REPORT-THE AGGREGATORS

```
library(readr)
data <- read_csv("country_vaccinations.csv")
```

```
## Rows: 86512 Columns: 15
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): country, iso_code, vaccines, source_name, source_website
## dbl  (9): total_vaccinations, people_vaccinated, people_fully_vaccinated, da...
## date (1): date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
country_vaccinations_by_manufacturer <- read_csv("country_vaccinations_by_manufacturer.csv")
```

```
## Rows: 35623 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (2): location, vaccine
## dbl  (1): total_vaccinations
## date (1): date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
other <- read_csv("countries of the world.csv")
```

```
## Rows: 227 Columns: 20
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (11): Country, Region, PopDensity, Coastline, Net migration, Phones, Ara...
## dbl  (3): Population, Area, GDP
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
options(warn = -1)
options(scipen = 10000)
options(repr.plot.width = 13.8, repr.plot.height = 9.2)

library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
```

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

```
library(scales)
```

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

```
library(RColorBrewer)
library(ggthemes)
library(lubridate)
```

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

```
library(ggrepel)
library(reshape)
```

```
##
## Attaching package: 'reshape'
##
## The following object is masked from 'package:lubridate':
##
##     stamp
##
## The following object is masked from 'package:dplyr':
##
##     rename
##
## The following objects are masked from 'package:tidyr':
##
##     expand, smiths
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(maps)
```

```
##
## Attaching package: 'maps'
##
## The following object is masked from 'package:purrr':
##
##     map
```

```
library(stringr)
library(ggcorrplot)
library(viridis)
```

```
## Loading required package: viridisLite
##
## Attaching package: 'viridis'
##
## The following object is masked from 'package:maps':
##
##     unemp
##
## The following object is masked from 'package:scales':
##
##     viridis_pal
```

```
annotate <- ggplot2::annotate

theme_michau <- theme(legend.position = "bottom", legend.direction = "horizontal", axis.text = element_
plot.caption = element_text(color = "gray65", face = "bold", size = 10), legend.text = element_text(size
axis.title = element_text(size = 15.9, face = "bold", color = "gray25"), legend.title = element_text(si
axis.line = element_line(size = 0.4), plot.title = element_text(size = 19.5), plot.subtitle = element_te
strip.text = element_text(size = 14.4, face = "bold"))
```

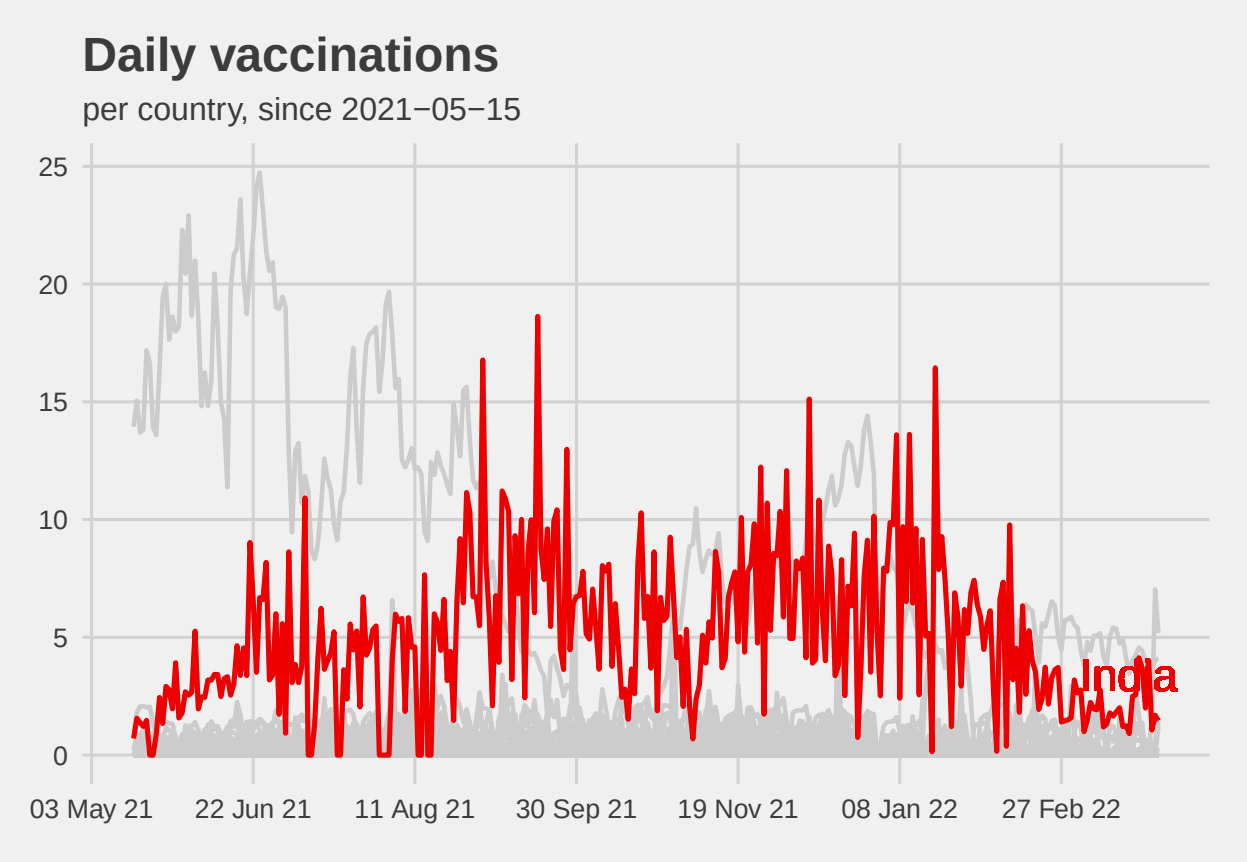
```
data <- data[,c("country", "total_vaccinations", "date", "people_vaccinated", "daily_vaccinations_raw",
                "people_vaccinated_per_hundred", "daily_vaccinations_per_million", "vaccines")]

data$date <- as.Date(data$date)

data$total_vaccinations[is.na(data$total_vaccinations)==T] <- 0
data$people_vaccinated[is.na(data$people_vaccinated)==T] <- 0
data$daily_vaccinations_raw[is.na(data$daily_vaccinations_raw)==T] <- 0
data$people_vaccinated_per_hundred[is.na(data$people_vaccinated_per_hundred)==T] <- 0
```

```r
data$daily_vaccinations_per_million[is.na(data$daily_vaccinations_per_million)==T] <- 0

head <- data[sample(1:nrow(data),5), ]
head[order(head$date),]
```

