431 Class 07

thomaselove.github.io/431

2020-09-15

Today's Data

NHANES data from 2011-12: a sample of 1000 adults.

- Fix a problem with the old `nh3` data I'd built in Classes 5 and 6
- Create `nh3` which solves this problem.
- How should we explore these data before modeling?
    - What can we learn about the center, spread, outliers, and shape of quantitative data?
    - Are these blood pressure data well described by a Normal distribution?
- How might we start to look at Associations between our quantities?
    - Scatterplots, Correlation, Linear Models, Smoothing

Loading our R Packages

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

theme_set(theme_bw())
```

We'll also load another package later that I'll keep secret for now.

Creating the old_nh2 data set (from last week)

```
set.seed(20200908)

old_nh2 <- NHANES %>%
    filter(SurveyYr == "2011_12") %>%
    select(ID, SurveyYr, Age, Height, Weight, BMI, Pulse,
           SleepHrsNight, BPSysAve, BPDiaAve, Gender,
           PhysActive, SleepTrouble, Smoke100,
           Race1, HealthGen, Depressed) %>%
    rename(SleepHours = SleepHrsNight, Sex = Gender,
           SBP = BPSysAve, DBP = BPDiaAve) %>%
    filter(Age > 20 & Age < 80) %>%
    drop_na() %>%
    slice_sample(., n = 1000) %>%
    clean_names()
```

But, there's a nuisance...

```
old_nh2 %>% arrange(id) %>% select(id:weight) %>% head(7)
```

```
# A tibble: 7 x 5
     id survey_yr   age height weight
  <int> <fct>     <int>  <dbl>  <dbl>
1 62180 2011_12      35   179.   89
2 62205 2011_12      28   171.   84.8
3 62205 2011_12      28   171.   84.8
4 62208 2011_12      38   169.   63.2
5 62220 2011_12      31   167.  113.
6 62222 2011_12      32   179    80.1
7 62222 2011_12      32   179    80.1
```

There are duplicate records in NHANES

To make this sample representative of the US, some subjects appear in the sample multiple times. Suppose we want to look only at distinct rows.

```r
nh_deduplicated <- NHANES %>%
    filter(SurveyYr == "2011_12") %>%
    select(ID, SurveyYr, Age, Height, Weight, BMI, Pulse,
           SleepHrsNight, BPSysAve, BPDiaAve, Gender,
           PhysActive, SleepTrouble, Smoke100,
           Race1, HealthGen, Depressed) %>%
    rename(SleepHours = SleepHrsNight, Sex = Gender,
           SBP = BPSysAve, DBP = BPDiaAve) %>%
    filter(Age > 20 & Age < 80) %>%
    drop_na() %>%
    distinct() # add this to avoid duplicate rows
```

```r
dim(nh_deduplicated)
```

```
[1] 1727   17
```

New version of data, without duplications

```
set.seed(20200914)

nh3 <- nh_deduplicated %>%
  slice_sample(n = 1000) %>%
  clean_names()
```

Today's Questions

1. How might we explore the blood pressure data to understand it better? In particular, does a Normal model fit our systolic and diastolic blood pressures well?
2. What is the nature of the association between systolic BP and diastolic BP in these NHANES subjects?

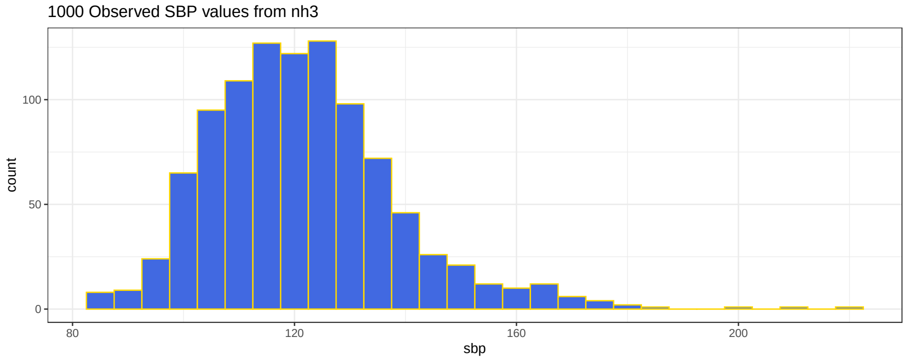
Today's Variables

Name	Description
`id`	Identifying code for each subject
`sbp`	Systolic Blood Pressure (mm Hg)
`dbp`	Diastolic Blood Pressure (mm Hg)

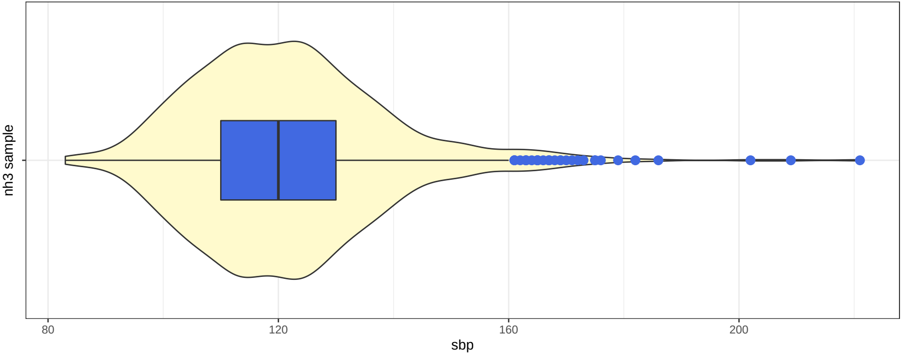
Plotting the `sbp` data to learn about center, spread, outliers, and shape

Histogram of Systolic BP values from nh3

```
ggplot(data = nh3, aes(x = sbp)) +
  geom_histogram(binwidth = 5,
                 fill = "royalblue", col = "gold") +
  labs(title = "1000 Observed SBP values from nh3")
```



Violin and Boxplot for `nh3` SBP data



min	Q1	median	Q3	max	mean	sd	n	missing
83	110	120	130	221	121.7	17.1	1000	0

Can we describe these data as being well-approximated by a Normal model?

What is a Normal Model?

By a Normal model, we mean that the data are assumed to be the result of selecting at random from a probability distribution called the Normal (or Gaussian) distribution, which is characterized by a bell-shaped curve.

- The Normal model is defined by establishing the values of two parameters: the mean and the standard deviation.

When is it helpful to assume our data follow a Normal model?

- When summarizing the data (especially if we want to interpret the mean and standard deviation)
- When creating inferences about populations from samples (as in a t test, or ANOVA)
- When creating regression models, it will often be important to make distributional assumptions about errors, for instance, that they follow a Normal model.

Does a Normal model fit our data "well enough"?

We evaluate whether a Normal model fits sufficiently well to our data on the basis of (in order of importance):

1. Graphs (DTDP) are the most important tool we have
   - There are several types of graphs available that are designed to (among other things) help us identify clearly several of the potential problems with assuming Normality.
2. Planned analyses after a Normal model decision is made
   - How serious the problems we see in graphs need to be before we worry about them changes substantially depending on how closely the later analyses we plan to do rely on the assumption of Normality.
3. Numerical Summaries are by far the least important even though they seem "easy-to-use" and "objective".

Does a Normal model fit well for my data?

The least important approach (even though it is seemingly the most objective) is the calculation of various numerical summaries.

Semi-useful summaries help us understand whether they match up well with the expectations of a normal model:

1. Assessing skewness with $skew_1$ (is the mean close to the median)?
2. Assessing coverage probabilities:

- In a Normal model, mean $\pm$ 1 standard deviation covers 68% of the data.
- In a Normal model, mean $\pm$ 2 standard deviations covers 95% of the data.
- In a Normal model, mean $\pm$ 3 standard deviations covers 99.7% of the data.

Quantifying skew with a simple $skew_1$ measure

$$skew_1 = \frac{mean - median}{standard\ deviation}$$

Interpreting $skew_1$ (for unimodal data)

- $skew_1 = 0$ if the mean and median are the same
- $skew_1 > 0.2$ indicates fairly substantial right skew
- $skew_1 < -0.2$ indicates fairly substantial left skew

Measuring skewness in the SBP values: nh3?

