% CI for $\mu_S - \mu_N$
Pooled t	0.82	(-8.4, 6.4)
Welch t	0.8	(-7.6, 5.6)
Bootstrap	$p > 0.100$	(-7.6, 5.3)

Procedure	p for $H_0 : psmed_S = psmed_N$	90% CI for S - N shift
Rank Sum	0.27	(-11, 2)

What conclusions should we draw, at $\alpha = 0.10$?

Example 3 (Comparing SBP by Sex) slides follow, for you to review on your own. It's very much like Example 1. The main difference is that we have no missing values in SBP or Sex in the `dm431` data.

dm431 Example 3. (Comparing SBP by Sex)

Estimate the difference in population mean systolic blood pressure among females as compared to males.

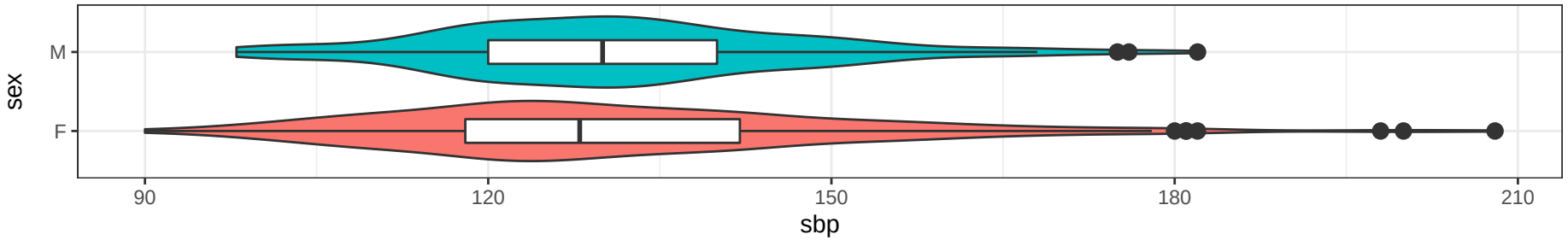
```
mosaic::favstats(sbp ~ sex, data = dm431) %>% kable(dig = 2)
```

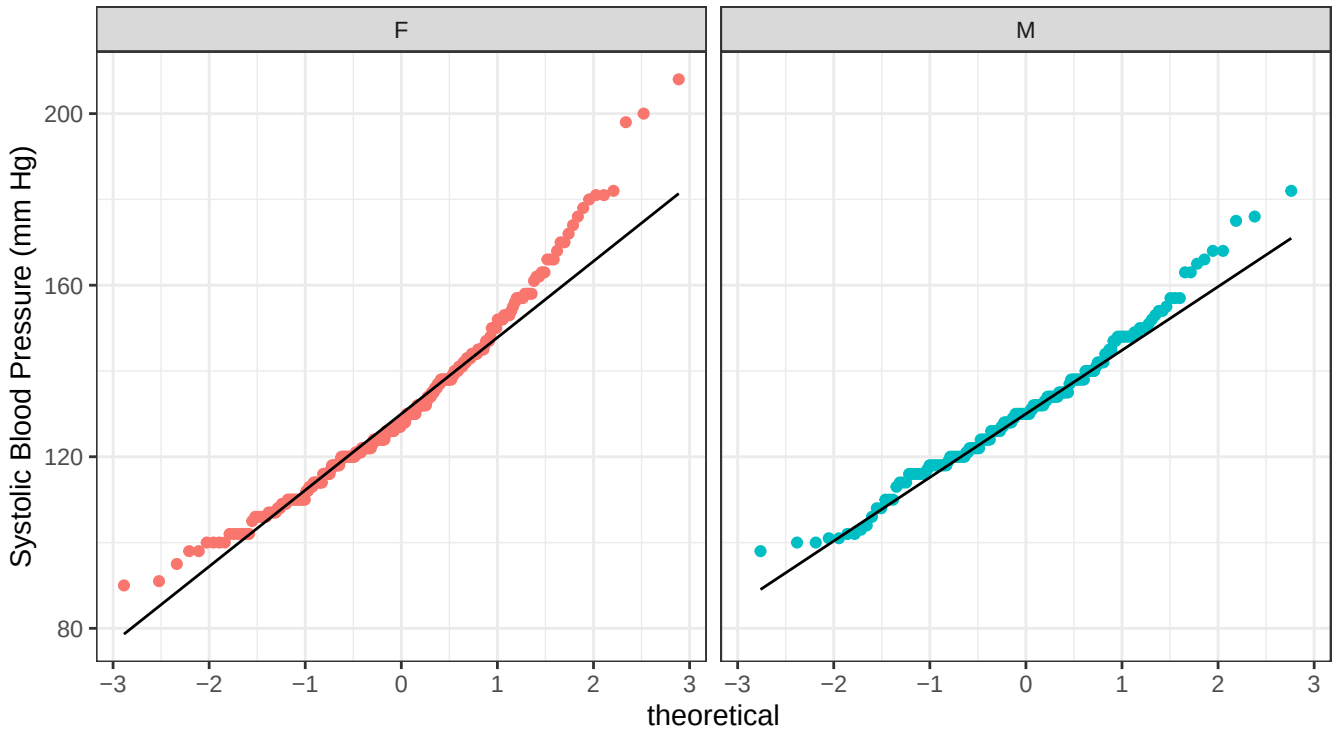
sex	min	Q1	median	Q3	max	mean	sd	n	missing
F	90	118	128	142	208	131.17	20.15	257	0
M	98	120	130	140	182	131.41	15.87	174	0

