431 Class 09

thomaselove.github.io/431

2021-09-21

Today's R Packages

```
library(broom) # for tidying up output
library(haven) # new today, importing files from SPSS
library(janitor)
library(knitr)
library(magrittr)
library(naniar)
library(patchwork)
library(readxl)
library(tidyverse)

theme_set(theme_bw())
```

Today's Data

Today, we'll use an SPSS file (.sav) to import the dm1000 data.

```
dm1000 <- read_sav("data/dm_1000.sav") %>%
  clean_names() %>%
  mutate(across(where(is.character), as_factor)) %>%
  mutate(across(where(is.labelled), as_factor)) %>%
  mutate(subject = as.character(subject))
```

- Note the next-to-last line in the code above, which is used to turn "labelled" variables (from SPSS) into factors in R.
- There are also functions called `read_sas()` and `read_xpt()` to read in SAS files, and `read_dta()` to read in Stata `.dta` files, available in the `haven` package.

The dm1000 tibble

```
# A tibble: 1,000 x 17
   subject   sbp   dbp insurance     age n_income    ht
   <chr>   <dbl> <dbl> <fct>       <dbl>    <dbl> <dbl>
 1 M-0001    145    70 Medicaid       55    29853  1.63
 2 M-0002    151    77 Commercial     52    31248  1.75
 3 M-0003    127    73 Medicare       69    23362  1.65
 4 M-0004    125    74 Medicaid       57    26033  1.63
 5 M-0005    120    73 Medicare       68    85374  1.69
 6 M-0006    127    75 Medicaid       56    31273  1.71
 7 M-0007    114    81 Commercial     54    25445  1.68
 8 M-0008    166   110 Medicare       45    67526  1.69
 9 M-0009    111    77 Medicare       61    15203  1.91
10 M-0010    146   102 Medicaid       63    17628  1.86
# ... with 990 more rows, and 10 more variables:
#   wt <dbl>, a1c <dbl>, ldl <dbl>, tobacco <fct>,
#   statin <dbl>, eye_exam <dbl>,
#   race_ethnicity <fct>, sex <fct>, county <fct>,
```

Describing the association of `sbp` and `dbp`

Numerical Summaries of sbp and dbp

```
mosaic::favstats(~ sbp, data = dm1000)
```

```
 min  Q1 median  Q3 max     mean       sd   n missing
  84 122    132 142 209 132.7746 17.95214 994       6
```

```
dm1000 %$% mosaic::favstats(~ dbp)
```

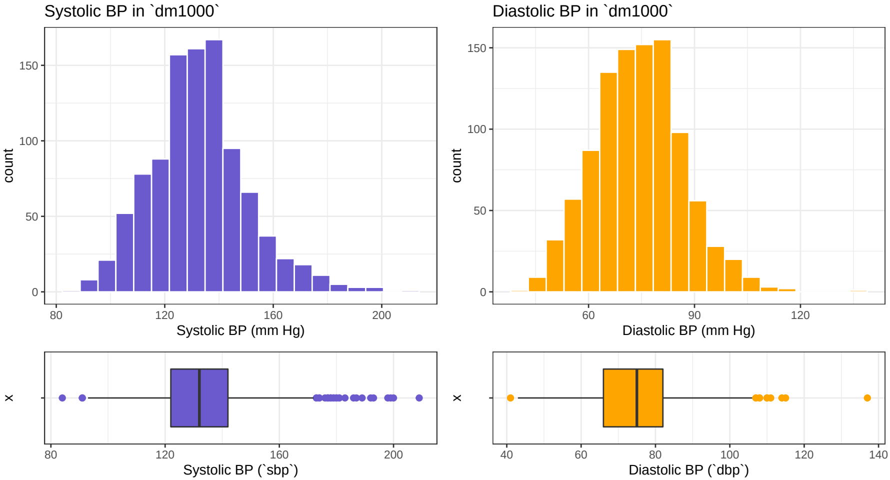
```
 min Q1 median Q3 max     mean       sd   n missing
  41 66     75 82 137 74.46378 12.42027 994       6
```

Are the same people missing sbp and dbp?