```
mosaic::favstats(~ sbp, data = nh3)
```

```
 min  Q1 median  Q3 max    mean       sd    n missing
  83 110    120 130 221 121.725 17.09677 1000       0
```

```
nh3 %>% summarize(skew1 = (mean(sbp) - median(sbp))/sd(sbp))
```

```
# A tibble: 1 x 1
  skew1
  <dbl>
1 0.101
```

What does this suggest?

Empirical Rule for a Normal Model

If the data followed a Normal distribution, perfectly, then about:

- 68% of the data would fall within 1 standard deviation of the mean
- 95% of the data would fall within 2 standard deviations of the mean
- 99.7% of the data would fall within 3 standard deviations of the mean

Remember that, regardless of the distribution of the data:

- Half of the data will fall below the median, and half above it.
- Half of the data will fall in the Interquartile Range (IQR).

How many SBPs are within 1 SD of the mean?

```
nh3 %>%
  count(sbp > mean(sbp) - sd(sbp),
        sbp < mean(sbp) + sd(sbp)) %>%
  kable()
```

sbp > mean(sbp) - sd(sbp)	sbp < mean(sbp) + sd(sbp)	n
FALSE	TRUE	134
TRUE	FALSE	131
TRUE	TRUE	735

How does this compare to the expectation under a Normal model? Remember that there are 1000 observations in `nh3`.

SBP and the mean ± 2 standard deviations rule?

The total sample size here is 1000.

```
nh3 %>%
  count(sbp > mean(sbp) - 2*sd(sbp),
        sbp < mean(sbp) + 2*sd(sbp)) %>%
  kable()
```

sbp > mean(sbp) - 2 * sd(sbp)	sbp < mean(sbp) + 2 * sd(sbp)	n
FALSE	TRUE	8
TRUE	FALSE	44
TRUE	TRUE	948

How does this compare to the expectation under a Normal model?

Hypothesis Testing to assess Normality

Don't. Graphical approaches are **far** better than hypothesis tests...

```r
nh3 %$% shapiro.test(sbp)
```

```

    Shapiro-Wilk normality test

data:  sbp
W = 0.95318, p-value < 2.2e-16
```

The very small *p* value (2.2×10^{-16} should be interpreted by you as meaning zero) indicates that the test finds some indications **against** adopting a Normal model for these data.

- Exciting, huh? But not actually all that useful, alas.

Why not test for Normality?

There are multiple hypothesis testing schemes (Kolmogorov-Smirnov, etc.) and each looks for one specific violation of a Normality assumption. None can capture the wide range of issues our brains can envision, and none by itself is great at its job.

- With any sort of reasonable sample size, the test is so poor at detecting non-normality compared to our eyes, that it finds problems we don't care about and ignores problems we do care about.
- And without a reasonable sample size, the test is essentially useless.

Whenever you *can* avoid hypothesis testing and instead actually plot the data, you should plot the data. Sometimes you can't plot (especially with really big data) but the test should be a last resort.

Can we simulate a Normal model for SBP?

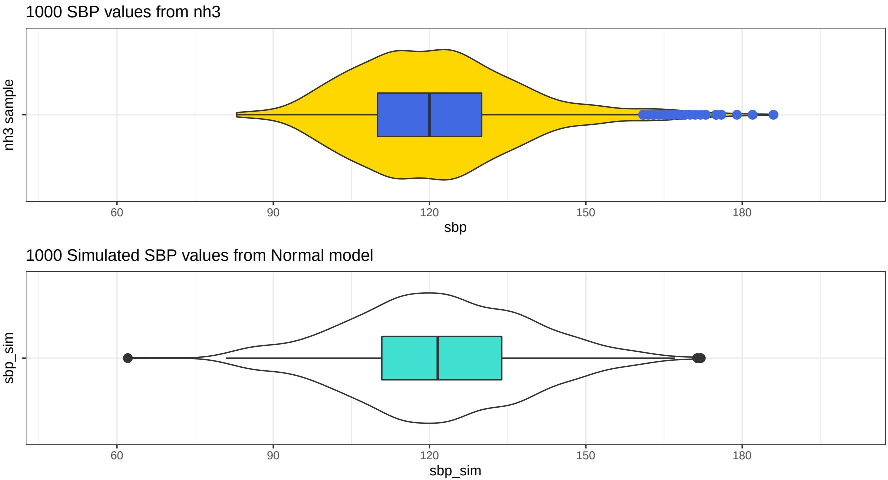
Simulate 1000 observations from a Normal distribution with the same mean and standard deviation as our nh3 systolic BP values. . .

```
set.seed(1234567)
sim1 <- tibble( sbp_sim =
                  rnorm(n = 1000, mean = 121.7, sd = 17.1))

