Version 1 Draft - 432 Homework 2 Answer Sketch

432 TAs

Due 2020-02-04. Version: 2020-02-01

Contents

Setup and Data Ingest

```
library(ggrepel)
library(viridis)
library(broom)
library(janitor)
library(caret)
library(magrittr)
library(here)
library(tidyverse)

ohc <- read_csv(here("/data/oh_counties_2017.csv")) %>%
    clean_names()
```

The `oh_counties_2017.csv` data set I have provided describes a series of variables, pulled from the data for the 88 counties of the the State of Ohio from the County Health Rankings report for 2017.

- You may also be interested in looking at the details of the 2017 Ohio Summary report (pdf), or at the Excel data file from which I created the `oh_counties_2017.csv` file.
- Note that later data sets are now available, but we will concentrate in this assignment on the 2017 results.

The available variables are listed below. Each variable describes data at the **COUNTY** level.

Variable	Description
fips	Federal Information Processing Standard code
county	name of County
years_lost_rate	age-adjusted years of potential life lost rate (per 100,000 population)
sroh_fairpoor	% of adults reporting fair or poor health (via BRFSS)
phys_days	mean number of reported physically unhealthy days per month
ment_days	mean number of reported mentally unhealthy days per mo
lbw_pct	% of births with low birth weight (< 2500 grams)
smoker_pct	% of adults that report currently smoking
obese_pct	% of adults that report body mass index of 30 or higher
food_env	indicator of access to healthy foods, in points (0 is worst, 10 is best)
inactive_pct	% of adults that report no leisure-time physical activity
exer_access	% of the population with access to places for physical activity
exc_drink	% of adults that report excessive drinking
alc_drive	% of driving deaths with alcohol involvement
sti_rate	Chlamydia cases / Population x 100,000
teen_births	Teen births / females ages 15-19 x 1,000
uninsured	% of people under age 65 without insurance
pcp_ratio	Population to Primary Care Physicians ratio
prev_hosp	Discharges for Ambulatory Care Sensitive Conditions/Medicare Enrollees x 1,000
hsgrads	High School graduation rate
unemployed	% of population age 16+ who are unemployed and looking for work
poor_kids	% of children (under age 18) living in poverty
income_ratio	Ratio of household income at the 80th percentile to income at the 20th percentile
associations	# of social associations / population x 10,000
pm2.5	Average daily amount of fine particulate matter in micrograms per cubic meter
h2oviol	Presence of a water violation: Yes or No
sev_housing	% of households with at least 1 of 4 housing problems: overcrowding, high housing costs, or lack of kitchen or plumbing facilities
drive_alone	% of workers who drive alone to work

Variable	Description
age.adj.mortality	premature age-adjusted mortality
dm_prev	% with a diabetes diagnosis
freq_phys_distress	% in frequent physical distress
freq_mental_distress	% in frequent mental distress
food_insecure	% who are food insecure
insuff_sleep	% who get insufficient sleep
health_costs	estimated mean health care costs
median_income	estimated median income
population	population size
age65plus	% of population who are 65 and over
african-am	% of population who are African-American
hispanic	% of population who are of Hispanic/Latino ethnicity
white	% of population who are White
female	% of population who are Female
rural	% of people in the county who live in rural areas

Question 1 (20 points)

Create a visualization (using R) based on some part of the `oh_counties_2017.csv` data set, and share it (the visualization and the R code you used to build it) with us. The visualization should be of a professional quality, describe information from at least three different variables listed above, include proper labels and a title, as well as a caption of no more than 50 words that highlights the key result. Although you may fit a model to help show patterns, your primary task is to show **the data** in a meaningful way, rather than to simply highlight the results of a model.

- You are welcome to find useful tools for visualizing data in R that we have yet to see in the slides in class, but which you've found in your reading of David Spiegelhalter's *The Art of Statistics* or in other venues.
- We will grade Question 1 strictly based on the quality of the visualization, its title and caption, in terms of being attractive, well-labeled and useful for representing the County Health Rankings data for Ohio, and how well it adheres to general principles for good visualizations we've seen in 431 and 432.

