Mehrabi et al. 2020. The global divide in data driven farming. Supplementary Information E: Mobile broadband affordability

Code by Juan Diego Martinez

2020-07-16

Contents

1 Aim of this document

The aim of this document is to provide reproducible code to process data on incomes, and the price of mobile broadband data, for different countries over the years 2015 to 2019. We disaggregate the cost of accessing 1GB pre-paid mobile data for 10 income groups for 82 countries in Africa, Asia, and Latin America and the Caribbean.

2 Methods

2.2 Data sources

1. **AffordabilityData__2015-2017.csv**: Price of Broadband Data (1GB mobile prepaid) as % of GNI per capita, 2015 - 2017. Source: Alliance for Affordable Internet A4AI.

2. **4AI-Mobile-Broadband-Pricing-2018Q4.csv**: Price of Broadband Data (1GB mobile prepaid) as % of GNI per capita as well as actual dollar value, 2018Q4. Source A4AI. IMPORTANT: The file needs to be downloaded first as *.xlsx* format, increase the number of displayed decimals in the numerical cells and then save as *.csv*. If the file is directly downloaded as *.csv* from their website some information will be lost.
3. **A4AI-Mobile-Broadband-Pricing-2019Q2.csv**: Price of Broadband Data (1GB mobile prepaid) as % of GNI per capita as well as actual dollar value, 2019Q2. Source A4AI. IMPORTANT: The file needs to be downloaded first as *.xlsx* format, increase the number of displayed decimals in the numerical cells and then save as *.csv*. If the file is directly downloaded as *.csv* from their website some information will be lost.

2.2 Methods Summary

To estimate the mean income of each decile in each country, we use the share of income held by the bottom 10% of that country based on World Bank Data (accessed through the package **'wbstats'**), in cases where no data for the specified years is available, we assume the inequality for each of those countries has remained the same since the last WB data point. We estimate the mean income of the bottom 10% (GNI_year_decile_1) of each country by multiplying the share of income held by bottom 10% (from WB data) with the GNI per capita (WB data, GNI, Atlas method current US$). Finally, A4AI recently released 2019Q2 data prices, to calculate affordability in that year we use 2018 GNI data from the World Bank. The 2019 values should thus be taken as preliminary estimates.

We focus on the 1st (poorest) decile (GNI_year_decile_1). To calculate the price of data (Price_data_year) for each country, we multiply Mobile_data*GNI_year for each country. Calculating Price_data_year/GNI_year_decile_1 gives us the price of Broadband Data (1GB mobile prepaid) as % of the poorest decile's mean income.

3 Reproducibility

3.1 Load packages

For the analysis in this document we will be using the `tidyverse` (Wickham 2019), `countrycode` (Arel-Bundock 2020), and `wbstats` (Piburn 2018a) packages.

```
library(tidyverse)
library(countrycode)

## Package to access the World
```

```
## Bank data API
library(wbstats)
```

We will also use the **R** packages `knitr` (Xie 2019) and `checkpoint` (Ooi 2019). The package `knitr` facilitates producing a dynamic document that contains all the steps required to analyze the data. The package `checkpoint` will install all package versions for you that we used in our analysis to avoid result discrepancies that may arise from software differences. Thus the reader is provided with all the code to fully reproduce the analysis, and adapt or repurpose it for another analyses.

```
library("checkpoint")
checkpoint(snapshotDate = "2020-07-15")
```

3.2 Load mobile broadband data

In this section we load the mobile broadband data from our latest download from A4AI. All income data is downloaded later through the World Bank API with **wbstats** package. (See 4.2 World Bank data)

```
Mobile_data_2018 <- read.csv("Data/A4AI-Mobile-Broadband-Pricing-2018Q4.csv")
Mobile_data_2019 <- read.csv("Data/A4AI-Mobile-Broadband-Pricing-2019Q2.csv")
Mobile_data_2015_2017 <- read.csv("Data/AffordabilityData_2015-2017.csv")
```