mosaic::favstats(~ sbp_sim, data = sim1) %>%
  kable(digits = 1)
```

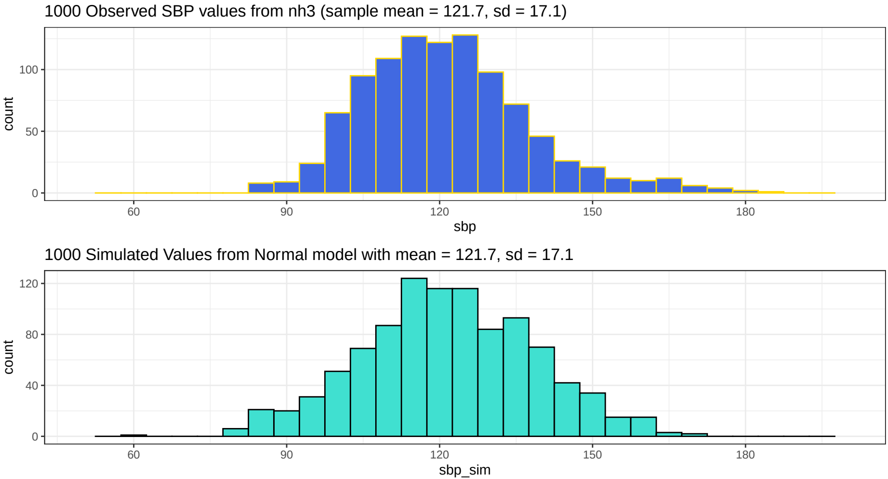
min	Q1	median	Q3	max	mean	sd	n	missing
62.1	110.8	121.6	133.9	172	121.6	17	1000	0

Comparing Boxplots of `nh3` and simulated SBP



- Does a Normal model look appropriate for describing the `nh3` SBP?

Comparing Histograms of nh3 and simulated SBP

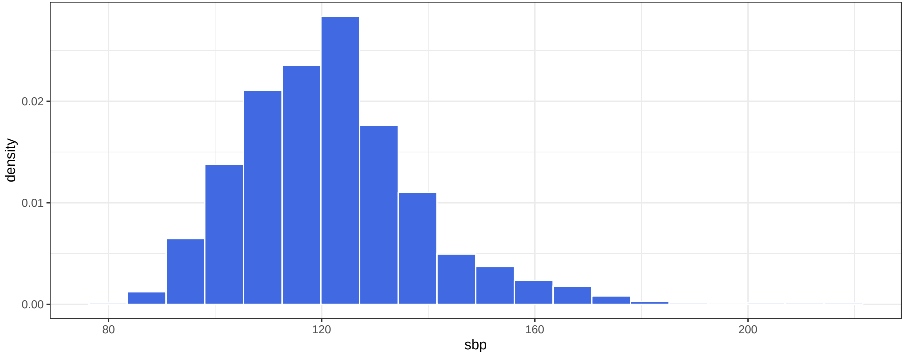


- Does a Normal model look appropriate for describing the nh3 SBP?

Rescale nh3 SBP histogram as density

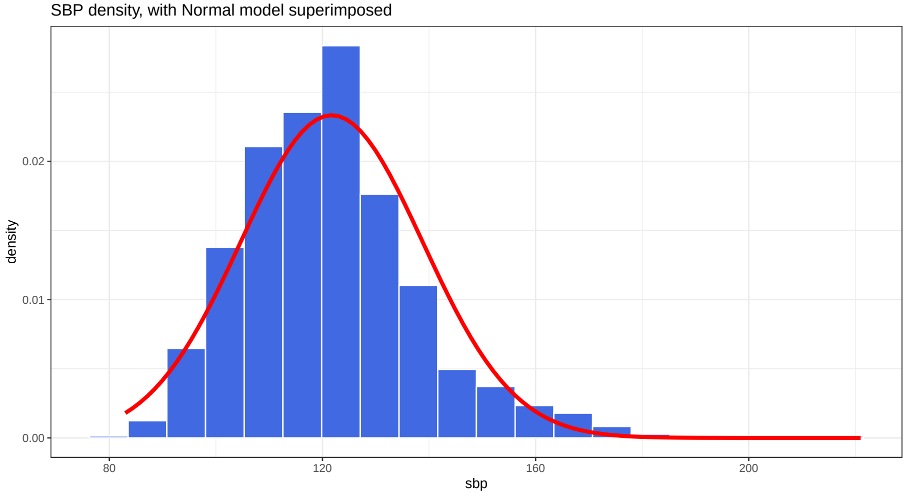
Suppose we want to rescale the histogram counts so that the bar areas integrate to 1. This will let us overlay a Normal density onto the results.

```
ggplot(nh3, aes(x = sbp)) +
  geom_histogram(aes(y = stat(density)), bins = 20,
                 fill = "royalblue", col = "white")
```



Density Function, with Normal superimposed

Now we can draw a Normal density curve on top of the rescaled histogram.



Code for plotting Histogram as Density function

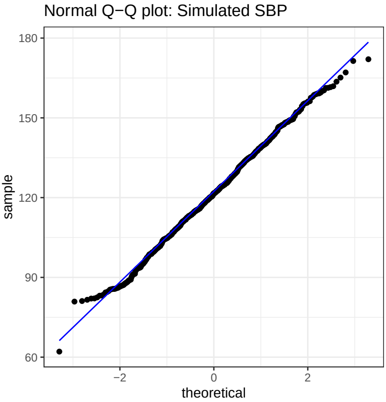
Including the superimposition of a Normal density on top of the histogram.

```r
ggplot(nh3, aes(x = sbp)) +
  geom_histogram(aes(y = stat(density)),
                 bins = 20,
                 fill = "royalblue", col = "white") +
  stat_function(fun = dnorm,
                args = list(mean = mean(nh3$sbp),
                            sd = sd(nh3$sbp)),
                col = "red", lwd = 1.5) +
  labs(title = "SBP density, with Normal model superimposed")
```

Using a Normal Q-Q plot

Normal Q-Q plot of our simulated data

Remember that these are draws from a Normal distribution, so this is what a sample of 1000 Normally distributed data points should look like.



The Normal Q-Q Plot

Tool to help assess whether the distribution of a single sample is well-modeled by the Normal.

- Suppose we have N data points in our sample.
- Normal Q-Q plot will plot N points, on a scatterplot.
    - Y value is the data value
    - X value is the expected value for that point in a Normal distribution

Using the Normal distribution with the same mean and SD as our sample, R calculates what the minimum value is expected to be, given a sample of size N, then the next smallest value, and so forth all the way up until the maximum value.

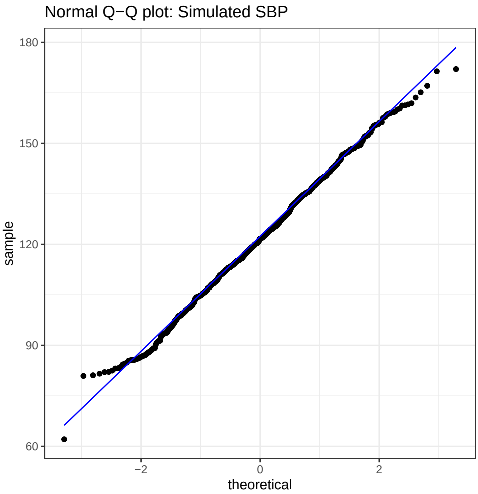
- X value in the Normal Q-Q plot is the value that a Normal distribution would take for that rank in the data set.
- We draw a line through Y = X, and points close to the line therefore match what we'd expect from a Normal distribution.

How do we create a Normal Q-Q plot?

For our simulated blood pressure data

