# 431 Class 11

github.com/THOMASELOVE/2019-431

2019-10-01

# Today's Agenda (Notes Chapters 16-17)

1. Working in an R Project, loading data with `here`
2. Statistical Inference and the `dm431` data
   - Point Estimates and Confidence Intervals for a Population's Mean
3. Group Work on Project Study A Proposal

- My Class 11 project directory has `R` and `data` subdirectories.
  - In the `R` subdirectory, I have the `Love-boost.R` script.
  - In the `data` subdirectory, I have `dm431.csv`, `dm431.xlsx`, and `dm431.Rds`.

# Methods for Loading/Saving Data

# Today's Packages

```
library(magrittr); library(janitor)
library(patchwork); library(here)
library(readxl); library(broom)
library(tidyverse)

source(here("R", "Love-boost.R"))
```

# Load the Data: Approach A

Load the data using `read_csv`. This renders all categorical variables as characters.

```
dm431a <- read_csv(here("data", "dm431.csv")) %>%
  clean_names(case = "upper_camel")

dm431a %>% slice(1:3) %>%
    select(Subject, Practice, Insurance, A1C, Sbp, SbpOld)
```

```
# A tibble: 3 x 6
  Subject Practice  Insurance     A1C   Sbp SbpOld
  <chr>   <chr>     <chr>       <dbl> <dbl>  <dbl>
1 S-001   Arlington Commercial    6.3    98    102
2 S-002   Bristol   Uninsured    11     162    100
3 S-003   Chester   Uninsured     8.7   154    138
```

# Load the Data: Approach B

Can we add to `read_csv` to get all variables that are imported as characters (except `subject`) changed to factors?

```
dm431b <- read_csv(here("data", "dm431.csv")) %>%
    clean_names(case = "snake") %>%
    mutate_if(is.character, as.factor) %>%
    mutate(subject = as.character(subject))

dm431b %>% slice(1:3) %>%
    select(subject, practice, insurance, a1c, sbp, sbp_old)
```

```
# A tibble: 3 x 6
  subject practice  insurance    a1c   sbp sbp_old
  <chr>   <fct>     <fct>      <dbl> <dbl>   <dbl>
1 S-001   Arlington Commercial   6.3    98     102
2 S-002   Bristol   Uninsured   11     162     100
3 S-003   Chester   Uninsured    8.7   154     138
```

# Saving the Approach B result

```
names(dm431b)
```

```
 [1] "subject"       "practice"      "age"
 [4] "race_eth"      "sex"           "a1c"
 [7] "insurance"     "income"        "sbp"
[10] "dbp"           "tobacco"       "ldl"
[13] "statin"        "a1c_old"       "insurance_old"
[16] "income_old"    "sbp_old"       "dbp_old"
[19] "tobacco_old"   "ldl_old"       "statin_old"
```

I like this version. Let's save it, to an `Rds` file.

```
saveRDS(dm431b, here("data", "dm431.Rds"))
```

# Load the Data: Approach C

We can use the `read.csv` followed by `tbl_df` approach to get all categorical variables imported as factors first.

```
dm431c <- read.csv("data/dm431.csv") %>%
    tbl_df %>%
    clean_names(case = "all_caps") %>%
    mutate(SUBJECT = as.character(SUBJECT))

dm431c %>% slice(1:3) %>%
    select(SUBJECT, PRACTICE, INSURANCE, A1C, SBP, SBP_OLD)
```

```
# A tibble: 3 x 6
  SUBJECT PRACTICE  INSURANCE    A1C   SBP SBP_OLD
  <chr>   <fct>     <fct>      <dbl> <int>   <int>
1 S-001   Arlington Commercial   6.3    98     102
2 S-002   Bristol   Uninsured   11     162     100
3 S-003   Chester   Uninsured    8.7   154     138
```

# Load the Data: Approach D

We can use the `read_xlsx` function from the `readxl` package to import directly from an Excel spreadsheet (.xlsx file).

```
dm431d <- read_xlsx("data/dm431.xlsx") %>%
  clean_names() # default is clean_names(case = "snake")

dm431d %>% slice(1:3) %>%
    select(subject, practice, insurance, a1c, sbp, sbp_old)
```

```
# A tibble: 3 x 6
  subject practice  insurance a1c           sbp sbp_old
  <chr>   <chr>     <chr>     <chr>       <dbl>   <dbl>
1 S-001   Arlington Commerci~ 6.3            98     102
2 S-002   Bristol   Uninsured 11            162     100
3 S-003   Chester   Uninsured 8.69999999~   154     138
```

Note what happens to the Hemoglobin A1c value for subject S-003.