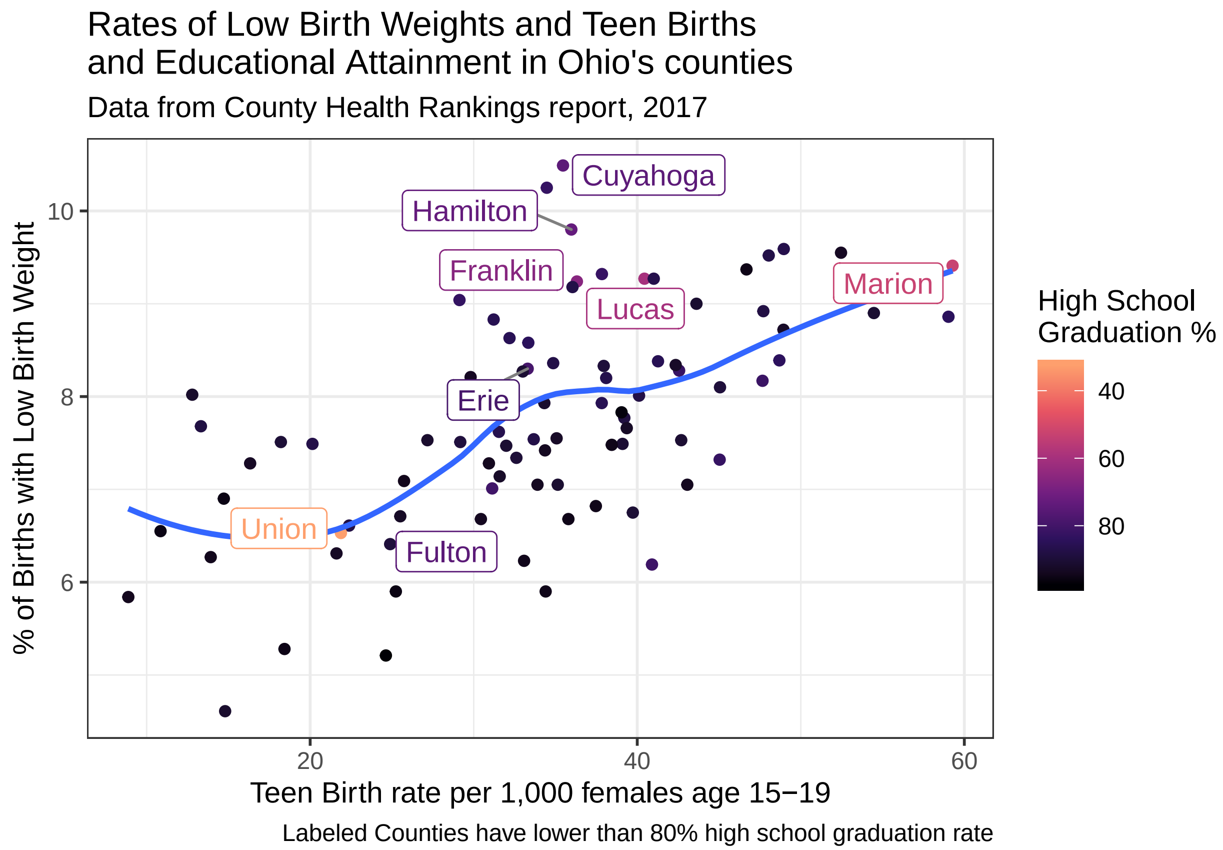
Answer for Question 1

There are, literally, thousands of appropriate options here, so it's difficult to write a sketch. Of course, that statement is true for all of these questions.