```
ggplot(sim1, aes(sample = sbp_sim)) +
  geom_qq() + # plot the points
  geom_qq_line(col = "blue") + # plot the Y = X line
  theme(aspect.ratio = 1) + # make the plot square
  labs(title = "Normal Q-Q plot: Simulated SBP")
```

Result, again. . .



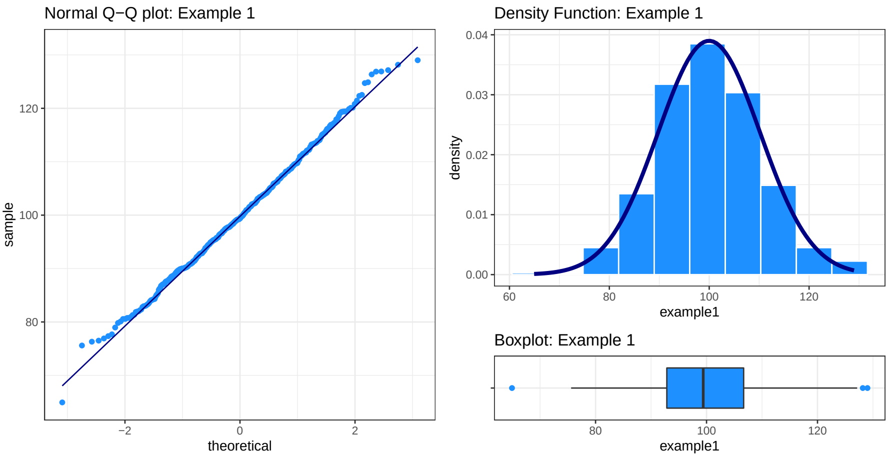
Interpreting the Normal Q-Q plot?

The Normal Q-Q plot can help us identify data as well approximated by a Normal distribution, or not, because of:

- skew (including distinguishing between right skew and left skew)
- behavior in the tails (which could be heavy-tailed [more outliers than expected] or light-tailed)

1. Normally distributed data are indicated by close adherence of the points to the diagonal reference line.
2. Skew is indicated by substantial curving (on both ends of the distribution) in the points away from the reference line (if both ends curve up, we have right skew; if both ends curve down, this indicates left skew)
3. An abundance or dearth of outliers (as compared to the expectations of a Normal model) are indicated in the tails of the distribution by an "S" shape or reverse "S" shape in the points.

Example 1: Data from a Normal Distribution

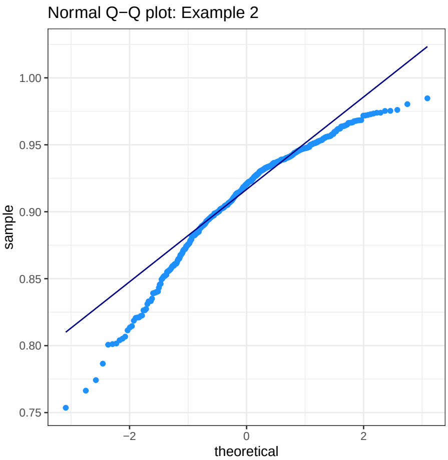


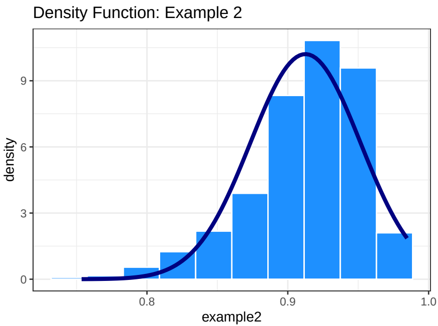
min	Q1	median	Q3	max	mean	sd	n	missing
64.9	92.8	99.4	106.7	129	100	10.2	500	0

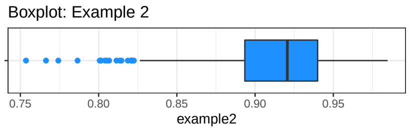
Does a Normal model fit well for my data?

1. Is a Normal Q-Q plot showing something close to a straight line, without clear signs of skew or indications of lots of outliers (heavy-tailedness)?
2. Does a boxplot, violin plot and/or histogram also show a symmetric distribution, where both the number of outliers is modest, and the distance of those outliers from the mean is modest?
