April 7, 2023

The results below are generated from an R script.

```
# Assignment: ASSIGNMENT 4.2.2  Housing Data Exercise
# Name: Ghanta, Madhavi
# Date: 2023-04-06

## Load the readxl package
library(readxl)

## Load the plyr package
library(plyr)

## Set the working directory to the root of your DSC 520 directory
setwd("C:/Users/mghan/Documents/dsc520")

##We interact with a few datasets in this course, one you are already familiar
##with, the 2014 American Community Survey and the second is a Housing dataset,
##that provides real estate transactions recorded from 1964 to 2016.  For this
##exercise, you need to start practicing some data transformation steps - which
##will carry into next week, as you learn some additional methods.  For this
##week, using either dataset (or one of your own - although I will let you know
##ahead of time that the Housing dataset is used for a later assignment, so not
##a bad idea for you to get more comfortable with now!), perform the following
##data transformations:

## Load the ‘data/week-6-housing.xlsx‘ to
housing_df <- read_excel("data/week-7-housing.xlsx")
str(housing_df)

## tibble [12,865 x 24] (S3: tbl_df/tbl/data.frame)
##  $ Sale Date               : POSIXct[1:12865], format: "2006-01-03" "2006-01-03" ...
##  $ Sale Price              : num [1:12865] 698000 649990 572500 420000 369900 ...
##  $ sale_reason             : num [1:12865] 1 1 1 1 1 1 1 1 1 1 ...
##  $ sale_instrument         : num [1:12865] 3 3 3 3 3 15 3 3 3 3 ...
##  $ sale_warning            : chr [1:12865] NA NA NA NA ...
##  $ sitetype                : chr [1:12865] "R1" "R1" "R1" "R1" ...
##  $ addr_full               : chr [1:12865] "17021 NE 113TH CT" "11927 178TH PL NE" "13315 174TH AVE N
##  $ zip5                    : num [1:12865] 98052 98052 98052 98052 98052 ...
##  $ ctyname                 : chr [1:12865] "REDMOND" "REDMOND" NA "REDMOND" ...
##  $ postalctyn              : chr [1:12865] "REDMOND" "REDMOND" "REDMOND" "REDMOND" ...
##  $ lon                     : num [1:12865] -122 -122 -122 -122 -122 ...
##  $ lat                     : num [1:12865] 47.7 47.7 47.7 47.6 47.7 ...
##  $ building_grade          : num [1:12865] 9 9 8 8 7 7 10 10 9 8 ...
##  $ square_feet_total_living: num [1:12865] 2810 2880 2770 1620 1440 4160 3960 3720 4160 2760 ...
##  $ bedrooms                : num [1:12865] 4 4 4 3 3 4 5 4 4 4 ...
```