What Dr. Love assumed most people would do is some sort of labeled scatterplot. Here's an example.

```r
ggplot(ohc, aes(x = teen_births, y = lbw_pct,
                col= hsgrads))  +
    geom_point() +
    geom_smooth(method = "loess", se = FALSE) +
    geom_label_repel(aes(
        label=ifelse(hsgrads<80, as.character(county),'')),
        segment.color = 'grey50') +
    theme_bw() +
    scale_color_viridis_c(end = 0.8, option = "magma",
        (name="High School \nGraduation %"),
```

```
        trans = "reverse") +
    labs(
        title = "Rates of Low Birth Weights and Teen Births\nand Educational Attainment in Ohio's count
        subtitle = "Data from County Health Rankings report, 2017",
        caption="Labeled Counties have lower than 80% high school graduation rate",
        y = "% of Births with Low Birth Weight",
        x = "Teen Birth rate per 1,000 females age 15-19")
```



Some people built maps, as well. That's great, but it's hard to show more than one variable at a time on a map of Ohio's counties. Dr. Love will share some of the plots we found to be most effective in class.

Grading Rubric: Question 1

- Award 17 points for a good effort and 18-20 points for an especially strong effort (one of the top 3-6 visualizations we saw).
- Anything that seems way off the mark should get 13 or less, and should be drawn to my attention, as should the graphs that get 18-20 points.
- Take off 1 point from the total if you find one or two typographical or syntax/grammar errors in this response.
- Take off 3 points from the total if you find three or more such errors.

I expect most people to score 14-17 points on this question.

Question 2 (20 points)

Write an essay (between 150 and 250 words) describing the creation and meaning of the visualization you created in Question 1, providing us with the context we need to understand why this is a useful visualization. In your short description, be sure to address:

- How does this visualization help its audience understand the data better?
- Why is this particular visualization effective, and what are the design features it uses that we can learn from to help us make more effective visualizations?

Answer 2

We don't write sketches for essay questions, but the takeaway from the labeled scatterplot provided above should draw attention to:

- the general pattern (% of low birth weight births generally rises with the teen birth rate) displayed,
- other features of the data (outlying counties at the high end also generally have lower educational attainment),
- plus identify the outliers (Union County, in terms of HS Graduation Rate, but also Cuyahoga in terms of low birth weight and Marion, in terms of high teen birth rate.)

Grading Rubric: Question 2

- Award 17 points for any essay that is within the word limit (or close) and answers the two questions reasonably well.
- Award 18-20 points for an essay that does a really nice job.
- Award 13 or fewer points to anything that doesn't address both questions, or is way off the mark.
- Take off 1 point from the total if you find one or two typographical or syntax/grammar errors in this response.
- Take off 3 points from the total if you find three or more such errors.

Question 3 (10 points)

Create a linear regression model to predict `obese_pct` as a function of `food_env` adjusting for `median_income`, and treating all three variables as quantitative, using all counties with complete data on those three variables. Specify and then carefully interpret the estimated coefficient of `food_env` and a 90% uncertainty interval around that estimate in context using nothing but complete English sentences. A model using main effects only, entered as linear predictors, will be sufficient.

Answer 3

The model we had in mind was

```
model1 <- lm(obese_pct ~ food_env + median_income, data = ohc)

tidy(model1, conf.int = TRUE, conf.level = 0.90)
```

```
# A tibble: 3 x 7
  term           estimate std.error statistic  p.value  conf.low conf.high
  <chr>             <dbl>     <dbl>     <dbl>    <dbl>     <dbl>     <dbl>
1 (Intercept)   32.8      3.17          10.4  1.00e-16 27.5      38.1
2 food_env       1.14     0.547          2.09 3.99e- 2  0.232     2.05
3 median_income -0.000166 0.0000363     -4.57 1.66e- 5 -0.000226 -0.000105
```

- The value of the estimated effect of a change of 1 point in `food_env` is an increase of 1.14 percentage points in `obese_pct`, with a 90% CI of (0.23, 2.05), assuming `median_income` is unchanged.

