

Average Drug Markup

Scott Cohn (RA)

4/27/2020

The following memo documents the creation of the average drug markup figure in *Microeconomics: Competition, Coordination and Conflict* by Dr. Samuel Bowles and Dr. Simon Halliday. The figure and data-wrangling were done in R. The included packages are in the code chunk below. All of the code used is presented here and runs cleanly as-is.

Differences

I note several differences between the original version of the figure and corrected version (presented here). First, the original version (henceforth, OG) used the *Estimated cost price (course)* column of Table 1 rather than the *Estimated cost price (Day) × 14*. This does not change the values significantly. Using either version is matter of commenting/uncommenting 1 line of code in the creation of the **genEst** dataframe in the bottom of the third code chunk. In the dataframe for Lopinavir/ritonavir (**lopRi**), I originally included the values for the “global fund”. That is now removed.

The rowwise markup ratios were the same in both my analysis and Dr. Tipoe’s analysis. However, the averages (mean and median) differed. I recoded how the averages are calculated and tested it on toy data. The output much more closely matches Dr. Tipoe’s numbers with minor differences. I’ve included a table with my output.

Small point: Dr. Tipoe deals does not include Bangladesh in the markup calculation because the markup is negative. This appears to be a function of the IF statements used in Excel. I include it (-0.29). This changes the mean of Chloroquine from 59.2 to 51.8 (their value to my value). This is the only negative markup I know of.

Technical documentation

```
knitr::opts_chunk$set(  
  echo = TRUE,  
  message = FALSE,  
  warning = FALSE  
)  
library(tidyverse)  
library(plyr)  
  
library(kableExtra) # for tables
```

Only one user-defined function is used. The markup ratio is calculated using **markupRatioFunc**, which takes in a arguments price, **p**, and cost, **c**, and returns the markup ratio.

```
# Takes in args price and cost  
# Returns markup ratio  
markupRatioFunc <- function(p, c){
```

```
(p - c) / c
}
```

The code chunk below is the manual input of data from the Hill et al (2020).¹ Each drug has an ID.

```
## Lopinavir/ritonavir (lopRi)
# 14-day treatment course (400 mg once daily)

lopRi <- tibble(
  "drugID" = 1,
  "country" = c("USA_pharm", "USA_va", "SWE", "TUR",
                "GBR", "FRA", "IND", "CHN", "ZAF"),
  "cost_lopRi" = c(503, 349, 172, 149, 144, 97, 40, 17, 15),
)

## Hydroxychloroquine (hyd)
# 14-day treatment course (400 mg once daily)

hyd <- tibble(
  "drugID" = 2,
  "country" = c("CHN", "USA_pharm", "MYS", "France",
                "GBR", "SWE", "BGD", "TUR", "USA_va", "IND"),
  "cost_hyd" = c(19, 18, 7, 5, 4, 3, 3, 3, 3, 2)
)

## Chloroquine (chl)
# 14-day treatment course (155 mg once daily)

chl <- tibble(
  "drugID" = 3,
  "country" = c("USA_pharm", "GBR", "CHN", "ZAF", "SWE", "MYS", "IND", "BGD"),
  "cost_chl" = c(93, 8, 5, 5, 4, 2, 1, 0.2)
)

## Azithromycin (azi)
# 14-day treatment course (500 mg once daily)

azi <- tibble(
  "drugID" = 4,
  "country" = c("USA_pharm", "FRA", "ZAF", "BRA", "USA_va",
                "SWE", "MYS", "GBR", "CHN", "BGD", "IND"),
  "cost_avi" = c(63, 44, 35, 19, 17, 16, 11, 11, 7, 5, 5)
)

## Sofosbuvir/daclatasvir (sofDa)
# 14-day treatment course (400/60 mg once daily)

sofDa <- tibble(
  "drugID" = 5,
  "country" = c("USA_va", "GBR", "FRA", "BRA", "BGD", "IND", "PAK"),
  "cost_sofDa" = c(18610, 7632, 4662, 4289, 166, 7, 6)
)
```