4 Data clean up

4.1 Mobile Data

The data is provided in the wrong format, here we transform string percentages into numerical percentages to undertake the analysis

```
Mobile_data_2015_2017[, 3] <- as.numeric(sub("%",
    "e-2", Mobile_data_2015_2017[,
        3]))
Mobile_data_2015_2017[, 4] <- as.numeric(sub("%",
    "e-2", Mobile_data_2015_2017[,
        4]))
Mobile_data_2015_2017[, 5] <- as.numeric(sub("%",
    "e-2", Mobile_data_2015_2017[,
        5]))
```

Assign shorter column names, add iso3 country codes to 2015-2017 dataset.

```
colnames(Mobile_data_2015_2017) <- c("Country",
    "Region", "Afford_2015", "Afford_2016",
    "Afford_2017")
Mobile_data_2015_2017$ISO3 <- countrycode(Mobile_data_2015_2017$Country,
    "country.name", "iso3c")
Mobile_data_2015_2017_long <- Mobile_data_2015_2017 %>%
    tidyr::pivot_longer(cols = starts_with("Afford"),
        names_to = c(".value",
            "year"), names_sep = "_") %>%
    mutate(year = as.integer(year))
```

Select necessary columns from 2018 and 2019 data, and calculate monthly and yearly income to compare with World Bank's most recent data.
To keep consistency with other years, further calculations will all be made with WB income data.

```
Mobile_data_2018 <- Mobile_data_2018 %>%
    select(ISO3, Cost_1_GB_USD,
        Cost_1_GB_Share_GNICM) %>%
    mutate(year = 2018)
Mobile_data_2019 <- Mobile_data_2019 %>%
    select(ISO3, Cost_1_GB_USD,
        Cost_1_GB_Share_GNICM) %>%
    mutate(year = 2019)
Mobile_data_2018_2019 <- bind_rows(Mobile_data_2018,
    Mobile_data_2019)
Mobile_data_2018_2019 <- Mobile_data_2018_2019 %>%
    mutate(Income_month = Cost_1_GB_USD/Cost_1_GB_Share_GNICM,
        Income = Income_month *
            12) %>% rename(Afford = Cost_1_GB_Share_GNICM)
```

Join all mobile broadband data sets, i.e. 2015_2017 with 2018_2019 data.

```
Mobile_data_long <- bind_rows(Mobile_data_2015_2017_long,
    Mobile_data_2018_2019) %>%
    mutate(Country = countrycode(ISO3,
        "iso3c", "country.name"),
        Region = countrycode(ISO3,
            "iso3c", "continent")) %>%
    arrange(Country, year)
```

4.2 World Bank data

4.2.1 Bottom 10% share of income Access the World Bank available information on share of income held by bottom 10% between 2015 and 2019 (or the latest). If no information is available between 2015 and 2019, older information will be used, but no older than the year 2000. If only some years in the target timeframe have info, we will fill the missing data with the closest year available, e.g. Argentina only has data for 2016 and 2017, we assume for 2018 the same share as 2017, and for 2015 the same as 2016.

```
## download data
bottom_10 <- wb(indicator = c("SI.DST.FRST.10"),
    startdate = 2000, enddate = 2019)

## filter for those countries
## that have mobile data info
bottom_10_with_mobile <- bottom_10 %>%
    filter(iso3c %in% Mobile_data_long$ISO3)
bottom_10$surveyyear <- bottom_10$date
bottom_10_latest <- bottom_10 %>%
    group_by(iso3c) %>% filter(date ==
    max(date)) %>% ungroup()
bottom_10_2015_2019 <- bottom_10 %>%
    filter(date %in% 2015:2019)

## bottom_10_latest_nr: those in
## bottom_10_latest that have NO
## RECENT (i.e. 2015-2019 data)
bottom_10_latest_nr <- bottom_10_latest %>%
    filter(!iso3c %in% bottom_10_2015_2019$iso3c)
```

```
## This next line is just a
## trick for the latter joining
## with WB_GNI, surveyyear saves
## the info on the actual year
## the info was sourced from
bottom_10_latest_nr$date <- "2015"
bottom_10_clean <- bind_rows(bottom_10_2015_2019,
    bottom_10_latest_nr) %>% arrange(country)
```