3. Do numerical measures match up with the expectations of a normal model?

Example 2: Data from a Left-Skewed Distribution

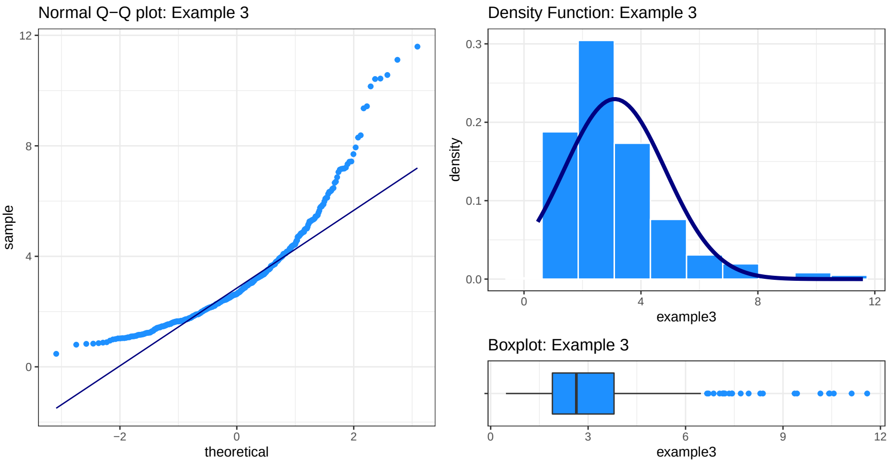






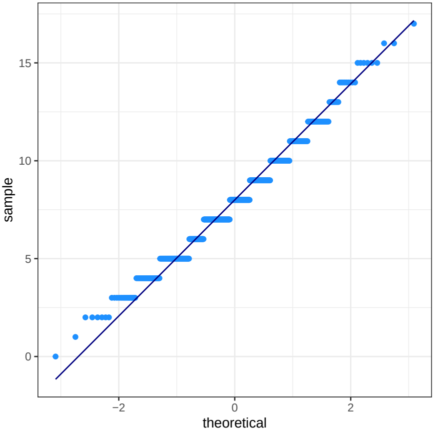
min	Q1	median	Q3	max	mean	sd	n	missing
0.8	0.9	0.9	0.9	1	0.9	0	500	0

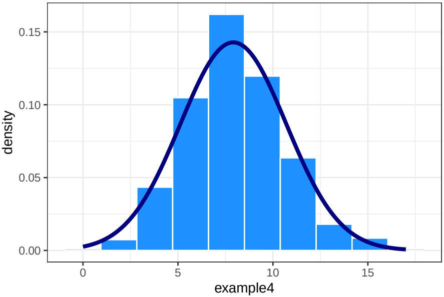
Example 3: Data from a Right-Skewed Distribution

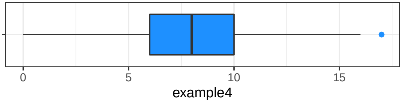


min	Q1	median	Q3	max	mean	sd	n	missing
0.5	1.9	2.6	3.8	11.6	3.1	1.7	500	0

Example 4: Discrete "Symmetric" Distribution

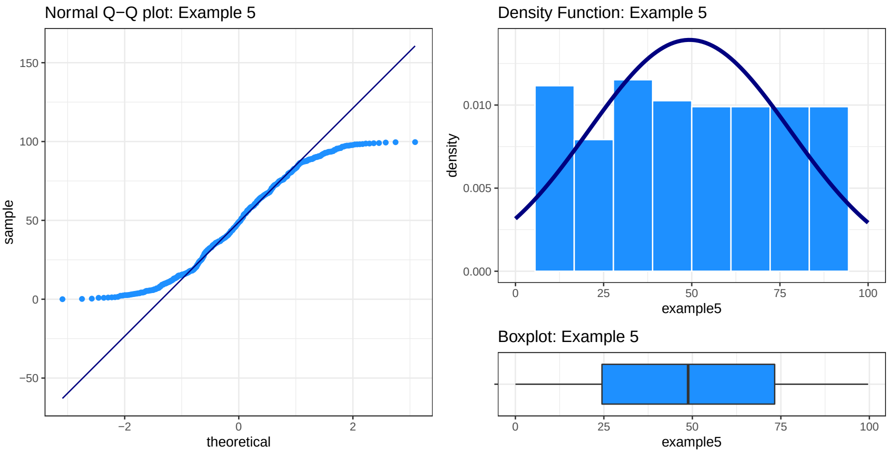

Normal Q−Q plot: Example 4


Density Function: Example 4


Boxplot: Example 4

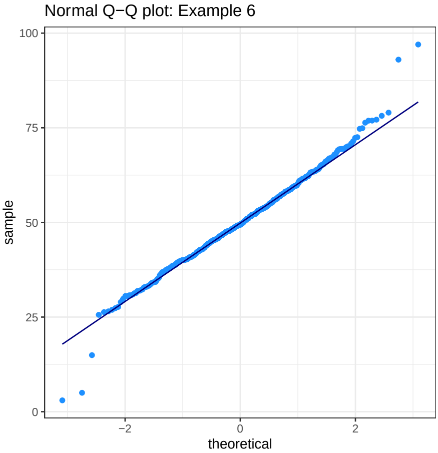
min	Q1	median	Q3	max	mean	sd	n	missing
0	6	8	10	17	7.9	2.8	500	0

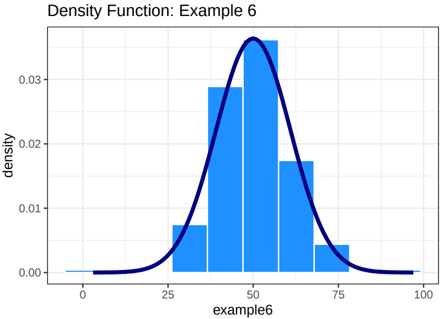
Example 5: Data from a Uniform Distribution

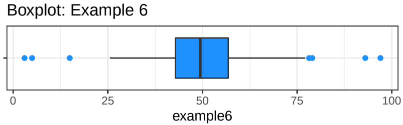


min	Q1	median	Q3	max	mean	sd	n	missing
0	24.5	48.8	73.2	99.6	49.3	28.7	500	0

Example 6: Symmetric data with outliers

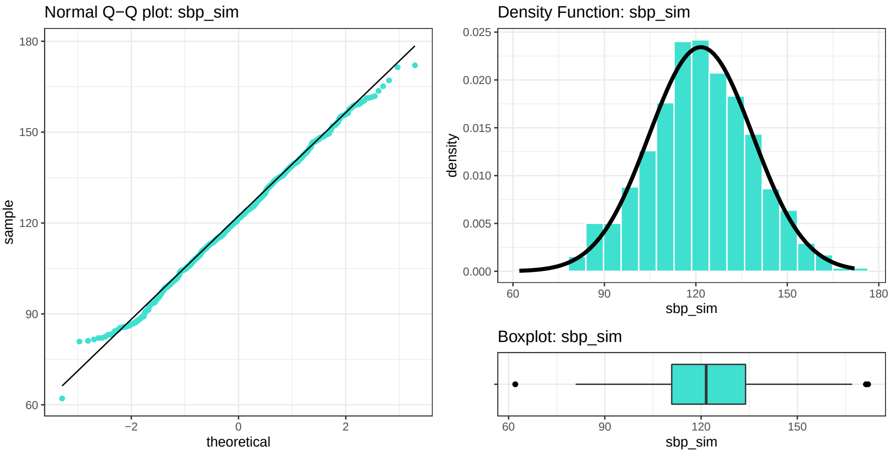






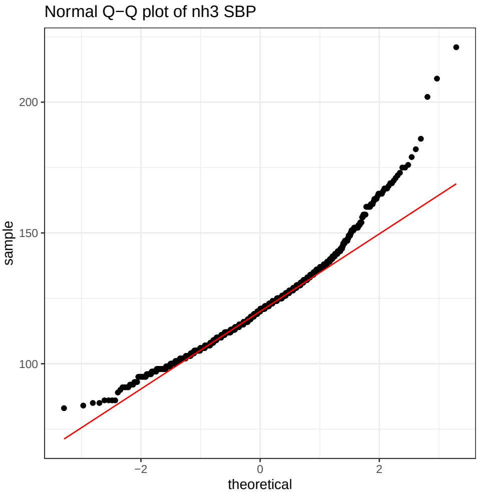
min	Q1	median	Q3	max	mean	sd	n	missing
3	42.8	49.4	56.8	97	50	11	500	0

Our 1000 simulated Systolic Blood Pressures

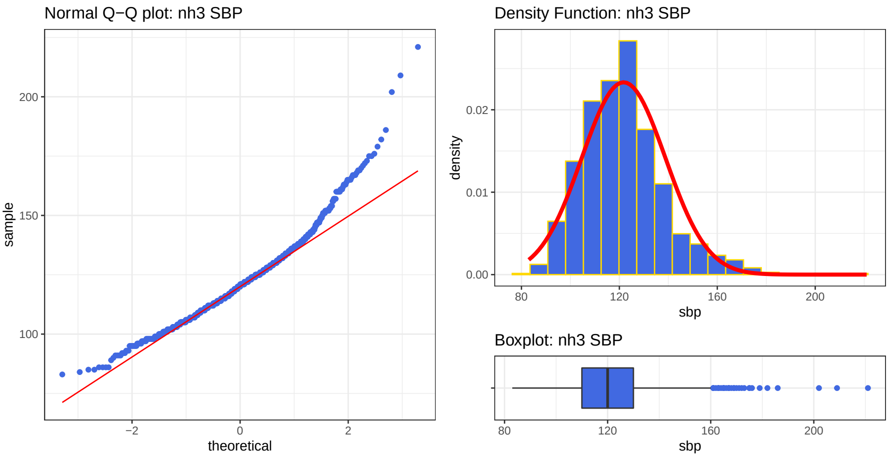


min	Q1	median	Q3	max	mean	sd	n	missing
62.1	110.8	121.6	133.9	172	121.6	17	1000	0

A Normal Q-Q Plot of the nh3 SBP data (n = 1000)



Our 1000 observed Systolic Blood Pressures



Normal Q−Q plot: nh3 SBP

Density Function: nh3 SBP

Boxplot: nh3 SBP

min	Q1	median	Q3	max	mean	sd	n	missing
83	110	120	130	221	121.7	17.1	1000	0

More Extensive Numerical Summaries?

We could try