```
##  $ bath_full_count         : num [1:12865] 2 2 1 1 1 2 3 2 2 1 ...
##  $ bath_half_count         : num [1:12865] 1 0 1 0 0 1 0 1 1 0 ...
##  $ bath_3qtr_count         : num [1:12865] 0 1 1 1 1 1 1 0 1 1 ...
##  $ year_built              : num [1:12865] 2003 2006 1987 1968 1980 ...
##  $ year_renovated          : num [1:12865] 0 0 0 0 0 0 0 0 0 0 ...
##  $ current_zoning          : chr [1:12865] "R4" "R4" "R6" "R4" ...
##  $ sq_ft_lot               : num [1:12865] 6635 5570 8444 9600 7526 ...
##  $ prop_type               : chr [1:12865] "R" "R" "R" "R" ...
##  $ present_use             : num [1:12865] 2 2 2 2 2 2 2 2 2 2 ...

summary(housing_df)

##    Sale Date                        Sale Price       sale_reason    sale_instrument
##  Min.   :2006-01-03 00:00:00.00   Min.   :    698   Min.   : 0.00   Min.   : 0.000
##  1st Qu.:2008-07-07 00:00:00.00   1st Qu.: 460000   1st Qu.: 1.00   1st Qu.: 3.000
##  Median :2011-11-17 00:00:00.00   Median : 593000   Median : 1.00   Median : 3.000
##  Mean   :2011-07-28 15:07:32.48   Mean   : 660738   Mean   : 1.55   Mean   : 3.678
##  3rd Qu.:2014-06-05 00:00:00.00   3rd Qu.: 750000   3rd Qu.: 1.00   3rd Qu.: 3.000
##  Max.   :2016-12-16 00:00:00.00   Max.   :4400000   Max.   :19.00   Max.   :27.000
##  sale_warning         sitetype          addr_full             zip5
##  Length:12865       Length:12865       Length:12865       Min.   :98052
##  Class :character   Class :character   Class :character   1st Qu.:98052
##  Mode  :character   Mode  :character   Mode  :character   Median :98052
##                                                           Mean   :98053
##                                                           3rd Qu.:98053
##                                                           Max.   :98074
##    ctyname           postalctyn             lon              lat        building_grade
##  Length:12865       Length:12865       Min.   :-122.2   Min.   :47.46   Min.   : 2.00
##  Class :character   Class :character   1st Qu.:-122.1   1st Qu.:47.67   1st Qu.: 8.00
##  Mode  :character   Mode  :character   Median :-122.1   Median :47.69   Median : 8.00
##                                        Mean   :-122.1   Mean   :47.68   Mean   : 8.24
##                                        3rd Qu.:-122.0   3rd Qu.:47.70   3rd Qu.: 9.00
##                                        Max.   :-121.9   Max.   :47.73   Max.   :13.00
##  square_feet_total_living    bedrooms      bath_full_count  bath_half_count
##  Min.   :  240            Min.   : 0.000   Min.   : 0.000   Min.   :0.0000
##  1st Qu.: 1820            1st Qu.: 3.000   1st Qu.: 1.000   1st Qu.:0.0000
##  Median : 2420            Median : 4.000   Median : 2.000   Median :1.0000
##  Mean   : 2540            Mean   : 3.479   Mean   : 1.798   Mean   :0.6134
##  3rd Qu.: 3110            3rd Qu.: 4.000   3rd Qu.: 2.000   3rd Qu.:1.0000
##  Max.   :13540            Max.   :11.000   Max.   :23.000   Max.   :8.0000
##  bath_3qtr_count   year_built   year_renovated    current_zoning       sq_ft_lot
##  Min.   :0.000   Min.   :1900   Min.   :   0.00   Length:12865       Min.   :    785
##  1st Qu.:0.000   1st Qu.:1979   1st Qu.:   0.00   Class :character   1st Qu.:   5355
##  Median :0.000   Median :1998   Median :   0.00   Mode  :character   Median :   7965
##  Mean   :0.494   Mean   :1993   Mean   :  26.24                      Mean   :  22229
##  3rd Qu.:1.000   3rd Qu.:2007   3rd Qu.:   0.00                      3rd Qu.:  12632
##  Max.   :8.000   Max.   :2016   Max.   :2016.00                      Max.   :1631322
##   prop_type          present_use
##  Length:12865       Min.   :  0.000
##  Class :character   1st Qu.:  2.000
##  Mode  :character   Median :  2.000
##                     Mean   :  6.598
##                     3rd Qu.:  2.000
##                     Max.   :300.000
```