4.2.2 GNI per capita Join the share of 10% data with the GNI per capita data (current US$, Atlas method). And calculate the GNI/capita of the bottom 10%.
As mentioned in the previous code chunk, here we fill the missing values with closest year with information available under the assumption that the share of income doesn't change dramatically from year to year. This avoids building a linear model to interpolate.

```
## download data
WB_GNI <- wb(indicator = c("NY.GNP.PCAP.CD"),
    startdate = 2015, enddate = 2019)

## filter for those countries
## that have mobile data info
WB_GNI_md <- WB_GNI %>% filter(iso3c %in%
    Mobile_data_long$ISO3)

## Fill missing data
WB_Atlas_GNI_bottom <- left_join(WB_GNI_md,
    bottom_10_clean, by = c("iso3c",
        "date"))
WB_Atlas_GNI_bottom <- WB_Atlas_GNI_bottom %>%
    group_by(iso3c) %>% fill(value.y,
    surveyyear, .direction = "down") %>%
    fill(value.y, surveyyear, .direction = "up")

WB_Atlas_GNI_bottom$Decile_1_current <- WB_Atlas_GNI_bottom$value.y *
    10/100 * WB_Atlas_GNI_bottom$value.x
## *10/100 is to account for
## share/value.y given in
## percentage
WB_Atlas_GNI_bottom <- WB_Atlas_GNI_bottom %>%
    mutate(date = as.integer(date))
```

4.3 Join A4AI broadband mobile data with World Bank income data

```
## Join data
joined <- left_join(Mobile_data_long,
    WB_Atlas_GNI_bottom, by = c(ISO3 = "iso3c",
        year = "date"))
## Clean columns and rename with
## more indicative names
joined <- joined %>% select(ISO3,
    Country, Region, year, Cost_1_GB_USD_month = Cost_1_GB_USD,
    Income_month, Income, Afford,
```

```
    value.x, value.y, surveyyear,
    Decile_1_current) %>% rename(GNI_WB_capita = value.x,
    share_bottom_10 = value.y,
    A4AI_Income = Income, A4AI_Income_month = Income_month,
    iso3c = ISO3) %>% group_by(iso3c) %>%
    fill(GNI_WB_capita, share_bottom_10,
        surveyyear, Decile_1_current,
        .direction = "down")  ## to use 2018 WB data for 2019

## Assign country, region, and
## continent columns
joined$Country <- countrycode(joined$iso3c,
    "iso3c", "country.name")
joined$Region <- countrycode(joined$iso3c,
    "iso3c", "region")
joined$Continent <- countrycode(joined$iso3c,
    "iso3c", "continent")

joined <- joined %>% arrange(Country,
    desc(year))
```

5 Affordability calculations

The following chunk makes the calculation for affordability consistent with the latest mobile pricing data A4AI reports, and it estimates price data from affordability in their 2015-2017 data for which they don't report direct mobile data costs.

```
joined <- joined %>% mutate(Price_data_year = Afford *
    GNI_WB_capita, Afford_decile_1 = ifelse(year <
    2018, Price_data_year/Decile_1_current,
    12 * Cost_1_GB_USD_month/Decile_1_current))
```

6 Write output file

In long format

```
write_csv(joined, "Data/Data_Affordability_decile_1_2015_2019_20200716_long.csv")
```

Convert data to wide format to write alternative output file

```
joined_wide <- joined %>% pivot_wider(names_from = year,
    values_from = c("Cost_1_GB_USD_month",
        "Afford_decile_1", "Afford",
        "Price_data_year", "GNI_WB_capita",
        "share_bottom_10", "Decile_1_current",
        "surveyyear", "A4AI_Income",
        "A4AI_Income_month")) %>%
    discard(~all(is.na(.)))  ## removes columns where there's only NA
```

In wide format

```
write.csv(joined_wide, "Data/Data_Affordability_decile_1_2015_2019_20200716.csv")
```

7 Summary

In this final chunk of code we summarize the number of countries with available data at A4AI and available data for income and share of bottom 10% from the World Bank.

```
## summary table for each year
knitr::kable(joined %>% group_by(year) %>%
    summarize(A4AI_coun3_count = n(),
        na = sum(is.na(Afford_decile_1)),
        non_na = sum(!is.na(Afford_decile_1))),
    align = "c", col.names = c("Year",
        "A4AI country count", "No income data",
        "Complete income data"))
```