```
nh3 %>% Hmisc::describe(sbp)
```

- but that will throw an error message, specifically `Error in describe.data.frame(., sbp) : object 'sbp' not found`. What is wrong? How can we fix that?

More Extensive Numerical Summaries?

We could try

```
nh3 %>% Hmisc::describe(sbp)
```

- but that will throw an error message, specifically `Error in describe.data.frame(., sbp) : object 'sbp' not found`. What is wrong? How can we fix that?
- We could drop the pipe and use $ notation, so `Hmisc::describe(nh3$sbp)`

More Extensive Numerical Summaries?

We could try

```
nh3 %>% Hmisc::describe(sbp)
```

- but that will throw an error message, specifically `Error in describe.data.frame(., sbp) : object 'sbp' not found`. What is wrong? How can we fix that?
- We could drop the pipe and use $ notation, so `Hmisc::describe(nh3$sbp)`
- Another option is to change the pipe (to the `%$%` pipe available in the `magrittr` package): `nh3 %$% Hmisc::describe(sbp)`

What do these summaries tell us?

```r
nh3 %$% Hmisc::describe(sbp)
```

```
sbp 
       n  missing distinct     Info     Mean      Gmd 
    1000        0       93        1    121.7    18.52 
     .05      .10      .25      .50      .75      .90 
      98      102      110      120      130      142 
     .95 
     152 

lowest :  83  84  85  86  89, highest: 182 186 202 209 221
```

- `Gmd` = Gini's mean difference (a robust measure of spread) = mean absolute difference between any pairs of observations. Larger `Gmd` indicates more spread.
- `Info` = a measure of relative information describing how "continuous" the data are. Higher `Info` indicates fewer ties.

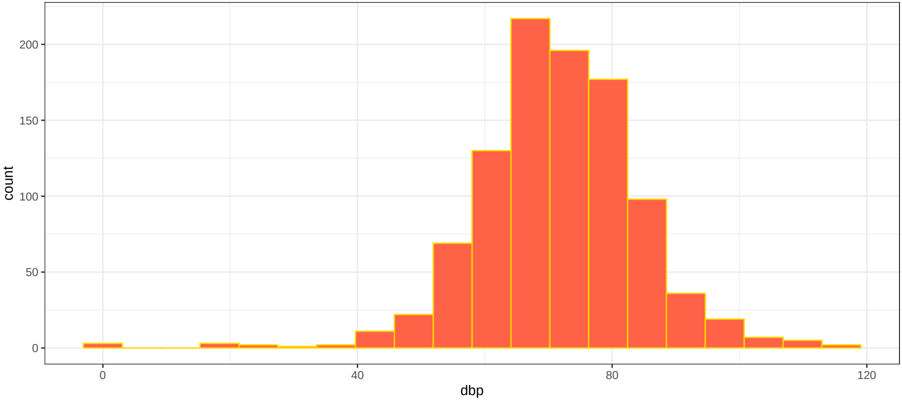
What Summaries to Report

It is usually helpful to focus on the shape, center and spread of a distribution. Bock, Velleman and DeVeaux provide some useful advice:

- If the data are skewed, report the median and IQR (or the three middle quantiles). You may want to include the mean and standard deviation, but you should point out why the mean and median differ. The fact that the mean and median do not agree is a sign that the distribution may be skewed. A histogram will help you make that point.
- If the data are symmetric, report the mean and standard deviation, and possibly the median and IQR as well.
- If there are clear outliers and you are reporting the mean and standard deviation, report them with the outliers present and with the outliers removed. The differences may be revealing. The median and IQR are not likely to be seriously affected by outliers.

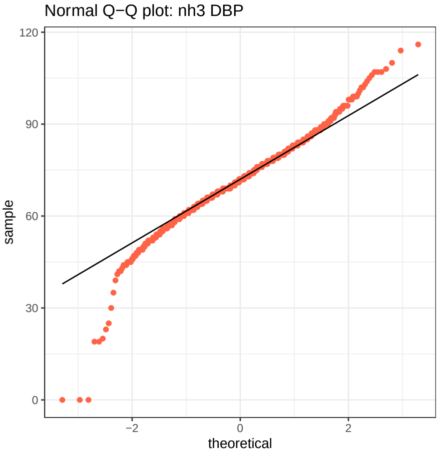
OK, what about Diastolic Blood Pressure?

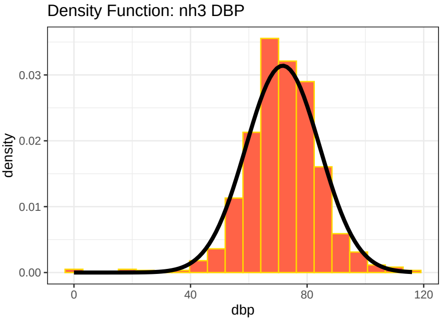
```
ggplot(data = nh3, aes(x = dbp)) +
  geom_histogram(bins = 20, fill = "tomato", col = "gold")
```

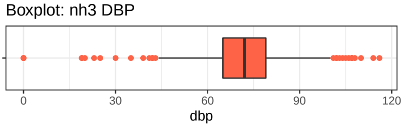


- We can generate the set of plots we've been using...

DBP in nh3: Center/Spread/Outliers/Shape?







min	Q1	median	Q3	max	mean	sd	n	missing
0	65	72	79	116	71.7	12.7	1000	0

Does a Normal model fit well for my data?

1. Is a Normal Q-Q plot showing something close to a straight line, without clear signs of skew or indications of lots of outliers (heavy-tailedness)?
2. Does a boxplot, violin plot and/or histogram also show a symmetric distribution, where both the number of outliers is modest, and the distance of those outliers from the mean is modest?
3. Do numerical measures match up with the expectations of a normal model?

Hmisc::describe for dbp?

```
nh3 %$% Hmisc::describe(dbp)
```

```
dbp
       n  missing distinct     Info     Mean      Gmd
    1000        0       78    0.999    71.66    13.58
     .05      .10      .25      .50      .75      .90
      52       57       65       72       79       86
     .95
      91

lowest :   0  19  20  23  25, highest: 107 108 110 114 116
```

What is a plausible diastolic blood pressure?

Stem-and-Leaf of dbp values?

```
stem(nh3$dbp)
```

```

  The decimal point is 1 digit(s) to the right of the |

   0 | 000
   1 | 99
   2 | 035
   3 | 059
   4 | 1223444555566777888999999
   5 | 0001111122222222233333334444444445555555666666666666677
   6 | 0000000000000000001111111111111111111112222222222222222
   7 | 0000000000000000000000000000000001111111111111111111111
   8 | 0000000000000000000000000000000001111111111111111111222
   9 | 0000000001111122222233444444555556666688889999
  10 | 012234567778
  11 | 046
```

Who are those people with tiny dbp values?

```
nh3 %>%
  filter(dbp < 40) %>%
  select(id, sbp, dbp)
```

```
# A tibble: 11 x 3
      id   sbp   dbp
   <int> <int> <int>
 1 71598    86    30
 2 68528   133    39
 3 64298   135     0
 4 64616   111    25
 5 65298   126    35
 6 62649   122    23
 7 70664   152     0
 8 69237   120     0
 9 68561   119    19
10 68908   129    20
```

Let's reset.