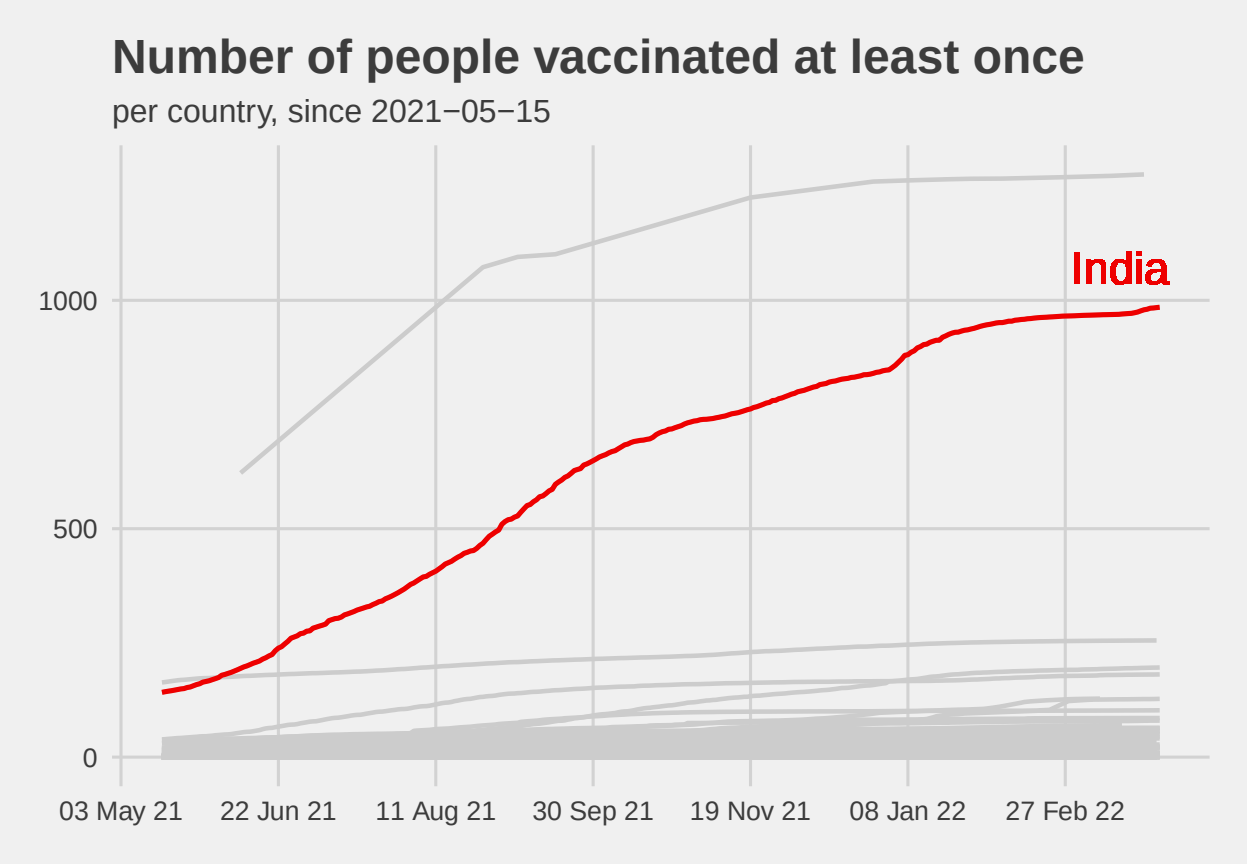
```
## # A tibble: 5 x 8
##   country             total~1 date       peopl~2 daily~3 peopl~4 daily~5 vacci~6
##   <chr>                 <dbl> <date>       <dbl>   <dbl>   <dbl>   <dbl> <chr>
## 1 Costa Rica                0 2021-01-02       0       0    0         47 Oxford~
## 2 Estonia               25896 2021-01-24   23482     161    1.77     838 Johnso~
## 3 Mauritius           1875132 2021-11-23  916524       0   72.0    10080 Covaxi~
## 4 Turks and Caicos I~       0 2021-12-19       0       0    0       3110 Pfizer~
## 5 Sao Tome and Princ~       0 2022-01-11       0       0    0       5708 Oxford~
## # ... with abbreviated variable names 1: total_vaccinations,
## #   2: people_vaccinated, 3: daily_vaccinations_raw,
## #   4: people_vaccinated_per_hundred, 5: daily_vaccinations_per_million,
## #   6: vaccines
```

```r
data$month <- month(data$date)
data$weekday <- weekdays(data$date)
data$percent_people <- data$people_vaccinated_per_hundred/100
```

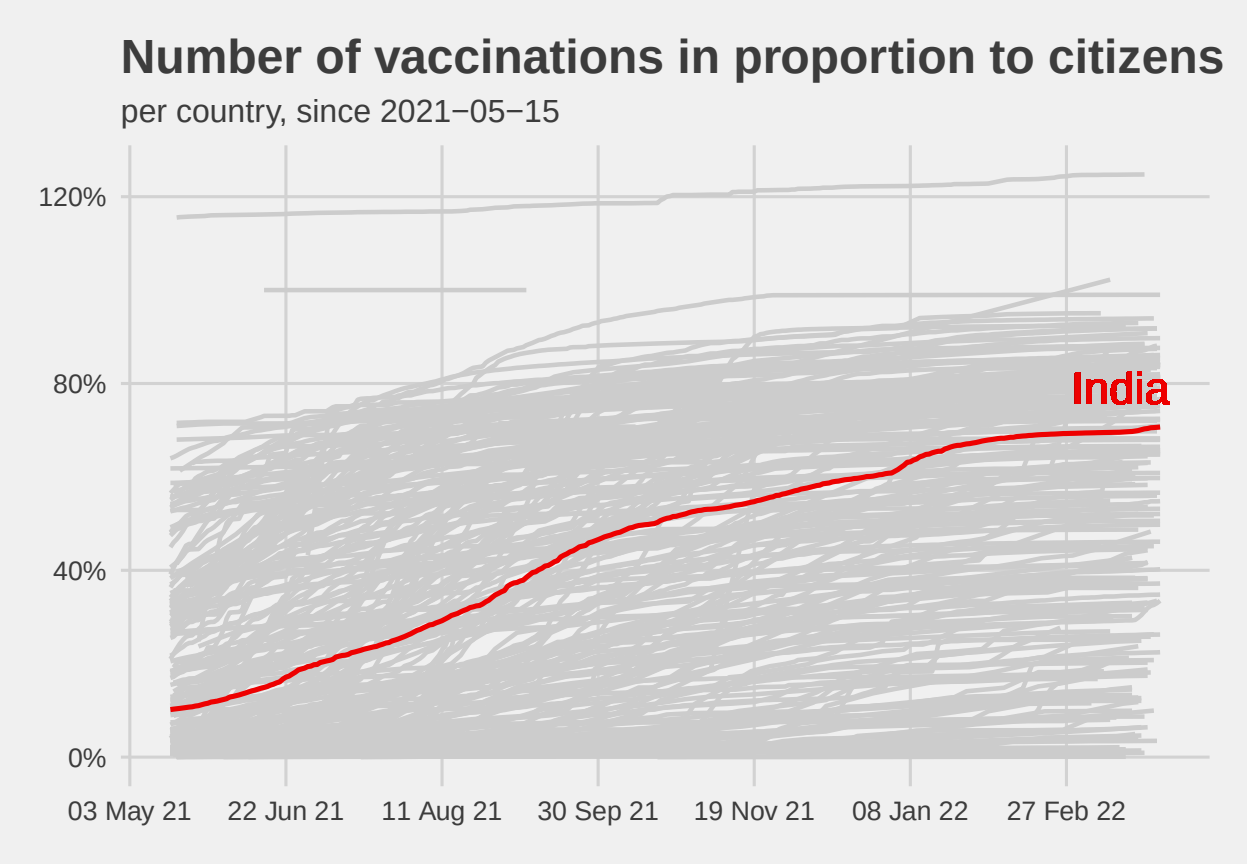
```r
full1 <- data %>%
  group_by(date, country) %>%
  filter(as.Date(date) > "2021-05-15")

ind1 <- data %>%
  group_by(date, country) %>%
  filter(country == "India") %>%
  filter(as.Date(date) > "2021-05-15")

full1$daily_vaccinations_raw <- full1$daily_vaccinations_raw/1000000
ind1$daily_vaccinations_raw <- ind1$daily_vaccinations_raw/1000000

ggplot()+
  geom_line(data = full1, aes(date, daily_vaccinations_raw, group = country), size = 0.8, colour = "gray
  geom_line(data = ind1, aes(date, daily_vaccinations_raw), size = 0.9, colour = "red2")+
  geom_text(data = ind1, aes(x = max(ind1$date), y = ind1$daily_vaccinations_raw[ind1$date==max(ind1$da
                             label = country), hjust = 0.8, vjust = -0.9, size = 5.9, color = "red2")+
  scale_x_date(date_labels = "%d %b %y", date_breaks = "50 days")+
  labs(x = "Date", y = "Vaccinations (in milions)", title = "Daily vaccinations", subtitle = "per counti
  theme_fivethirtyeight()
```




```
full2 <- data %>%
  group_by(date, country) %>%
  filter(people_vaccinated>0) %>%
  filter(as.Date(date) > "2021-05-15")

ind2 <- data %>%
  group_by(date, country) %>%
  filter(country == "India") %>%
  filter(people_vaccinated>0) %>%
  filter(as.Date(date) > "2021-05-15")

full2$people_vaccinated <- full2$people_vaccinated/1000000
ind2$people_vaccinated <- ind2$people_vaccinated/1000000

ggplot()+
  geom_line(data = full2, aes(date, people_vaccinated, group = country), size = 0.8, colour = "gray80")+
  geom_line(data = ind2, aes(date, people_vaccinated), size = 0.9, colour = "red2")+
  geom_text(data = ind2, aes(x = max(ind2$date), y = ind2$people_vaccinated[ind2$date==max(ind2$date)],
                             label = country), hjust = 0.9, vjust = -0.7, size = 5.9, color = "red2")+
  scale_x_date(date_labels = "%d %b %y", date_breaks = "50 days")+
  labs(x = "Date", y = "Vaccinations (cumulated, in milions)", title = "Number of people vaccinated at
       subtitle = "per country, since 2021-05-15")+
  theme_fivethirtyeight()
```