# Approach E: Reading in a saved .Rds

We can also read in the .Rds file (R data set) with factors enabled properly that we built earlier in R, and saved with saveRDS.

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

dm431 %>% slice(1:3) %>%
    select(subject, practice, insurance, a1c, sbp, sbp_old)
```

```
# A tibble: 3 x 6
  subject practice  insurance    a1c   sbp sbp_old
  <chr>   <fct>     <fct>      <dbl> <dbl>   <dbl>
1 S-001   Arlington Commercial   6.3    98     102
2 S-002   Bristol   Uninsured   11     162     100
3 S-003   Chester   Uninsured    8.7   154     138
```

That's the version we'll use.

# How do we load in other types of files?

Take a look at the RStudio Data Import Cheat Sheet

- `readr` package (part of the core tidyverse) helps with rectangular data sets from .csv, .tsv, .fxf (fixed width files), web logs, tabular files, and other delimited files.
- `haven` to read in SPSS, Stata and SAS files
- `readxl` for excel files (.xls as well as .xlsx)
- `DBI` for databases
- `xml2` for XML
- `httr` for Web APIs
- `rvest` for HTML (Web scraping)

# Describing the `sbp` data within `dm431`

# Systolic Blood Pressure in the dm431 data

Here, I will look at systolic blood pressure values from a sample of 431 adult patients living in Northeast Ohio between the ages of 31 and 70, who have a diagnosis of diabetes, as gathered in the `dm431.csv` data file.

- These data are simulated to mirror some details from real data gathered by *Better Health Partnership*.
- The `dm431` data contains multitudes, but for now, we're just looking at 431 systolic blood pressure values, gathered in the `sbp` variable.

**In the Course Notes (See Chapters 16-18)**

I don't use the `dm431` data in the Part B notes. In Chapter 16 I look at a study of serum zinc levels, and then, I present methods for estimating first a population mean (Chapter 17), and then a population proportion (Chapter 18) from those data. That's what we'll do this week.

# Summarizing sbp in the dm431 data

Today, we're focused on our sample of 431 systolic blood pressure values captured at a particular moment in time.

