Here I’ll draw a version of producing various visualizations of the progress of COVID-19 in R and ggplot. The main difference is that instead of ordering the panels alphabetically by country, I’ll order them from highest to lowest current reported cases.

Here’s the figure we’ll end up with:



Cumulative reported COVID-19 cases to date, top 50 Countries

There are two small tricks. First, getting *all* the data to show (in grey) in each panel while highlighting just *one* country. Second, for reasons of space, moving the panel labels (in ggplot’s terminology, the strip labels) inside the panels, in order to tighten up the space a bit. Doing this is really the same trick both times, viz, creating a some mini-datasets to use for particular layers of the plot.

The code for this (including code to pull the data) is attached in the repository. There’s a new function in the repo, though.

We’ll start with the data mostly cleaned and organized.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | > cov\_case\_curve  # A tibble: 1,165 x 9  # Groups: iso3 [94]  date cname iso3 cases deaths cu\_cases cu\_deaths days\_elapsed end\_label  <date> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <drtn> <chr>  1 2020-01-19 China CHN 136 1 216 3 0 days NA  2 2020-01-20 China CHN 19 0 235 3 1 days NA  3 2020-01-21 China CHN 151 3 386 6 2 days NA  4 2020-01-22 China CHN 140 11 526 17 3 days NA  5 2020-01-23 China CHN 97 0 623 17 4 days NA  6 2020-01-24 China CHN 259 9 882 26 5 days NA  7 2020-01-25 China CHN 441 15 1323 41 6 days NA  8 2020-01-26 China CHN 665 15 1988 56 7 days NA  9 2020-01-27 China CHN 787 25 2775 81 8 days NA  10 2020-01-28 China CHN 1753 25 4528 106 9 days NA  # … with 1,155 more rows |

Then we pick out the top 50 countries, isolating their maximum case value. The code here is a bit inefficient as I keep having to recode some of the country names in the mini-datasets. There are other inefficiencies too, but oh well. I’ll clean them up later.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32 | top\_50 <- cov\_case\_curve %>%  group\_by(cname) %>%  filter(cu\_cases == max(cu\_cases)) %>%  ungroup() %>%  top\_n(50, cu\_cases) %>%  select(iso3, cname, cu\_cases) %>%  mutate(days\_elapsed = 1,  cu\_cases = max(cov\_case\_curve$cu\_cases) - 1e4,  cname = recode(cname, `United States` = "USA",  `Iran, Islamic Republic of` = "Iran",  `Korea, Republic of` = "South Korea",  `United Kingdom` = "UK"))  top\_50  # A tibble: 50 x 4  iso3 cname cu\_cases days\_elapsed  <chr> <chr> <dbl> <dbl>  1 ARG Argentina 75991 1  2 AUS Australia 75991 1  3 AUT Austria 75991 1  4 BEL Belgium 75991 1  5 BRA Brazil 75991 1  6 CAN Canada 75991 1  7 CHL Chile 75991 1  8 CHN China 75991 1  9 CZE Czech Republic 75991 1  10 DNK Denmark 75991 1  # … with 40 more rows |

This gives us our label layer. We’ve set days\_elapsed and cu\_cases values to the same thing for every country, because these are the x and y locations where the country labels will go.

Next, a data layer for the grey line traces and a data layer for the little endpoints at the current case-count value.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | cov\_case\_curve\_bg <- cov\_case\_curve %>%  select(-cname) %>%  filter(iso3 %in% top\_50$iso3)  cov\_case\_curve\_endpoints <- cov\_case\_curve %>%  filter(iso3 %in% top\_50$iso3) %>%  mutate(cname = recode(cname, `United States` = "USA",  `Iran, Islamic Republic of` = "Iran",  `Korea, Republic of` = "South Korea",  `United Kingdom` = "UK")) %>%  group\_by(iso3) %>%  filter(cu\_cases == max(cu\_cases)) %>%  select(cname, iso3, days\_elapsed, cu\_cases) %>%  ungroup() |

We drop cname in the cov\_case\_curve\_bg layer, because we’re going to facet by that value with the main dataset in a moment. That’s the trick that allows the traces for all the countries to appear in each panel.

And now we can draw the plot. I really need to fix that country recode

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56 | cov\_case\_sm <- cov\_case\_curve %>%  filter(iso3 %in% top\_50$iso3) %>%  mutate(cname = recode(cname, `United States` = "USA",  `Iran, Islamic Republic of` = "Iran",  `Korea, Republic of` = "South Korea",  `United Kingdom` = "UK")) %>%  ggplot(mapping = aes(x = days\_elapsed, y = cu\_cases)) +  # The line traces for every country, in every panel  geom\_line(data = cov\_case\_curve\_bg,  aes(group = iso3),  size = 0.15, color = "gray80") +  # The line trace in red, for the country in any given panel  geom\_line(color = "firebrick",  lineend = "round") +  # The point at the end. Bonus trick: some points can have fills!  geom\_point(data = cov\_case\_curve\_endpoints,  size = 1.1,  shape = 21,  color = "firebrick",  fill = "firebrick2"  ) +  # The country label inside the panel, in lieu of the strip label  geom\_text(data = top\_50,  mapping = aes(label = cname),  vjust = "inward",  hjust = "inward",  fontface = "bold",  color = "firebrick",  size = 2.1) +  # Log transform and friendly labels  scale\_y\_log10(labels = scales::label\_number\_si()) +  # Facet by country, order from high to low  facet\_wrap(~ reorder(cname, -cu\_cases), ncol = 5) +  labs(x = "Days Since 100th Confirmed Case",  y = "Cumulative Number of Cases (log10 scale)",  title = "Cumulative Number of Reported Cases of COVID-19: Top 50 Countries",  subtitle = paste("Data as of", format(max(cov\_curve$date), "%A, %B %e, %Y")),  Data: <https://www.ecdc.europa.eu/>") +  theme(plot.title = element\_text(size = rel(1), face = "bold"),  plot.subtitle = element\_text(size = rel(0.7)),  plot.caption = element\_text(size = rel(1)),  # turn off the strip label and tighten the panel spacing  strip.text = element\_blank(),  panel.spacing.x = unit(-0.05, "lines"),  panel.spacing.y = unit(0.3, "lines"),  axis.text.y = element\_text(size = rel(0.5)),  axis.title.x = element\_text(size = rel(1)),  axis.title.y = element\_text(size = rel(1)),  axis.text.x = element\_text(size = rel(0.5)),  legend.text = element\_text(size = rel(1)))  ggsave("figures/cov\_case\_sm.png",  cov\_case\_sm, width = 10, height = 12, dpi = 300) |