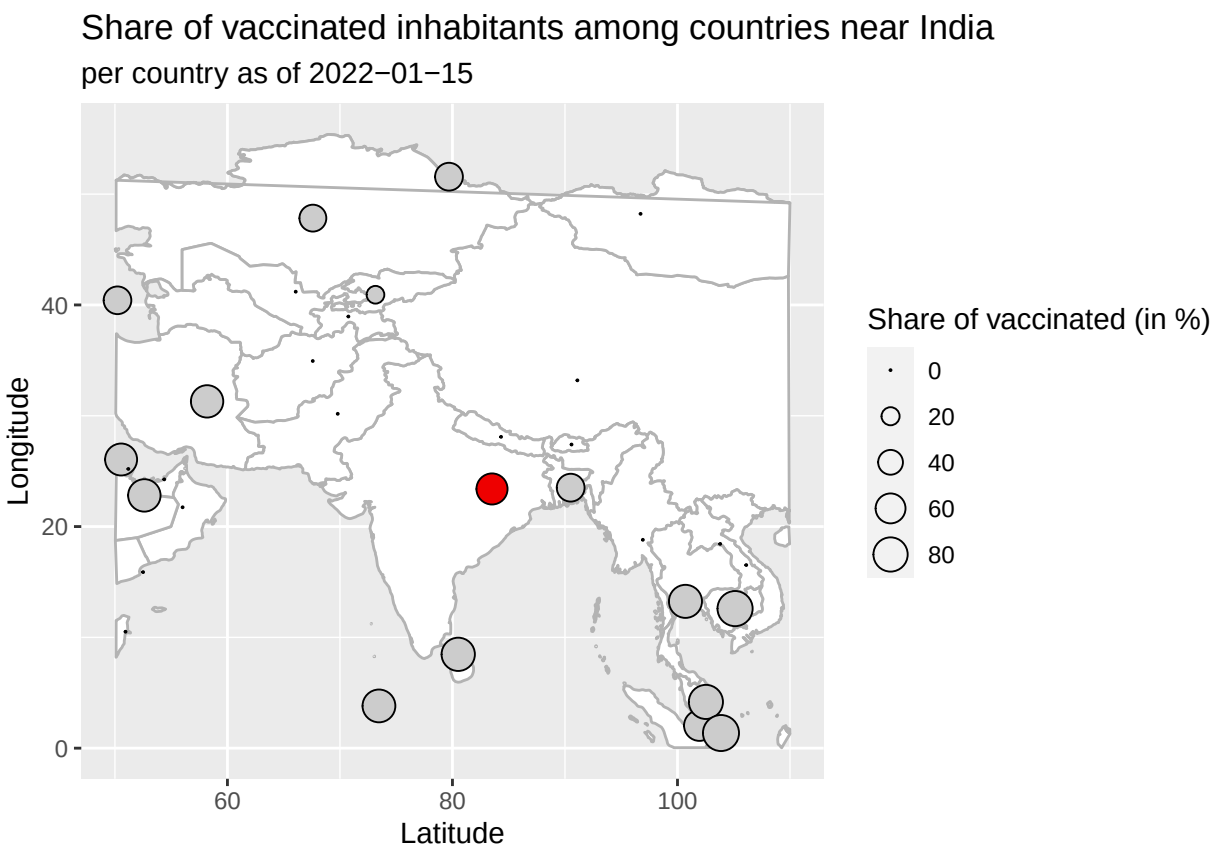



```
ggplot()+
  geom_line(data = full2, aes(date, percent_people, group = country), size = 0.8, colour = "gray80")+
  geom_line(data = ind2, aes(date, percent_people), size = 0.9, colour = "red2")+
  geom_text(data = ind2, aes(x = max(ind2$date), y = ind2$percent_people[ind2$date==max(ind2$date)],
                             label = country), hjust = 0.9, vjust = -0.7, size = 5.9, color = "red2")+
  scale_x_date(date_labels = "%d %b %y", date_breaks = "50 days")+
  scale_y_continuous(labels = scales::percent_format(accuracy = 1))+
  labs(x = "Date", y = "Vaccinations (%)",
       title = "Number of vaccinations in proportion to citizens",
       subtitle = "per country, since 2021-05-15")+
  theme_fivethirtyeight()
```




```
world_map <- map_data("world") %>%
  filter(between(lat, 0,60)) %>%
  filter(between(long, 50, 110))

country <- data %>%
  filter(date == "2022-01-15")%>%
  group_by(country) %>%
  select(country, people_vaccinated_per_hundred)

names(country)[1] <- "region"

world_map <- world_map %>%
  left_join(country, by = "region")

world_map2 <- world_map %>%
  select(long, lat, region, people_vaccinated_per_hundred) %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat), people_vaccinated_per_hundred = mean(people_vaccinated_
            .groups = 'drop')

world_map2$india <- ifelse(world_map2$region=="India","red2","gray80")

ggplot()+
  geom_polygon(data = world_map, aes(x = long, y = lat, group = group), fill = "white", colour = "gray7
  geom_point(data = world_map2, aes(x = long, y = lat, size = people_vaccinated_per_hundred),
             shape = 21, fill = world_map2$india)+
```

```
  scale_size(range = c(0, 6))+
  scale_fill_manual(values = c("red2", "gray80"))+
  labs(x = "Latitude", y = "Longitude",
       title = "Share of vaccinated inhabitants among countries near India",
       size = "Share of vaccinated (in %)", subtitle = "per country as of 2022-01-15")
```




```
  theme_fivethirtyeight()+
  theme(legend.position = "right", legend.direction = "vertical")
```

```
## List of 93
##  $ line                      :List of 6
##   ..$ colour       : chr "black"
##   ..$ size         : num 0.545
##   ..$ linetype     : num 1
##   ..$ lineend      : chr "butt"
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi FALSE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ rect                      :List of 5
##   ..$ fill         : Named chr "#F0F0F0"
##   .. ..- attr(*, "names")= chr "Light Gray"
##   ..$ colour       : logi NA
##   ..$ size         : num 0.545
##   ..$ linetype     : num 0
##   ..$ inherit.blank: logi FALSE
```

```
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ text                        :List of 11
##   ..$ family       : chr "sans"
##   ..$ face         : chr "plain"
##   ..$ colour       : Named chr "#3C3C3C"
##   .. ..- attr(*, "names")= chr "Dark Gray"
##   ..$ size         : num 12
##   ..$ hjust        : num 0.5
##   ..$ vjust        : num 0.5
##   ..$ angle        : num 0
##   ..$ lineheight   : num 0.9
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : logi FALSE
##   ..$ inherit.blank: logi FALSE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ title                       : NULL
##  $ aspect.ratio                : NULL
##  $ axis.title                  : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ axis.title.x                :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 3points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x.top            :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 3points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x.bottom         : NULL
##  $ axis.title.y                :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
```

```
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : num 90
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 3points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.y.left         : NULL
##  $ axis.title.y.right        :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : num -90
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 3points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text                 :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi FALSE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x               :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 2.4points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x.top           :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
```

```
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 2.4points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x.bottom        : NULL
##  $ axis.text.y               :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 1
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 2.4points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.y.left          : NULL
##  $ axis.text.y.right         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 2.4points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.ticks                : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ axis.ticks.x              : NULL
##  $ axis.ticks.x.top          : NULL
##  $ axis.ticks.x.bottom       : NULL
##  $ axis.ticks.y              : NULL
##  $ axis.ticks.y.left         : NULL
##  $ axis.ticks.y.right        : NULL
##  $ axis.ticks.length         : 'simpleUnit' num 3points
##   ..- attr(*, "unit")= int 8
##  $ axis.ticks.length.x       : NULL
##  $ axis.ticks.length.x.top   : NULL
##  $ axis.ticks.length.x.bottom: NULL
```

```
##  $ axis.ticks.length.y       : NULL
##  $ axis.ticks.length.y.left  : NULL
##  $ axis.ticks.length.y.right : NULL
##  $ axis.line                 : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ axis.line.x               : NULL
##  $ axis.line.x.top           : NULL
##  $ axis.line.x.bottom        : NULL
##  $ axis.line.y               : NULL
##  $ axis.line.y.left          : NULL
##  $ axis.line.y.right         : NULL
##  $ legend.background         :List of 5
##   ..$ fill         : NULL
##   ..$ colour       : logi NA
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi FALSE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ legend.margin             : 'margin' num [1:4] 6points 6points 6points 6points
##   ..- attr(*, "unit")= int 8
##  $ legend.spacing            : 'simpleUnit' num 12points
##   ..- attr(*, "unit")= int 8
##  $ legend.spacing.x          : NULL
##  $ legend.spacing.y          : NULL
##  $ legend.key                :List of 5
##   ..$ fill         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ legend.key.size           : 'simpleUnit' num 1.2lines
##   ..- attr(*, "unit")= int 3
##  $ legend.key.height         : NULL
##  $ legend.key.width          : NULL
##  $ legend.text               :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ legend.text.align         : NULL
##  $ legend.title              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
```

```
##   ..$ hjust        : num 0
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ legend.title.align        : NULL
##  $ legend.position           : chr "right"
##  $ legend.direction          : chr "vertical"
##  $ legend.justification      : chr "center"
##  $ legend.box                : chr "vertical"
##  $ legend.box.just           : NULL
##  $ legend.box.margin         : 'margin' num [1:4] 0cm 0cm 0cm 0cm
##   ..- attr(*, "unit")= int 1
##  $ legend.box.background     : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ legend.box.spacing        : 'simpleUnit' num 12points
##   ..- attr(*, "unit")= int 8
##  $ panel.background          :List of 5
##   ..$ fill         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ panel.border              :List of 5
##   ..$ fill         : logi NA
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi FALSE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ panel.spacing             : 'simpleUnit' num 6points
##   ..- attr(*, "unit")= int 8
##  $ panel.spacing.x           : NULL
##  $ panel.spacing.y           : NULL
##  $ panel.grid                :List of 6
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi FALSE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ panel.grid.major          :List of 6
##   ..$ colour       : Named chr "#D2D2D2"
##   .. ..- attr(*, "names")= chr "Medium Gray"
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ lineend      : NULL
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi FALSE
```

