# Mehrabi et al. 2020. The global divide in data driven farming. Supplementary Information A: Analysis.

Code by Zia Mehrabi

November 1, 2020

## Contents

# 1 Aim of document

The aim of this document is to provide the Supplementary Information for the analysis reported in Mehrabi, Z. et al. The global divide in data-driven farming. https://doi.org/10.1038/s41893-020-00631-0. This document was created with `knitr` [14], a software that combines the typesetting system LaTeX and the statistical computing language **R**, and allows for reproduction of our results.

This document is split into nine sections. In section 2 we illustrate how we make our analysis fully reproducible. Section 3 explains a bit of background to the data we are working with and shows how we manipulated the data to get it ready for analysis. Sections 4 - 8 walks step by step through the analysis that underpins the results presented in our paper; and finally, section 9 reprints the the underlying network data used in the analysis.

# 2 Checkpoint

For the analysis in this document we will be using the following packages: `knitr` [14], `checkpoint` [6], `raster` [4], `sp` [7], `maptools` [2], `rgdal` [1], `dplyr` [10], `ggplot2` [11], `ggridges` [12], `mgcv` [13], `scam` [8], `rworldmap` [9], `car` [3].

We call the **R** package `checkpoint` [6]. This will create a local library on your computer and install a copy of the packages required by this project as they existed on CRAN on the specified snapshot date, and update the **R** session to use these packages. This helps make our analysis fully reproducible on your machine.

```
checkpoint(snapshotDate = "2020-07-30")
```

Note that the **R** version used here is 3.6.1 (2019-07-05). Using other versions of **R** should not have any influence on the results obtained.

# 3 Data

There are a variety of data sets used in this analysis. Except for proprietary products, all other data are prepared and processed for this analysis in the accompanying Supplementary Information directories B-E, see these folders for further details. Briefly, the key data sets analysed in this document include:

- Global mobile network coverage in 2018 for a total of 910 entities running 2391 networks in 229 countries and territories, representing 2G (CDMA, 1XRTT, GSM, GPRS, EDGE) 3G (EVDO, UMTS, HDPA/+), and 4G (WiMax, LTE), developed and supplied by Mosaik LLC (since acquired by Ookla, https://www.ookla.com/).
- A geospatial data set of global farm size distributions built from national level data on farm size distributions and sub national data on field size distributions (see Supplementary Information B for more information).
- A library of geospatial data with global extent, including croplands with nitrogen deficiencies, and extreme aridity (annual precipitation <250mm), hotspots of yield gaps, rainfall-dependent croplands, the number of food insecure, the number of people infected by Plasmodium falciparum, and the number of people with low human development index scores (see Supplementary Information for more information).
- A harmonized data set of farming and non-farming household-level mobile ownership in 70 countries, 5429 sub national units; and farming and non-farming internet access in 48 countries and 4905 sub national units (see Supplementary Information D for more information).

- The cost of 1GB pre-paid mobile as a percentage of income from the Alliance for Affordable Internet (https://a4ai.org/) for 83 countries in Africa, Asia, and Latin America and the Caribbean, disaggregated by income groups within each country (see Supplementary Information E for more information).

# 4 Main Text Figure 1a

## 4.1 Data preparation

In this section we plot the availability of 2G/3G/4G network coverage on croplands globally. First we read in a rasterized version of the original shapefiles supplied by Mosaik, which are 10 $km^2$ layers, for each of the three aggregate networks types.

```
twg <- raster("Data/Network/2g.tif")
thg <- raster("Data/Network/3g.tif")
fourg <- raster("Data/Network/4g.tif")
```

Next we read in the cropland data on fractional crop area, this was used in Supplementary Information B, so we can read it from that directory. It is in the same crs and resolution as the network coverage data.

```
cropland.a <- raster("../SI_B/data/Ramankutty_2008_cropland/CroplandPastureArea2000_Geotiff/cropland2000
res(cropland.a) <- res(twg)
cropland.a <- extend(cropland.a, fourg)
extent(cropland.a) <- extent(twg)
```

Then we create a single network layer that we will use for mapping, which prioritizes vizualizing the highest generation technology available at a given location.

```
tech.all.val <- ifelse(!is.na(values(fourg)), "4G", ifelse(!is.na(values(thg)),
    "3G", ifelse(!is.na(values(twg)), "2G", NA)))

tech.all.val <- as.factor(tech.all.val)
tech.all <- twg  #create raster object to store valuses
values(tech.all) <- tech.all.val
```

For plotting we create equal area versions of the network data and cropland for mapping, although our analysis will be done in lat long (which is the native crs of the majority of the data products used in the analysis). We match the extent of the two layers, and write the resulting network layer to disk for future people wishing to reproduce our map.

```
# convert to equal area to map with cropland.
crop.eck4 <- projectRaster(cropland.a, res = c(8439, 8439), crs = "+proj=eck4 +datum=WGS84 +ellps=WGS84
    method = "bilinear", over = T)

tech.all.eck4 <- projectRaster(tech.all, res = c(8439, 8439),
    crs = "+proj=eck4 +datum=WGS84 +ellps=WGS84 +towgs84=0,0,0",
    method = "ngb", over = T)

tech.all.eck4 <- extend(tech.all.eck4, crop.eck4)
tech.all.eck4 <- setExtent(tech.all.eck4, crop.eck4, keepres = TRUE)

tech.masked <- ifelse(values(crop.eck4) > 0.01, values(tech.all.eck4),
    NA)
values(tech.all.eck4) <- tech.masked
```

```
writeRaster(tech.all.eck4, "Data/Out/tech.all.eck4.tif", overwrite = T)
```

We then binarize the cropland area layer and write it to disk, this we will use as a backdrop for the network coverage map.

```
cropbin.eck4 <- crop.eck4
values(cropbin.eck4) <- ifelse(values(crop.eck4) > 0.01, 1, 0)
writeRaster(cropbin.eck4, "Data/Out/cropbin.eck4.tif", overwrite = T)
```

## 4.2 Figure 1a

We plot this network coverage on croplands.

```
cols <- c("#1b9e77ff", "#d95f02ff", "#7570b3ff", "gray")
leg.txt <- c("4G", "3G", "2G", "None")
plot(cropbin.eck4, col = c("white", "gray"), colNA = "black",
    legend = F, axes = FALSE, box = FALSE)
plot(tech.all.eck4, col = cols[1:3], add = TRUE, legend = F)
legend("bottomleft", inset = 0.1, legend = leg.txt, col = cols,
    pch = 15, box.col = "white", bg = "black", text.col = "white",
    cex = 1, title = "Service")
```

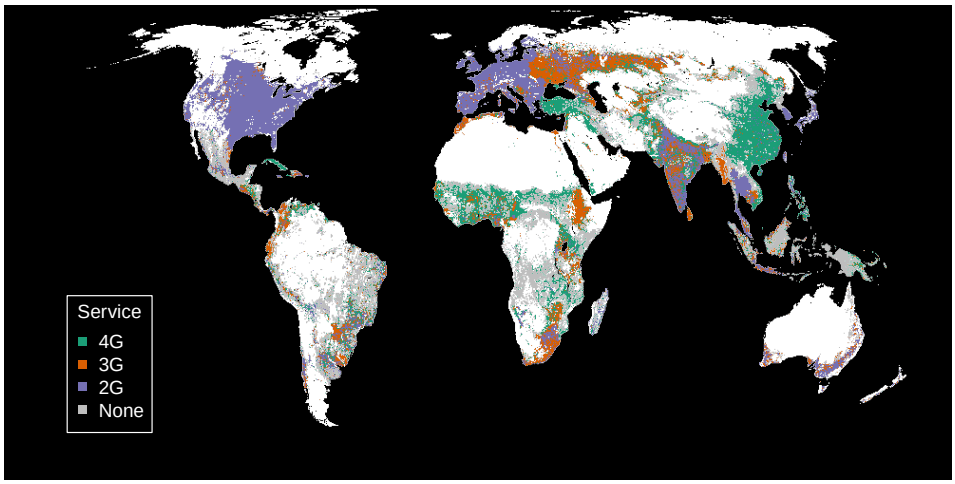


Figure 1: Network coverage across croplands

## 4.3 Insights

We complement the plot with an assessment of the coverage in percentage terms.We create a common dataframe for all of the network and cropland layers.

```
Area <- area(twg)
cropland <- Area * cropland.a  #total area in km2 in each grid cell.
dat.rast <- stack(twg, thg, fourg, cropland, cropland.a)  #make a common DF
all.dat.f1 <- data.frame(coordinates(dat.rast), values(dat.rast))
colnames(all.dat.f1) <- c("x", "y", "2G", "3G", "4G", "cropland",
    "cropland.a")
```

Then we estimate the area of cropland for each generation of technology.

```
all.dat.f1$`2G` <- ifelse(all.dat.f1$`2G` > 0, all.dat.f1$cropland,
    NA)
all.dat.f1$`3G` <- ifelse(all.dat.f1$`3G` > 0, all.dat.f1$cropland,
    NA)
all.dat.f1$`4G` <- ifelse(all.dat.f1$`4G` > 0, all.dat.f1$cropland,
    NA)
fourgcrop <- sum(all.dat.f1$`4G`, na.rm = T)/sum(all.dat.f1$cropland,
    na.rm = T)
thgcrop <- sum(all.dat.f1$`3G`, na.rm = T)/sum(all.dat.f1$cropland,
    na.rm = T)
twgcrop <- sum(all.dat.f1$`2G`, na.rm = T)/sum(all.dat.f1$cropland,
    na.rm = T)
```

Below we print the technology coverage over global croplands. We can see that as shown in the map, coverage of 2G is almost double that of 4G networks on croplands globally.

```
technology <- c("2G", "3G", "4G")
coverage_cropland <- c(twgcrop, thgcrop, fourgcrop)
data.frame(technology = technology, coverage_cropland = coverage_cropland)
```

```
  technology coverage_cropland
1         2G              0.84
2         3G              0.62
3         4G              0.42
```

We also check coverage by different world regions. First we need to add a region variable to the data to group by.

```
all.dat.f1.sub <- subset(all.dat.f1, cropland.a >= 0.01)
coords2country = function(points) {
    countriesSP <- getMap(resolution = "low")
    pointsSP = SpatialPoints(points, proj4string = CRS(proj4string(countriesSP)))
    indices = over(pointsSP, countriesSP)
    indices$REGION  # returns the continent (7 continent model)
}
all.dat.f1.sub$region <- coords2country(all.dat.f1.sub[, 1:2])
```

Then we use this new region variable to summarize percent coverage by each technology by each region.

```
region.over <- all.dat.f1.sub %>% group_by(region) %>% summarise(c.area = sum(cropland,
    na.rm = T), `2G` = sum(`2G`, na.rm = T)/c.area, `3G` = sum(`3G`,
    na.rm = T)/c.area, `4G` = sum(`4G`, na.rm = T)/c.area) %>%
    na.omit()

region.over
```

```
# A tibble: 6 x 5
  region          c.area  `2G`  `3G`   `4G`
```

```
  <fct>            <dbl> <dbl> <dbl>  <dbl>
1 Africa        2129406. 0.745 0.333 0.0920
2 Asia          5342067. 0.849 0.460 0.293
3 Australia      316503. 0.231 0.759 0.502
4 Europe        3271951. 0.986 0.881 0.539
5 North America 2206951. 0.935 0.995 0.989
6 South America 1578931. 0.615 0.432 0.267
```