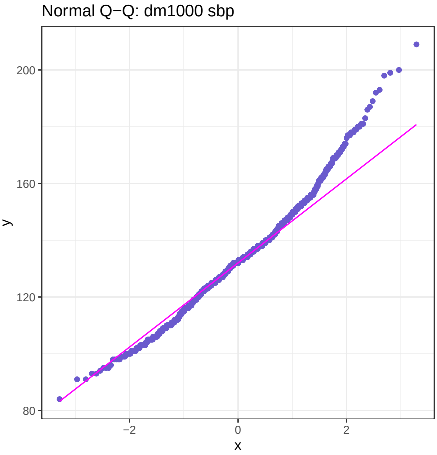
```
dm1000 %>% select(sbp, dbp) %>%
  miss_case_summary()
```

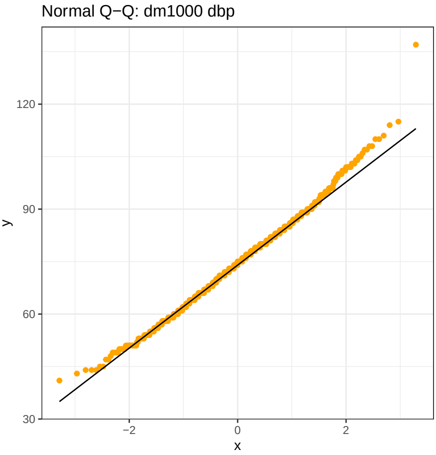
```
# A tibble: 1,000 x 3
    case n_miss pct_miss
   <int>  <int>    <dbl>
 1   107      2      100
 2   230      2      100
 3   284      2      100
 4   385      2      100
 5   440      2      100
 6   970      2      100
 7     1      0        0
 8     2      0        0
 9     3      0        0
10     4      0        0
# ... with 990 more rows
```

Distributions of `sbp` and `dbp`



Normal model for `sbp` and `dbp`?

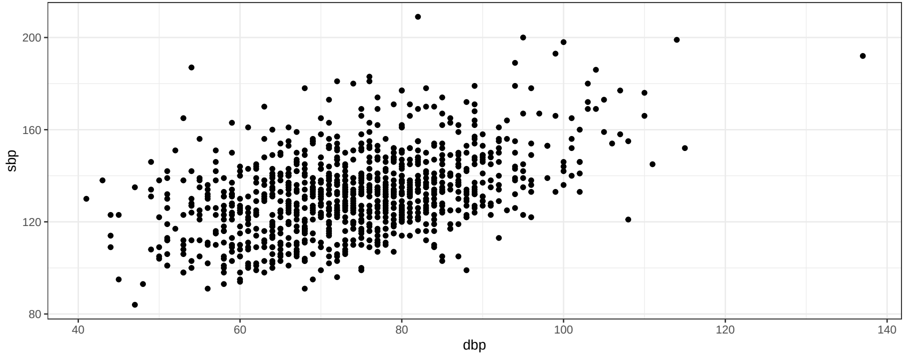




How closely associated are sbp and dbp?

```
ggplot(data = dm1000, aes(x = dbp, y = sbp)) +
  geom_point()
```

```
Warning: Removed 6 rows containing missing values
(geom_point).
```