- What is the outcome here?
- What are the two exposure groups we are comparing?
- What are the sample means, $\bar{x}_F$ and $\bar{x}_M$?
- Point estimate of the difference in population means, $\mu_F - \mu_M$?

DTDP for Example 3. (Comparing SBP by Sex)

Example 3. Comparing SBP by sex in our dm431 data





Linear Model for Example 3 (slide A)

Estimate the difference in population mean systolic blood pressure among females as compared to males.

```
m1 <- lm(sbp ~ sex, data = dm431)

tidy(m1, conf.int = T, conf.level = 0.90) %>% kable(dig = 2)
```

term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	131.17	1.16	113.41	0.00	129.26	133.07
sexM	0.24	1.82	0.13	0.89	-2.76	3.24

- What can we learn from this output?
  - What is the sample mean `sbp` for females?
  - What is the sample mean `sbp` for males?
  - The point estimate for $\mu_F - \mu_M$ is ...

Linear Model for Example 3 (slide B)

Estimate the difference in population mean systolic blood pressure among females as compared to males.

```
m1 <- lm(sbp ~ sex, data = dm431)

tidy(m1, conf.int = T, conf.level = 0.90) %>% kable(dig = 2)
```

term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	131.17	1.16	113.41	0.00	129.26	133.07
sexM	0.24	1.82	0.13	0.89	-2.76	3.24

- What can we learn from this output?
  - The point estimate for $\mu_F - \mu_M$ is -0.24
  - The 90% confidence interval for $\mu_F - \mu_M$ is (-**3.24**, **2.76**)

Building a Pooled t CI: Example 3

1. Best approach: use indicator variable regression
2. Also: direct call to t test with pooled variance estimate

```
t.test(sbp ~ sex, data = dm431, alt = "two.sided", mu = 0,
       var.equal = TRUE, conf.level = 0.90)
```

```
    Two Sample t-test

data:  sbp by sex
t = -0.13225, df = 429, p-value = 0.8949
alternative hypothesis: true difference in means is not equal
90 percent confidence interval:
 -3.241344  2.759883
sample estimates:
mean in group F mean in group M
       131.1673        131.4080
```

t test can be tidied

```
t1 <- tidy(t.test(sbp ~ sex, data = dm431,
                  var.equal = TRUE, conf.level = 0.90))
```

- `conf.level` must be specified to `t.test`. Otherwise, it uses 0.95.