```
nh3_new <- nh3 %>%
  filter(dbp > 39)

nrow(nh3)
```

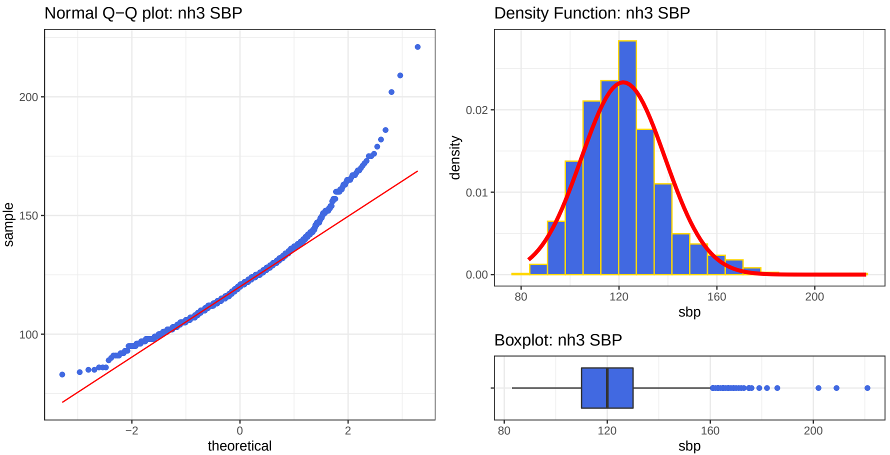
```
[1] 1000
```

```
nrow(nh3_new)
```

```
[1] 989
```

We'll work with `nh3_new` for the rest of today.

nh3_new: Systolic Blood Pressure



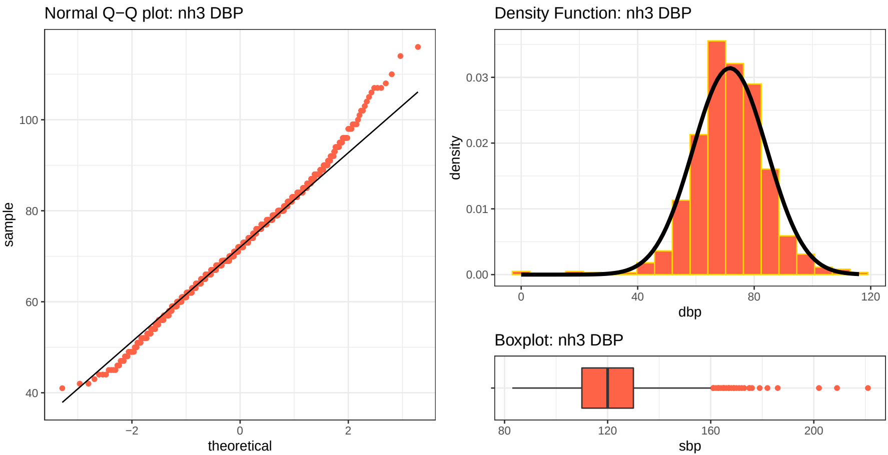
Normal Q−Q plot: nh3 SBP

Density Function: nh3 SBP

Boxplot: nh3 SBP

min	Q1	median	Q3	max	mean	sd	n	missing
83	110	120	130	221	121.7	17.1	1000	0

nh3_new: Diastolic Blood Pressure



min	Q1	median	Q3	max	mean	sd	n	missing
0	65	72	79	116	71.7	12.7	1000	0

Summing Up: Does a Normal Model fit well?

If a Normal model fits our data well, then we should see the following graphical indications:

1. A histogram that is symmetric and bell-shaped.
2. A boxplot where the box is symmetric around the median, as are the whiskers, without a serious outlier problem.
3. A normal Q-Q plot that essentially falls on a straight line.

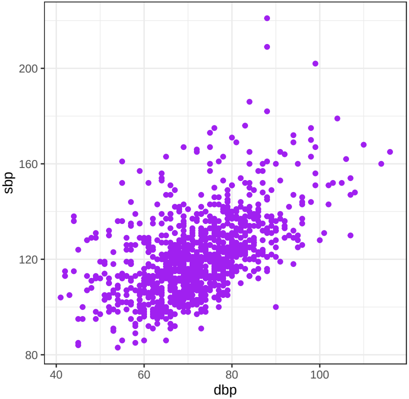
As for numerical summaries, we'd like to see

4. The mean and median within 0.2 standard deviation of each other.
5. No real evidence of too many outlier candidates (more than 5% starts to get us concerned about a Normal model)
6. No real evidence of individual outliers outside the reasonable range for the size of our data (we might expect about 3 observations in 1000 to fall more than 3 standard deviations away from the mean.)

How can we describe the relationship between SBP and DBP?

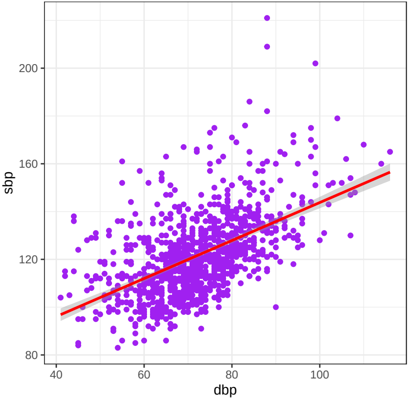
Scatterplot to study the SBP-DBP association

```
ggplot(nh3_new, aes(x = dbp, y = sbp)) +
  geom_point(col = "purple") +
  theme(aspect.ratio = 1) # make the plot square for slide
```



Try to predict sbp from dbp?

```
ggplot(nh3_new, aes(x = dbp, y = sbp)) +
  geom_point(col = "purple") + theme(aspect.ratio = 1) +
  geom_smooth(method = "lm", formula = y ~ x,
              col = "red", se = TRUE)
```



What should we conclude from this scatterplot?

1. Each of the 989 subjects in `nh3_new` is represented by one of the points.
2. The association appears generally to be "positive" or "direct" in that as `dbp` increases, so does `sbp`.
3. The association appears fairly linear, in that most of the points cluster around a pattern fit well by a straight line.
4. The association is not particularly strong in that the range of `sbp` values associated with any particular `dbp` value is still pretty wide.

   - If we know someone's `dbp`, that should help us make better predictions of their sbp, but perhaps not much better than if we didn't know their `dbp`.

5. I don't see any substantial outliers that are far away from all other points.

What line is being fit there?

Least Squares Regression Line (a linear model) to predict sbp using dbp

```
m1 <- lm(sbp ~ dbp, data = nh3_new)
m1
```

```

Call:
lm(formula = sbp ~ dbp, data = nh3_new)

Coefficients:
(Intercept)          dbp
     64.270        0.795
```

Model m1 is **sbp = 76.13 + 0.62 dbp**.

- What do the slope and intercept mean?

Linear Model `m1`: sbp = 76.13 + 0.62 dbp

76.13 is the intercept = predicted value of `sbp` when `dbp` = 0.

- Is that reasonable in this setting?

0.62 is the slope = predicted change in `sbp` per 1 unit change in `dbp`

- What are the units here?
- What does the fact that this estimated slope is positive mean?
- What would the line look like if the slope was negative? What if the slope was zero?

Does R like this linear model?

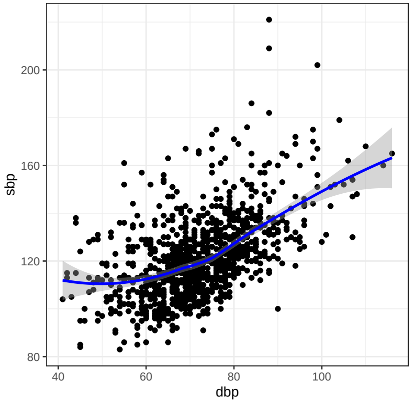
```
tidy(m1) %>% kable(digits = 2)
```

term	estimate	std.error	statistic	p.value
(Intercept)	64.27	2.96	21.70	0
dbp	0.80	0.04	19.63	0

Yes. Wow. It **really** does. Look at those *p* values!

Does a linear model really work well here?

```
ggplot(nh3_new, aes(x = dbp, y = sbp)) +
  geom_point() + theme(aspect.ratio = 1) +
  geom_smooth(method = "loess", formula = y ~ x,
              col = "blue", se = TRUE)
```



How much of the variation in sbp does m1 capture?