Grading Rubric: Question 3

- Estimating the result correctly is worth 3 points.
- The other seven points are awarded if the student has an excellent description of the effect size. That description should:
  - describe a comparison between two *counties*, with the *same median income*, who differ by 1 *point* in `food_env` and come to a conclusion about the *size* and the *direction* of the effect on `obese_pct`, while using *appropriate units* for `food_env` and `obese_pct` and indicating the *statistical significance* of the result at the *10% significance level* (or *90% confidence level*) with reference to the confidence interval.
  - the seven key elements are:
    1. this model describes counties, not subjects/individuals/whatever
    2. holding the same median income
    3. comparison of two counties with differing food environments with the correct units for the food environment measure
    4. description of effect on an outcome with the correct units for the outcome
    5. size of effect specified to match their model
    6. direction of effect specified to match their model
    7. significance of effect and level of confidence/significance
- Showing the effect graphically is a nice, but ungraded, touch.

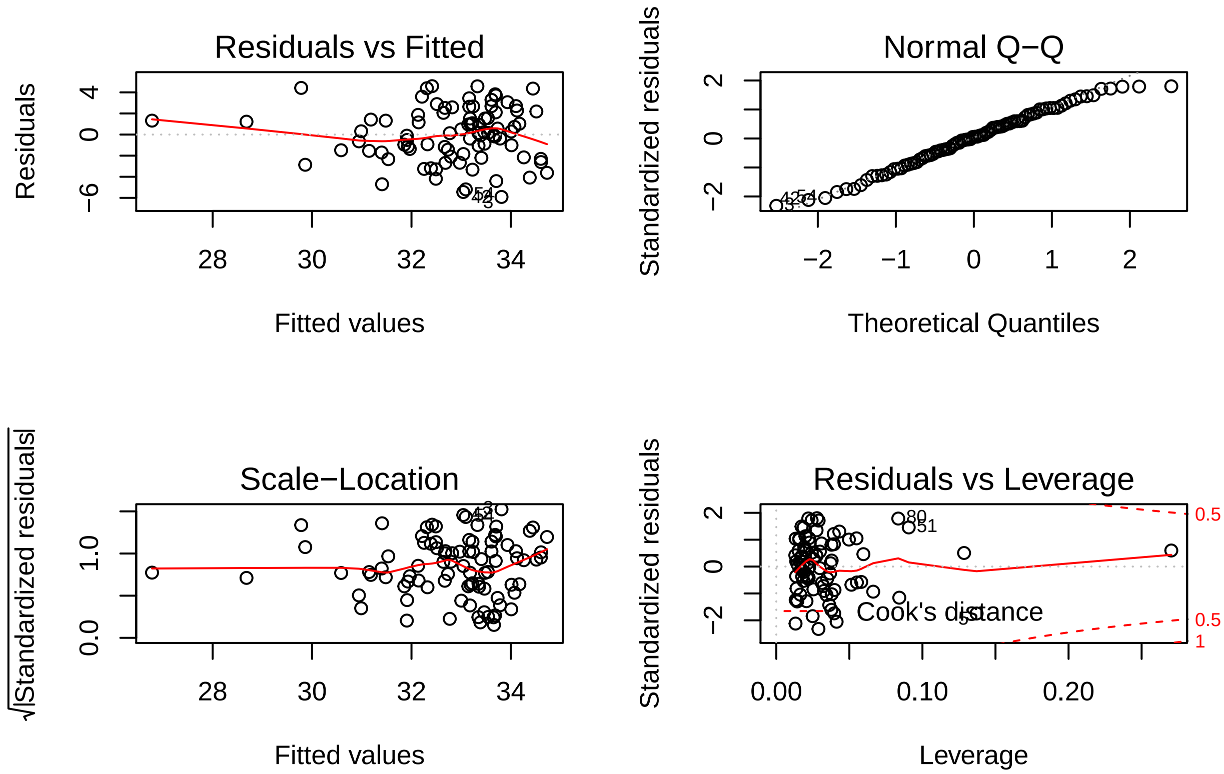
Question 4 (10 points)

Evaluate the quality of the model you fit in Question 3 in terms of adherence to regression modeling assumptions, through the specification and written evaluation of residual plots and other diagnostics. What might be done to improve the fit of the model you've developed in Question 3? Identify by name any outlying counties, in terms of the relationship you're studying, and explain why they are flagged as outliers.