# 5 Main Text Figure 1b

## 5.1 Data preparation

In this section we plot the availability of network coverage across different farm size classes. First we read in the farm size data. We created 100 samples for this product, but the variation between each is minimal, so here we simply run with one of them here. The data was also created with an equal area projection. However as we run the analysis in lat long so we reproject the farm size data into lat long. We then append it to the data frame for figure 1.

```
farm.size <- raster("../SI_B/raster_out_forPublication/farmSize_agarea_20201101_1.tif")
resValue <- 0.0833333

farm.size.wgs <- projectRaster(farm.size, res = c(resValue, resValue),
    crs = "+proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0",
    method = "ngb", over = T)

farm.size.wgs <- extend(farm.size.wgs, twg)
extent(farm.size.wgs) <- extent(twg)
all.dat.f1$farmsize <- values(farm.size.wgs)
```

Here we pull off only cropland areas greater than 1% to match the same threshold used in the creation of the farm size layer.

```
all.dat.f1.crop <- subset(all.dat.f1, cropland.a >= 0.01)
```

Now we create a new farm size variable for plotting, we combine some of the the 10-20 and 20-50 bins as the latter has poor representation in pixel counts in the global farm size product due to the assignment algorithm. We also create a >200 bin for large farm sizes as threshold bins for larger farms are typically represented differently by different country surveys.

```
all.dat.f1.crop$farmsize.new <- factor(car::recode(all.dat.f1.crop$farmsize,
    "1='0-1';
                                    2='1-2';
                                    5='2-5';
                                    10='5-10';
                                    c(20,50)='10-50';
                                    c(100)='50-100';
                                    c(200)='100-200';
                                    c(500,1000,5000)='>200';"),
    levels = c("0-1", "1-2", "2-5", "5-10", "10-50", "50-100",
        "100-200", ">200"))
```

We compute coverage for each farm size class, and convert the dataframe from wide to long for plotting.

```
fs.over <- all.dat.f1.crop %>% group_by(farmsize.new) %>% summarise(c.area = sum(cropland,
    na.rm = T), `2G` = sum(`2G`, na.rm = T)/c.area, `3G` = sum(`3G`,
    na.rm = T)/c.area, `4G` = sum(`4G`, na.rm = T)/c.area) %>%
    mutate(eq = c(1:8, NA)) %>% na.omit()

fs.over.l <- fs.over %>% tidyr::gather("2G", "3G", "4G", key = "Service",
    value = "value")
```

## 5.2 Figure 1b

Here we plot the results of coverage across farm size classes. As we can see there is a marked decline in high speed services for smaller farms, while basic 2G services are available across all farm size classes.

```
ggplot(fs.over.l, aes(eq,value*100, colour=Service))+ # geom_point()+
   geom_line()+
   geom_point()+
     geom_line()+
     scale_colour_manual(values = c("#1b9e77ff","#d95f02ff", "#7570b3ff"))+
   theme_bw()+
    theme(plot.background = element_blank(),
          panel.grid.major = element_blank(),
          panel.grid.minor = element_blank(),
        strip.background = element_rect(colour = "white", fill = "white"),
        plot.title = element_text(size = 12, face="bold",hjust = 0.5))+
  ylab("Percentage cover")+
 xlab("Farm size class (hectares)")+
  scale_x_continuous(breaks= 1:8, labels= c("1" ="0-1", "2" ="1-2",
                                            "3" ="2-5","4" ="5-10",
                                          "5" ="10-50","6" ="50-100",
                                          "7"="100-200","8"=">200"))
```

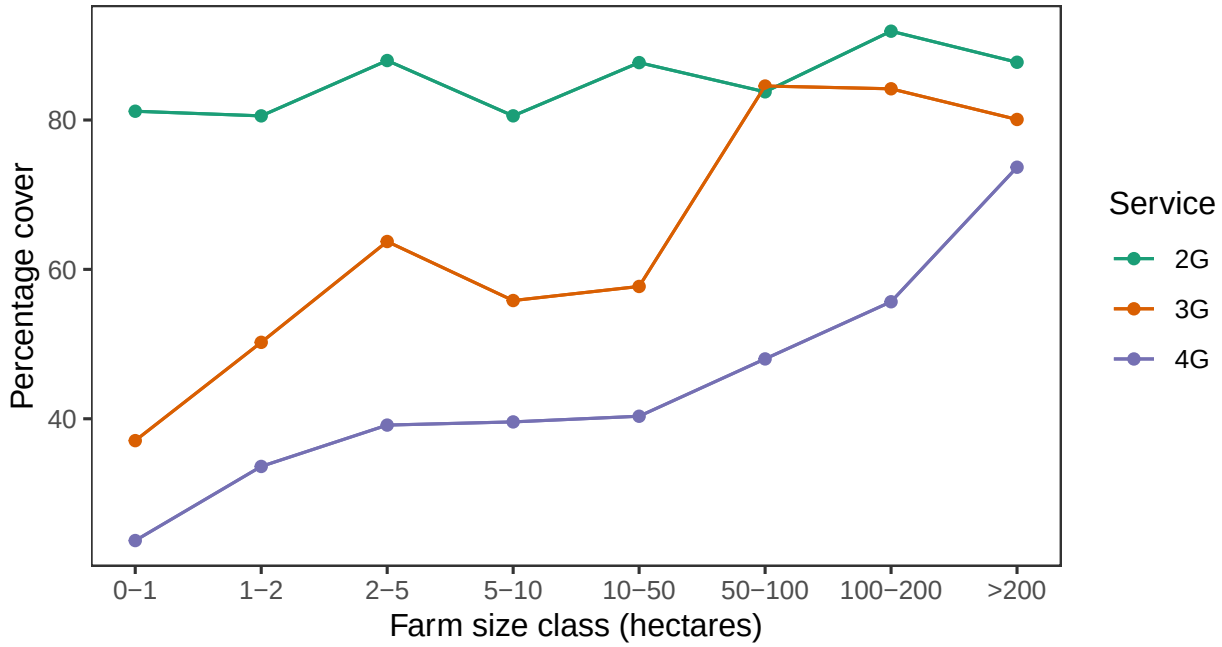


Figure 2: Network coverage across farm size classes

### 5.3 Insights

Here we print the technology coverage over different farm size classes from the figure. Overall, coverage of 4G on small farms (<1ha) is around 1/3 of that of larger farms >200ha. This is a basic reproduction of the data underlying the graphic. Overall, coverage of 4G on small farms (<1ha) is around 1/3 of that of larger farms >200ha

```
fs.over
```

```
# A tibble: 8 x 6
  farmsize.new   c.area  `2G`  `3G`  `4G`    eq
  <fct>           <dbl> <dbl> <dbl> <dbl> <int>
1 0-1          4460327. 0.812 0.371 0.237     1
2 1-2          1050981. 0.806 0.502 0.336     2
3 2-5           785585. 0.880 0.637 0.392     3
4 5-10         1635267. 0.806 0.558 0.396     4
5 10-50         509127. 0.877 0.577 0.403     5
6 50-100       2983166. 0.838 0.845 0.480     6
7 100-200      1096281. 0.919 0.842 0.557     7
8 >200         2226172. 0.877 0.801 0.737     8
```

# 6 Main Text Table 1

In this section we assess network coverage across agricultural areas and human populations of greatest need for data driven interventions, using a geospatial library of demand layers developed in Supplementary Information C.

## 6.1 Data preparation

We write a function to pull in each data layer and estimate the coverage for each. We note some of the population data is in higher resolution (e.g. 0.04167 degrees) than the agricultural data (e.g. 0.083 degrees) so we aggregate it by a factor of 2.

```
service.stack <- stack(twg, thg, fourg)  #set up the service data
coverage <- function(r1, r2) {
    r2 <- raster(r2)
    r2 <- if (round(res(r2)[1], digits = 3) == 0.083) {
        # aggregate if needed
        r2
    } else {
        aggregate(r2, round((0.083)/res(r2)[1]), FUN = sum)
    }
    r2
    dat.rast <- stack(r1, r2)  #stack
    all.dat <- data.frame(coordinates(dat.rast), values(dat.rast))
    colnames(all.dat) <- c("x", "y", "2G", "3G", "4G", "value")
    all.dat$`2G` <- ifelse(all.dat$`2G` > 0, all.dat$value, NA)
    all.dat$`3G` <- ifelse(all.dat$`3G` > 0, all.dat$value, NA)
    all.dat$`4G` <- ifelse(all.dat$`4G` > 0, all.dat$value, NA)
    total = sum(all.dat$value, na.rm = T)
    n2G = sum(all.dat$`2G`, na.rm = T)
    n3G = sum(all.dat$`3G`, na.rm = T)
    n4G = sum(all.dat$`4G`, na.rm = T)
    dat <- data.frame(`2G` = n2G/total, `3G` = n3G/total, `4G` = n4G/total)
    rownames(dat) <- names(r2)
    names(dat) <- names(dat)
    return(dat)
}
```

Then we use this function to estimate coverage for each overlay.

```
overlays <- list.files(c("../SI_C/RESULTS/AREA_BASED", "../SI_C/RESULTS/POP_BASED"),
    full.names = T)
coverage.est <- do.call(rbind, lapply(overlays, function(x) coverage(service.stack,
    x)))
```

## 6.2 Table 1

And finally print the results. We find significant and important gaps in high-speed (i.e. 3G/4G) services in areas affected by nitrogen-deficient cropping areas and severe yield gaps, areas most dependent on rainfall for crop production, and in arid environments (<250mm annual rainfall). We also find that gaps in coverage for individuals affected by food insecurity globally (as proxied by childhood stunting,) and in Africa (as proxied by childhood stunting, wasting and underweight). Similarly we find gaps for people with malaria, and

worryingly populations with low human development indices (e.g. those with the worst access to healthcare, education, and income).

```
data.frame(coverage.est)
```

```
                      X2G  X3G  X4G
aridity              0.80 0.37 0.17
greenH2O             0.93 0.71 0.54
Ndeficit             0.79 0.60 0.22
yieldgap             0.82 0.61 0.21
malaria_ppl_glob     0.82 0.37 0.17
shdi_ppl_glob        0.81 0.39 0.10
stunt_ppl_AFR        0.87 0.52 0.22
stunt_ppl_glob       0.93 0.61 0.45
underweight_ppl_AFR 0.86 0.47 0.19
wasting_ppl_AFR      0.87 0.52 0.25
```

# 7 Main Text Figure 2

## 7.1 Data Preparation

There are two kinds of access data used for creating this figure. The first are cell mobile ownership data and the second are internet access data, both for farming and non-farming households. Both data sets were compiled from nationally representative data sets of cell phone ownership and internet access and explained in Supplementary Information D. The list of files created in this supplement is printed below. Details of the sources can be found in the accompanying SurveyList file.

```
list.files("../SI_D/output data/")
```

```
[1] "Mehrabietal2020_InternetDataset.csv"
[2] "Mehrabietal2020_MobilePhoneDataset.csv"
[3] "Mehrabietal2020_PanelDataset.csv"
[4] "Mehrabietal2020_SurveyList.csv"
[5] "README_Mehrabietal2020_InternetDataset.rtf"
[6] "README_Mehrabietal2020_MobilePhoneDataset.rtf"
[7] "README_Mehrabietal2020_PanelDataset.rtf"
[8] "README_Mehrabietal2020_SurveyList.rtf"
[9] "SI_notebook_revised_01May2020.pdf"
```

One challenge with these data is that is is collected in different years. One option would simply be to present the data "as is", as is often done in data harmonization efforts, but this would pose significant challenges to interpretation of differences between countries or regions, because technology growth has been very fast over the past decades. Instead, to facilitate easier comparisons, we scale these data to a common year as detailed below.

