STA 3100 Programming with Data: Assignment 030

(65 points)

Importing Data and Working with Data Frames and Dates

Note: To prevent echoing of messages and warnings in chunk output, at times you may find it useful to set one or both of the chunk options `message = FALSE` and `warning = FALSE`.

Guitars

1. The file `Data/guitars2.csv` contains information about a small collection of guitars.

   (a) (10 pts) Use the readr function `read_csv()` to read the file into R as a data frame (tibble) names `guitars`. The entries in the columns "Purchase Price" and "Insured Value" should be read as numbers and the entries in the column "Purchase Date" should be read as dates. You must specify this using the optional `col_types` argument of the `read_csv()` function (see sections 11.3.1 and 11.4.2 of r4ds). Note that `Purchase Date`, `Purchase Price`, and `Insured Value` are non-syntactic names (because they contain spaces), so you need to surround them with backticks when refering to them in R code.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.8     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.1
## v readr   2.1.2     v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
guitars <- read_csv("C:/Users/camid/Documents/UF/SOPHOMORE YR/FALL2022/STA3100/data frames/Data/guitars2
  "Purchase Price"= col_number(),
  "Insured Value" = col_number(),
  "Purchase Date" = col_date())
  )
```

```
## Warning: One or more parsing issues, see `problems()` for details
```

```
   When you have done this part correctly, you will still receive a warning about "parsing issues".  Wha
```

The source of this problem is that there is a new line that the parser is reading. This is causing the warning to pop up.

(b) (5 pts) Use the dplyr function `rename()` to rename the columns "Weight with Case", "Weight of Case", "Purchase Date", "Purchase Price", "Purchased New", and "Insured Value" to "wt_w_case", "wt_case", "prch_date", "prch_price", "prch_new", and "ins_val", respectively.

(c) (5 pts) Now change all the column names to lower case. To do this you can either use the dplyr function `rename_with()` or the base R function `names()`. Note that two functions that can be used to change character strings to lower case are `str_to_lower()` from the tidyverse package `stringr` and `tolower()` from base R.

(d) (5 pts) For the guitar with serial number 616619, set the purchase date to July 2, 1977. *Hint*: A base R solution is fairly straightforward. However, if you want to do this in a tidyverse way, you could use `mutate()` with the `replace()` function, taking the index argument of `replace()` to be `serial == "616619"`.

(e) (5 pts) Use the dplyr function `mutate()` with the lubridate functions `year()` and `month()` to create columns `prch_year` and `prch_month` containing the year of purchase (numeric) and the month of purchase (factor with abbreviated month names), respectively.

Consumer Price Index

6. The file `Data/cpi.cvs` contains the values of the Consumer Price Index (CPI) from January 1947 through September 2022.to

   (a) (5 pts) Use the readr function `read_csv()` to read the file into R as a data frame (tibble) named `cpi`.

   (b) (5 pts) Remove the columns `HALF1` and `HALF2` from the `cpi` data frame.

   (c) (5 pts) Now use the tidyr function `pivot_longer()` to change `cpi` the data frame to have columns `Year`, `Month`, and `CPI`. *Hint*: Set the `cols` argument to `!Year`, the `names_to` argument to `"Month"`, and the `values_to` argument to `"CPI"`.

   (d) (5 pts) Use the dplyr functions `group_by()` and `summarize()` (or `summarise()`) to create a data frame `cpi_by_year` containing the average (mean) CPI for each year from 2013 though 2022. *Note*: Because the CPIs for October 2022 through December 2022 are not available yet, you will get `NA` for 2022 unless you set `na.rm = TRUE` in the `mean()` function. Either solution will be accepted.

Adjusting for Inflation

3. To adjust a dollar amount `x` at time 1 to an equivalent dollar amount at time 2, one multiplies `x` by the ratio of the CPI at time 2 to the CPI at time 1, so `(CPI2 / CPI1) * x`. Indeed, this is exactly the calculation done by the Bureau of Labor Statistic's online CPI Inflation Calculator.

   (a) (5 pts) Use the dplyr function `left_join()` to add a column `prch_cpi` to the guitars data frame containing CPI for the month and year that the guitar was purchased.

   (b) (5 pts) As you can see from the data, the CPI for September 2022 is 296.808. Use the dplyr function `mutate()` to add a column named `adj_prch_price` to the `guitars` data frame giving the inflation adjusted purchase prices of the guitars as of September 2022.

   (c) (5 pts) Use the dplyr functions `select()` and `arrange()` to display the model, year, purchase date, purchase price, adjusted purchase price, and insured value for the guitars, ordered by date of purchase.