```r
head(housing_df)
```

```
## # A tibble: 6 x 24
##   `Sale Date`         `Sale Price` sale_reason sale_instrument sale_warning sitetype
##   <dttm>                     <dbl>       <dbl>           <dbl> <chr>        <chr>
## 1 2006-01-03 00:00:00       698000           1               3 <NA>         R1
## 2 2006-01-03 00:00:00       649990           1               3 <NA>         R1
## 3 2006-01-03 00:00:00       572500           1               3 <NA>         R1
## 4 2006-01-03 00:00:00       420000           1               3 <NA>         R1
## 5 2006-01-03 00:00:00       369900           1               3 15           R1
## 6 2006-01-03 00:00:00       184667           1              15 18 51        R1
## # i 18 more variables: addr_full <chr>, zip5 <dbl>, ctyname <chr>, postalctyn <chr>,
## #   lon <dbl>, lat <dbl>, building_grade <dbl>, square_feet_total_living <dbl>,
## #   bedrooms <dbl>, bath_full_count <dbl>, bath_half_count <dbl>, bath_3qtr_count <dbl>,
## #   year_built <dbl>, year_renovated <dbl>, current_zoning <chr>, sq_ft_lot <dbl>,
## #   prop_type <chr>, present_use <dbl>
```

```r
# Use the apply function on a variable in your dataset
avg_price <- apply(housing_df['Sale Price'],2,mean)
avg_price
```

```
## Sale Price
##   660737.7
```

```r
# Use the aggregate function on a variable in your dataset
#Calculate the average sales price for each city
groupCityPrice <- aggregate(housing_df$"Sale Price" ~ housing_df$ctyname,housing_df,mean)
head(groupCityPrice)
```

```
##   housing_df$ctyname housing_df$"Sale Price"
## 1           REDMOND                644803.2
## 2         SAMMAMISH                972480.3
```

```r
groupDatePrice <- aggregate(housing_df$"Sale Price" ~ housing_df$"Sale Date",housing_df,mean)
head(groupDatePrice)
```

```
##   housing_df$"Sale Date" housing_df$"Sale Price"
## 1             2006-01-03                482509.5
## 2             2006-01-04                624592.1
## 3             2006-01-05                655475.0
## 4             2006-01-06                677475.0
## 5             2006-01-09                436750.0
## 6             2006-01-10                497631.0
```

```r
#Calculate the total number of full baths in the homes by the year they were built
groupBathYear <- aggregate(housing_df$bath_full_count ~ housing_df$year_built,housing_df,sum)
head(groupBathYear)
```

```
##   housing_df$year_built housing_df$bath_full_count
## 1                  1900                         9
## 2                  1903                         1
## 3                  1905                         3
## 4                  1906                         1
## 5                  1909                         1
## 6                  1910                         1
```

```r
# Use the plyr function on a variable in your dataset - more specifically, I want to see you split some
#install.packages("plyr")
library(plyr)

#Finding total number of rooms in each house

#Checking if any NA values for room information
any(is.na(housing_df$bedrooms))
```

```