### 7.1.1 Internet access

The world bank provide national data on internet ownership as the share of the population, which we will use to center all sub national internet access estimates to a common year in the following section. Here we read in the necessary data to do that.

First we read in the sub national internet access data. We note that this, and the cell access data have a variable called "region", which is the term used by IPUMS and DHS for sub national units. To avoid confusion, throughout the code when referring to supra-national regions, we use the pre-fix "sp".

```
net <- read.csv("../SI_D/data/Mehrabietal2020_InternetDataset.csv")
```

We then subset the data to represent the proportions of people in each location with internet access.

```
net.sub <- subset(net, internet == TRUE)
```

We add in the world bank codes to the sub national data because we will be joining it with national world bank data later.

```
net.sub$country.new <- countrycode::countrycode(net.sub$country,
    "country.name", "wb", warn = FALSE, custom_match = c(Dominicanrepublic = "DOM"))  #add in DOM manua
```

We check the number of sub national regions in each country that we have data for. One survey – Argentina has only farmers. Other locations have very few (e.g. <3) farming regions in the surveys. We drop these cases here.

```
checkcov <- net.sub %>% group_by(iso, year, ag) %>% summarize(count = n())
few.reg <- subset(checkcov$iso, checkcov$ag == 1 & checkcov$count <
    3)
```

```
onlyag <- setdiff(subset(checkcov$iso, checkcov$ag == 1), subset(checkcov$iso,
    checkcov$ag == 0))
net.sub <- subset(net.sub, !(iso %in% c(few.reg, onlyag)))
```

We then read in the national level trends in internet access. This data are downloaded from the world bank website: https://data.worldbank.org/indicator/it.net.user.zs on Sept 26th 2018. They represent the proportion of a nation's population using the internet. The most recent year in the data with full coverage is 2016.

```
net.nat <- data.frame(readxl::read_excel("Data/National/Internet/API_IT.NET.USER.ZS_DS2_en_excel_v2_1013
    sheet = 1, skip = 2))
```

We subset the national data by the countries represented in the sub national data.

```
net.nat.sub <- subset(net.nat, Country.Code %in% net.sub$country.new)
```

Next we transform the national data from wide to long.

```
net.nat.sub.l <- net.nat.sub %>% tidyr::gather(Year, Percent,
    X1960:X2017)
net.nat.sub.l$Year <- as.numeric(sub(".", "", net.nat.sub.l$Year))
net.nat.sub.l <- subset(net.nat.sub.l, Year %in% (1990:2016))
```

### 7.1.2 Centering internet access

Next we fit a generalized additive model (gam) to the national internet access data to get smoothed time series predictions of the proportion of people accessing internet in each country for each year. We set the smoothing parameter $\gamma$, which controls the trade-off between smoothness of the estimated function and fidelity to the data, to 1.5 to avoid possible over fitting of the smoothness selection.

We will use this model to identify the percentage point differences in internet access between the observed year in the sub national data, and a common reference year across all data, then use this percentage point difference to scale the observed ownership within the farming and non-farming populations across countries to a common year.

```
net.nat.sub.l$Country.Code <- as.factor(net.nat.sub.l$Country.Code)
net.nat.sub.l$prop <- net.nat.sub.l$Percent/100

full.mod.net <- gam(prop ~ s(Year, by = Country.Code, bs = "cs") +
    Country.Code, data = net.nat.sub.l, gamma = 1.5, family = betar,
    link = "logit")
```

Initially we check functional form of the predictions for each country. We note that these fits look good, and do not suggest a need for increasing the basis dimension.

```
fitbyvarnet <- data.frame(cbind(full.mod.net$model, fit = fitted.values(full.mod.net),
    response = full.mod.net$y))

ggplot(fitbyvarnet, aes(Year, response)) + geom_point() + geom_line(aes(Year,
    fit), colour = "red") + facet_wrap(~Country.Code)
```

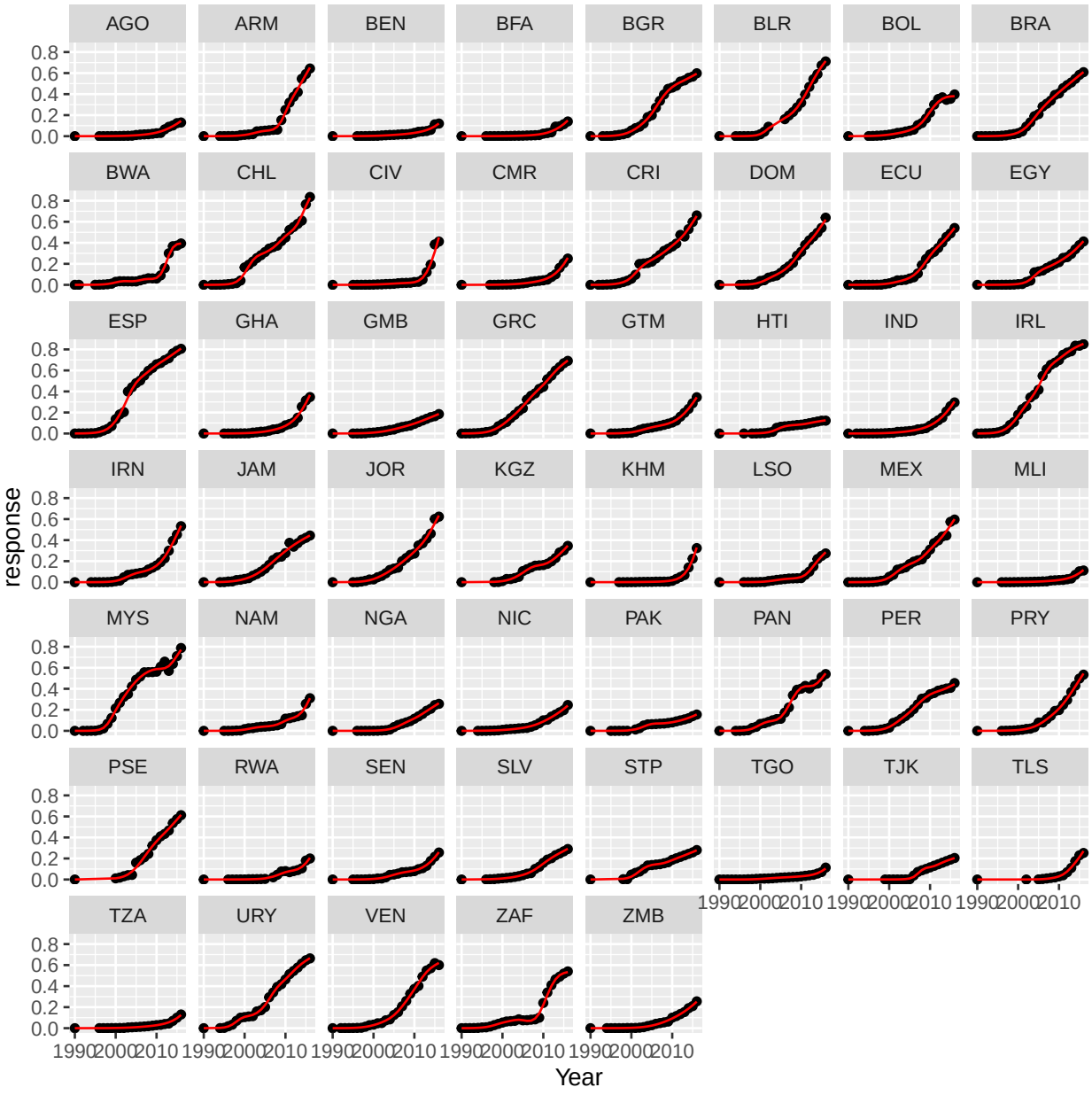


Figure 3: Trends in national internet access

We a check of the goodness of fit for each country, which shows the predictions are extremely tight for this model, and suitable for our purposes of centering the data.

```
ggplot(fitbyvarnet, aes(fit, response)) + geom_jitter() + geom_smooth(method = "lm",
    colour = "black", se = F) + facet_wrap(~Country.Code)
```

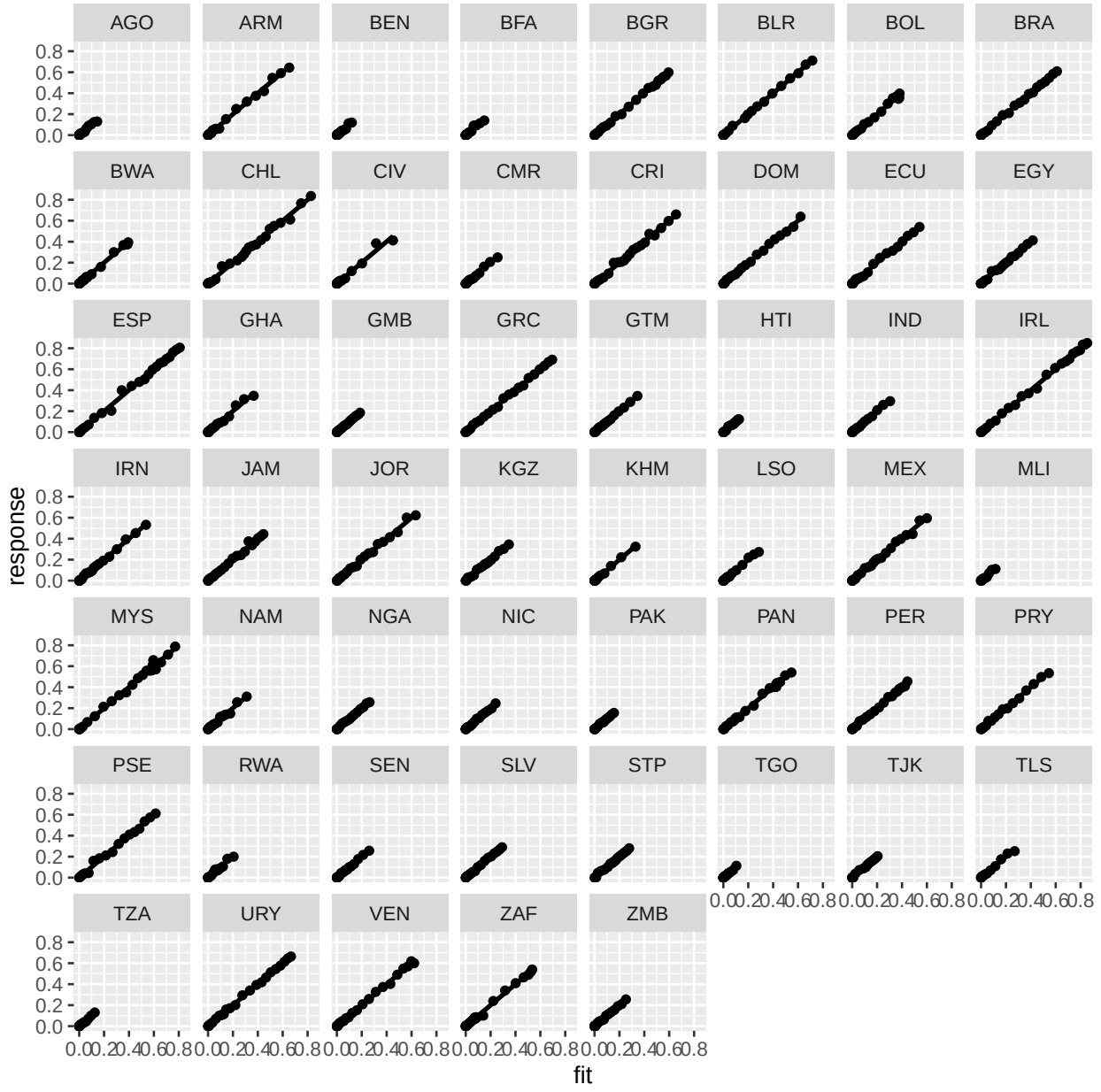


Figure 4: Goodness of fit for model on internet access trends

We use this model to create a data frame predicting out for the range of years in the sub national data and extrapolating for two years ahead of the data set to 2018. First we create a data frame.

```
countries <- unique(net.nat.sub.l$Country.Code)
rge <- 2018 - 1990 + 1  #range
year <- rep(1990:2018, length(countries))  #yr
c <- rep(countries, times = 1, each = rge)  #group
newdat <- data.frame(prop = NA, Year = year, Country.Code = c)  #new data
# sapply(full.mod.net$smooth, '[[', 'label') #check terms
```