Elements of `t1` are printed below (after rearrangement)

method	alternative	estimate1	estimate2
Two Sample t-test	two.sided	131.17	131.41

estimate	conf.low	conf.high	statistic	parameter	p.value
-0.24	-3.24	2.76	-0.13	429	0.89

Building the Welch t CI: Example 3

- Sensible approach when assuming Normal populations is OK, but we don't want to assume the two populations have the same variance (as pooled t requires)

```
t.test(sbp ~ sex, data = dm431, alt = "two.sided", mu = 0,
       conf.level = 0.90)
```

```
    Welch Two Sample t-test

data:  sbp by sex
t = -0.13838, df = 419.27, p-value = 0.89
alternative hypothesis: true difference in means is not equal
90 percent confidence interval:
 -3.108582  2.627120
sample estimates:
mean in group F mean in group M
       131.1673        131.4080
```

Welch t test can also be tidied

```
t2 <- tidy(t.test(sbp ~ sex, data = dm431,
                  conf.level = 0.90))
```

- We must specify `conf.level` in the `t.test` unless we want 0.95.

Elements of `t2` are printed below (after rearrangement)

method	alternative	estimate1	estimate2
Welch Two Sample t-test	two.sided	131.17	131.41

estimate	conf.low	conf.high	statistic	parameter	p.value
-0.24	-3.11	2.63	-0.14	419.27	0.89

Wilcoxon-Mann-Whitney Rank Sum: Example 3

```
wilcox.test(sbp ~ sex, data = dm431,
            conf.int = TRUE, conf.level = 0.90)
```

```
    Wilcoxon rank sum test with continuity
    correction

data:  sbp by sex
W = 21329, p-value = 0.4167
alternative hypothesis: true location shift is not equal to 0
90 percent confidence interval:
 -4.000050  1.999972
sample estimates:
difference in location
              -1.99992
```

Rank Sum test can also be tidied

```
w3 <- tidy(wilcox.test(sbp ~ sex, data = dm431,
                       conf.int = TRUE, conf.level = 0.90))
```

- Specify `conf.int` and `conf.level` in the `wilcox.test`.

Elements of `w3` are printed below (after rearrangement)

method	alternative	statistic
Wilcoxon rank sum test with continuity correction	two.sided	21328.5

estimate	conf.low	conf.high	p.value
-2	-4	2	0.42

Using `bootdif` to compare mean(SBP) by Sex

So, to compare systolic BP (our outcome) across the two levels of sex (our grouping factor) for the adult patients with diabetes in NE Ohio, run. . .

```
set.seed(431431)
boot4 <- dm431 %$% bootdif(sbp, sex, conf.level = 0.90)
boot4
```

```
Mean Difference            0.05            0.95
      0.2407308      -2.6226195       3.1868901
```

- This CI estimates $\mu_M - \mu_F$: observe the listed sample mean difference for the necessary context.
- Invert the signs to estimate $\mu_F - \mu_M$.
- Again the p value must be larger than 0.10 since 0 is in the 90% CI.

Results for the SBP and Sex Study

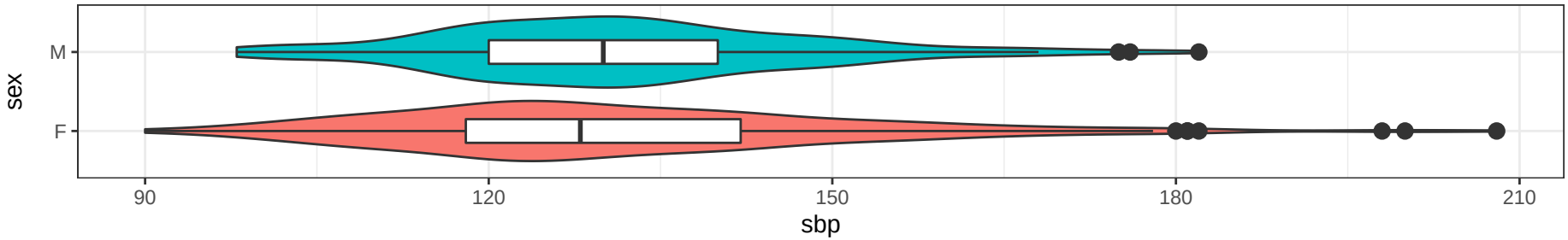
Procedure	p for $H_0 : \mu_F = \mu_M$	90% CI for $\mu_F - \mu_M$
Pooled t	0.8949	(-3.24, 2.76)
Welch t	0.89	(-3.11, 2.63)
Bootstrap	$p > 0.100$	(-3.19, 2.62)