```
mosaic::favstats(~ sbp, data = dm431)
```

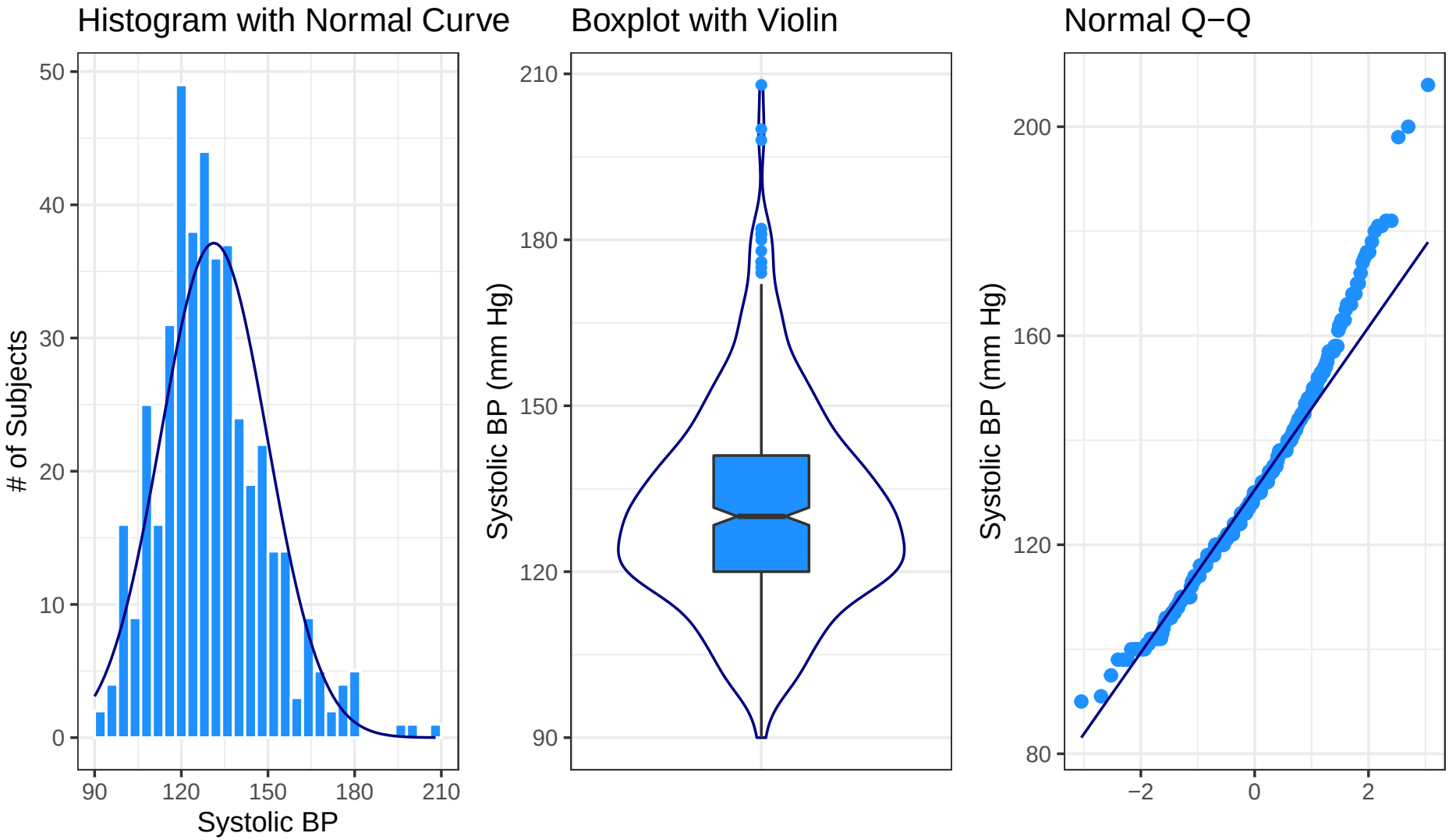
| min | Q1 | median | Q3 | max | mean | sd | n | missing |
|---|---|---|---|---|---|---|---|---|
| 90 | 120 | 130 | 141 | 208 | 131.2645 | 18.52038 | 431 | 0 |

The next slide provides some key graphical displays of the sbp data.

- Does a Normal model seem reasonable for these data?

# Graphical Summaries: sbp in dm431

Systolic BP (mm Hg) for 431 NE Ohio Adults with Diabetes



# Fundamentals of Statistical Inference

# Something Happened! Is this Signal or Noise?

Very often, sample data indicate that something has happened. . .

- the proportion of people who respond to this treatment has changed
- the mean value of this measure appears to have changed

Before we get too excited, it's worth checking whether the apparent result might possibly be the result of random sampling error.

Statistics provides a number of tools for reaching an informed choice (informed by sample information, of course) including confidence intervals and hypothesis tests (p values), in particular.

# Key Questions: Making Inferences From A Sample

1. What is the population about which we aim to make an inference?
2. What is the sample available to us to make that inference?
    - Who are the individuals fueling our inference?
    - What data are available to make an inference?
3. Why might this sample not represent the population?

# Point Estimation and Confidence Intervals

The basic theory of estimation can be used to indicate the probable accuracy and potential for bias in estimating based on limited samples.

- A **point estimate** provides a single best guess as to the value of a population or process parameter.
- A **confidence interval** can convey how much error one must allow for in a given estimate.

The key tradeoffs are

- cost vs. precision, and
- precision vs. confidence in the correctness of the statement.

Often, if we are dissatisfied with the width of the confidence interval and want to make it smaller, we have to reconsider the sample – larger samples produce shorter intervals.

# Defining a Confidence Interval

A confidence interval for a population or process mean uses data from a sample (and perhaps some additional information) to identify a range of potential values for the population mean, which, if certain assumptions hold, can be assumed to provide a reasonable estimate for the true population mean.

A confidence interval consists of:

1. An interval estimate describing the population parameter of interest (here the population mean), and
2. A probability statement, expressed in terms of a confidence level.

# Our Goal in this Situation

Suppose that we are willing to assume that the systolic blood pressures across the entire population of NE Ohio adults ages 31-70 living with diabetes follows a Normal distribution (and so, summarizing it with a mean, called $\mu$, is a rational choice.)

Suppose that we are also willing to assume that the 431 adults contained in the `dm431` tibble are a random sample from that complete population. While we know the sample mean of these 431 adults, we don't know $\mu$, the mean across **all** NE Ohio adults ages 31-70 living with diabetes. So we need to estimate it.

Our first inferential goal will be to produce a **confidence interval for the true (population) mean** of all adults with diabetes ages 31-70 living in NE Ohio based on this sample.

# Starting with An Answer

