DATA 607, Project Two: Data Transformation - Laptop Data

Kevin Kirby

2024-09-29

Overview

This is one of three distinct files created for project two of the Fall 2024 edition of DATA 607. This project asked me to pick three datasets posted by other students in the class and use it in a tidying and review exercise. The below is based on a laptop dataset posted by TH. I added the file to my CUNY GCP bucket and enabled public download.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.3     v tibble    3.2.1
## v purrr     1.0.2     v tidyr     1.3.1
## v readr     2.1.5
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(readr)
library(stringr)
library(caret)
```

```
## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:purrr':
##
##     lift
```

```r
library(scales)
```

```
##
## Attaching package: 'scales'
##
## The following object is masked from 'package:purrr':
##
##     discard
##
## The following object is masked from 'package:readr':
##
##     col_factor
```

```r
laptop_data_th <- read.csv("https://storage.googleapis.com/data_science_masters_files/2024_fall/data_60
```

Tidying the data

For the cleanup, the discussion board posted said: "the dataset has issues with missing data, duplicates, inconsistencies, and inaccuracies. Missing rows and '?' entries can be addressed through data imputation or removal, while duplicates should be dropped to avoid skewed results."

The below handles missing and duplicate values. It also replaces the weight value for a laptop weighing less than 1 pound with the median weight for a laptop from the same brand. I also removed the Type Name field with all the strange values because I saw no value in it.

```r
tidy_laptop_data <- laptop_data_th %>%
  rename_with(tolower) %>%
  rename(
    brand = company,
    size = inches,
    screen_res = screenresolution,
    op_sys = opsys
  ) %>%
  mutate(cpu_gpu = paste(cpu, gpu, sep = "; ")) %>%
  select(
    brand,
    op_sys,
    size,
    price,
    cpu_gpu,
    memory,
    ram,
    screen_res,
    weight
  ) %>%
  drop_na() %>%
  mutate(
    weight = str_trim(weight),
    weight = str_remove_all(weight, "[^0-9.]"),
    weight = as.numeric(weight),
    weight_lb = weight * 2.20462
  )


median_weight_by_brand <- tidy_laptop_data %>%
```

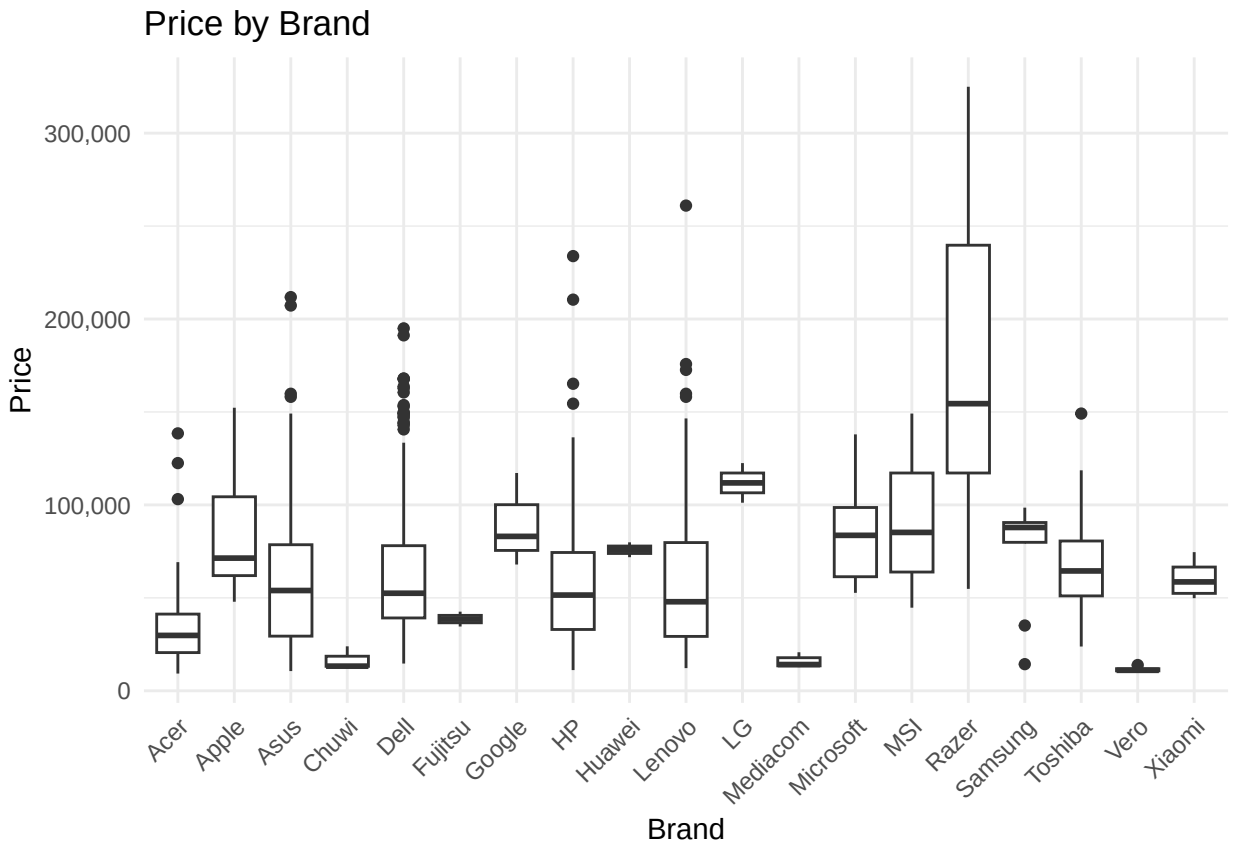
```
  group_by(brand) %>%
  summarise(median_weight = median(weight, na.rm = TRUE))