We predict from the model to center sub-national data on a common year, assuming the trajectories of each sub population will follow the same trajectory (note, we relax this assumption for the cell phone access data where we have time series of access dis-aggregated by farming and non-farming populations).

```
pr <- predict(full.mod.net, newdat, se.fit = T)
newdat.p <- cbind(fit = pr$fit, newdat)
newdat.p$fit.r <- full.mod.net$family$linkinv(newdat.p$fit)  #fit on respose
```

Next we create variables to match the predictions with the sub national data, and then match up the predictions for the observed year and for 2018.

```
newdat.p$cy <- paste(newdat.p$Country.Code, newdat.p$Year, sep = "_")  #create matches
net.sub$cy <- paste(net.sub$iso, net.sub$year, sep = "_")
net.sub$fit.obs <- newdat.p[match(net.sub$cy, newdat.p$cy), ]$fit.r  #add  predicted value observed yea
net.sub$cy.2018 <- paste(net.sub$iso, 2018, sep = "_")  #add match for 2018
net.sub$fit.2018 <- newdat.p[match(net.sub$cy.2018, newdat.p$cy),
    ]$fit.r  #add predicted for 2018
```

We then identify the percentage point difference in the national data in access, and add that percentage point difference to the observed sub national data, to create a scaled version of the observed values for all countries centered on 2018.

```
net.sub$diff.fit <- net.sub$fit.2018 - net.sub$fit.obs
net.sub$prop.scaled <- net.sub$prop + net.sub$diff.fit
# summary(net.sub$prop.scaled>1) #<1% of the data is
# predicted to reach total ownership by 2018.
net.sub$prop.scaled.trunc <- ifelse(net.sub$prop.scaled > 1,
    1, net.sub$prop.scaled)  #so we truncate the ownership for those cases
net.sub$perc <- round(net.sub$prop.scaled.trunc * 100)  #create percentage response
```

Finally, we add in some variables for plotting, first we add in the region variable, and remove European data.

```
getspregion <- function(x, format) {
    sub.cont <- countrycode::countrycode(x, paste(format), "region")
    spregion <- car::recode(sub.cont, "c('Australia and New Zealand') = 'Aus.NZ';
c('Northern America') = 'N. America';
c('Eastern Europe', 'Western Europe', 'Northern Europe','Southern Europe') = 'Europe';
c('Southern Africa','Eastern Africa', 'Middle Africa','Northern Africa','Western Africa') = 'Africa';
c('Southern Asia','Eastern Asia','Central Asia', 'Middle Asia','Northern Asia','Western Asia','South-Eas
c('Central America','South America', 'Caribbean') = 'LAm.Carib'")
}

net.sub$spregion <- getspregion(net.sub$iso, "iso3c")
net.sub <- subset(net.sub, !(spregion == "Europe"))
# net.sub<-net.sub[-which(net.sub$spregion == 'Europe'),]
net.sub$country2 <- countrycode::countrycode(net.sub$iso, "iso3c",
```

```
    "country.name")
```

We also add the farm id variable.

```
net.sub$ag.new <- car::recode(net.sub$ag, "0='Non-farmer';
                      1='Farmer'")
summary(factor(net.sub$ag.new))
```

```
    Farmer Non-farmer
      4745       5064
```

### 7.1.3 Cell access

For cell phone access we use supra-national regional trends in access to center country level trends on a common year. Unlike for internet access, national cell ownership time series do not exist. The World Bank distribute cell subscriptions data as a share of the population, but these data are fraught with problems – the most obvious being that mobile subscription data, which is available for mobiles, poorly reflect per capita access. To generate the percentage point increase from the year of the data to present we instead use panel data sets for countries in supra-national regions (e.g. Africa, Asia, Latin America  Caribbean) to scale all access for countries in those regions to a common year.

First we read in the non-panel cell access data.

```
cell <- read.csv("../SI_D/data/Mehrabietal2020_MobilePhoneDataset.csv")
```

We note that this data set contain multiple surveys for the same country from different sources (due to updates), so we select the most recent survey for each country here. We also remove Argentinian data which contain only agricultural responses.

```
mx.cell <- cell %>% group_by(country) %>% summarize(year.mx = max(year)) %>%
    mutate(nat.yr = paste(country, year.mx))
cell$nat.yr <- paste(cell$country, cell$year)
cell.sub <- subset(cell, nat.yr %in% mx.cell$nat.yr)
onlyag <- setdiff(subset(cell.sub$iso, cell.sub$ag == 1), subset(cell.sub$iso,
    cell.sub$ag == 0))  #remove Argentina
cell.sub <- subset(cell.sub, !(iso %in% onlyag))
```

We then subset the data to represent the proportions of people in each location with cell access.

```
cell.sub <- subset(cell.sub, cell == TRUE)
```

Next, we read in the panel data sets.

```
panel <- read.csv("../SI_D/data/Mehrabietal2020_PanelDataset.csv")
```

Next we get the subset the data by mobile phone owners only.

```
panel.sub <- subset(panel, mobile == TRUE)
panel.all <- panel.sub
```

Then we add the new farmer coding to this panel data set.

```
panel.all$ag.new <- car::recode(panel.all$ag, "0='Non-farmer';
                      1='Farmer'")
summary(factor(panel.all$ag.new))  #check
```

```
    Farmer Non-farmer
       603        603
```

We add supra-national regional identifiers for each country.

```
panel.all$spregion <- getspregion(panel.all$country, "country.name")
```

Finally, we note that there are some admin unit coverage inconsistency across different waves of the DHS surveys. While there are possible ways to account include these sampling frame differences in later modelling steps when estimating country level, here for simplicity we simply remove admin units which are not represented throughout the time series.

```
reg.to.drop <- panel.all %>% mutate(cr = as.factor(paste(country,
    region))) %>% group_by(cr, country) %>% summarize(count = n()/2) %>%
    filter(count < 3) %>% select(cr)

panel.all$cr <- as.factor(paste(panel.all$country, panel.all$region))
panel.all <- subset(panel.all, !(cr %in% reg.to.drop$cr))
panel.all$cr <- factor(panel.all$cr)
```

### 7.1.4 Centering cell access

Here we use the average trends in access across countries in a supra-national region (e.g. Africa, Asia, Latin America  Caribbean) to center the cell access data across a wider number of countries in those regions on a given year. We do not investigate the error in regional estimates introduced by this convenience sampling (although this could be improved on in future iterations of this data set). Due to the sparsity of the data we will rely on theory of technology diffusion, and fit shape constrained models to estimate the regional increase in cell phone access.

First, we anchor the time series for the span of the prediction window is contained in the bounds of 0-1 within a reasonable range of time. We anchor the time series using mobile subscription data accessed from the World Bank on Aug 8th 2019 from https://data.worldbank.org/indicator/it.net.user.zs. We divide the subscription indicator by a arbitrary constant of three (equivalent to three subscriptions per person) to arrive at an ownership anchor. First we check the date ranges.

```
cell.script <- read.csv("Data/National/API_IT.CEL.SETS.P2_DS2_en_csv_v2_54223/API_IT.CEL.SETS.P2_DS2_en_
cell.script.sub <- subset(cell.script, Country.Code %in% panel.all$iso)
cell.script.2000 <- data.frame(iso = cell.script.sub$Country.Code,
    year = 2000, prop = (cell.script.sub$X2000)/300)
```

We then identify the range of years for each survey, create a duplicate data frame to insert the anchor data, assign the anchor data to it, and then bind this to the panel data set.

```
minmax <- panel.all %>% group_by(spregion, country, region, ag.new) %>%
    summarize(miny = min(year), maxy = max(year), dif = miny -
        maxy, n = n()) %>% mutate(crminy = paste(country, region,
    miny))  #identify min, max year for surveys

panel.all$cry <- paste(panel.all$country, panel.all$region, panel.all$year)
panel.min <- subset(panel.all, cry %in% minmax$crminy)  #duplicate data.frame for anchor data
panel.min$prop <- cell.script.2000[match(panel.min$iso, cell.script.2000$iso),
    ]$prop
panel.min$year <- cell.script.2000[match(panel.min$iso, cell.script.2000$iso),
    ]$year
panel.min$weight <- (panel.min$denom/panel.min$scale) * panel.min$prop  #add in numerator (needed for m
panel.all <- rbind(panel.all, panel.min)  #bind
```

Finally we create a variable that interacts the continent with the farmer variable.

```
panel.all$ca <- as.factor(paste(panel.all$spregion, panel.all$ag.new,
    sep = "_"))
```

Then we set up the model. We use a binomial distribution with a logit link to model these data. We constrain the fit using a penalized monotonically increasing function to represent the expected dynamics of adoption.

```
panel.all$country <- as.factor(panel.all$country)
panel.all$cr <- factor(panel.all$cr)

panel.all$success <- round(panel.all$weight)
panel.all$fail <- (round(panel.all$weight)/panel.all$prop) *
    (1 - panel.all$prop)

full.mod.cell <- scam(cbind(success, fail) ~ s(year, by = ca,
    bs = "mpi") + ca + s(country, bs = "re"), data = panel.all,
    family = binomial)
```

We assess the response fitted vs response for each country modeled. The fits look reasonable.

```
fitbyvar <- data.frame(cbind(full.mod.cell$model, fit = fitted.values(full.mod.cell),
    response = full.mod.cell$y, byvar = paste(full.mod.cell$model$ca,
        full.mod.cell$model$country), id = rownames(full.mod.cell$model)))

ggplot(fitbyvar, aes(fit, response)) + geom_jitter() + geom_smooth(method = "lm",
    colour = "black", se = F) + facet_wrap(~byvar)
```