```
model1 <- lm(sbp ~ 1, data = dm431)

tidy(model1, conf.int = TRUE, conf.level = 0.90) %>%
  select(term, estimate, std.error, conf.low, conf.high) %>%
  knitr::kable(digits = 2)
```

| term | estimate | std.error | conf.low | conf.high |
|---|---|---|---|---|
| (Intercept) | 131.26 | 0.89 | 129.79 | 132.74 |

- Our point estimate for the population mean SBP ($\mu$) is 131.26 mm Hg.
- Our 90% confidence interval is (129.79, 132.74) mm Hg for $\mu$.

# A 90% Confidence Interval for $\mu$

- Our 90% confidence interval estimate for $\mu$ turns out to be (129.79, 132.74) mm Hg. How do we interpret this result?
- Some people think this means that there is a 90% chance that the true mean of the population, $\mu$, falls between 129.79 and 132.74 mm Hg.

# A 90% Confidence Interval for $\mu$

- Our 90% confidence interval estimate for $\mu$ turns out to be (129.79, 132.74) mm Hg. How do we interpret this result?
- Some people think this means that there is a 90% chance that the true mean of the population, $\mu$, falls between 129.79 and 132.74 mm Hg.
- That's not correct. Why not?

# A 90% Confidence Interval for $\mu$

- Our 90% confidence interval estimate for $\mu$ turns out to be (129.79, 132.74) mm Hg. How do we interpret this result?
- Some people think this means that there is a 90% chance that the true mean of the population, $\mu$, falls between 129.79 and 132.74 mm Hg.
- That's not correct. Why not?
- The population mean $\mu$ is a constant **parameter** of the population of interest. That constant is not a random variable, and does not change. So the actual probability of the population mean falling inside that range is either 0 or 1.

# So what do we have confidence in?

Our confidence is in our process.

- It's in the sampling method (random sampling) used to generate the data, and in the assumption that the population follows a Normal distribution.
- It's captured in our accounting for one particular type of error (called *sampling error*) in developing our interval estimate, while assuming all other potential sources of error are negligible.

So what is a more appropriate interpretation of our 90% confidence interval for $\mu$?

# A somewhat better interpretation

- Our 90% confidence interval for $\mu$ is (129.79, 132.74) mm Hg.

If we used this same method to sample data from the true population of adults ages 31-70 with diabetes in NE Ohio and built 100 such 90% confidence intervals, then 90 of them would contain the true population mean. We don't know whether this one interval we built contains $\mu$, though.

- We call 100(1 - $\alpha$)%, here, 90%, or 0.90, the *confidence* level, and
- $\alpha = 10\%$, or 0.10 is called the *significance* level.

If we had instead built a series of 100 different 95% confidence intervals, then about 95 of them would contain the true value of $\mu$.

# Available Methods

To build a point estimate and confidence interval for the population mean, we could use

1. A **t-based** estimate and confidence interval, available from an intercept-only linear model, or (equivalently) from a t test.
    - This approach will require an assumption that the population comes from a Normal distribution.
2. A **bootstrap** confidence interval, which uses resampling to estimate the population mean.
    - This approach won't require the Normality assumption, but has some other constraints.
3. A **Wilcoxon signed rank** approach, but that won't describe the mean, only a pseudo-median.
    - This also doesn't require the Normality assumption, but no longer describes the population mean unless the data can at least be assumed to be symmetric.

# Population Mean Estimation using the t distribution

What do we need? (Besides a computer running R.)

1. An assumption that the data in our sample come from a population that follows a Normal distribution.
2. An assumption that random sampling from the population is a good model for how the data were collected.
    - We assume samples were taken from the population independently, and they have identical distributions.
3. A pre-specified confidence level 100*(1 - $\alpha$) for our confidence interval.
4. The sample itself, to determine the sample size $n$ (of non-missing values), the sample mean $\bar{x}$ and the sample standard deviation $s_x$.
    - These will let us calculate:
        - our point estimate of the population mean $\mu$
        - the standard error of the sample mean
        - the margin of error (half-width) of our confidence interval

# Building a 90% Confidence Interval for $\mu$

- If we want 90% confidence, this means 100*(1 - $\alpha$) = 90, and thus we have our significance level $\alpha = 0.10$.
- Our point estimate of the population mean $\mu$ is the sample mean $\bar{x}$.
- We'll also need the sample size, $n = 431$, and the sample standard deviation $s_x$.