## [1] FALSE
```

```r
any(is.na(housing_df$bath_full_count))
```

```
## [1] FALSE
```

```r
any(is.na(housing_df$bath_half_count))
```

```
## [1] FALSE
```

```r
any(is.na(housing_df$bath_3qtr_count))
```

```
## [1] FALSE
```

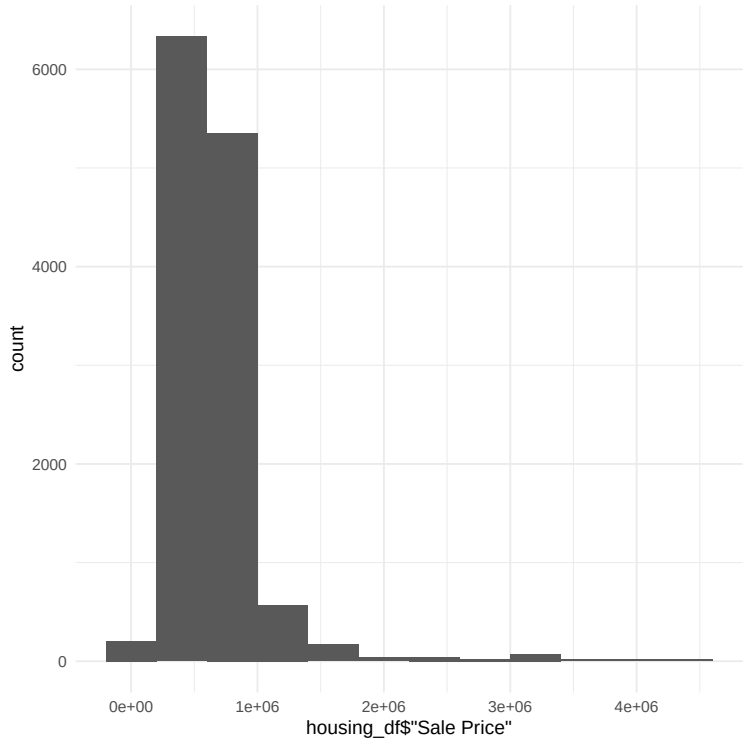
```r
#Take out NA citynames
housing_df$ctyname[is.na(housing_df$ctyname)] <- 'Not Stated'
head(housing_df)
```

```
## # A tibble: 6 x 24
##   `Sale Date`         `Sale Price` sale_reason sale_instrument sale_warning sitetype
##   <dttm>                     <dbl>       <dbl>           <dbl> <chr>        <chr>
## 1 2006-01-03 00:00:00       698000           1               3 <NA>         R1
## 2 2006-01-03 00:00:00       649990           1               3 <NA>         R1
## 3 2006-01-03 00:00:00       572500           1               3 <NA>         R1
## 4 2006-01-03 00:00:00       420000           1               3 <NA>         R1
## 5 2006-01-03 00:00:00       369900           1               3 15           R1
## 6 2006-01-03 00:00:00       184667           1              15 18 51        R1
## # i 18 more variables: addr_full <chr>, zip5 <dbl>, ctyname <chr>, postalctyn <chr>,
## #   lon <dbl>, lat <dbl>, building_grade <dbl>, square_feet_total_living <dbl>,
## #   bedrooms <dbl>, bath_full_count <dbl>, bath_half_count <dbl>, bath_3qtr_count <dbl>,
## #   year_built <dbl>, year_renovated <dbl>, current_zoning <chr>, sq_ft_lot <dbl>,
## #   prop_type <chr>, present_use <dbl>
```

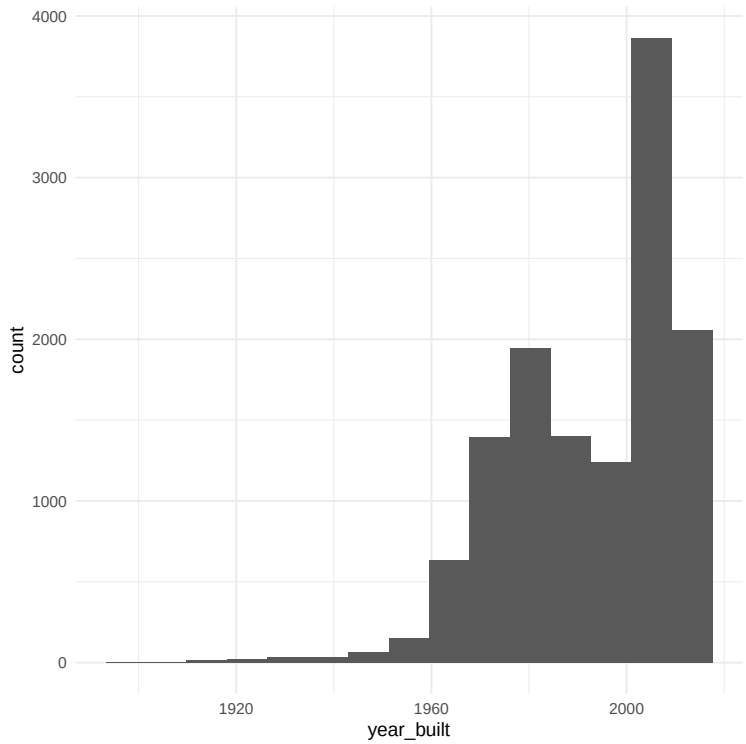
```r
#Only keep sales with city names listed
housing <- housing_df[housing_df$ctyname != 'Not Stated', ]
head(housing)
```

```
## # A tibble: 6 x 24
##   `Sale Date`         `Sale Price` sale_reason sale_instrument sale_warning sitetype
##   <dttm>                     <dbl>       <dbl>           <dbl> <chr>        <chr>
## 1 2006-01-03 00:00:00       698000           1               3 <NA>         R1
## 2 2006-01-03 00:00:00       649990           1               3 <NA>         R1
## 3 2006-01-03 00:00:00       420000           1               3 <NA>         R1
## 4 2006-01-03 00:00:00       369900           1               3 15           R1
## 5 2006-01-04 00:00:00       650000           1               3 <NA>         R1
## 6 2006-01-04 00:00:00       599950           1               3 <NA>         R1
## # i 18 more variables: addr_full <chr>, zip5 <dbl>, ctyname <chr>, postalctyn <chr>,
```

```
## #   lon <dbl>, lat <dbl>, building_grade <dbl>, square_feet_total_living <dbl>,
## #   bedrooms <dbl>, bath_full_count <dbl>, bath_half_count <dbl>, bath_3qtr_count <dbl>,
## #   year_built <dbl>, year_renovated <dbl>, current_zoning <chr>, sq_ft_lot <dbl>,
## #   prop_type <chr>, present_use <dbl>

#Calculate number of rooms
housing$numRooms <-  with(housing,(bedrooms + bath_full_count + bath_half_count + bath_3qtr_count))
head(housing$numRooms)

## [1] 7 7 5 5 6 6

# Check distributions of the data
library(ggplot2)

#Distribution of 'Sale Price' is negatively skewed -- more prices at the lower end of the distribution
#Would have to look at the real estate in these cities/zip codes around these dates and check the city r
#However, most of the houses are in the range from $0 - $1,000,000 which is pretty standard
#I need to look into the houses around $0, because that would be impossible for an object you are purcha
ggplot(housing_df, aes(x=housing_df$'Sale Price')) + geom_histogram(bins=12)