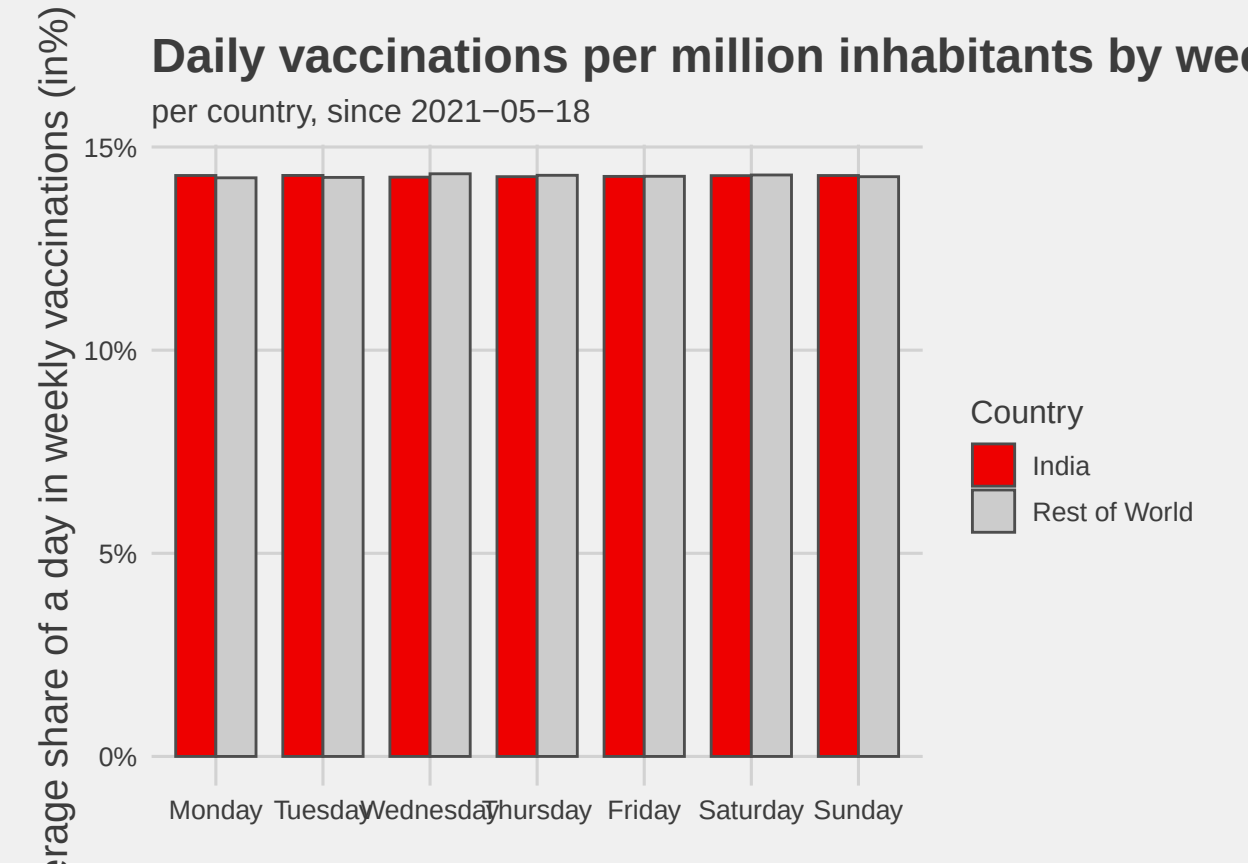
```
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ panel.grid.minor          : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ panel.grid.major.x        : NULL
##  $ panel.grid.major.y        : NULL
##  $ panel.grid.minor.x        : NULL
##  $ panel.grid.minor.y        : NULL
##  $ panel.ontop               : logi FALSE
##  $ plot.background           :List of 5
##   ..$ fill         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ plot.title                :List of 11
##   ..$ family       : NULL
##   ..$ face         : chr "bold"
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 1.5
##   ..$ hjust        : num 0
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 6points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi FALSE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.title.position       : chr "panel"
##  $ plot.subtitle             :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : num 0
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 6points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.caption              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : num 1
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 6points 0points 0points 0points
```

```
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.caption.position     : chr "panel"
##  $ plot.tag                  :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 1.2
##   ..$ hjust        : num 0.5
##   ..$ vjust        : num 0.5
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ plot.tag.position         : chr "topleft"
##  $ plot.margin               : 'simpleUnit' num [1:4] 1lines 1lines 1lines 1lines
##   ..- attr(*, "unit")= int 3
##  $ strip.background          :List of 5
##   ..$ fill         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi FALSE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ strip.background.x        : NULL
##  $ strip.background.y        : NULL
##  $ strip.placement           : chr "inside"
##  $ strip.text                :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : 'rel' num 0.8
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 4.8points 4.8points 4.8points 4.8points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ strip.text.x              : NULL
##  $ strip.text.y              :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : num -90
```

```
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ strip.switch.pad.grid     : 'simpleUnit' num 3points
##   ..- attr(*, "unit")= int 8
##  $ strip.switch.pad.wrap     : 'simpleUnit' num 3points
##   ..- attr(*, "unit")= int 8
##  $ strip.text.y.left         :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : NULL
##   ..$ angle        : num 90
##   ..$ lineheight   : NULL
##   ..$ margin       : NULL
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  - attr(*, "class")= chr [1:2] "theme" "gg"
##  - attr(*, "complete")= logi TRUE
##  - attr(*, "validate")= logi TRUE
```