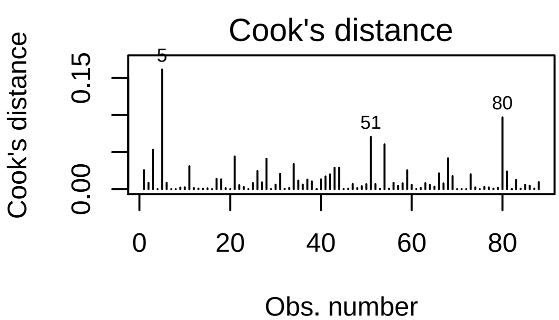
Answer 4

- You have created a linear regression model in Question 3 that predicts `obese_pct` as a function of `food_env` adjusting for `median_income`. To evaluate the quality of the model in terms of adherence to regression modeling assumptions, plotting the model might be a good idea. Remember, you are trying to check the following four assumptions: linearity, constant variance (Homoscedasticity), normality and independence. Based on the plots below, the residual plots look reasonable for linearity, normality and constant variance and there are no influential values or outliers.

```r
par(mfrow=c(2,2))
plot(model1)
```




```r
plot(model1, 4)
```



- Assessing the fit of the model: the current model explains about 19% of the variation in `obese_pct`. To improve the model, you may consider adding more predictor variables and using an automated procedure like stepwise regression to find a better alternative.

```r
m1 <- glance(model1) %>%
    mutate(name="mod_1") %>%
    select(name, r.squared, adj.r.squared, sigma, AIC, BIC) %>%
    knitr::kable(digits = c(0,4,4,4,4,4))
m1
```

name	r.squared	adj.r.squared	sigma	AIC	BIC
mod_1	0.208	0.1894	2.5829	421.6915	431.6008

Grading Rubric: Question 4

- 1.5 points for evaluating each of the four main plots (1, 2, 3, and 5)
- 3 points for correctly identifying the amount of variation (19%) in `obese_pct` explained by the model.
- 1 point for making a reasonable suggestion on how to improve the model

Question 5 (20 points)

Create a logistic regression model to predict the presence of a water violation (as contained in `h2oviol`) on the basis of `sev_housing` and `pm2.5`. Specify and then carefully interpret the estimated coefficient of `sev_housing` and a 90% uncertainty interval around that estimate in context using nothing but complete English sentences. A model using main effects only, entered as linear predictors, will be sufficient.

Answer 5

The model we had in mind was

```
model2 <- glm(h2oviol == "Yes" ~ sev_housing + pm2_5, data = ohc, family = binomial())

tidy(model2, conf.int = TRUE, conf.level = 0.90, exponentiate = TRUE)
```

```
# A tibble: 3 x 7
  term        estimate std.error statistic p.value    conf.low conf.high
  <chr>          <dbl>     <dbl>     <dbl>   <dbl>       <dbl>     <dbl>
1 (Intercept)  0.00303     5.36      -1.08     0.280 0.000000311     17.5
2 sev_housing  1.03        0.0869     0.300    0.764 0.889            1.19
3 pm2_5        1.60        0.467      1.01     0.313 0.752            3.55
```

- The value of the estimated effect of a change of 1 percentage point in `sev_housing` is an increase by a factor of 1.026 in the odds of a water violation, with a 90% CI of (0.89, 1.19), assuming `pm2_5` is unchanged.

Grading Rubric: Question 5

- Estimating the result correctly is worth 6 points.
- The other fourteen points are awarded if the student has an excellent description of the effect size. That description should:
  - describe a comparison between two *counties*, with the *same pm2_5*, who differ by 1 *percentage point* in `sev_housing` and come to a conclusion about the *size* and the *direction* of the effect on `h2oviol` in terms of an odds ratio, while using *appropriate units* for all variables, referring to odds changes, rather than changes in risk or probability and indicating the *statistical significance* of the result at the *10% significance level* (or *90% confidence level*) with reference to the confidence interval.
  - the seven key elements (2 points each) are:
    1. this model describes counties, not subjects/individuals/whatever
    2. holding the same pm2.5 or pm2_5 (depending on whether they cleaned the names)
    3. comparison of two counties with differing percentages meeting the `sev_housing` standard
    4. description of effect on the `h2oviol` odds
    5. size of effect specified to match their model

6. direction of effect specified to match their model (which should be about the odds of having a violation)
7. (lack of) significance of effect and level of confidence/significance

- Showing the effect graphically is a nice, but ungraded, touch.