## Warning:  Use of `housing_df$"Sale Price"` is discouraged.
## i Use `Sale Price` instead.
```




```
#The distribution of the variable, year_built, is positively skewed
#Most of the houses were built anytime from 1970s to current day which I also think is pretty standard
#For the older houses, they will probably need some renovations eventually and may even need to be knoc
#Many old houses, as old as the 1920s, would most likely not be livable anymore
ggplot(housing_df, aes(x=year_built)) + geom_histogram(bins = 15)
```




```
# Identify if there are any outliers
#Summary function
summary(housing_df)
```

```
##    Sale Date                        Sale Price       sale_reason    sale_instrument
##  Min.   :2006-01-03 00:00:00.00   Min.   :    698   Min.   : 0.00   Min.   : 0.000
##  1st Qu.:2008-07-07 00:00:00.00   1st Qu.: 460000   1st Qu.: 1.00   1st Qu.: 3.000
##  Median :2011-11-17 00:00:00.00   Median : 593000   Median : 1.00   Median : 3.000
##  Mean   :2011-07-28 15:07:32.48   Mean   : 660738   Mean   : 1.55   Mean   : 3.678
##  3rd Qu.:2014-06-05 00:00:00.00   3rd Qu.: 750000   3rd Qu.: 1.00   3rd Qu.: 3.000
##  Max.   :2016-12-16 00:00:00.00   Max.   :4400000   Max.   :19.00   Max.   :27.000
##  sale_warning         sitetype          addr_full             zip5
##  Length:12865       Length:12865       Length:12865       Min.   :98052
##  Class :character   Class :character   Class :character   1st Qu.:98052
##  Mode  :character   Mode  :character   Mode  :character   Median :98052
##                                                           Mean   :98053
##                                                           3rd Qu.:98053
##                                                           Max.   :98074
##    ctyname           postalctyn             lon              lat        building_grade
##  Length:12865       Length:12865       Min.   :-122.2   Min.   :47.46   Min.   : 2.00
##  Class :character   Class :character   1st Qu.:-122.1   1st Qu.:47.67   1st Qu.: 8.00
##  Mode  :character   Mode  :character   Median :-122.1   Median :47.69   Median : 8.00
##                                        Mean   :-122.1   Mean   :47.68   Mean   : 8.24
##                                        3rd Qu.:-122.0   3rd Qu.:47.70   3rd Qu.: 9.00
##                                        Max.   :-121.9   Max.   :47.73   Max.   :13.00
##  square_feet_total_living    bedrooms      bath_full_count  bath_half_count
##  Min.   :  240            Min.   : 0.000   Min.   : 0.000   Min.   :0.0000
##  1st Qu.: 1820            1st Qu.: 3.000   1st Qu.: 1.000   1st Qu.:0.0000
##  Median : 2420            Median : 4.000   Median : 2.000   Median :1.0000
##  Mean   : 2540            Mean   : 3.479   Mean   : 1.798   Mean   :0.6134
```

```
##  3rd Qu.: 3110            3rd Qu.: 4.000   3rd Qu.: 2.000   3rd Qu.:1.0000
##  Max.   :13540            Max.   :11.000   Max.   :23.000   Max.   :8.0000
##  bath_3qtr_count   year_built   year_renovated    current_zoning       sq_ft_lot
##  Min.   :0.000   Min.   :1900   Min.   :   0.00   Length:12865       Min.   :    785
##  1st Qu.:0.000   1st Qu.:1979   1st Qu.:   0.00   Class :character   1st Qu.:   5355
##  Median :0.000   Median :1998   Median :   0.00   Mode  :character   Median :   7965
##  Mean   :0.494   Mean   :1993   Mean   :  26.24                      Mean   :  22229
##  3rd Qu.:1.000   3rd Qu.:2007   3rd Qu.:   0.00                      3rd Qu.:  12632
##  Max.   :8.000   Max.   :2016   Max.   :2016.00                      Max.   :1631322
##   prop_type          present_use
##  Length:12865       Min.   :  0.000
##  Class :character   1st Qu.:  2.000
##  Mode  :character   Median :  2.000
##                     Mean   :  6.598
##                     3rd Qu.:  2.000
##                     Max.   :300.000
```

```
#The house that has a sales price of $698 is an outlier
#Looking at the summary info for this variable, the minimum value is $698 and the maximum value is $4.4
#When looking at the data for the $698 dollar homes, they are relatively high in square feet and have m
#which is inplausible for a house that is so cheap
#Also, the first quartile for this variable starts at $460,000 which puts $698 way below the first range
minPrice <- housing_df[housing_df$`Sale Price` == 698,]
minPrice$square_feet_total_living
```

```
## [1] 5830 1040
```

```
minPrice$bedrooms
```

```
## [1] 4 3
```

```
#Looking at 240 square foot total living
#This home is also an outlier ... 240 square feet total living with a sales price of $687,500 is crazy!