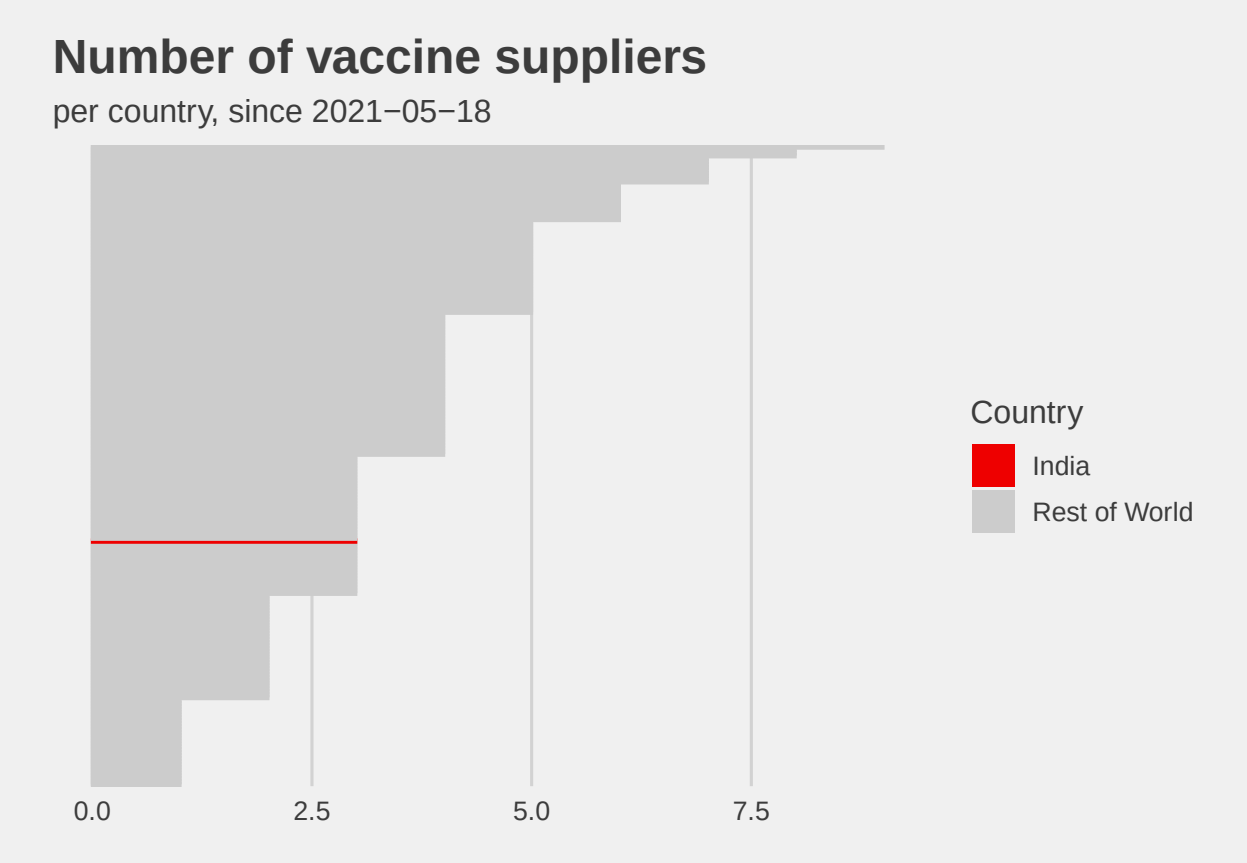
```
full3 <- data %>%
  filter(as.Date(date) > "2021-05-18") %>%
  select(weekday, country, daily_vaccinations_per_million) %>%
  group_by(weekday, country) %>%
  summarise(nd = mean(daily_vaccinations_per_million), .groups = 'drop')

full3$india <- ifelse(full3$country=="India","India", "Rest of World")

full3ind <- full3 %>%
  filter(india=="India")

full3row <- full3 %>%
  filter(india=="Rest of World")

full3row <- full3row %>%
  group_by(weekday) %>%
  summarise(nd = mean(nd), .groups = 'drop')

full3row$country <- "Rest of World"
full3ind <- full3ind[,1:3]
full3row <- full3row[,c(1,3,2)]

full3ind <- as.data.frame(full3ind)
full3ind$nd <- full3ind$nd/sum(full3ind$nd)

full3row <- as.data.frame(full3row)
full3row$nd <- full3row$nd/sum(full3row$nd)
```

```r
full3 <- rbind(full3ind, full3row)

ggplot(full3, aes(weekday, nd, fill = country))+
  geom_bar(stat = "identity",  width = 0.75, position=position_dodge(), col = "gray30")+
  scale_x_discrete(limit = c("Monday","Tuesday","Wednesday","Thursday","Friday","Saturday","Sunday"))+
  scale_fill_manual(values = c("red2","gray80"))+
  scale_y_continuous(labels = scales::percent_format(accuracy = 1))+
  labs(x = "Weekday", y = "Average share of a day in weekly vaccinations (in%)",
       title = "Daily vaccinations per million inhabitants by weekday",
       fill = "Country", subtitle = "per country, since 2021-05-18")+
  theme_fivethirtyeight()+
  theme(legend.position = "right", legend.direction = "vertical", axis.title.y = element_text(size = 15
```




```r
data$number_of_vac <- str_count(data$vaccines, ",")+1

vacx <- data %>%
  filter(as.Date(date) > "2021-05-18") %>%
  group_by(country, number_of_vac) %>%
  summarise(avg = mean(number_of_vac), .groups = 'drop') %>%
  filter(avg>0)

vacx$india <- ifelse(vacx$country=="India", "India", "Rest of World")

ggplot(vacx, aes(reorder(country, +avg), avg, fill = india, colour = india))+
  geom_bar(stat = "identity",  width = 0.9)+
```

```
  scale_fill_manual(values = c("red2", "gray80"))+
  scale_colour_manual(values = c("red2", "gray80"), guide = "none")+
  coord_flip()+
  labs(y = "Number of suppliers", x = "Country", title = "Number of vaccine suppliers", fill = "Country
       subtitle = "per country, since 2021-05-18")+
  theme_fivethirtyeight()+
  theme(legend.position = "right", legend.direction = "vertical", axis.text.y = element_text(size = 0),
        panel.grid.major.y = element_blank())
```

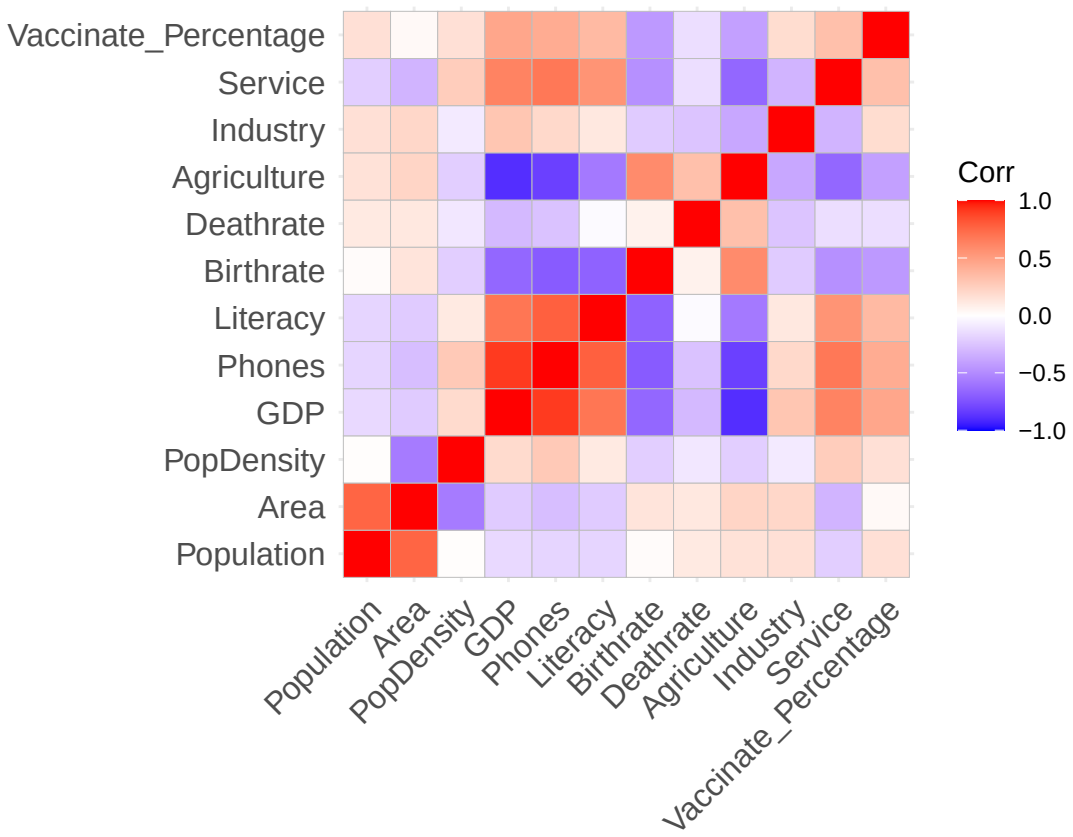



```
other <- other[,c("Country", "Population", "Area", "PopDensity", "GDP",
                  "Phones", "Literacy", "Birthrate", "Deathrate", "Agriculture", "Industry", "Service")]

names(other) <- c("region", "Population", "Area", "PopDensity", "GDP", "Phones", "Literacy", "Birthrate
                  "Deathrate", "Agriculture", "Industry", "Service")

other$region <- gsub(" ", "", other$region)
other$PopDensity <- as.numeric(gsub(",", ".", other$PopDensity))
other$Phones <- as.numeric(gsub(",", ".", other$Phones))
other$Literacy <- as.numeric(gsub(",", ".", other$Literacy))
other$Birthrate <- as.numeric(gsub(",", ".", other$Birthrate))
other$Deathrate <- as.numeric(gsub(",", ".", other$Deathrate))
other$Agriculture <- as.numeric(gsub(",", ".", other$Agriculture))
other$Industry <- as.numeric(gsub(",", ".", other$Industry))
other$Service <- as.numeric(gsub(",", ".", other$Service))
```