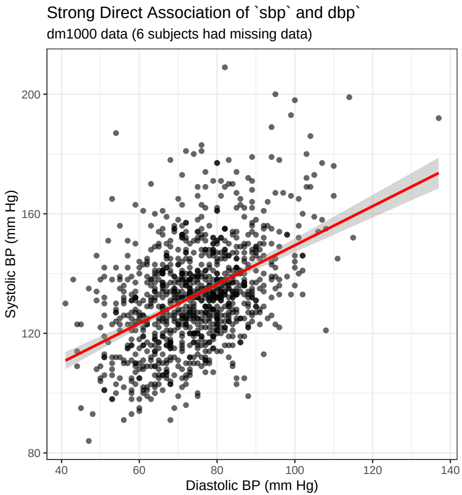


Improving the scatterplot (code)

```
dm1000 %>% filter(complete.cases(sbp, dbp)) %>%
ggplot(data = ., aes(x = dbp, y = sbp)) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", col = "red",
              formula = y ~ x, se = TRUE) +
  theme(aspect.ratio = 1) +
  labs(x = "Diastolic BP (mm Hg)",
       y = "Systolic BP (mm Hg)",
       title = "Strong Direct Association of `sbp` and dbp`",
       subtitle = "dm1000 data (6 subjects had missing data)")
```

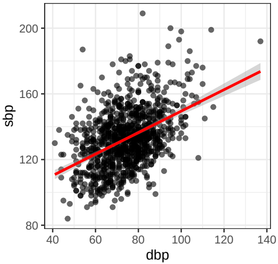
- What am I doing in these lines of code?

Higher DBP is associated with Higher SBP



- One point for each of the 994 subjects with known SBP and DBP...

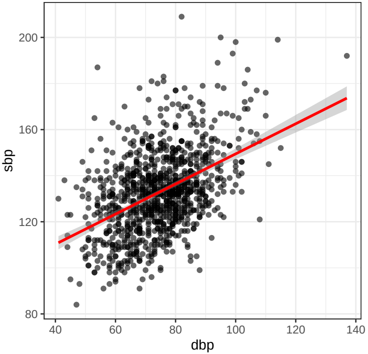
What are we looking for in this plot?



Is the association. . .

1. **Linear or Non-Linear?** (is there a curve here?)
2. **Direction?** (as X increases, what happens to Y?)
3. **Outliers?** (far away on X, or Y, or the combination?)
4. **Strength?** (points closely clustered together around a line?)

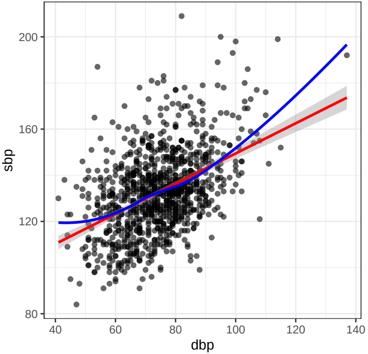
What might we conclude here?



1. **Linear?**: The points roughly follow the straight line's path.

- Do you see any clear signs of a curve?
- Would adding a loess smooth help us?

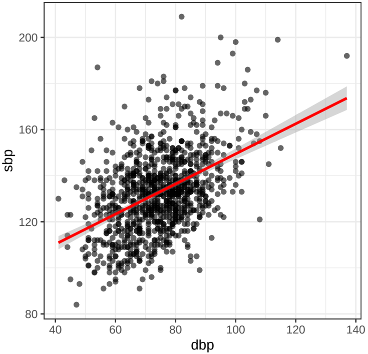
What might we conclude here?



1. **Linear?**

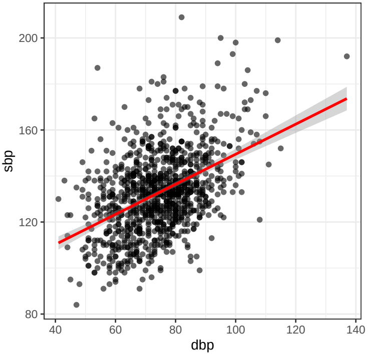
- The loess smooth (in blue) suggests a potential curve
- Is it overreacting to the highly leveraged point (`dbp` = 140)?

What might we conclude here?