#From looking at the other homes that have a sales price of $687,500 or higher, 240 square feet is the m
#240 is quite different than 2700, which marks this home also as an outlier
minSqFt <- housing_df[housing_df$square_feet_total_living == 240,]
head(minSqFt)
```

```
## # A tibble: 1 x 24
##   `Sale Date`         `Sale Price` sale_reason sale_instrument sale_warning sitetype
##   <dttm>                     <dbl>       <dbl>           <dbl> <chr>        <chr>
## 1 2016-12-10 00:00:00       687500           1               3 <NA>         R1
## # i 18 more variables: addr_full <chr>, zip5 <dbl>, ctyname <chr>, postalctyn <chr>,
## #   lon <dbl>, lat <dbl>, building_grade <dbl>, square_feet_total_living <dbl>,
## #   bedrooms <dbl>, bath_full_count <dbl>, bath_half_count <dbl>, bath_3qtr_count <dbl>,
## #   year_built <dbl>, year_renovated <dbl>, current_zoning <chr>, sq_ft_lot <dbl>,
## #   prop_type <chr>, present_use <dbl>
```

```
squareFtPrice <- housing_df[housing_df$'Sale Price' >= 687500,]
summary(squareFtPrice$square_feet_total_living)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     240    2700    3250    3312    3720   13540
```

```
# Create at least 2 new variables
```

```
#New variable for total number of bathroom(s)
housing_df$totalBath <- with(housing_df,(bath_full_count + bath_half_count + bath_3qtr_count))
head(housing_df$totalBath)

## [1] 3 3 3 2 2 4

any(is.na(housing_df$year_built))

## [1] FALSE

any(is.na(housing_df$year_renovated))

## [1] FALSE

#New variable for true age of house -- subtract current year, 2021, from the year that the house was bu
housing_df$houseAge <- with(housing_df,(2021 - year_built))
head(housing_df$houseAge)

## [1] 18 15 34 53 41 16
```

The R session information (including the OS info, R version and all packages used):

```
sessionInfo()

## R version 4.2.2 (2022-10-31 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 22621)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.utf8  LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8 LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] plyr_1.8.8     readxl_1.4.2   pastecs_1.3.21 ggplot2_3.4.1  tidyr_1.3.0
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.10      pillar_1.9.0     compiler_4.2.2   cellranger_1.1.0 highr_0.10
##  [6] tools_4.2.2      boot_1.3-28      evaluate_0.20    lifecycle_1.0.3  tibble_3.2.1
## [11] gtable_0.3.3     pkgconfig_2.0.3  rlang_1.1.0      cli_3.6.1        rstudioapi_0.14
## [16] xfun_0.38        withr_2.5.0      dplyr_1.1.1      knitr_1.42       generics_0.1.3
## [21] vctrs_0.6.1      grid_4.2.2       tidyselect_1.2.0 glue_1.6.2       R6_2.5.1
## [26] fansi_1.0.4      purrr_1.0.1      farver_2.1.1     magrittr_2.0.3   scales_1.2.1
## [31] colorspace_2.1-0 labeling_0.4.2   utf8_1.2.3       tinytex_0.44     munsell_0.5.0

Sys.time()

## [1] "2023-04-07 23:07:13 PDT"
```