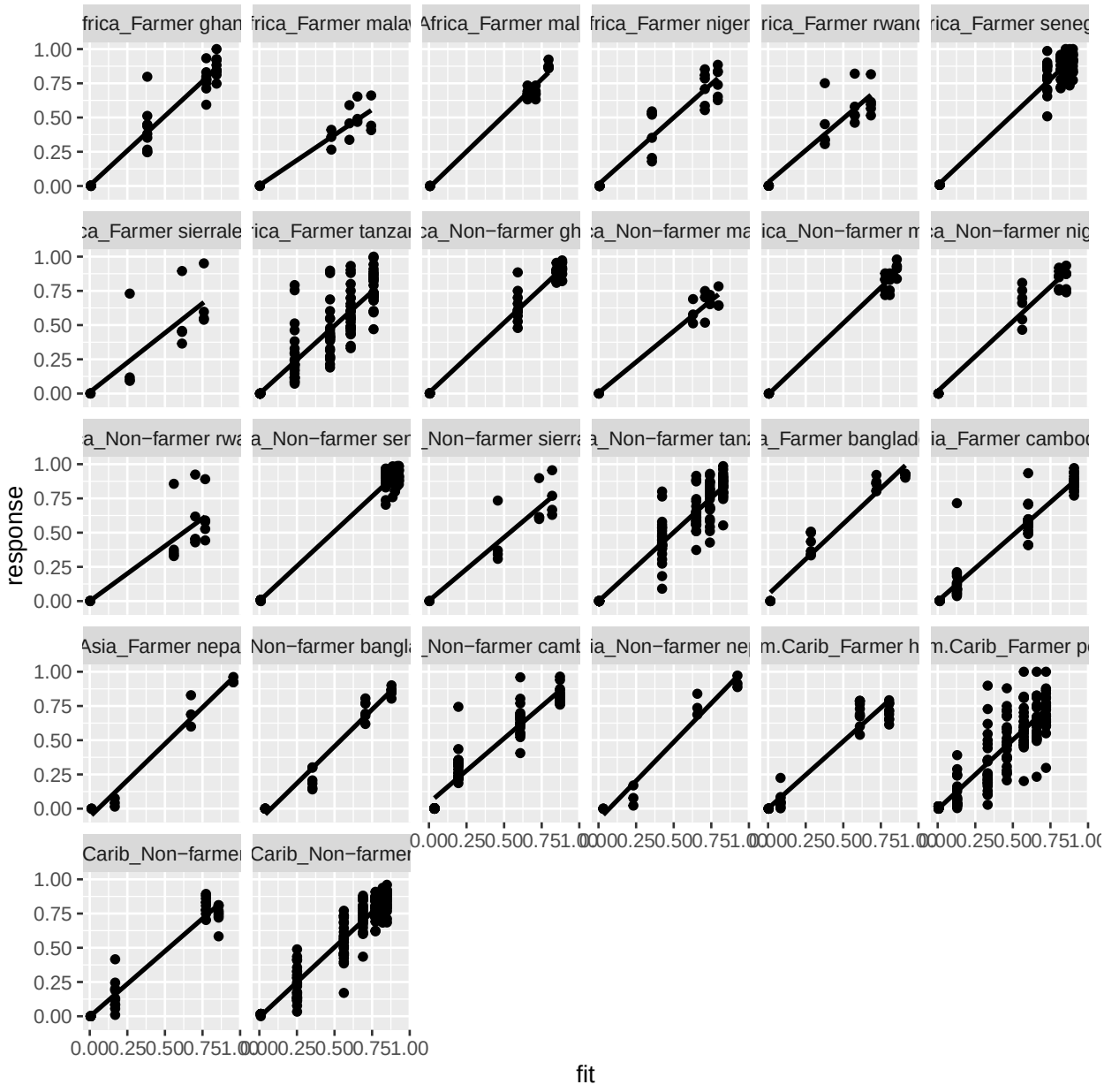


Figure 5: Goodness of fit for cell phone access

We also check the smooth functions across the range of predictions on the response scale. To do this we create a new data frame with predicted values for the range of the data, including an extrapolation to the year 2018.

```
newdat <- expand.grid(country = unique(panel.all$country), year = c(min(cell.sub$year -
    10):max(cell.sub$year + 1)), ca = unique(panel.all$ca), prop = NA)
newdat$ca.country <- paste(newdat$country, newdat$ca)
newdat <- subset(newdat, ca.country %in% paste(panel.all$country,
    panel.all$ca))
newdat$prop <- predict(full.mod.cell, newdat, se.fit = T)$fit  #get fit,#NB: exclude doesn't work on sc
newdat$fit.r <- full.mod.cell$family$linkinv(newdat$prop)
newdat$ag.new <- ifelse(grepl("_Non-farmer", newdat$ca), "Non-farmer",
```

```
    "Farmer")
newdat$spregion <- ifelse(grepl("Asia", newdat$ca), "Asia", ifelse(grepl("Africa",
    newdat$ca), "Africa", "LAm.Carib"))
```

And then we create plots of the smooths. The smooth functions also look reasonable when predicted at the country level, and so we will use them for time centering these data.

```
ggplot(panel.all, aes(year, prop, colour = country)) + geom_point(aes(year,
    prop, group = as.factor(cr)), lwd = 0.3) + geom_line(aes(year,
    prop, group = as.factor(cr)), lwd = 0.3, alpha = 0.2) + geom_line(data = newdat,
    aes(year, fit.r)) + facet_wrap(country ~ ag.new + spregion) +
    theme_bw() + theme(legend.position = "None")
```

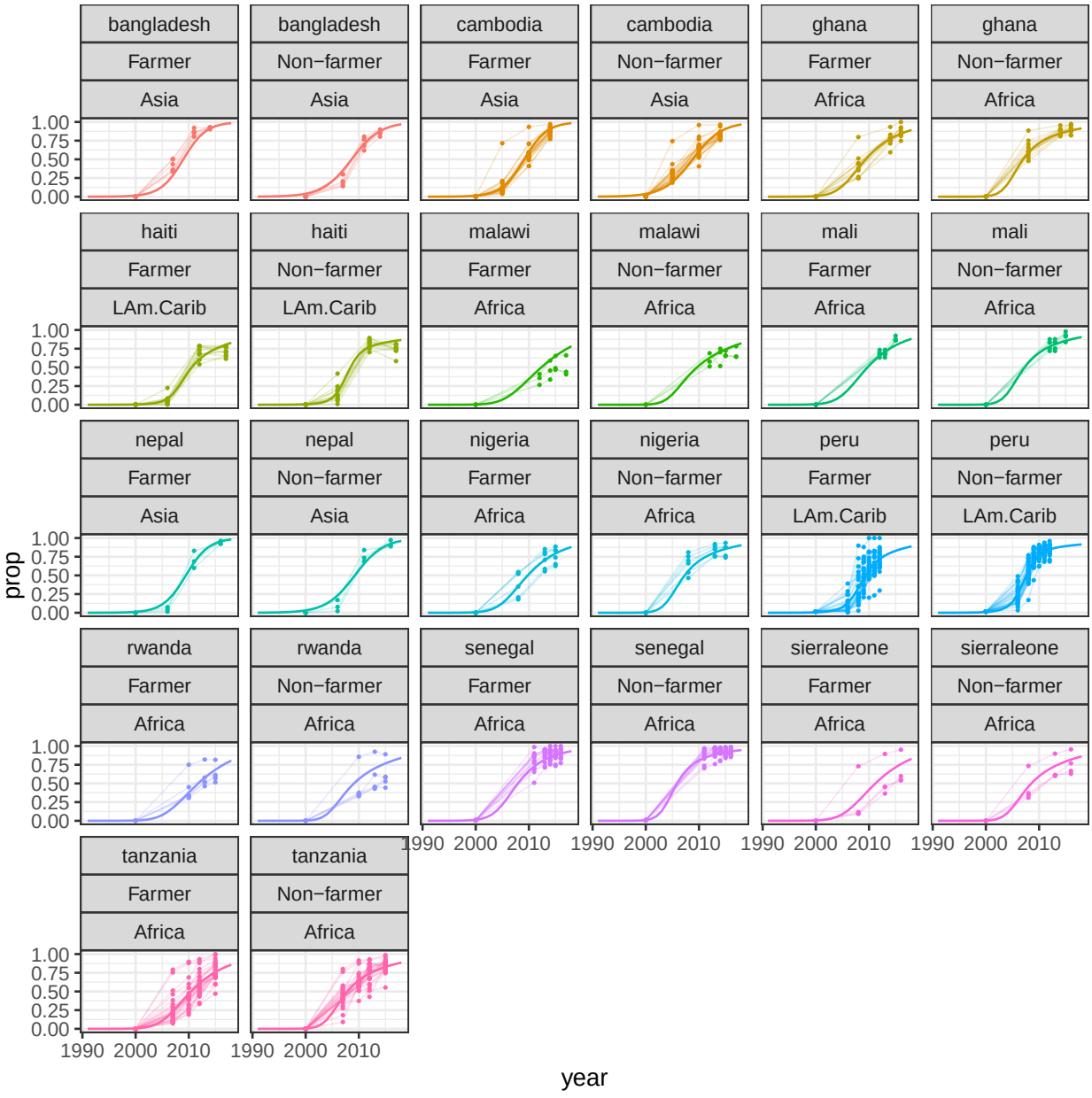


Figure 6: Time series of cell ownership for select countries

Next we average over the predictions for each supranational region to get regional time series to center the rest of the data with. We have to take this step because the `exclude` argument in `predict.gam()` does not work for `predict.scam()`.

```
newdat.av <- newdat %>% group_by(year, ca, ag.new, spregion) %>%
    summarize(fit.r = mean(fit.r))
```

Next we need to add some new variables – first the ag.new varible.

```
cell.sub$ag.new <- car::recode(cell.sub$ag, "0='Non-farmer';
                        1='Farmer'")
```

Second we add the supra-national region variable. We remove Europe because we will not be using Europe for this analysis.

```
cell.sub$spregion <- getspregion(cell.sub$country, "country.name")
cell.sub <- cell.sub[-which(cell.sub$spregion == "Europe"), ]
cell.sub$country2 <- countrycode::countrycode(cell.sub$iso, "iso3c",
    "country.name")
```

We then create matching variables and then match up the predictions for the observed year of the survey and for 2018.

```
newdat.av$cay <- paste(newdat.av$ca, newdat.av$year, sep = "_")  #create matches
cell.sub$cay <- paste(cell.sub$spregion, cell.sub$ag.new, cell.sub$year,
    sep = "_")
cell.sub$fit.obs <- newdat.av[match(cell.sub$cay, newdat.av$cay),
    ]$fit.r  #add continental predicted value observed year
cell.sub$cay.2018 <- paste(cell.sub$spregion, cell.sub$ag.new,
    2018, sep = "_")  #add match for 2018
cell.sub$fit.2018 <- newdat.av[match(cell.sub$cay.2018, newdat.av$cay),
    ]$fit.r  #add predicted for 2018
```

And finally we find the percentage point difference from the supra-national regional trends in access and add those differences to the original survey proportions to create a centered cell access data on the year 2018. We note that based on this scaling around 30% of the sub populations are expected to have achieved 100% access by this year, and so we truncate accordingly.

```
cell.sub$diff.fit <- cell.sub$fit.2018 - cell.sub$fit.obs
cell.sub$prop.scaled <- cell.sub$prop + cell.sub$diff.fit
summary(cell.sub$prop.scaled > 1)  #~ 30% of the data is predicted to reach total ownership
```

```
   Mode   FALSE    TRUE
logical    7175    3684
```

```
cell.sub$prop.scaled.trunc <- ifelse(cell.sub$prop.scaled > 1,
    1, cell.sub$prop.scaled)  #so we truncate the ownership
cell.sub$perc <- round(cell.sub$prop.scaled.trunc * 100)  #create percentage response.
```

## 7.2 Figure 2

Here we create the plot for figure 2. First we create a data frame for plotting, including the creation of variables to order countries alphabetically within supra-national regions.

```
net.sub$order.reg <- as.numeric(paste(as.numeric(as.factor(net.sub$spregion)),
    1/as.numeric(as.factor(net.sub$country2)), sep = ""))

cell.sub$order.reg <- as.numeric(paste(as.numeric(as.factor(cell.sub$spregion)),
```

```
    1/as.numeric(as.factor(cell.sub$country2)), sep = ""))

perc <- c(cell.sub$perc, net.sub$perc)

type <- c(rep("cell", nrow(cell.sub)), rep("internet", nrow(net.sub)))
country <- c(cell.sub$country2, net.sub$country2)
order.reg <- c(cell.sub$order.reg, net.sub$order.reg)
ag.new <- c(cell.sub$ag.new, net.sub$ag.new)
spregion <- c(cell.sub$spregion, net.sub$spregion)
admin.unit <- c(paste(cell.sub$country, cell.sub$region), paste(net.sub$country,
    net.sub$region))
all.dat.f2 <- data.frame(spregion = spregion, perc = perc, order.reg = order.reg,
    country = country, type = type, ag.new = ag.new, admin.unit = admin.unit)
```