¹A `tibble` is a fancy R dataframe.

```

## Pirfenidone (pirf)
# 28-day treatment course (801 mg three times daily)

pirf <- tibble(
  "drugID" = 6,
  "country" = c("USA_pharm", "USA_va", "GBR", "ZAF",
                "FRA", "SWE", "TUR", "CHN", "BGD", "IND"),
  "cost_pirf" = c(9606, 6513, 2561, 2490, 2344, 2196, 1499, 1379, 124, 100)
)

## Tocilizumab (toc)
# Single IV dose (560 mg)

toc <- tibble(
  "drugID" = 7,
  "country" = c("USA_pharm", "CHN", "USA_va", "GBR",
                "IND", "BGD", "TUR", "EGY", "ZAF", "PAK"),
  "cost_toc" = c(3383, 1950, 1948, 914, 806, 690, 650, 606, 566, 510)
)

## General Estimates
# Single course
genEst <- tibble(
  "drugID" = seq(1:7),
  "drugName" = c("Lopinavir/ritonavir", "Hydroxychloroquine",
                 "Chloroquine", "Azithromycin", "Sofosbuvir/daclatasvir",
                 "Pirfenidone", "Tocilizumab"),
  "cost_genEst" = c(4, 1, 0.3, 1.4, 5, 31, NA)
  "cost_genEst" = c(3.92, 1.12, 0.28, 1.4, 5.46, 30.52, NA) # 1/day*14
)

```

There is a dataframe for each drug, including the general estimate. To verify that the data was input correctly for each drug, run up to this step. The following chunk does a full-join for all of the drugs, including the general estimate. The wide dataframe was gathered (reshape in Stata) using key-value pairs into `dfCost`. Then rows with NA values were dropped and excess columns filtered out.

```

df <- as_tibble(join_all(list(azi, chl, hyd, lopRi, pirf, sofDa, toc),
  by = c('country', 'drugID'), type = 'full')
)

dfCost <- left_join(df, genEst, "drugID") %>%
  gather(3:9, key = "drugKey", value = "drugCostByCountry") %>%
  drop_na("drugCostByCountry") %>%
  drop_na("cost_genEst") %>%
  select("drugID", "drugName", "country", "cost_genEst", "drugCostByCountry")

```

At this stage, no calculations were done. These data are entered, merged, and cleaned. This is another point to verify data-entry. No values will have been changed from the initial drug dataframes.

The first 6 lines of code in the next chunk calculate the markup ratio (rounded to 2 digits). These values are consistent with Dr. Tipoe's calculations. The mean and median are calculated and put in dataframes named `df_mean` and `df_med`. Since the mean and median values are not altered from here, opening these dataframes shows the drug name and associated average.

```

# cbind markup ratio
dfCost <- dfCost %>%
  cbind(markupRatio = round(markupRatioFunc(
    p = dfCost[['drugCostByCountry']],
    c = dfCost[['cost_genEst']],
    digits = 2)
  )

# find mean/median + rename cols
df_mean <- aggregate(dfCost$markupRatio, list(dfCost$drugName), mean) %>%
  dplyr::rename("mean" = "x") %>%
  dplyr::rename("drugName" = "Group.1")

df_med <- aggregate(dfCost$markupRatio, list(dfCost$drugName), median) %>%
  dplyr::rename("median" = "x") %>%
  dplyr::rename("drugName" = "Group.1")

# left join averages into single df by drugName
dfCostAvg <- left_join(df_mean, df_med, by = "drugName")

# Table (not part of analysis)
kable(dfCostAvg, "latex", booktabs = T) %>% kable_styling(latex_options = "striped")

```

drugName	mean	median
Azithromycin	14.1300	10.430
Chloroquine	51.7675	15.075
Hydroxychloroquine	4.9820	2.125
Lopinavir/ritonavir	41.1200	35.730
Pirfenidone	93.4030	73.375
Sofosbuvir/daclatasvir	924.4829	784.530

```

# left join averages with dfCost by drugName then reshape wide-to-long
dfCostAvg <- left_join(dfCostAvg, dfCost, by = "drugName") %>%
  gather(2:3, key = "avgType", value = "avgVal")