Question 6 (10 points)

Produce and interpret the meaning of a confusion matrix which describes the quality of the predictions you have developed in Question 5. How well does the model you fit make predictions about `h2oviol`?

Answer 6

Get predictions for all subjects in our data

The first step is to get the predictions using the `augment` function from the `broom` library. Then, we can create a confusion matrix to get and present it in standard epidemiological format, as in the first example, or we can use the `ConfusionMatrix` from the `carat` library to produce more useful summaries. Based on the findings, our model has a sensitivity of 35.7%, specificity of 71.7%, positive predictive value of 53.6% and a negative predictive value of 55.0%.

- If the county actually had a water violation, our model predicts that 35.7% of the time.
- If the county actually didn't have a water violation, our model predicts that 71.7% of the time.
- Our predictions of water violations were correct 53.6% of the time.
- Our predictions of not having a water violation were correct 55.0% of the time.

```
model2_aug <- augment(model2, ohc, type.predict = "response")
```

Making the confusion matrix

```
confuseMat_small <- model2_aug %>%
  mutate(h2oviol_act = factor(h2oviol == "Yes"),
     h2oviol_pre = factor(.fitted >= 0.5),
     h2oviol_act = fct_relevel(h2oviol_act, "TRUE"),
     h2oviol_pre = fct_relevel(h2oviol_pre, "TRUE")) %$%
  table(h2oviol_pre, h2oviol_act)

confuseMat_small
```

```
           h2oviol_act
h2oviol_pre TRUE FALSE
      TRUE    15    13
      FALSE   27    33
```

```
confuseMat <- model2_aug %$% confusionMatrix(
  data = factor(.fitted >= 0.5),
  reference = factor(h2oviol == "Yes"),
  positive = "TRUE")

tidy(confuseMat) %>% knitr::kable(digits = 3)
```

term	class	estimate	conf.low	conf.high	p.value
accuracy	NA	0.545	0.436	0.652	0.375
kappa	NA	0.076	NA	NA	0.040
sensitivity	TRUE	0.357	NA	NA	NA
specificity	TRUE	0.717	NA	NA	NA
pos_pred_value	TRUE	0.536	NA	NA	NA
neg_pred_value	TRUE	0.550	NA	NA	NA
precision	TRUE	0.536	NA	NA	NA
recall	TRUE	0.357	NA	NA	NA
f1	TRUE	0.429	NA	NA	NA
prevalence	TRUE	0.477	NA	NA	NA
detection_rate	TRUE	0.170	NA	NA	NA
detection_prevalence	TRUE	0.318	NA	NA	NA
balanced_accuracy	TRUE	0.537	NA	NA	NA

Grading Rubric: Question 6

- 6 points for create a correct confusion matrix
- 1 point for each correct interpretation of the main descriptors of the predictions (sensitivity, specificity, PPV, NPV)