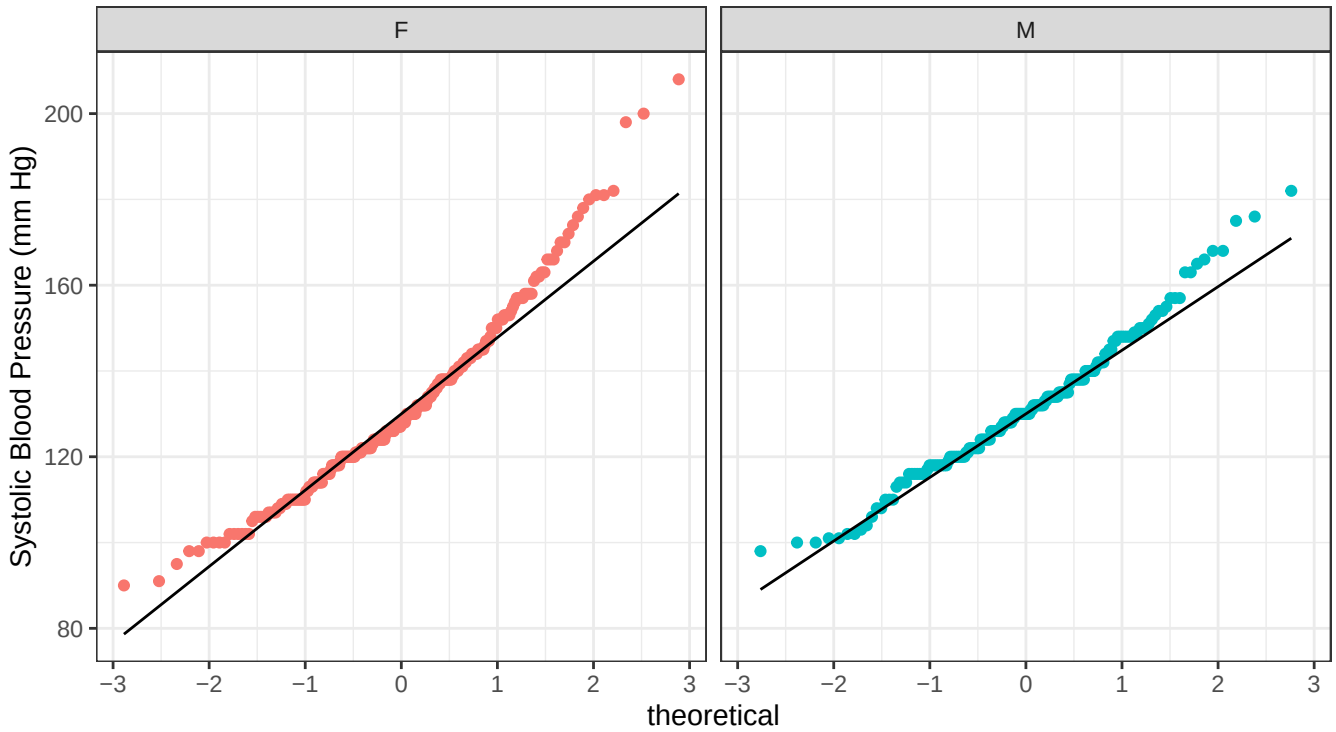
Procedure	p for $H_0 : psmed_F = psmed_M$	90% CI for F - M shift
Rank Sum	0.4167	(-4, 2)

Which method should we use?

Systolic BP, within groups defined by sex

Example 3. Comparing SBP by sex in our dm431 data





Results for the SBP and Sex Study

Procedure	p for $H_0 : \mu_F = \mu_M$	90% CI for $\mu_F - \mu_M$
Pooled t	0.8949	(-3.24, 2.76)
Welch t	0.89	(-3.11, 2.63)
Bootstrap	$p > 0.100$	(-3.19, 2.62)

Procedure	p for $H_0 : psmed_F = psmed_M$	90% CI for F - M shift
Rank Sum	0.4167	(-4, 2)

What conclusions should we draw, at $\alpha = 0.10$?