```

Below is the code that creates the figure. This is included for completeness and contains no further manipulations in data.

```

# avg markup by drug across ALL countries
drugNames <- c("Pirfenidone", "Lopinavir/ritonavir",
  "Chloroquine", "Azithromycin")
drugNameDisease <- c("Pirfenidone \n (Pulmonary Fibrosis)",
  "Lopinavir/ritonavir \n (HIV-1)",
  "Chloroquine \n (Malaria)",
  "Azithromycin \n (Antibiotic)")

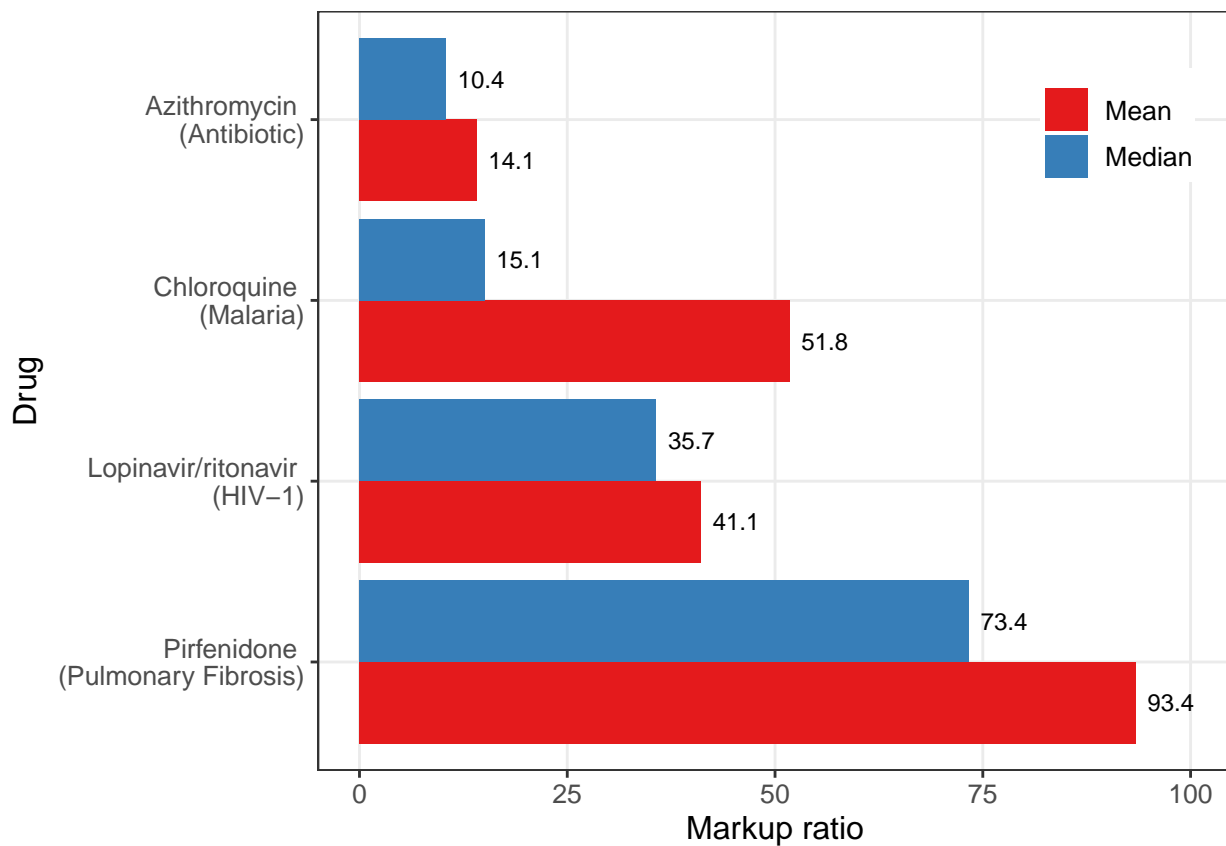
dfCostAvg %>%
  #filter(country != "USA_va") %>%
  filter(drugID != 5) %>% # sofo/dac (remove due to size discrep.)
  filter(drugID != 2) %>% # hydrox (remove by request of S.B.)
  mutate(drugName = factor(drugName, levels = drugNames)) %>%
  ggplot(aes(x = factor(drugName), y = avgVal, fill = avgType)) +

```

```

geom_bar(stat = "identity", position = "dodge") +
# scale_x_discrete(labels = str_wrap(drugNameDisease, width = 15)) + # auto-wrap (cluttered)
scale_x_discrete(label = drugNameDisease) +
scale_y_continuous(limits = c(0, 100)) +
scale_fill_brewer(name = "", labels = c("Mean", "Median"), palette = "Set1") +
labs(y = "Markup ratio", x = "Drug") +
coord_flip() +
geom_text(aes(label = round(avgVal, 1)), position = position_dodge(width = 0.9),
          hjust = -0.25, size = 3, check_overlap = TRUE) +
theme_bw() +
theme(legend.title = element_blank(),
      panel.grid.minor = element_blank(),
      text = element_text(size = 12),
      legend.position = c(.875, .85),
      legend.margin = margin(1, 1, 1, 1))

```



```

ggsave("avgMarkupAllCountries.pdf", width = 6, height = 3, units = "in")

```