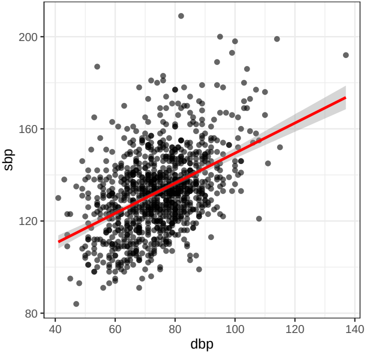
2. **Direction?**
   - As `dbp` increases, so does `sbp`, generally.
   - Slope of the regression line is positive.

What might we conclude here?



1. **Linear?**: No strong evidence of a meaningful curve.
2. **Direction?**: As `dbp` increases, so does `sbp`, generally.
3. **Outliers?**: A few (out of 1000) worth another look, probably.

What might we conclude here?



4. **Strength?**: Does this association seem very strong?

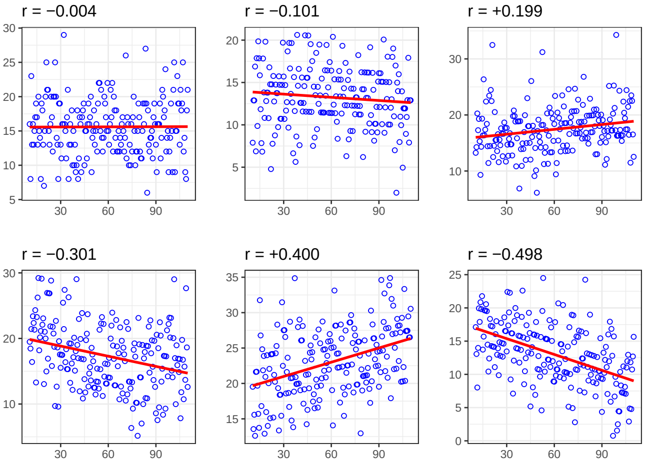
- `sbp` values associated with any particular `dbp` value range widely.
- If we know the `dbp`, that should help us make better predictions of `sbp`, but how much better than if we didn't know `dbp`?
- What might the **correlation** of `sbp` and `dbp` might be?

Summarizing Strength with the Pearson Correlation

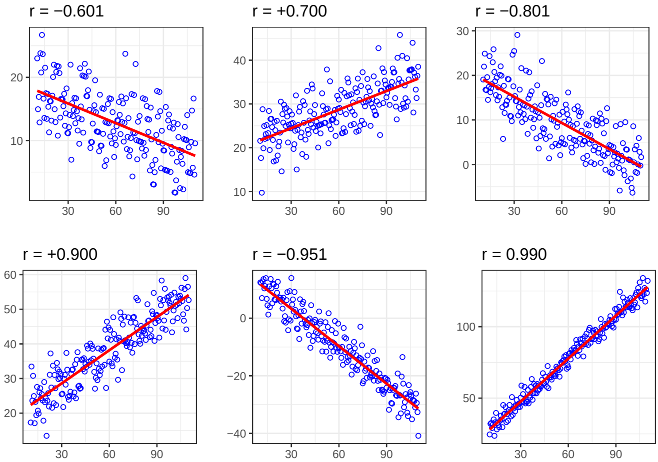
The Pearson correlation (abbreviated r) ranges from -1 to $+1$.

- 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.

Gaining Some Insight into Correlation



Some Stronger Correlations



(Pearson) Correlation Coefficients for sbp and dbp

```
dm1000 %$% cor(sbp, dbp)
```

```
[1] NA
```

```
dm1000 %>%
  filter(complete.cases(sbp, dbp)) %$%
  cor(sbp, dbp)
```

```
[1] 0.4521072
```

```
dm1000 %$% cor(sbp, dbp, use = "complete.obs")
```

```
[1] 0.4521072
```

- What does this correlation imply about a linear fit to the data?

What line is being fit in our model m1?