We check the data with a simple plot, where each point represents a different sub national unit, with colours differentiated by farming and sub national populations. While this plot is useful to show the underlying data points, the overlap between points makes the distribution of the sub national access patterns difficult to see.

```
ggplot(all.dat.f2, aes(x = perc, y = reorder(country, order.reg),
    color = ag.new)) + geom_jitter(alpha = 0.2) + xlim(0, 100) +
    scale_y_discrete(expand = c(0.01, 0)) + scale_fill_manual(values = c("#d95f02ff",
    "#7570b3ff"), labels = c("Farmer", "Non-Farmer")) + scale_color_manual(values = c("#d95f02ff",
    "#7570b3ff")) + ylab("") + xlab("Percentage of households") +
    facet_wrap(~type) + theme_bw()
```


Figure 7: Subnational distributions of cellphone and internet access for farming and non-farming households

One option for displaying these data would be to complement them with a harmonized shapefile, and create four maps, for access to mobiles and internet and for farming and non-farming populations. Here we simply fit Gaussian probability density functions for each farming and non-farming sub-population using the access data from administrative unit in each country. This allows for a quick visualisation of major differences (although carries its own trade-offs). For further insights, we create numerical summaries below for the raw data underlying the plot.

```
ggplot(all.dat.f2, aes(x = perc, y = reorder(country, order.reg),
    height = ..density.., color = "white", fill = ag.new)) +
    geom_density_ridges(scale = 2.5, stat = "density", bw = 5,
        color = "white", alpha = 0.7, kernal = "guassian") +
    xlim(0, 100) + scale_y_discrete(expand = c(0.01, 0)) + scale_fill_manual(values = c("#d95f02ff",
    "#7570b3ff"), labels = c("Farmer", "Non-Farmer")) + scale_color_manual(values = c("#d95f02ff",
    "#7570b3ff"), guide = "none") + theme(legend.position = "none") +
    ylab("") + xlab("Percentage of households") + theme_ridges(center = TRUE) +
    facet_wrap(~type)
```

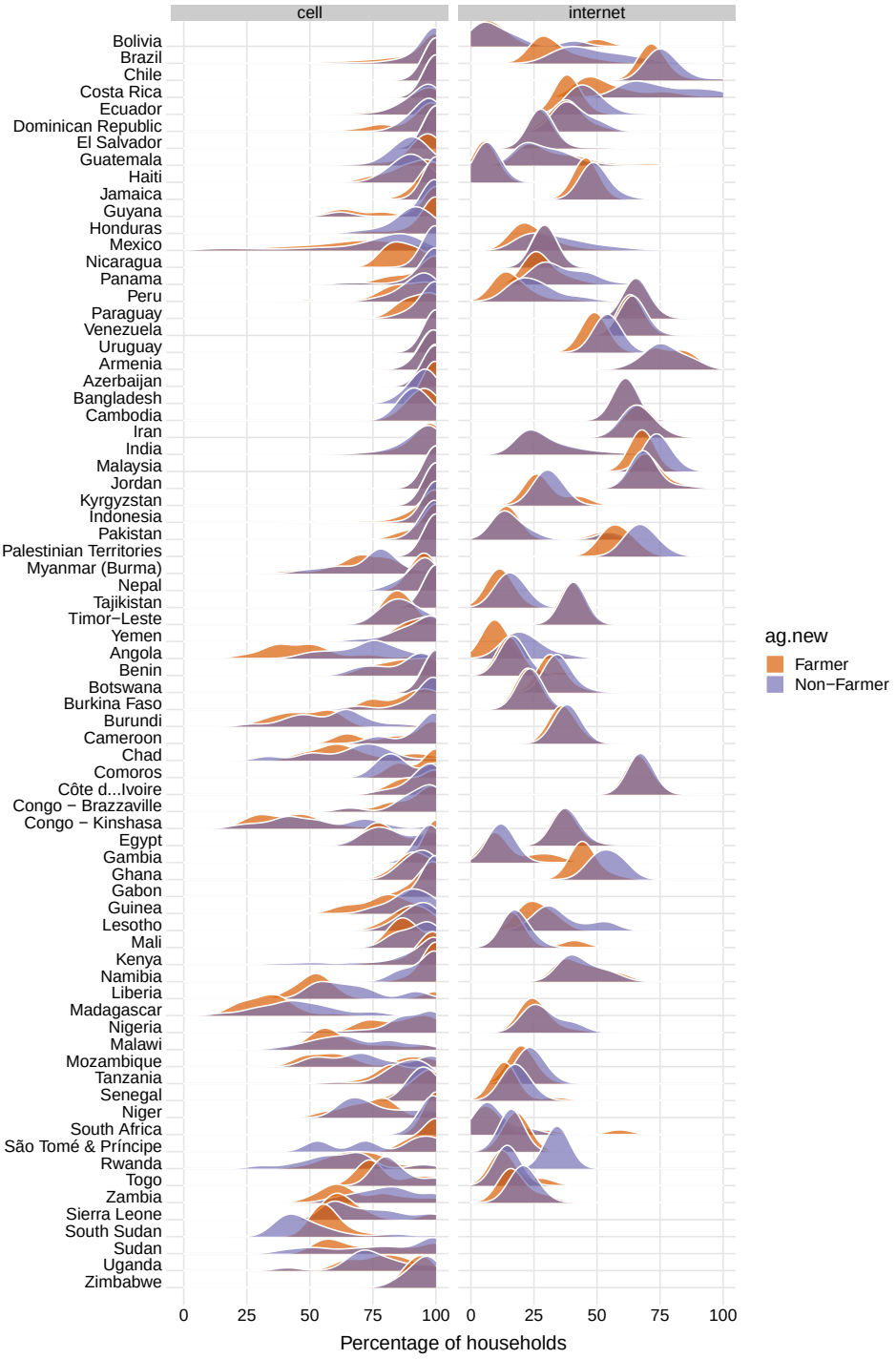


Figure 8: Density plots of subnational distributions of cellphone and internet access for farming and non-farming households

## 7.3 Insights

Here we assess the average access values by country and by supra-national region for further insight into the data underlying figure 2. First we check the total sub national units in each region for each technology, note the large numbers for Latin American region, which are driven by census surveys for Brazil and Mexico which are representative at the municipality level.

```
admin.count <- all.dat.f2 %>% group_by(type, spregion) %>% summarize(n.admin.units = n()/2)
admin.count
```

```
# A tibble: 6 x 3
# Groups:   type [2]
  type     spregion  n.admin.units
  <fct>    <fct>             <dbl>
1 cell     Africa             941
2 cell     Asia               848
3 cell     LAm.Carib         3640.
4 internet Africa             522.
5 internet Asia               931
6 internet LAm.Carib         3452
```

Then we average the access metrics by country. We just print the header here, but you can retrieve the full table by re-running this code.

```
country.av <- all.dat.f2 %>% group_by(spregion, country, type,
    ag.new) %>% summarize(mean = mean(perc), sd = sd(perc), n.admin.units = n()) %>%
    as.data.frame()

country.av[1:10, ]
```

```
   spregion  country     type     ag.new mean   sd n.admin.units
1    Africa   Angola     cell     Farmer   49 17.0            18
2    Africa   Angola     cell Non-farmer   72 13.1            18
3    Africa   Angola internet     Farmer   11  6.0            18
4    Africa   Angola internet Non-farmer   21  7.2            18
5    Africa    Benin     cell     Farmer   88 10.1            12
6    Africa    Benin     cell Non-farmer   88 10.9            12
7    Africa    Benin internet     Farmer   17  5.4            12
8    Africa    Benin internet Non-farmer   17  2.3            11
9    Africa Botswana     cell     Farmer  100  1.5            21
10   Africa Botswana     cell Non-farmer  100  0.0            21
```

We then average the access metrics by supra-national region, using the averages at the country level.

```
cont.av.country <- country.av %>% group_by(spregion, type, ag.new) %>%
    summarize(meanr = mean(mean), minr = min(mean), maxr = max(mean),
        n.countries = n()) %>% as.data.frame()
cont.av.country
```

```
    spregion     type     ag.new meanr minr maxr n.countries
1     Africa     cell     Farmer    80 33.5  100          38
2     Africa     cell Non-farmer    82 45.3  100          38
3     Africa internet     Farmer    27 10.8   67          20
4     Africa internet Non-farmer    29  9.2   67          20
5       Asia     cell     Farmer    95 71.3  100          15
6       Asia     cell Non-farmer    95 73.9  100          15
7       Asia internet     Farmer    49 11.6   78          11
8       Asia internet Non-farmer    50 15.8   77          11
```

```
9  LAm.Carib     cell     Farmer    93 62.7  100          17
10 LAm.Carib     cell Non-farmer    95 73.1  100          17
11 LAm.Carib internet     Farmer    38  5.4   73          17
12 LAm.Carib internet Non-farmer    42  6.5   77          17
```

We average the access metrics by supra-national region, here by pooling.These pooled estimates are very similar to the unpooled ones.

```
spregion.av.sub <- all.dat.f2 %>% group_by(spregion, type, ag.new) %>%
    summarize(mean = mean(perc), LQT = quantile(perc, probs = 0.25),
        UQT = quantile(perc, probs = 0.75), n.admin.units = n())

spregion.av.sub
```

```
# A tibble: 12 x 7
# Groups:   spregion, type [6]
   spregion  type     ag.new      mean   LQT   UQT n.admin.units
   <fct>     <fct>    <fct>      <dbl> <dbl> <dbl>         <int>
 1 Africa    cell     Farmer      78.3    64    98           941
 2 Africa    cell     Non-farmer  80.2    70    99           941
 3 Africa    internet Farmer      29.5    19    36           491
 4 Africa    internet Non-farmer  32.2    23    37           552
 5 Asia      cell     Farmer      94.8    93   100           848
 6 Asia      cell     Non-farmer  94.2    91   100           848
 7 Asia      internet Farmer      40.9    23    61           935
 8 Asia      internet Non-farmer  40.4    23    61           927
 9 LAm.Carib cell     Farmer      80.0    67   100          3641
10 LAm.Carib cell     Non-farmer  86.3    81   100          3640
11 LAm.Carib internet Farmer      33.1    23    38          3319
12 LAm.Carib internet Non-farmer  43.4    30    55          3585
```

The above tables are a bit unwieldy for finding results, so we select of countries with the lowest average access statistics below. First we show the 10 countries with lowest access metric for cell phones.

```
lowest.cell <- subset(country.av, type == "cell" & ag.new ==
    "Farmer")
lowest.net <- subset(country.av, type == "internet" & ag.new ==
    "Farmer")
lowest.cell[order(lowest.cell$mean)[1:10], ]
```

```
     spregion          country type ag.new mean   sd n.admin.units
59     Africa       Madagascar cell Farmer   34  9.7            22
29     Africa Congo - Kinshasa cell Farmer   44 17.1            26
1      Africa           Angola cell Farmer   49 17.0            18
17     Africa          Burundi cell Farmer   51 14.8            18
97     Africa      South Sudan cell Farmer   58  8.4            42
57     Africa          Liberia cell Farmer   60 20.4             6
211 LAm.Carib           Mexico cell Farmer   63 23.4          1654
61     Africa           Malawi cell Farmer   65 15.7            32
23     Africa             Chad cell Farmer   67 16.5            21
91     Africa     Sierra Leone cell Farmer   68 14.3            14
```