Year	A4AI country count	No income data	Complete income data
2015	61	7	54
2016	61	6	55
2017	61	5	56
2018	99	12	87
2019	99	12	87

8 Session Info

```r
sessionInfo()
```

```
## R version 3.6.1 (2019-07-05)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Mojave 10.14.6
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_CA.UTF-8/en_CA.UTF-8/en_CA.UTF-8/C/en_CA.UTF-8/en_CA.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] wbstats_0.2       countrycode_1.1.1 forcats_0.4.0     stringr_1.4.0
##  [5] dplyr_1.0.2       purrr_0.3.3       readr_1.3.1       tidyr_1.0.0
##  [9] tibble_3.0.3      ggplot2_3.3.2     tidyverse_1.3.0
##
## loaded via a namespace (and not attached):
##  [1] tidyselect_1.1.0 xfun_0.11        haven_2.2.0      colorspace_1.4-1
##  [5] vctrs_0.3.4      generics_0.0.2   htmltools_0.4.0  yaml_2.2.0
##  [9] rlang_0.4.7      pillar_1.4.6     glue_1.4.2       withr_2.1.2
## [13] DBI_1.0.0        dbplyr_1.4.2     modelr_0.1.5     readxl_1.3.1
## [17] lifecycle_0.2.0  munsell_0.5.0    gtable_0.3.0     cellranger_1.1.0
## [21] rvest_0.3.5      evaluate_0.14    knitr_1.26       curl_4.3
## [25] fansi_0.4.0      highr_0.8        broom_0.7.0      Rcpp_1.0.4
## [29] scales_1.1.0     backports_1.1.5  formatR_1.7      jsonlite_1.6
## [33] fs_1.3.1         hms_0.5.2        digest_0.6.23    stringi_1.4.3
## [37] grid_3.6.1       cli_2.0.0        tools_3.6.1      magrittr_1.5
## [41] crayon_1.3.4     pkgconfig_2.0.3  ellipsis_0.3.0   xml2_1.2.2
## [45] reprex_0.3.0     lubridate_1.7.4  assertthat_0.2.1 rmarkdown_1.18
## [49] httr_1.4.1       rstudioapi_0.10  R6_2.4.1         compiler_3.6.1
```

9 Bibliography

Arel-Bundock, Vincent. 2020. *Countrycode: Convert Country Names and Country Codes*. https://CRAN.R-project.org/package=countrycode.

Arel-Bundock, Vincent, Nils Enevoldsen, and CJ Yetman. 2018. "Countrycode: An R Package to Convert Country Names and Country Codes." *Journal of Open Source Software* 3 (28): 848. https://doi.org/10.21105/joss.00848.

Ooi, Hong. 2019. *Checkpoint: Install Packages from Snapshots on the Checkpoint Server for Reproducibility*. https://CRAN.R-project.org/package=checkpoint.

Piburn, Jesse. 2018a. *Wbstats: Programmatic Access to Data and Statistics from the World Bank Api*. https://CRAN.R-project.org/package=wbstats.

———. 2018b. *Wbstats: Programmatic Access to the World Bank Api*. Oak Ridge, Tennessee: Oak Ridge National Laboratory. https://www.ornl.gov/division/csed/gist.

Wickham, Hadley. 2019. *Tidyverse: Easily Install and Load the 'Tidyverse'*. https://CRAN.R-project.org/package=tidyverse.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

Xie, Yihui. 2014. "Knitr: A Comprehensive Tool for Reproducible Research in R." In *Implementing Reproducible Computational Research*, edited by Victoria Stodden, Friedrich Leisch, and Roger D. Peng. Chapman; Hall/CRC. http://www.crcpress.com/product/isbn/9781466561595.

———. 2015. *Dynamic Documents with R and Knitr*. 2nd ed. Boca Raton, Florida: Chapman; Hall/CRC. https://yihui.name/knitr/.

———. 2019. *Knitr: A General-Purpose Package for Dynamic Report Generation in R*. https://CRAN.R-project.org/package=knitr.