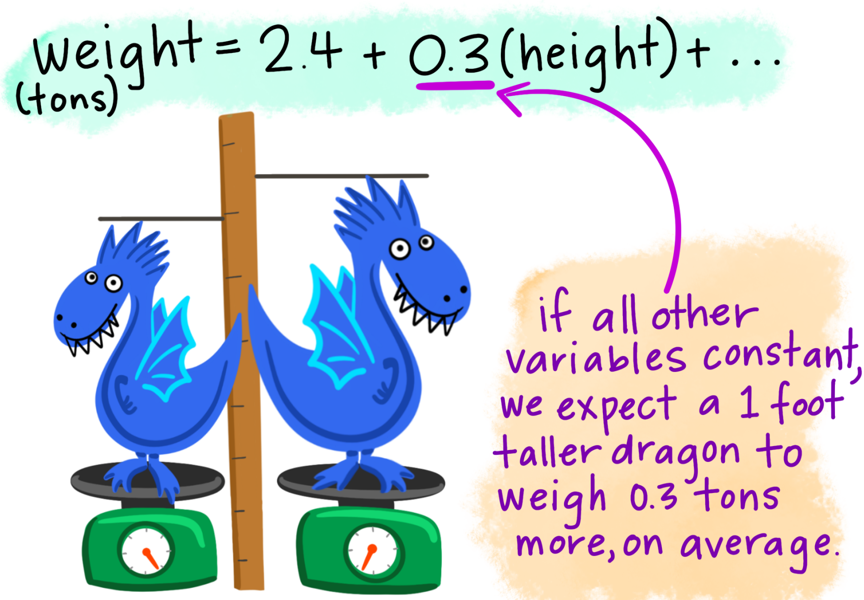
Least Squares Regression Line (a linear model) to predict sbp using dbp

```
m1 <- lm(sbp ~ dbp, data = dm1000)
m1
```

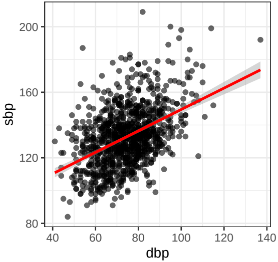
```
Call:
lm(formula = sbp ~ dbp, data = dm1000)

Coefficients:
(Intercept)          dbp
    84.1147       0.6535
```

Model m1 is **sbp = 84.11 + 0.65 dbp**.

What does the slope mean?



Linear Model `m1`: sbp = 84.11 + 0.65 dbp

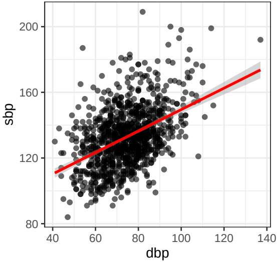


84.11 is the intercept = predicted value of `sbp` when `dbp` = 0.

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

- What does the model predict for `sbp` for a subject with `dbp` = 100?
- What if the subject had `dbp` = 99? 101? 110?

Linear Model `m1`: sbp = 84.11 + 0.65 dbp



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

0.65 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 would the line look like if the slope was zero?

Confidence Intervals for Regression Coefficients

We'll use the `tidy()` function from the `broom` package.

```
tidy(m1, conf.int = TRUE, conf.level = 0.90) %>%
  select(term, estimate, std.error, conf.low, conf.high) %>%
  kable(digits = 4)
```

term	estimate	std.error	conf.low	conf.high
(Intercept)	84.1147	3.0901	79.0271	89.2022
dbp	0.6535	0.0409	0.5861	0.7209

- How might we interpret the confidence interval for the slope of `dbp`?
  - Remember that the slope is the change in `sbp` per 1 unit change in `dbp` according to our model `m1`.
- How might we interpret the intercept term in model `m1`?