And then we print the 10 countries with the lowest access metrics for internet.

```
lowest.net[order(lowest.net$mean)[1:10], ]
```

```
     spregion             country     type ag.new mean   sd n.admin.units
203 LAm.Carib               Haiti internet Farmer  5.4  1.3            11
```

```
3      Africa              Angola internet Farmer 10.8  6.0            18
161      Asia          Tajikistan internet Farmer 11.6  2.6             5
95     Africa        South Africa internet Farmer 14.4 18.5             9
169 LAm.Carib             Bolivia internet Farmer 14.5 18.0             6
89     Africa             Senegal internet Farmer 14.7  5.7            14
107    Africa                Togo internet Farmer 14.8  6.5             6
43     Africa              Gambia internet Farmer 15.0  9.9             7
7      Africa               Benin internet Farmer 16.8  5.4            12
85     Africa So Tom & Prncipe internet Farmer 18.0  2.8             2
```

We also order internet access in African countries here, from lowest to highest.

```
lowest.net.Af <- subset(country.av, type == "internet" & spregion ==
    "Africa" & ag.new == "Farmer")
lowest.net.Af[order(lowest.net.Af$mean), ]
```

```
    spregion             country     type ag.new mean   sd n.admin.units
3     Africa              Angola internet Farmer   11  6.0            18
95    Africa        South Africa internet Farmer   14 18.5             9
89    Africa             Senegal internet Farmer   15  5.7            14
107   Africa                Togo internet Farmer   15  6.5             6
43    Africa              Gambia internet Farmer   15  9.9             7
7     Africa               Benin internet Farmer   17  5.4            12
85    Africa So Tom & Prncipe internet Farmer   18  2.8             2
113   Africa              Zambia internet Farmer   18  5.6            10
103   Africa            Tanzania internet Farmer   20  2.7           113
65    Africa                Mali internet Farmer   20 10.1             6
15    Africa        Burkina Faso internet Farmer   23  1.3             5
77    Africa             Nigeria internet Farmer   26  5.0            35
55    Africa             Lesotho internet Farmer   26  5.1            10
11    Africa            Botswana internet Farmer   32  3.4            19
81    Africa              Rwanda internet Farmer   34   NA             1
21    Africa            Cameroon internet Farmer   37  2.9            12
37    Africa               Egypt internet Farmer   38  6.6           180
71    Africa             Namibia internet Farmer   44  8.7            12
47    Africa               Ghana internet Farmer   45  3.8            10
33    Africa       Cte dIvoire internet Farmer   67  2.8            10
```

Finally we work out the average differences in internet and cell access scores between farmers and non-farmers at the country level

```
differences <- all.dat.f2 %>% mutate(type.ag = paste(type, ag.new,
    sep = "")) %>% select(type.ag, admin.unit, spregion, perc,
    country) %>% tidyr::spread(key = type.ag, value = perc) %>%
    mutate(cell.diff = `cellNon-farmer` - cellFarmer, net.diff = `internetNon-farmer` -
        internetFarmer) %>% group_by(spregion, country) %>% summarize(mean.cell.diff = mean(cell.diff,
    na.rm = T), mean.net.diff = mean(net.diff, na.rm = T))
```

And then we print the countries with the highest average gap between farmers and non-farmers in access, first those for mobile.

```
differences[order(-differences$mean.cell.diff)[1:10], ]
```

```
# A tibble: 10 x 4
# Groups:   spregion [2]
   spregion  country          mean.cell.diff mean.net.diff
   <fct>     <fct>                     <dbl>         <dbl>
```

```
 1 Africa    Angola                   23.6         10.7
 2 LAm.Carib Nicaragua                12.1          0.182
 3 Africa    Madagascar               11.7        NaN
 4 Africa    Congo - Kinshasa         11.2        NaN
 5 Africa    Zambia                   11.1          2.9
 6 LAm.Carib Mexico                   10.4          9.53
 7 Africa    Guinea                    9.38       NaN
 8 Africa    Cameroon                  7.25         1.36
 9 Africa    Burundi                   6.67       NaN
10 Africa    Liberia                   6.33       NaN
```

And then print those countries with the highest gaps for internet.

```
differences[order(-differences$mean.net.diff)[1:10], ]
```

```
# A tibble: 10 x 4
# Groups:   spregion [3]
   spregion  country                 mean.cell.diff mean.net.diff
   <fct>     <fct>                            <dbl>         <dbl>
 1 LAm.Carib Costa Rica                       0.182         18.5
 2 LAm.Carib Brazil                           4.15          15.0
 3 Africa    Angola                          23.6           10.7
 4 Africa    Lesotho                          3             10.1
 5 Asia      Palestinian Territories          0              9.55
 6 LAm.Carib Mexico                          10.4            9.53
 7 Africa    Ghana                            1.7            8.8
 8 LAm.Carib Panama                           3.17           8.19
 9 LAm.Carib Ecuador                          0.974          7.47
10 LAm.Carib Peru                             4.28           6.72
```

## 7.4 Data coverage

Here we quickly estimate the coverage of the access data underlying Figure 2 by the number of farmers globally. We draw from Lowder's compilation of farming households globally [5]. We note that these data, do not cover all countries, and vary based on time span, but do give us a rough idea of how many farming households our data set is likely representing for these three regions globally.

```
Lowd <- readxl::read_excel("Data/National/Nfarms/1-s2.0-S0305750X15002703-mmc1 (1).xlsx",
    sheet = "WEB APPENDIX table 1", skip = 2)
Lowd$country2 <- countrycode::countrycode(Lowd$Country, "country.name",
    "country.name")
Lowd$spregion <- getspregion(Lowd$country2, "country.name")
Lowd <- subset(Lowd, spregion %in% c("Asia", "Africa", "LAm.Carib"))
Lowd <- Lowd %>% select(country2, spregion, `Total Number of Holdings`) %>%
    as.data.frame()
Lowd$`Total Number of Holdings` <- as.numeric(Lowd$`Total Number of Holdings`)
Lowd.sub <- subset(Lowd, country2 %in% all.dat.f2$country)
all <- Lowd %>% group_by(spregion) %>% summarize(farmHH = sum(`Total Number of Holdings`,
    na.rm = T))
sub <- Lowd.sub %>% group_by(spregion) %>% summarize(farmHH = sum(`Total Number of Holdings`,
    na.rm = T))
all$perc <- sub$farmHH/all$farmHH
all
```

```
# A tibble: 3 x 3
```

```
  spregion     farmHH  perc
  <chr>         <dbl> <dbl>
1 Africa     59063730 0.753
2 Asia      443103890 0.456
3 LAm.Carib  21081218 0.883
```

# 8 Main Text Figure 3

Here we dis-aggregate the cost of accessing mobile data for 10 income groups for 83 countries in Africa, Asia, and Latin America and the Caribbean and show that for the poorest 10% of the population the cost of fully engaging in the digital economy is still prohibitively high. We utilize the input data processed Supplementary Information E.

## 8.1 Data preparation

First we read in the processed data here. We then add a common supra-national region variable used in previous figures, and remove countries outside regions of interest.

```
cost <- read.csv("../SI_E/Data/Data_Affordability_decile_1_2015_2019_20200714.csv")
cost$all <- ifelse(!is.na(cost$Afford_decile_1_2015) & !is.na(cost$Afford_decile_1_2016) &
    !is.na(cost$Afford_decile_1_2017) & !is.na(cost$Afford_decile_1_2018),
    T, F)
cost$spregion <- getspregion(cost$Country, "country.name")
cost <- subset(cost, spregion %in% c("LAm.Carib", "Asia", "Africa"))
```

To account for differences in sampling effort, we estimate the difference (a bias estimate) between the median regional cost of data in 2018 (the year with most complete coverage) and the median regional cost in 2018 when coverage equaled that of the years between 2015 and 2018. We then bias corrected the time series by adjusting the median costs for years between 2015 and 2018 to known differences observed in 2018.

We note that the A4AI also published Q2 data for costs for 2019, but there is no income data to accompany these new estimates for, and so here only work with 2018 Q4.

```
cost$bias.2015<-(ifelse(is.na(cost$Afford_decile_1_2015), NA, cost$Afford_decile_1_2018))
cost$bias.2016<-(ifelse(is.na(cost$Afford_decile_1_2016), NA, cost$Afford_decile_1_2018))
cost$bias.2017<-(ifelse(is.na(cost$Afford_decile_1_2017), NA, cost$Afford_decile_1_2018))

cost.adj<-cost %>% group_by(spregion)%>%

  summarize(
    ##get expected estimates for 2018 given different country coverages
            median2018=median(Afford_decile_1_2018, na.rm=T),
            median2015.adj=median(bias.2015, na.rm=T),
            median2016.adj=median(bias.2016, na.rm=T),
            median2017.adj=median(bias.2017, na.rm=T),

    ##get observed estimates for each year
            median2015=median(Afford_decile_1_2015, na.rm=T),
            median2016=median(Afford_decile_1_2016, na.rm=T),
            median2017=median(Afford_decile_1_2017, na.rm=T),

    ##correct the observed estimates in 2015-2017 based on the bias
            median2015.cor=median2015+(median2018-median2015.adj),
            median2016.cor=median2016+ (median2018-median2016.adj),
            median2017.cor=median2017+ (median2018-median2017.adj),
            median2018.cor=median2018,

            )  %>%

  tidyr::gather(class, access, median2015.cor:median2018.cor) %>%
  mutate(year=c(rep(2015, 3), rep(2016, 3), rep(2017,3), rep(2018,3)))
```

## 8.2 Figure 3

Now we recreate figure 3 in the manuscript.

```
layout(matrix(1:2, ncol = 2, byrow = T))
cost$label <- ifelse(cost$Country %in% c("Honduras", "Central African Republic",
    "Chad", "Congo - Kinshasa", "Nepal", "Guinea-Bissau"), as.character(cost$iso3c),
    NA)
ggplot(cost, aes(spregion, Afford_decile_1_2018 * 100, colour = spregion,
    label = label)) + geom_jitter() + geom_boxplot(outlier.shape = NA) +
    scale_colour_manual(values = c("#d95f02ff", "#7570b3ff",
        "#1b9e77ff")) + theme_bw() + ggrepel::geom_text_repel() +
    theme(plot.background = element_blank(), panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(), strip.background = element_rect(colour = "white",
            fill = "white")) + ylab("Cost of 1GB dara (% of income)") +
    ylim(0, 240) + xlab("") + theme(legend.position = "none")

ggplot(cost.adj, aes(year, access * 100, colour = spregion)) +
    geom_line() + geom_point() + scale_colour_manual(values = c("#d95f02ff",
    "#7570b3ff", "#1b9e77ff")) + theme_bw() + theme(plot.background = element_blank(),
    panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
    strip.background = element_rect(colour = "white", fill = "white")) +
    theme(legend.position = "none") + ylab("") + xlab("")
```