Session Information

```r
sessioninfo::session_info()
```

```
- Session info ---------------------------------------------------------------
 setting  value
 version  R version 3.6.2 (2019-12-12)
 os       macOS Catalina 10.15.2
 system   x86_64, darwin15.6.0
 ui       X11
 language (EN)
 collate  en_US.UTF-8
 ctype    en_US.UTF-8
 tz       America/New_York
 date     2020-02-01

- Packages -------------------------------------------------------------------
 package      * version    date       lib source
 assertthat     0.2.1      2019-03-21 [1] CRAN (R 3.6.0)
 backports      1.1.5      2019-10-02 [1] CRAN (R 3.6.0)
 broom        * 0.5.4      2020-01-27 [1] CRAN (R 3.6.2)
 caret        * 6.0-85     2020-01-07 [1] CRAN (R 3.6.0)
 cellranger     1.1.0      2016-07-27 [1] CRAN (R 3.6.0)
 class          7.3-15     2019-01-01 [1] CRAN (R 3.6.2)
 cli            2.0.1      2020-01-08 [1] CRAN (R 3.6.0)
 codetools      0.2-16     2018-12-24 [1] CRAN (R 3.6.2)
 colorspace     1.4-1      2019-03-18 [1] CRAN (R 3.6.0)
 crayon         1.3.4      2017-09-16 [1] CRAN (R 3.6.0)
 data.table     1.12.8     2019-12-09 [1] CRAN (R 3.6.0)
```

```
DBI              1.1.0      2019-12-15 [1] CRAN (R 3.6.0)
dbplyr           1.4.2      2019-06-17 [1] CRAN (R 3.6.0)
digest           0.6.23     2019-11-23 [1] CRAN (R 3.6.0)
dplyr          * 0.8.3      2019-07-04 [1] CRAN (R 3.6.0)
e1071            1.7-3      2019-11-26 [1] CRAN (R 3.6.0)
evaluate         0.14       2019-05-28 [1] CRAN (R 3.6.0)
fansi            0.4.1      2020-01-08 [1] CRAN (R 3.6.0)
farver           2.0.3      2020-01-16 [1] CRAN (R 3.6.0)
forcats        * 0.4.0      2019-02-17 [1] CRAN (R 3.6.0)
foreach          1.4.7      2019-07-27 [1] CRAN (R 3.6.0)
fs               1.3.1      2019-05-06 [1] CRAN (R 3.6.0)
generics         0.0.2      2018-11-29 [1] CRAN (R 3.6.0)
ggplot2        * 3.2.1      2019-08-10 [1] CRAN (R 3.6.0)
ggrepel        * 0.8.1      2019-05-07 [1] CRAN (R 3.6.0)
glue             1.3.1      2019-03-12 [1] CRAN (R 3.6.0)
gower            0.2.1      2019-05-14 [1] CRAN (R 3.6.0)
gridExtra        2.3        2017-09-09 [1] CRAN (R 3.6.0)
gtable           0.3.0      2019-03-25 [1] CRAN (R 3.6.0)
haven            2.2.0      2019-11-08 [1] CRAN (R 3.6.0)
here           * 0.1        2017-05-28 [1] CRAN (R 3.6.0)
highr            0.8        2019-03-20 [1] CRAN (R 3.6.0)
hms              0.5.3      2020-01-08 [1] CRAN (R 3.6.0)
htmltools        0.4.0      2019-10-04 [1] CRAN (R 3.6.0)
httr             1.4.1      2019-08-05 [1] CRAN (R 3.6.0)
ipred            0.9-9      2019-04-28 [1] CRAN (R 3.6.0)
iterators        1.0.12     2019-07-26 [1] CRAN (R 3.6.0)
janitor        * 1.2.1      2020-01-22 [1] CRAN (R 3.6.0)
jsonlite         1.6        2018-12-07 [1] CRAN (R 3.6.0)
knitr            1.27       2020-01-16 [1] CRAN (R 3.6.0)
labeling         0.3        2014-08-23 [1] CRAN (R 3.6.0)
lattice        * 0.20-38    2018-11-04 [1] CRAN (R 3.6.0)