Obtaining R^2 and some Regression Fit Summaries

We'll use the `glance()` function, also from the `broom` package.

```
glance(m1) %>%
  select(nobs, r.squared, adj.r.squared, AIC, BIC) %>%
  kable(digits = c(0, 4, 4, 1, 1))
```

nobs	r.squared	adj.r.squared	AIC	BIC
994	0.2044	0.2036	8339.3	8354

- `nobs` = # of observations actually used to fit the model
- R^2 = "r-squared" is the square of the Pearson correlation r.
    - Recall we had $r = 0.4521$ for the association of `sbp` and `dbp`.
    - Squaring r, we get 0.2044.
- R^2 can be interpreted as the percentage of variation in `sbp` that `m1` accounts for with `dbp`

Interpreting R^2 and other Regression Summaries

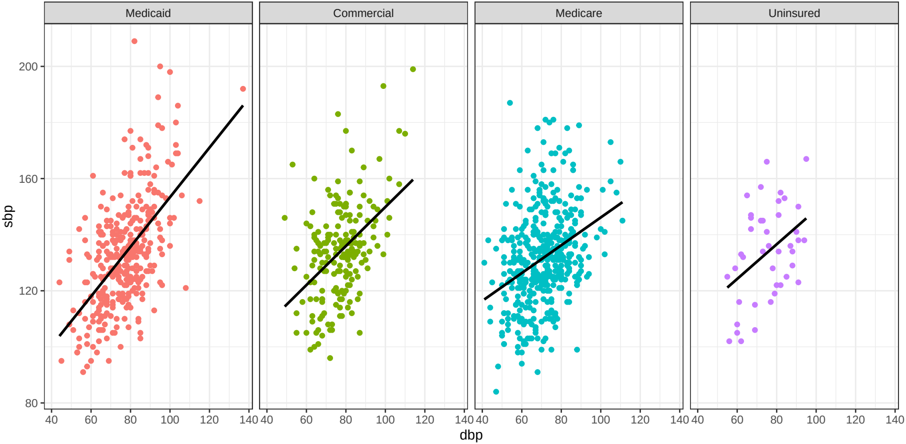
```
glance(m1) %>%
  select(nobs, r.squared, adj.r.squared, AIC, BIC) %>%
  kable(digits = c(0, 4, 4, 1, 1))
```

nobs	r.squared	adj.r.squared	AIC	BIC
994	0.2044	0.2036	8339.3	8354

- R^2 is also the proportionate reduction in error (as measured by sum of squared errors) in our predictions made using `m1` as compared to an "intercept only" regression model where we simply predict the mean of `sbp` for any subject, regardless of their `dbp`.
- Adjusted R^2, AIC and BIC will become relevant as we compare multiple models for the same outcome.

sbp-dbp association in insurance subgroups?

- Different linear model for `sbp` using `dbp` in each `insurance` category.



Code for previous slide

```
dm1000 %>% filter(complete.cases(sbp, dbp, insurance)) %>%
  ggplot(data = ., aes(x = dbp, y = sbp, col = insurance)) +
  geom_point() +
  geom_smooth(method = "lm", col = "black",
              formula = y ~ x, se = FALSE) +
  guides(col = "none") +
  facet_wrap(~ insurance, nrow = 1)
```

Does sbp-dbp correlation vary by insurance?

```
dm1000 %>%
  filter(complete.cases(insurance, sbp, dbp)) %>%
  group_by(insurance) %>%
  summarize(n = n(), pearson_r = cor(sbp, dbp), r.squared = pe
  kable(digits = 3)
```

insurance	n	pearson_r	r.squared
Medicaid	330	0.577	0.332
Commercial	193	0.452	0.204
Medicare	429	0.346	0.120
Uninsured	42	0.413	0.171

- How might we fit a linear model within each insurance type?
- Which of those models would have the largest R^2?