```
mosaic::favstats(~ sbp, data = dm431)
```

```
 min  Q1 median  Q3 max     mean       sd   n missing
  90 120    130 141 208 131.2645 18.52038 431       0
```

So $\bar{x} = 131.26$ and $s_x = 18.52$, and we have $\alpha = 0.10$.

# The Standard Error of a Sample Mean

The standard error, generally, is the name we give to the standard deviation associated with any particular parameter estimate.

- If we are using a sample mean based on a sample of size $n$ to estimate a population mean, the **standard error of that sample mean** is $\sigma/\sqrt{n}$, where $\sigma$ is the standard deviation of the measurements in the population.
- We often estimate this particular standard error with its sample analogue, $s_x/\sqrt{n}$, where $s_x$ is the sample standard deviation.
- Other statistics have different standard errors.
  - For $p$, the sample proportion, $\sqrt{p(1-p)/n}$ is the standard error using a sample of size $n$.
  - For $r$, the sample Pearson correlation, $\sqrt{\frac{1-r^2}{n-2}}$ is the standard error using $n$ pairs of observations.

# Standard Error of the Mean for the SBP data

```
dm431 %$% psych::describe(sbp) %>%
  select(n, mean, sd, se)
```

```
     n   mean    sd   se
X1 431 131.26 18.52 0.89
```

The standard deviation of the SBP data turns out to be 18.52, with $n = 431$ observations, so we estimate the standard error of the mean is

$$SE_{mean}(\text{SBP}) = \frac{SD(\text{SBP})}{\sqrt{n}} = \frac{18.52}{\sqrt{431}} = 0.89$$

This standard error will play an important role in the development of our confidence interval using the t distribution.

# Confidence Interval for a population mean

We can build a 100(1-$\alpha$)% confidence interval using the $t$ distribution, using the sample mean $\bar{x}$, the sample size $n$, and the sample standard deviation $s_x$. The two-sided 100(1-$\alpha$)% confidence interval (based on a $t$ test) is:

$$\bar{x} \pm t_{\alpha/2,n-1}(\frac{s_x}{\sqrt{n}})$$

where $t_{\alpha/2,n-1}$ is the value that cuts off the top $\alpha/2$ percent of the $t$ distribution, with $n-1$ degrees of freedom.

We obtain the relevant cutoff value in R by substituting in values for `alphaover2` and `n-1` into the following line of R code:

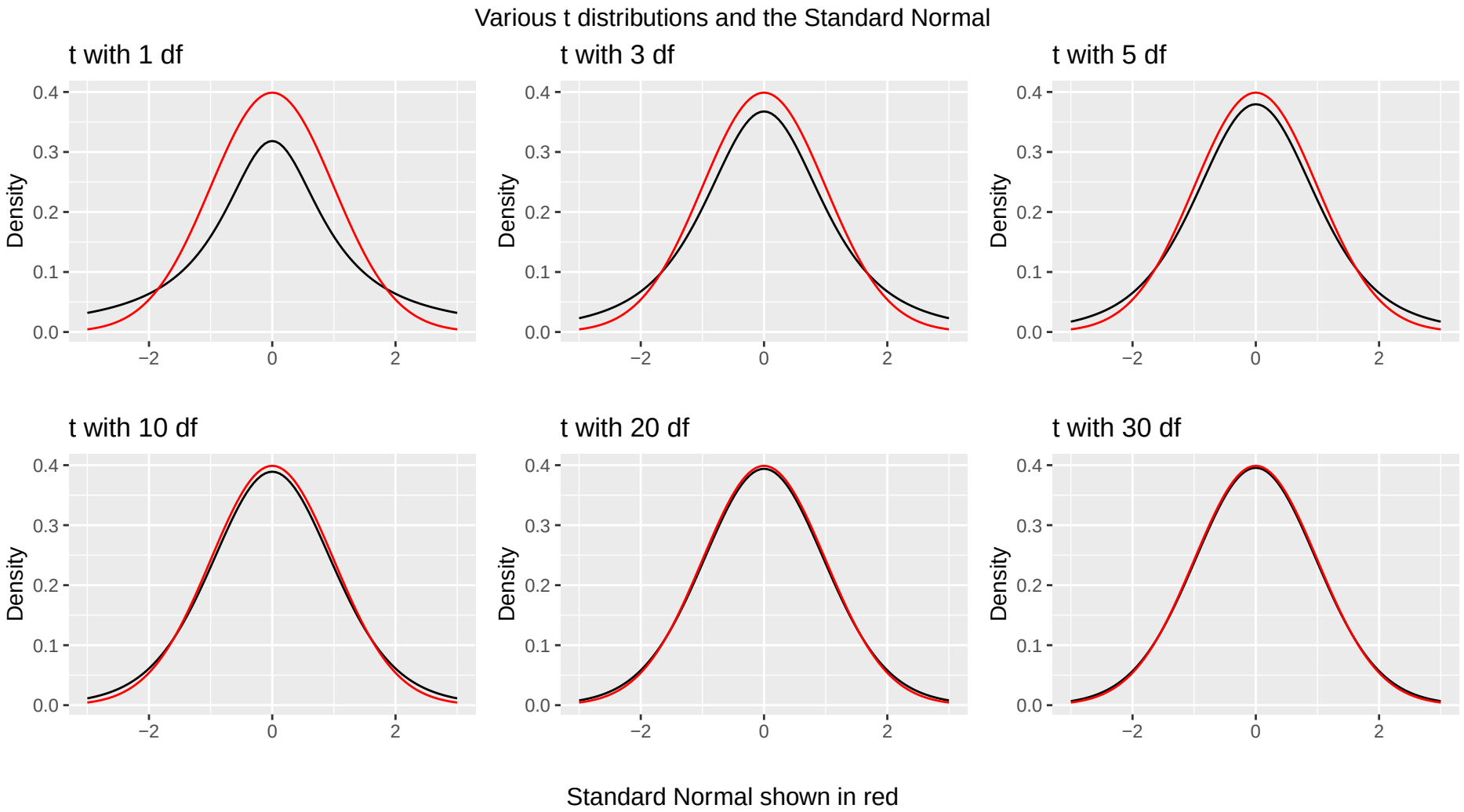
```
qt(alphaover2, df = n-1, lower.tail=FALSE)
```

# Student's t distribution

Student's t distribution looks a lot like a Normal distribution, when the sample size is large. Unlike the normal distribution, which is specified by two parameters, the mean and the standard deviation, the t distribution is specified by one parameter, the degrees of freedom.

- t distributions with large numbers of degrees of freedom are more or less indistinguishable from the standard Normal distribution.
- t distributions with smaller degrees of freedom (say, with df $< 30$, in particular) are still symmetric, but are more outlier-prone than a Normal distribution.

# Six t Distributions and a Standard Normal



# "Hand-Crafting" the 90% confidence interval for $\mu$

Let's build a 90% confidence interval for the true mean SBP across the entire population of NE Ohio adults ages 31-70 with diabetes.

| $\alpha$ | $n$ | $\bar{x}$ | $s_x$ | SE($\bar{x}$) |
|---|---|---|---|---|
| 0.10 | 431 | 131.26 | 18.52 | 0.89 |

The two-sided 100(1-$\alpha$)% confidence interval (based on a $t$ test) is: $\bar{x} \pm t_{\alpha/2,n-1}(s/\sqrt{n})$, or

- The 90% CI for $\mu$ is 131.26 $\pm$ $t_{0.10/2,431-1}$ (0.89)
  - To calculate the t cutoff value for $\alpha = 0.10$ and $n = 431$, we use

`qt(0.10/2, df = 431-1, lower.tail=FALSE)` = 1.648405

- So the 90% CI for $\mu$ is 131.26 $\pm$ 1.6484 x 0.89, or
- 131.26 $\pm$ 1.47, or (129.79, 132.73)

# Getting R to build a CI for $\mu$

Happily, R does all of this work, and with less inappropriate rounding.