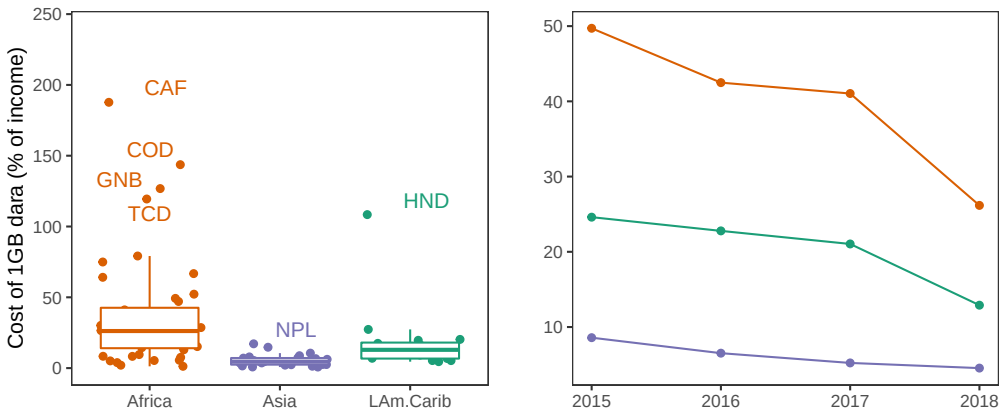


Figure 9: Mobile broadband affordability

## 8.3 Insights

Then we obtain some complementary insights for the affordability statistics.First we check the total number of countries used in this analysis.

```
sum(!is.na(cost$Afford_decile_1_2018))
```

```
[1] 83
```

Second we identify the countries with the lowest affordability metrics.

```
cost2018 <- cost %>% select(Afford_decile_1_2018, spregion, Country) %>%
    na.omit()

cost2018[order(-cost2018$Afford_decile_1_2018)[1:10], ]
```

```
   Afford_decile_1_2018  spregion                 Country
16                 1.88    Africa Central African Republic
22                 1.44    Africa         Congo - Kinshasa
39                 1.27    Africa            Guinea-Bissau
17                 1.19    Africa                     Chad
42                 1.08 LAm.Carib                 Honduras
78                 0.79    Africa             Sierra Leone
90                 0.75    Africa                     Togo
56                 0.67    Africa               Madagascar
57                 0.64    Africa                   Malawi
7                  0.52    Africa                    Benin
```

Finally we reproduce figure 3b in table form.

```
cost.adj %>% select(access, spregion, year) %>% mutate(access = access *
    100)
```

```
# A tibble: 12 x 3
   access spregion   year
    <dbl> <chr>     <dbl>
 1  49.7  Africa     2015
 2   8.59 Asia       2015
 3  24.6  LAm.Carib  2015
 4  42.5  Africa     2016
 5   6.52 Asia       2016
 6  22.8  LAm.Carib  2016
 7  41.0  Africa     2017
 8   5.23 Asia       2017
 9  21.0  LAm.Carib  2017
10  26.2  Africa     2018
11   4.54 Asia       2018
12  12.9  LAm.Carib  2018
```

# 9 Appendix

Below we plot the full coverage of network data. First we check the crs, extent and resolution.

```
# tech.raw<-brick(twg,thg,fourg)
tech.raw <- brick(twg, thg, fourg)
names(tech.raw) <- c("2G", "3G", "4G")
tech.raw[tech.raw > 0] <- 1
tech.raw
```

```
class      : RasterBrick
dimensions : 2160, 4320, 9331200, 3  (nrow, ncol, ncell, nlayers)
resolution : 0.083, 0.083  (x, y)
extent     : -180, 180, -90, 90  (xmin, xmax, ymin, ymax)
crs        : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0
source     : memory
names      : X2G, X3G, X4G
min values :   1,   1,   1
max values :   1,   1,   1
```

Then we plot out the global coverage maps.

```
par(mfrow = c(3, 1), mai = c(0, 0.01, 0.01, 0.01))
plot(tech.raw$X2G, col = cols[1], legend = F, colNA = "black",
    axes = FALSE, box = FALSE)
plot(tech.raw$X3G, col = cols[2], legend = F, colNA = "black",
    axes = FALSE, box = FALSE)
plot(tech.raw$X4G, col = cols[3], legend = F, colNA = "black",
    axes = FALSE, box = FALSE)
```

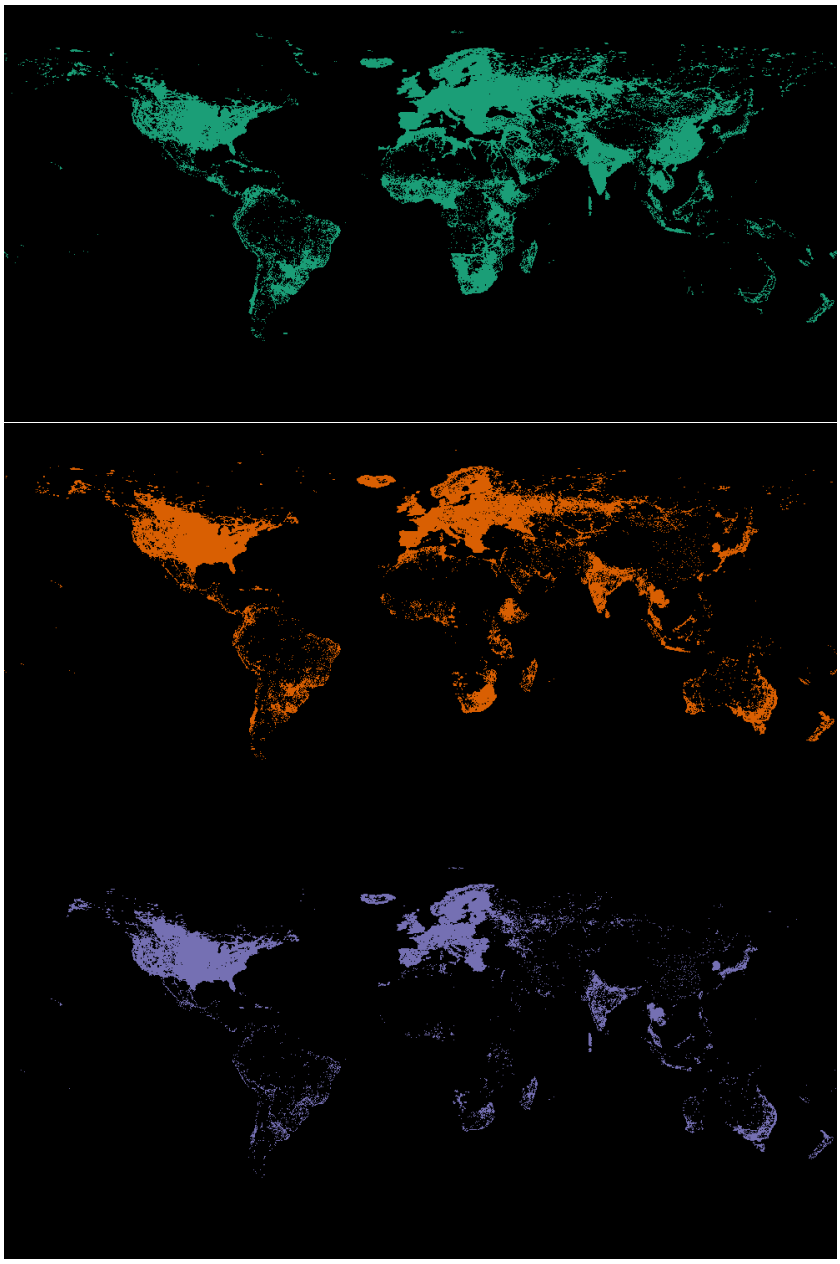

Figure 10: Global mobile network coverage. Green= 2G, Orange= 3G, Purple=4G. Source: Mosaik LLC

## Session information

```
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] rworldmap_1.3-6  scam_1.2-5       mgcv_1.8-28      nlme_3.1-140
 [5] magick_2.4.0     gridExtra_2.3    cowplot_1.0.0    ggridges_0.5.1
 [9] ggplot2_3.3.2    dplyr_1.0.2      rgdal_1.4-8      maptools_0.9-9
[13] plyr_1.8.4       raster_3.1-5     sp_1.4-1         ncdf4_1.17
[17] checkpoint_0.4.7 knitr_1.26

loaded via a namespace (and not attached):
 [1] spam_2.5-1       tidyselect_1.1.0 xfun_0.11        purrr_0.3.3
 [5] splines_3.6.1    lattice_0.20-41  colorspace_1.4-1 vctrs_0.3.4
 [9] generics_0.0.2   utf8_1.1.4       rlang_0.4.7      pillar_1.4.6
[13] foreign_0.8-71   glue_1.4.2       withr_2.1.2      lifecycle_0.2.0
[17] stringr_1.4.0    fields_10.3      dotCall64_1.0-0  munsell_0.5.0
[21] gtable_0.3.0     codetools_0.2-16 evaluate_0.14    labeling_0.3
[25] fansi_0.4.0      highr_0.8        Rcpp_1.0.4       scales_1.1.0
[29] formatR_1.7      farver_2.0.1     digest_0.6.23    stringi_1.4.3
[33] ggrepel_0.8.2    grid_3.6.1       cli_2.0.0        tools_3.6.1
[37] magrittr_1.5     maps_3.3.0       tibble_3.0.3     crayon_1.3.4
[41] pkgconfig_2.0.3  ellipsis_0.3.0   Matrix_1.2-17    assertthat_0.2.1
[45] rstudioapi_0.10  R6_2.4.1         compiler_3.6.1
```

# References


[1] Roger Bivand, Tim Keitt, and Barry Rowlingson. *rgdal: Bindings for the 'Geospatial' Data Abstraction Library*. R package version 1.4-8. 2019. URL: https://CRAN.R-project.org/package=rgdal.

[2] Roger Bivand and Nicholas Lewin-Koh. *maptools: Tools for Handling Spatial Objects*. R package version 0.9-9. 2019. URL: https://CRAN.R-project.org/package=maptools.

[3] John Fox, Sanford Weisberg, and Brad Price. *car: Companion to Applied Regression*. R package version 3.0-5. 2019. URL: https://CRAN.R-project.org/package=car.

[4] Robert J. Hijmans. *raster: Geographic Data Analysis and Modeling*. R package version 3.3-13. 2020. URL: https://CRAN.R-project.org/package=raster.

[5] Sarah K. Lowder, Jakob Skoet, and Terri Raney. "The Number, Size, and Distribution of Farms, Smallholder Farms, and Family Farms Worldwide". In: *World Development* 87.C (2016), pp. 16–29. URL: https://EconPapers.repec.org/RePEc:eee:wdevel:v:87:y:2016:i:c:p:16-29.

[6] Hong Ooi. *checkpoint: Install Packages from Snapshots on the Checkpoint Server for Reproducibility*. R package version 0.4.7. 2019. URL: https://CRAN.R-project.org/package=checkpoint.

[7] Edzer Pebesma and Roger Bivand. *sp: Classes and Methods for Spatial Data*. R package version 1.4-1. 2020. URL: https://CRAN.R-project.org/package=sp.

[8] Natalya Pya. *scam: Shape Constrained Additive Models*. R package version 1.2-5. 2019. URL: https://CRAN.R-project.org/package=scam.

[9] Andy South. *rworldmap: Mapping Global Data*. R package version 1.3-6. 2016. URL: https://CRAN.R-project.org/package=rworldmap.

[10] Hadley Wickham et al. *dplyr: A Grammar of Data Manipulation*. R package version 1.0.2. 2020. URL: https://CRAN.R-project.org/package=dplyr.

[11] Hadley Wickham et al. *ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics*. R package version 3.3.2. 2020. URL: https://CRAN.R-project.org/package=ggplot2.

[12] Claus O. Wilke. *ggridges: Ridgeline Plots in 'ggplot2'*. R package version 0.5.1. 2018. URL: https://CRAN.R-project.org/package=ggridges.

[13] Simon Wood. *mgcv: Mixed GAM Computation Vehicle with Automatic Smoothness Estimation*. R package version 1.8-28. 2019. URL: https://CRAN.R-project.org/package=mgcv.

[14] Yihui Xie. *knitr: A General-Purpose Package for Dynamic Report Generation in R*. R package version 1.26. 2019. URL: https://CRAN.R-project.org/package=knitr.