Model for subjects with Medicare insurance?

```
m2_medicare <- dm1000 %>%
  filter(insurance == "Medicare") %>%
  filter(complete.cases(sbp, dbp)) %$%
  lm(sbp ~ dbp)

tidy(m2_medicare, conf.int = TRUE, conf.level = 0.90) %>%
  select(term, estimate, conf.low, conf.high) %>%
  kable(digits = 3)
```

term	estimate	conf.low	conf.high
(Intercept)	96.589	88.889	104.290
dbp	0.495	0.388	0.603

Glancing at the Medicare-Only Model

```
glance(m2_medicare) %>%
  select(r.squared, nobs)
```

```
# A tibble: 1 x 2
  r.squared  nobs
      <dbl> <int>
1     0.120   429
```

Model including both dbp and insurance?

```
m3 <-
  dm1000 %>%
  filter(complete.cases(sbp, dbp, insurance)) %$%
  lm(sbp ~ dbp * insurance)

glance(m3) %>% select(nobs, r.squared, adj.r.squared) %>%
  kable(digits = c(0, 3, 3))
```

nobs	r.squared	adj.r.squared
994	0.222	0.217

Coefficients of Model m3

```
tidy(m3) %>% select(term, estimate, std.error) %>%
  kable(digits = 3)
```

term	estimate	std.error
(Intercept)	64.948	5.395
dbp	0.884	0.069
insuranceCommercial	15.337	9.613
insuranceMedicare	31.641	7.107
insuranceUninsured	22.247	17.604
dbp:insuranceCommercial	-0.188	0.123
dbp:insuranceMedicare	-0.389	0.094
dbp:insuranceUninsured	-0.267	0.231

- What does this model imply for Medicare subjects?

Understanding the `m3` model

Model `m3` predicts `sbp` using

```
  64.948                + 0.884 `dbp`
+ 31.641 Medicare - 0.389 `dbp` * Medicare
+ 15.337 Commer.  - 0.188 `dbp` * Commer.
+ 22.247 Medicaid - 0.267 `dbp` * Medicaid
```

1. What is the resulting equation for a Medicare subject?

Understanding the m3 model

Model m3 predicts sbp using

```
  64.948                + 0.884 `dbp`
+ 31.641 Medicare - 0.389 `dbp` * Medicare
+ 15.337 Commer.  - 0.188 `dbp` * Commer.
+ 22.247 Medicaid - 0.267 `dbp` * Medicaid
```

What is the resulting equation for a Medicare subject?

```
sbp = (64.948 + 31.641) + (0.884 - 0.389) * dbp
          sbp = 96.589 + 0.495 dbp
```

- This matches the result we obtained running the sbp on dbp regression for the Medicare subjects alone in model m2_medicare.

Understanding the m3 model

Again, model m3 predicts sbp using

```
  64.948              + 0.884 `dbp`
+ 31.641 Medicare - 0.389 `dbp` * Medicare
+ 15.337 Commer.  - 0.188 `dbp` * Commer.
+ 22.247 Medicaid - 0.267 `dbp` * Medicaid
```

Insurance	Predicted sbp
Medicare	96.589 + 0.495 dbp
Commercial	(64.948 + 15.337) + (0.884 - 0.188) dbp
Commercial	or, 80.285 + 0.696 dbp
Medicaid	87.195 + 0.617 dbp
Uninsured	64.948 + 0.884 dbp

Which model shows better fit to the data?

```
g1 <- glance(m1) %>%
  mutate(m_name = "m1 (dbp only)")
g3 <- glance(m3) %>%
  mutate(m_name = "m3 (dbp * insurance)")

bind_rows(g1, g3) %>%
  select(m_name, nobs, r.squared, adj.r.squared, AIC, BIC) %>%
  kable(digits = c(0, 0, 3, 3, 0, 0))
```

m_name	nobs	r.squared	adj.r.squared	AIC	BIC
m1 (dbp only)	994	0.204	0.204	8339	8354
m3 (dbp * insurance)	994	0.222	0.217	8329	8373

- Model `m3` has better R^2, and adjusted R^2; better AIC, but worse BIC.
- IGNORING: regression assumptions, and predictions in new data...

Coming Up

- More with your favorite movies
- Associations between categorical variables