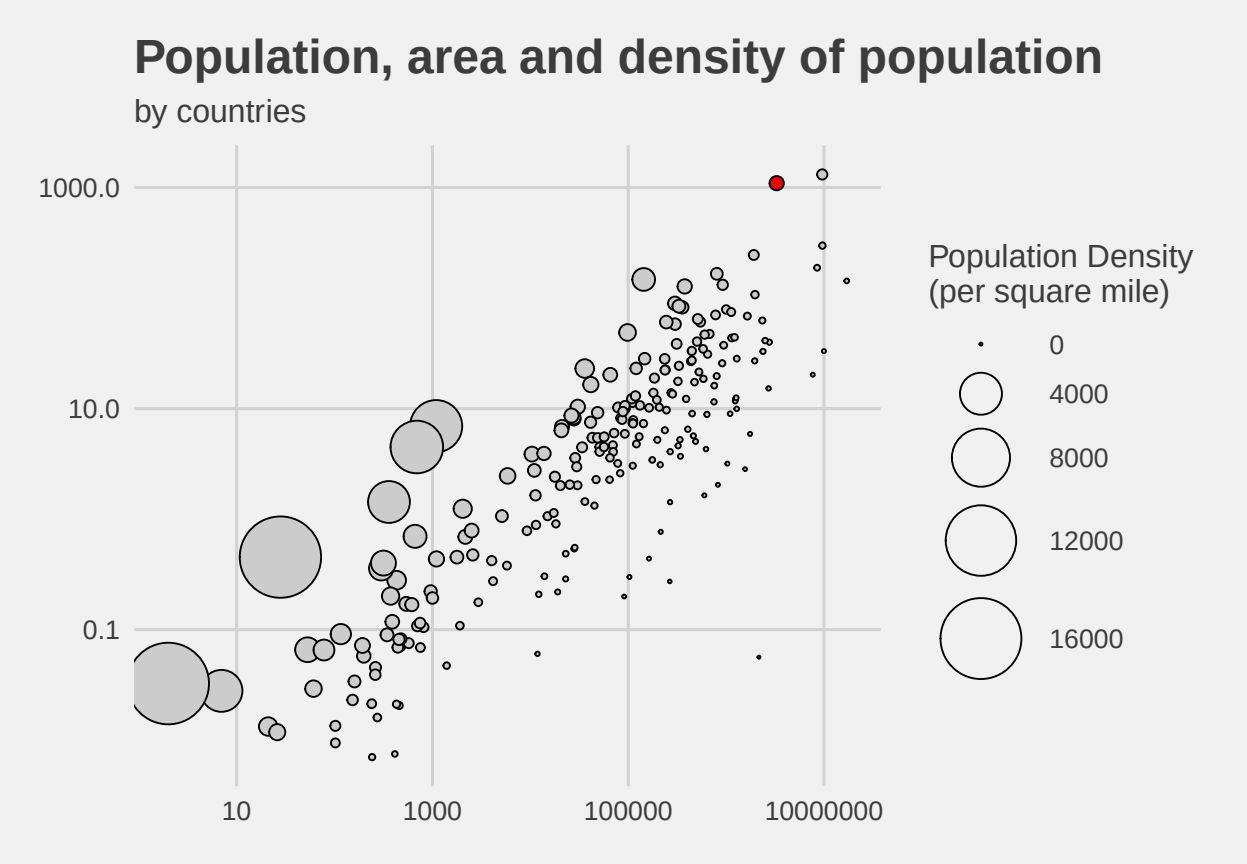
```
country <- as.data.frame(country)
country$region <- as.character(country$region)

other <- merge(other, country, by = "region", all.x = T, )
names(other)[13] <- "Vaccinate_Percentage"

head2 <- other[sample(1:nrow(other),5), ]
head2
```

```
##        region Population   Area PopDensity   GDP Phones Literacy Birthrate
## 56   Djibouti     486530  23000       21.2  1300   22.8      679      3953
## 60    Ecuador   13547510 283560       47.8  3300  125.6      925      2229
## 51    Croatia    4494749  56542       79.5 10600  420.4      985       961
## 121 Lithuania    3585906  65200       55.0 11400  223.4      996       875
## 199    Taiwan   23036087  35980      640.3 23400  591.0      961      1256
##     Deathrate Agriculture Industry Service Vaccinate_Percentage
## 56       1931       0.179    0.225   0.596                0.00
## 60        423       0.070    0.312   0.618               81.02
## 51       1148       0.070    0.308   0.622               55.87
## 121      1098       0.055    0.325   0.620               71.93
## 199       648       0.018    0.259   0.723                0.00
```

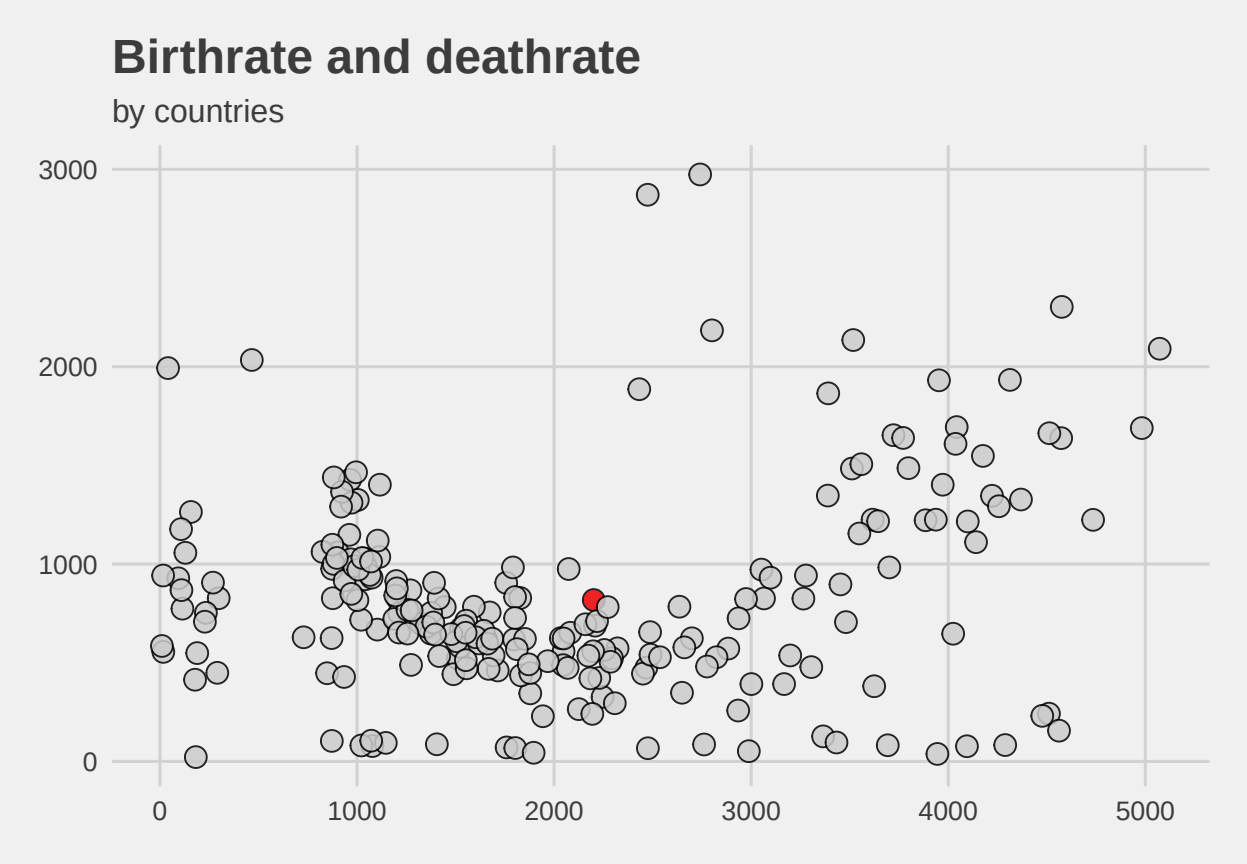
```
core <- cor(other[,c(2:ncol(other))], method = "spearman", use = "complete.obs")

options(repr.plot.width = 20, repr.plot.height = 20)

ggcorrplot(core)
```