tidy_laptop_data <- tidy_laptop_data %>%
  left_join(median_weight_by_brand, by = "brand") %>%
  mutate(
    weight = case_when(
      is.na(weight) ~ median_weight,
      weight < 1.0 ~ median_weight,
      TRUE ~ weight
    ),
    weight_lb = weight * 2.20462
  ) %>%
  select(-median_weight)
```

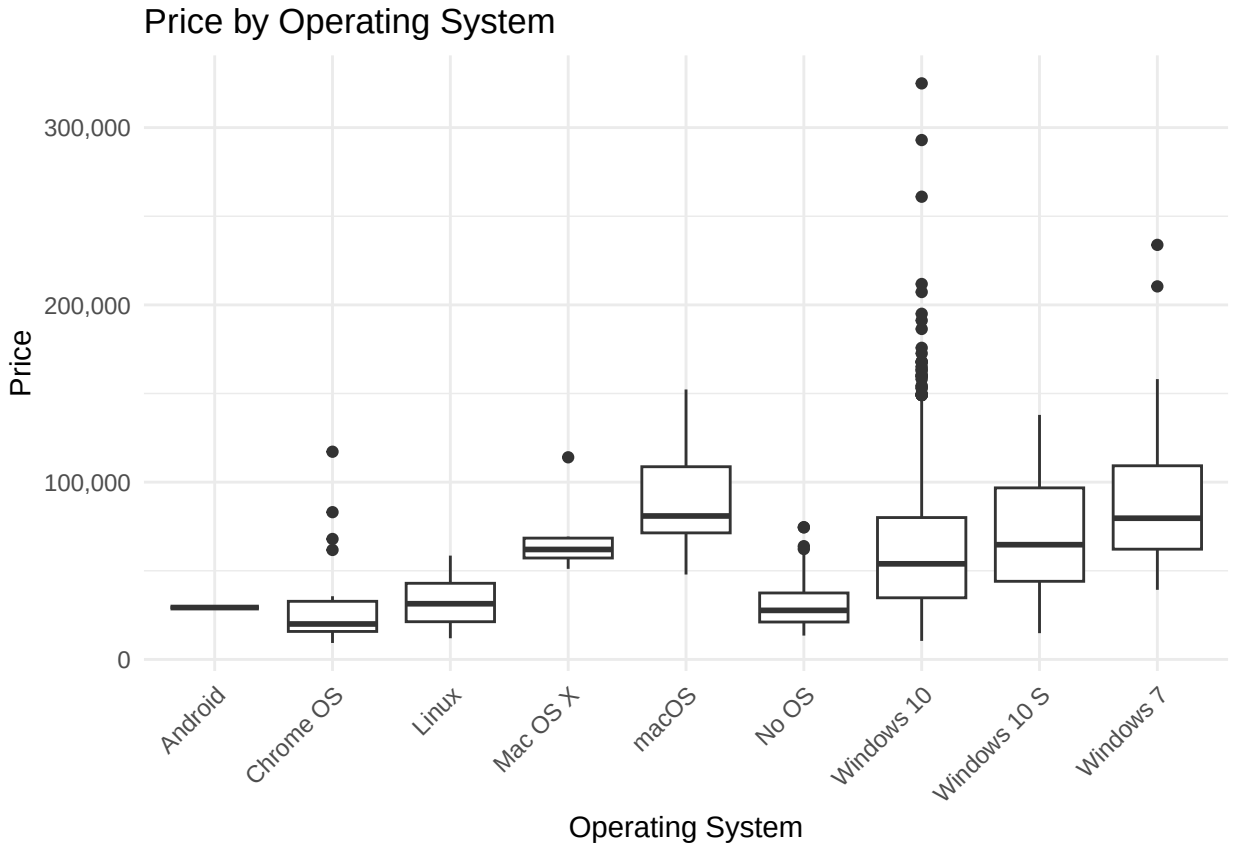
Analysis

Here's a box-plot of the different price ranges you see across the different brands and operating systems. This is where I started to focus in on prices and realize something wasn't adding up.

```
ggplot(tidy_laptop_data, aes(x = brand, y = price)) +
  geom_boxplot() +
  labs(
    title = "Price by Brand",
    x = "Brand",
    y = "Price"
  ) +
  scale_y_continuous(labels = label_number(big.mark = ",")) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```




```
ggplot(tidy_laptop_data, aes(x = op_sys, y = price)) +
  geom_boxplot() +
  labs(
    title = "Price by Operating System",
    x = "Operating System",
    y = "Price"
  ) +
  scale_y_continuous(labels = label_number(big.mark = ",")) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Conclusion

I'm a bit skeptical about this dataset due to how wild these numbers are. My guess is that these prices are in some unknown foreign dollar rather than USD. I checked the original data source and couldn't find anything additional. This data here is probably proportionally accurate but not accurate at the raw price level.