lava             1.6.6      2019-08-01 [1] CRAN (R 3.6.0)
lazyeval         0.2.2      2019-03-15 [1] CRAN (R 3.6.0)
lifecycle        0.1.0      2019-08-01 [1] CRAN (R 3.6.0)
lubridate        1.7.4      2018-04-11 [1] CRAN (R 3.6.0)
magrittr       * 1.5        2014-11-22 [1] CRAN (R 3.6.0)
MASS             7.3-51.5   2019-12-20 [1] CRAN (R 3.6.0)
Matrix           1.2-18     2019-11-27 [1] CRAN (R 3.6.2)
ModelMetrics     1.2.2.1    2020-01-13 [1] CRAN (R 3.6.0)
modelr           0.1.5      2019-08-08 [1] CRAN (R 3.6.0)
munsell          0.5.0      2018-06-12 [1] CRAN (R 3.6.0)
nlme             3.1-143    2019-12-10 [1] CRAN (R 3.6.0)
nnet             7.3-12     2016-02-02 [1] CRAN (R 3.6.0)
pillar           1.4.3      2019-12-20 [1] CRAN (R 3.6.0)
pkgconfig        2.0.3      2019-09-22 [1] CRAN (R 3.6.0)
plyr             1.8.5      2019-12-10 [1] CRAN (R 3.6.0)
pROC             1.16.1     2020-01-14 [1] CRAN (R 3.6.0)
prodlim          2019.11.13 2019-11-17 [1] CRAN (R 3.6.0)
purrr          * 0.3.3      2019-10-18 [1] CRAN (R 3.6.0)
R6               2.4.1      2019-11-12 [1] CRAN (R 3.6.0)
Rcpp             1.0.3      2019-11-08 [1] CRAN (R 3.6.0)
readr          * 1.3.1      2018-12-21 [1] CRAN (R 3.6.0)
readxl           1.3.1      2019-03-13 [1] CRAN (R 3.6.0)
recipes          0.1.9      2020-01-07 [1] CRAN (R 3.6.0)
```

```
 reprex          0.3.0      2019-05-16 [1] CRAN (R 3.6.0)
 reshape2        1.4.3      2017-12-11 [1] CRAN (R 3.6.0)
 rlang           0.4.4      2020-01-28 [1] CRAN (R 3.6.2)
 rmarkdown       2.1        2020-01-20 [1] CRAN (R 3.6.2)
 rpart           4.1-15     2019-04-12 [1] CRAN (R 3.6.2)
 rprojroot       1.3-2      2018-01-03 [1] CRAN (R 3.6.0)
 rstudioapi      0.10       2019-03-19 [1] CRAN (R 3.6.0)
 rvest           0.3.5      2019-11-08 [1] CRAN (R 3.6.0)
 scales          1.1.0      2019-11-18 [1] CRAN (R 3.6.0)
 sessioninfo     1.1.1      2018-11-05 [1] CRAN (R 3.6.0)
 snakecase       0.11.0     2019-05-25 [1] CRAN (R 3.6.0)
 stringi         1.4.5      2020-01-11 [1] CRAN (R 3.6.0)
 stringr       * 1.4.0      2019-02-10 [1] CRAN (R 3.6.0)
 survival        3.1-8      2019-12-03 [1] CRAN (R 3.6.0)
 tibble        * 2.1.3      2019-06-06 [1] CRAN (R 3.6.0)
 tidyr         * 1.0.2      2020-01-24 [1] CRAN (R 3.6.0)
 tidyselect      1.0.0      2020-01-27 [1] CRAN (R 3.6.2)
 tidyverse     * 1.3.0      2019-11-21 [1] CRAN (R 3.6.0)
 timeDate        3043.102   2018-02-21 [1] CRAN (R 3.6.0)
 utf8            1.1.4      2018-05-24 [1] CRAN (R 3.6.0)
 vctrs           0.2.2      2020-01-24 [1] CRAN (R 3.6.0)
 viridis       * 0.5.1      2018-03-29 [1] CRAN (R 3.6.0)
 viridisLite   * 0.3.0      2018-02-01 [1] CRAN (R 3.6.0)
 withr           2.1.2      2018-03-15 [1] CRAN (R 3.6.0)
 xfun            0.12       2020-01-13 [1] CRAN (R 3.6.0)
 xml2            1.2.2      2019-08-09 [1] CRAN (R 3.6.0)
 yaml            2.2.0      2018-07-25 [1] CRAN (R 3.6.0)

[1] /Library/Frameworks/R.framework/Versions/3.6/Resources/library
```