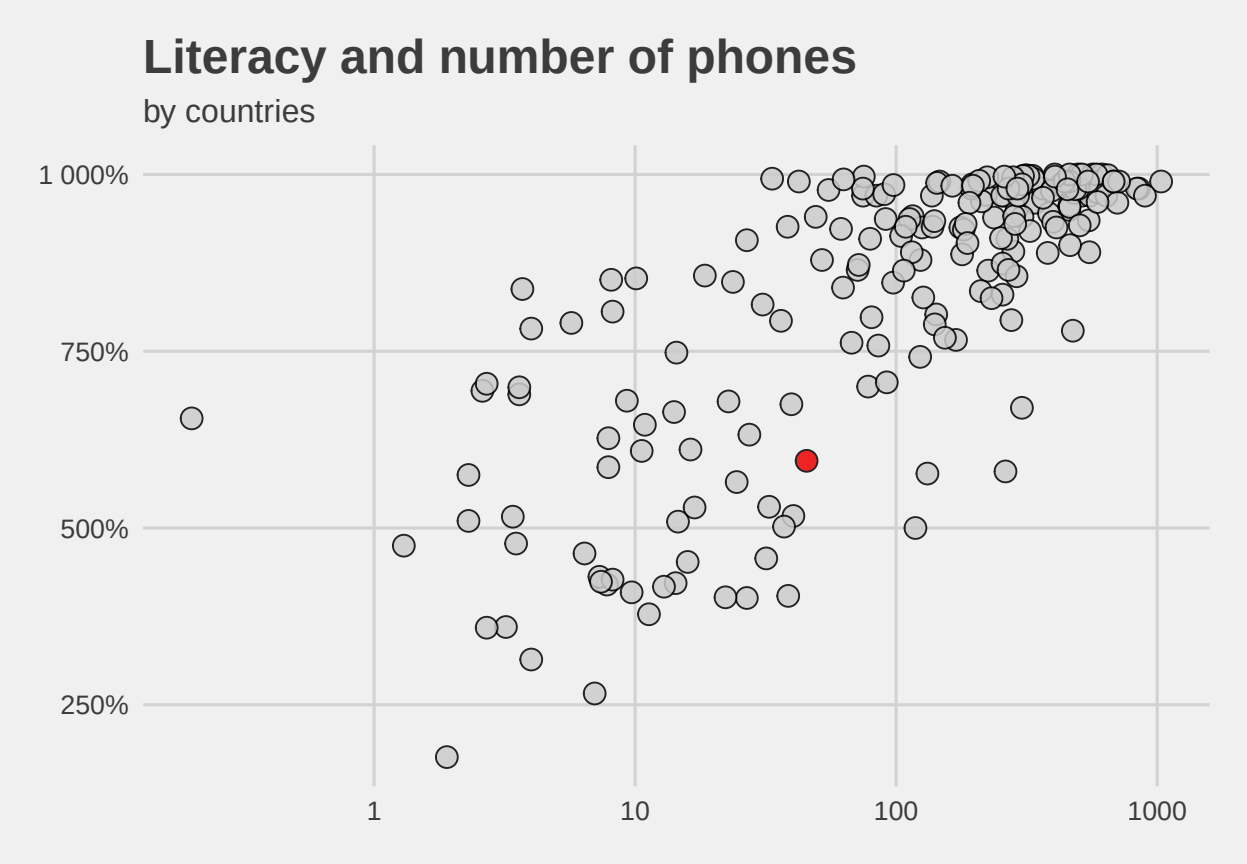

```
other$india <- ifelse(other$region=="India","India", "Rest of World")

options(repr.plot.width = 13.8, repr.plot.height = 9.2)

ggplot(other, aes(Area, Population/1000000, fill = india, size = PopDensity))+
  geom_point(shape = 21)+
  scale_fill_manual(values = c("red2", "gray80"), guide = "none")+
  scale_size(range = c(0.2, 14))+
  scale_y_log10()+
  scale_x_log10()+
  labs(y = "Population (in milions, logarithmic scale)", x = "Area (in km2, logarithmic scale)", title =
       subtitle = "by countries", size = "Population Density\n(per square mile)")+
  theme_fivethirtyeight()+
  theme(legend.position = "right", legend.direction = "vertical")
```



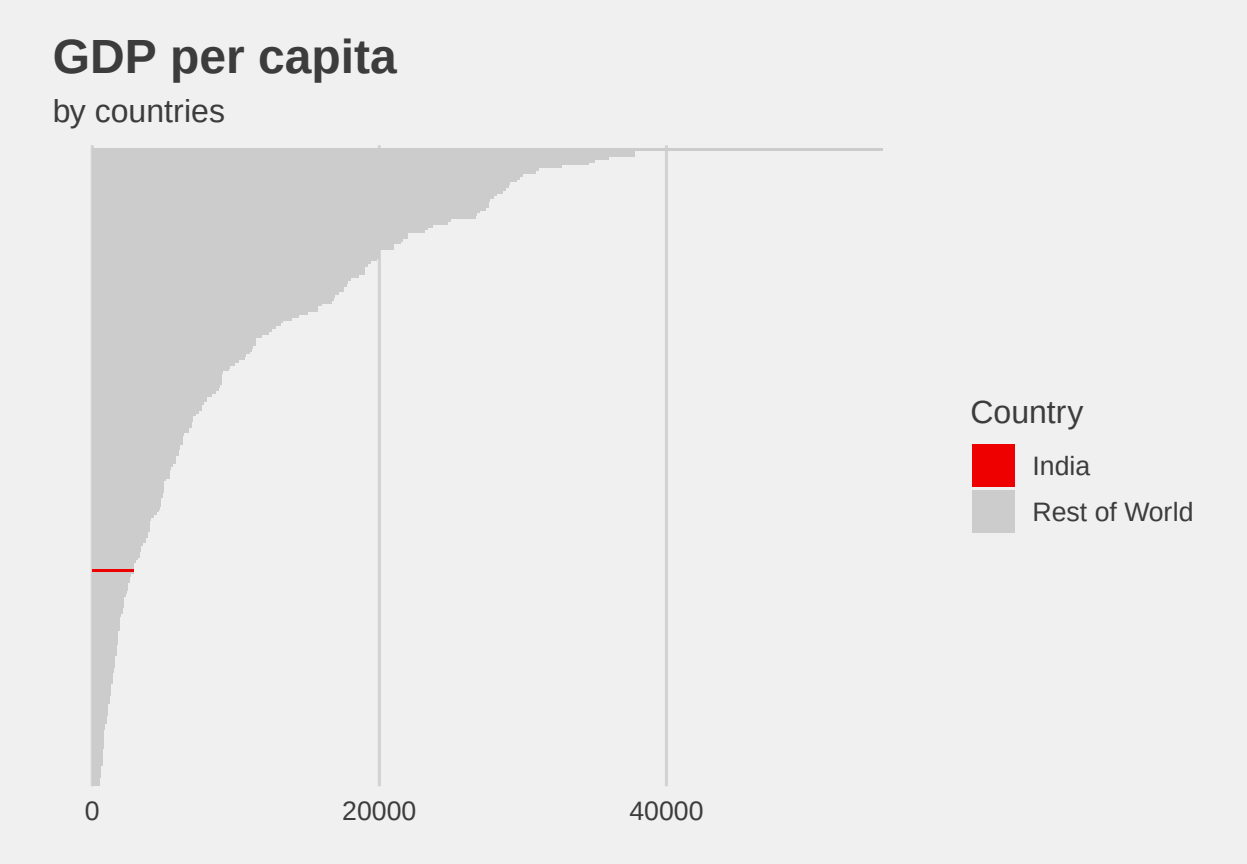

```
ggplot(other, aes(Birthrate, Deathrate, fill = india))+
  geom_point(shape = 21, size = 3.5, alpha = 0.85)+
  scale_fill_manual(values = c("red2", "gray80"), guide = "none")+
  labs(y = "Deathrate", x = "Birthrate", title = "Birthrate and deathrate", fill = "Country",
       subtitle = "by countries")+
  theme_fivethirtyeight()+
  theme(legend.position = "right", legend.direction = "vertical")
```