```
t1 <- dm431 %$% t.test(sbp, conf.level = 0.90,
                       alternative = "two.sided")

t1
```

```
    One Sample t-test

data:  sbp
t = 147.14, df = 430, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
90 percent confidence interval:
 129.794 132.735
sample estimates:
mean of x
 131.2645
```

# Summarizing the Confidence Interval

```
tidy(t1) %>% # from broom package
  select(estimate, conf.low, conf.high, method, alternative)
```

| estimate <dbl> | conf.low <dbl> | conf.high <dbl> | method <chr> | alternative <chr> |
|---|---|---|---|---|
| 131.2645 | 129.794 | 132.735 | One Sample t-test | two.sided |

Our 90% confidence interval for the true population mean SBP in NE Ohio adults with diabetes, based on our sample of 431 patients, is (129.8, 132.7) mm Hg[1].

[1] Since the actual SBP values are integers, we should probably include no more than one additional significant figure in our confidence interval.

# We've Seen This Result Before

This intercept-only linear regression model yields the same estimates.

```
model1 <- lm(sbp ~ 1, data = dm431)
tidy(model1, conf.int = TRUE, conf.level = 0.90) %>%
  select(term, estimate, std.error, conf.low, conf.high) %>%
  knitr::kable(digits = 2)
```

| term | estimate | std.error | conf.low | conf.high |
|---|---|---|---|---|
| (Intercept) | 131.26 | 0.89 | 129.79 | 132.74 |

- Our point estimate for the population mean SBP ($\mu$) will be 131.26 mm Hg based on the dm431 sample.
- Our 90% confidence interval estimate for $\mu$ turns out to be (129.79, 132.74) mm Hg.

# What if we want a two-sided 95% CI instead?

The `t.test` function in R has an argument to specify the desired confidence level.

```
t.test(dm431$sbp, conf.level = 0.95, alt = "two.sided")
```

```
    One Sample t-test

data:  dm431$sbp
t = 147.14, df = 430, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 129.5111 133.0179
sample estimates:
mean of x
 131.2645
```

# Using Different Levels of Confidence

Below, we see two-sided confidence intervals for various levels of $\alpha$.

| Confidence Level | $\alpha$ | Two-Sided Interval Estimate for SBP Population Mean, $\mu$ | Point Estimate for SBP Population Mean, $\mu$ |
|---|---|---|---|
| 80% or 0.80 | 0.20 | (130.1, 132.4) | 131.3 |
| 90% or 0.90 | 0.10 | (129.8, 132.7) | 131.3 |
| 95% or 0.95 | 0.05 | (129.5, 133) | 131.3 |
| 99% or 0.99 | 0.01 | (129, 133.6) | 131.3 |

What is the relationship between the confidence level and the width of the confidence interval in the table?

# One-sided vs. Two-sided Confidence Intervals

In some situations, we are concerned with either an upper limit for the population mean $\mu$ or a lower limit for $\mu$, but not both.

If we, as before, have a sample of size $n$, with sample mean $\bar{x}$ and sample standard deviation $s$, then:

- The upper bound for a one-sided 100(1-$\alpha$)% confidence interval for the population mean is $\mu \leq \bar{x} + t_{\alpha,n-1}(\frac{s}{\sqrt{n}})$, with lower "bound" $-\infty$.
- The corresponding lower bound for a one-sided 100(1 - $\alpha$) CI for $\mu$ would be $\mu \geq \bar{x} - t_{\alpha,n-1}(\frac{s}{\sqrt{n}})$, with upper "bound" $\infty$.

# One-Sided CI for $\mu$