```r
glance(m1) %>% select(r.squared, sigma, p.value)
```

```
# A tibble: 1 x 3
  r.squared sigma  p.value
      <dbl> <dbl>    <dbl>
1     0.281  14.5 1.05e-72
```

Who can we make predictions for using model m1?

We sampled 1000 observations from a group, and then dropped those with dbp below 40, leaving n = 989.) How many people in total would be eligible?

```
nh3_eligible <- nh_deduplicated %>%
  clean_names() %>%
  filter(dbp > 39)

dim(nh3_eligible)
```

```
[1] 1709   17
```

```
dim(nh3_new)
```

```
[1] 989  17
```

Can we look at the other (1709 - 989 = 720) subjects?

```r
nh3_therest <- anti_join(nh3_eligible, nh3_new, by = "id")

dim(nh3_therest)
```

```
[1] 720  17
```

Can we use m1 to predict SBP in our nh3_therest data?

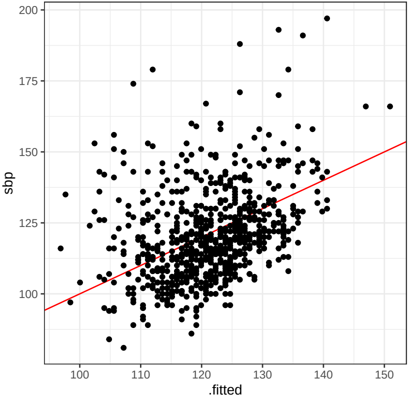
```
new720_nh3 <- augment(m1, newdata = nh3_therest)

new720_nh3 %>% select(id, sbp, dbp, .fitted, .resid) %>%
  head() %>% kable(digits = 2)
```

id	sbp	dbp	.fitted	.resid
62172	103	72	121.51	-18.51
62180	107	66	116.74	-9.74
62199	110	65	115.95	-5.95
62205	122	87	133.44	-11.44
62223	105	69	119.13	-14.13
62228	114	74	123.10	-9.10

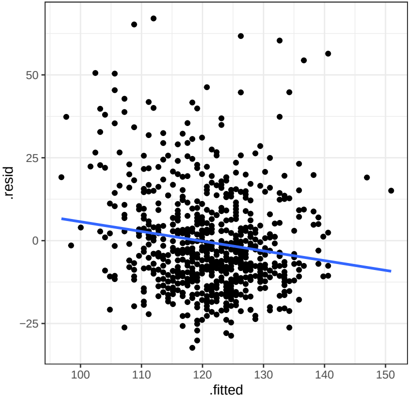
Out of Sample: Actual SBP vs. Fitted SBP by `m1` (n = 720)

```
ggplot(new720_nh3, aes(x = .fitted, y = sbp)) +
  geom_abline(slope = 1, intercept = 0, col = "red") +
  geom_point() + theme(aspect.ratio = 1)
```



Out of Sample: Residuals vs. Fitted SBP by `m1` (n = 720)

```r
ggplot(new720_nh3, aes(x = .fitted, y = .resid)) +
  geom_point() + theme(aspect.ratio = 1) +
  geom_smooth(method = "lm", formula = y ~ x, se = FALSE)
```



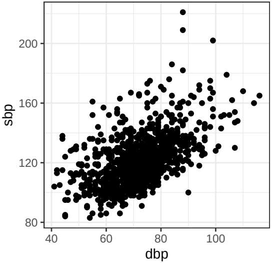
Numerical Summary: Pearson Correlation

The Pearson correlation ranges from -1 to $+1$.

- The closer the absolute value of the correlation is to 1, the stronger a linear fit will be to the data, (in a limited sense).
- A strong positive correlation (near $+1$) will indicate a strong model with a positive slope.
- A strong negative correlation (near -1) will indicate a strong linear model with a negative slope.
- A weak correlation (near 0) will indicate a poor fit for a linear model, although a non-linear model may still fit the data quite well.

Correlation in our sbp-dbp scatterplot?

```
ggplot(nh3_new, aes(x = dbp, y = sbp)) +
  geom_point() + theme(aspect.ratio = 1)
```



```
nh3_new %$% cor(sbp, dbp)
```

```
[1] 0.5299471
```

What does a correlation of +0.53 imply about a linear fit to the data?

How is the r-squared (r^2)?

```
glance(m1) %>% select(r.squared) %>% kable()
```

r.squared
0.2808439

- R-squared describes the proportion of the variation in `sbp` accounted for by the linear model `m1` using `dbp`.
- About 28% in this case. Is that good?
- Why is this called R-squared? What is the R?

```
nh3_new %$% cor(sbp, dbp)
```

```
[1] 0.5299471
```

```
nh3_new %$% cor(sbp, dbp)^2
```

```
[1] 0.2808439
```

Is this the only linear model R can fit to these data?

Nope.

```
library(rstanarm)
```

```
Loading required package: Rcpp

This is rstanarm version 2.21.1

- See https://mc-stan.org/rstanarm/articles/priors for changes

- Default priors may change, so it's safest to specify priors,

- For execution on a local, multicore CPU with excess RAM we r

  options(mc.cores = parallel::detectCores())
```

Fit linear model using stan_glm?

```r
m2 <- stan_glm(sbp ~ dbp, data = nh3_new)
```

```
SAMPLING FOR MODEL 'continuous' NOW (CHAIN 1).
Chain 1:
Chain 1: Gradient evaluation took 0 seconds
Chain 1: 1000 transitions using 10 leapfrog steps per transiti
Chain 1: Adjust your expectations accordingly!
Chain 1:
Chain 1:
Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
```

Bayesian fitted linear model for our sbp data

```r
print(m2)
```

```
stan_glm
 family:       gaussian [identity]
 formula:      sbp ~ dbp
 observations: 989
 predictors:   2
------
            Median MAD_SD
(Intercept) 64.3    3.1  
dbp          0.8    0.0  

Auxiliary parameter(s):
      Median MAD_SD
sigma 14.5    0.3  

------
```

Is the Bayesian model (with default prior) very different from our `lm` in this situation?

```r
coef(m1) # fit with lm
```

```
(Intercept)         dbp 
 64.2697456   0.7950429 
```

```r
coef(m2) # stan_glm with default priors
```

```
(Intercept)         dbp 
   64.31678     0.79492 
```

How about a robust linear model?

```r
# requires installation of robust package
# which I doubt we'll use again

m3 <- robust::lmRob(sbp ~ dbp, data = nh3_new)
```

```
Registered S3 method overwritten by 'fit.models':
  method       from 
  vcov.default Hmisc

Registered S3 methods overwritten by 'robust':
  method              from      
  plot.covfm          fit.models
  print.covfm         fit.models
  summary.covfm       fit.models
  print.summary.covfm fit.models

rlm is already registered in the fit.models registry

covfm is already registered in the fit.models registry
```

Or another kind of robust linear model?

```r
# requires installation of robustbase package
# which I doubt we'll ever use again

m4 <- robustbase::lmrob(sbp ~ dbp, data = nh3_new)

tidy(m4) %>% kable(digits = 3)
```

term	estimate	std.error	statistic	p.value
(Intercept)	64.632	3.070	21.055	0
dbp	0.768	0.042	18.475	0