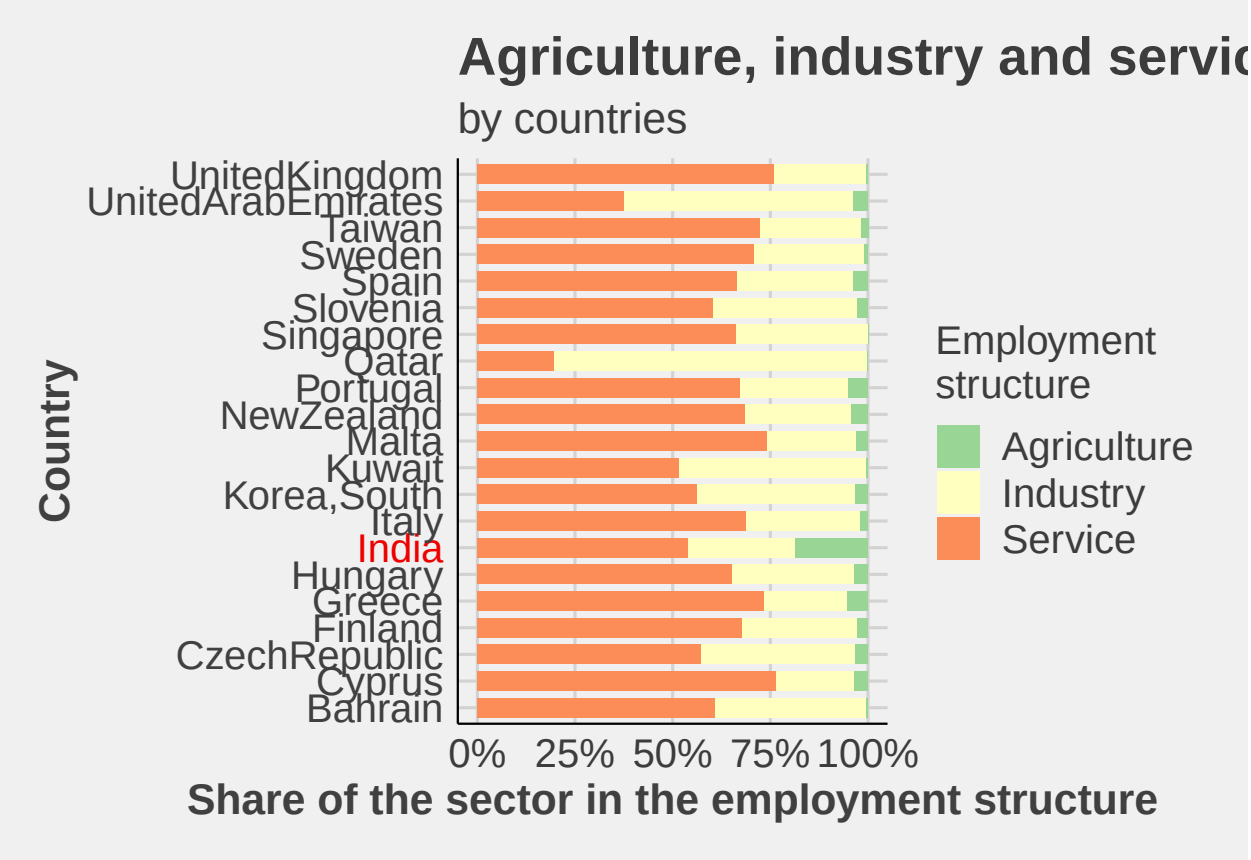

```
ggplot(other, aes(Phones, Literacy/100, fill = india))+
  geom_point(shape = 21, size = 3.5, alpha = 0.85)+
  scale_fill_manual(values = c("red2", "gray80"), guide = "none")+
  scale_x_log10()+
  scale_y_continuous(labels = scales::percent_format(accuracy = 1))+
  #annotate(geom = "curve", xend = 500, y = 0.82, x = 800, yend = 0.95, curvature = 0.25, arrow = arrow
  #annotate("text", x = 800, y = 0.8, label = "India", size = 5.5, colour = "red2", fontface = 2)+
  labs(y = "Literacy (in %)", x = "Phones (per 1000 inhabitants, logarithmic scale)", title = "Literacy
       fill = "Country", subtitle = "by countries")+
  theme_fivethirtyeight()+
  theme(legend.position = "right", legend.direction = "vertical")
```




```
ggplot(other, aes(reorder(region, + GDP), GDP, fill = india, size = india))+
  geom_bar(stat = "identity", width = 0.9)+
  scale_fill_manual(values = c("red2", "gray80"))+
  scale_colour_manual(values = c("red2", "gray80"))+
  scale_size_manual(values = c(7,2), guide = "none")+
  coord_flip()+
  labs(y = "GDP per capita (in $)", x = "Country", title = "GDP per capita", fill = "Country",
       subtitle = "by countries")+
  theme_fivethirtyeight()+
  theme(legend.position = "right", legend.direction = "vertical", axis.text.y = element_text(size = 0),
        panel.grid.major.y = element_blank())
```




```
gosp <- other %>%
  select(region, Agriculture, Industry, Service) %>%
  filter(is.na(Agriculture)==F) %>%
  filter(is.na(Industry)==F) %>%
  filter(is.na(Service)==F) %>%
  melt(., id.vars = "region")

gosp <- gosp[gosp$region %in% c("India", "Sweden", "Italy", "Singapore", "Taiwan", "UnitedArabEmirates"
"Spain", "NewZealand", "Qatar","Greece", "Cyprus", "Kuwait", "Slovenia", "Portugal", "Korea,South",
"Malta", "Bahrain", "Finland", "CzechRepublic", "Hungary", "UnitedKingdom"),]

gosp$india <- ifelse(gosp$region=="India","India", "Rest of World")
a <- ifelse(gosp$region=="India","red2", "gray25")

ggplot(gosp, aes(region, weight = value))+
  geom_bar(aes(fill = variable), width = 0.75)+
  scale_fill_brewer(palette = "Spectral", direction = -1)+
  scale_colour_manual(values = c("red2", "gray45"), guide = "none")+
  scale_y_continuous(labels = scales::percent_format(accuracy = 1))+
  coord_flip()+
  labs(x = "Country", y = "Share of the sector in the employment structure", title = "Agriculture, indus
       colour = "Country", fill = "Employment\nstructure", subtitle = "by countries")+
  theme_fivethirtyeight()+
  theme_michau+
  theme(legend.position = "right", legend.direction = "vertical", axis.text.y = element_text(colour = a
```


Agriculture, industry and servic
by countries
UnitedKingdom
UnitedArabEmirates
Taiwan
Sweden
Spain
Slovenia
Singapore
Qatar
Portugal
NewZealand
Malta
Kuwait
Korea,South
Italy
India
Hungary
Greece
Finland
CzechRepublic
Cyprus
Bahrain
Country
0%
25%
50%
75%
100%
Share of the sector in the employment structure
Employment structure
Agriculture
Industry
Service