```
t.test(dm431$sbp, conf.level = 0.90, alt = "greater")
```

```
    One Sample t-test

data:  dm431$sbp
t = 147.14, df = 430, p-value < 2.2e-16
alternative hypothesis: true mean is greater than 0
90 percent confidence interval:
 130.1195      Inf
sample estimates:
mean of x
 131.2645
```

# Another One-Sided CI for $\mu$

```
t.test(dm431$sbp, conf.level = 0.90, alt = "less")
```

```
    One Sample t-test

data:  dm431$sbp
t = 147.14, df = 430, p-value = 1
alternative hypothesis: true mean is less than 0
90 percent confidence interval:
     -Inf 132.4095
sample estimates:
mean of x
 131.2645
```

# Relationship between One-Sided and Two-Sided CIs

Note the relationship between the *two-sided* 80% confidence interval, and the *one-sided* 90% confidence interval.

| Confidence Level | $\alpha$ | Type of Interval | Interval Estimate for Population Mean SBP, $\mu$ |
|---|---|---|---|
| 80% or 0.80 | 0.20 | Two-Sided | (130.12, 132.41) |
| 90% or 0.90 | 0.10 | One Sided ($>$) | $\mu > 130.12$ |

Why does this happen?

# Why, indeed?

- The 90% two-sided interval is placed so as to cut off the top 5% of the distribution with its upper bound, and the bottom 5% of the distribution with its lower bound.
- The 95% "less than" one-sided interval is placed so as to have its upper bound cut off the top 5% of the distribution.

| Confidence Level | $\alpha$ | Type of Interval | Interval Estimate for Population Mean SBP, $\mu$ |
|---|---|---|---|
| 90% or 0.90 | 0.10 | Two-Sided | (129.79, 132.74) |
| 95% or 0.95 | 0.05 | One Sided ($<$) | $\mu < 132.74$ |

# Interpreting the Result

(129.79, 132.74) mm Hg. is a 90% two-sided confidence interval for the population mean SBP among NE Ohio adults with diabetes. How can we interpret that?

- Our point estimate for the true population mean SBP among NE Ohio adults with diabetes is 131.26 mm Hg. The values in the interval (129.79, 132.74) represent a reasonable range of estimates for the true population mean SBP among NE Ohio adults with diabetes, and we are 90% confident that this method of creating a confidence interval will produce a result containing the true population mean SBP among NE Ohio adults ages 31-70 with diabetes.
- Were we to draw 100 samples of size 431 from the population described by this sample, and use each such sample to produce a confidence interval in this manner, approximately 90 of those confidence intervals would cover the true population mean SBP among NE Ohio adults ages 31-70 with diabetes.

# Assumptions of a t-based Confidence Interval

*"Begin challenging your assumptions. Your assumptions are your windows on the world. Scrub them off every once in awhile or the light won't come in." (Alan Alda)*

1. Sample is drawn at random from the population or process.
2. Samples are drawn independently from each other from a population or process whose distribution is unchanged during the sampling process.
3. Population or process follows a Normal distribution.

**Can we drop any of these assumptions?**

Only if we're willing to consider alternative inference methods.

# Coming Up . . .

We'll show you how to find an appropriate confidence interval describing the center of a population without having to assume that population has a Normal distribution.

- Using the **bootstrap** to create a confidence interval for the population mean without assuming a Normal distribution for the population
- **Wilcoxon rank sum** approach to create a confidence interval for the population pseudo-median without assuming a Normal distribution for the population
  - But this does require understanding what the pseudo-median is. . .

I've put the R code in the next two slides. . .

# Bootstrap 90% confidence interval

```
set.seed(20191001)
Hmisc::smean.cl.boot(dm431$sbp, conf.int = .90, B = 1000)
```

```
    Mean     Lower     Upper
131.2645 129.7702 132.7847
```

# Wilcoxon rank sum based 90% confidence interval

```
wilcox.test(dm431$sbp, conf.int = TRUE, conf.level = 0.90)
```

```
    Wilcoxon signed rank test with continuity
    correction

data:  dm431$sbp
V = 93096, p-value < 2.2e-16
alternative hypothesis: true location is not equal to 0
90 percent confidence interval:
 129.0 131.5
sample estimates:
(pseudo)median
           130
```