Homework 4

Walid Medani

2021-03-27

```css
.chunkcolor {
background-color: MistyRose;
}
```

```r
chinaprojects<- read_csv("Source_Data.csv") %>%
    rename(year = transactions_start_year)  %>%
    dplyr::select(project_id, project_title, recipients, recipients_iso3, ad_sector_codes, ad_sector_nam
        mutate(hover = paste0(recipients, "\n$", total_commitments))
options(scipen = 999)
```

World Map

I chose this world map visualization to showcase China's financing to
countries abroad between 2000-2014. A map is an engaging visual-
ization tool to show data that happens across different locations and
time, while also adding depth and simplicity. I used the plotly package
to perform this task since it felt the most intuitive for creating interac-
tive graphics and didn't require the need to connect to an external api
such as Google or Openstreemap.

I wish I was able to customize the visualizations of the map more. I
wanted to create a darker map background however most of the doc-
umentation and themes for plotly was within Python. Lastly since
plotly produces an HTML I wasn't able to produce a .png of the graph
despite using orcas.

```r
worldmap <- plot_geo(chinaprojects,
                locationmode = 'ISO-3',
                frame = ~year) %>%
  add_trace(locations = ~recipients_iso3,
            z = ~total_commitments,
            color = ~total_commitments, colorscale = 'Viridis',
            text = ~hover,
            hoverinfo = 'text') %>%
  colorbar(title = "Total Spending (USD)", tickprefix = '$') %>%
  layout(title = "Chinese Financing Abroad\n2000 - 2014") %>%
  config(displayModeBar = TRUE)
worldmap
```

Choropleth

I chose to dig deeper into visualizing maps by making a choropleth map of African countries that received financing from China. A choropleth is a great way to visualize how variables vary across a geographic region since it essentially functions as a "heat map".

I had difficulty with adjusting the color spectrum in order to make it more visually distinct. I also had trouble with shortening the legend labels and also giving them a prefix of "$" like I did with plotly above.

```r
countrytotal<- chinaprojects %>%
  group_by(recipients_iso3) %>%
  summarise(AmountReceived = sum(total_commitments, na.rm=TRUE))

africa <- st_read("africa.geo.json") %>%
  rename(recipients_iso3 = iso_a3)

## Reading layer `africa.geo' from data source `C:\Users\W\Documents\R projects\DACSS-601\homework 4 fol
## Simple feature collection with 54 features and 66 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -25.36042 ymin: -46.96575 xmax: 51.41704 ymax: 37.3452
## Geodetic CRS:  WGS 84

map_and_data <- inner_join(africa, countrytotal, by="recipients_iso3")

ggplot(map_and_data) +
  geom_sf(aes(fill=AmountReceived)) +
  scale_fill_viridis_c(name= 'Financing Received (USD)',
                       guide=guide_legend(
                       legend.position = 'right',
                        label.hjust = 1,
                        keywidth = 1,
                        keyheight = 2
                       )) +
  labs(title = "African Recipients of Chinese Financing",
       subtitle = "2000 - 2014") +
    theme(title = element_text(face = 'bold'))
```
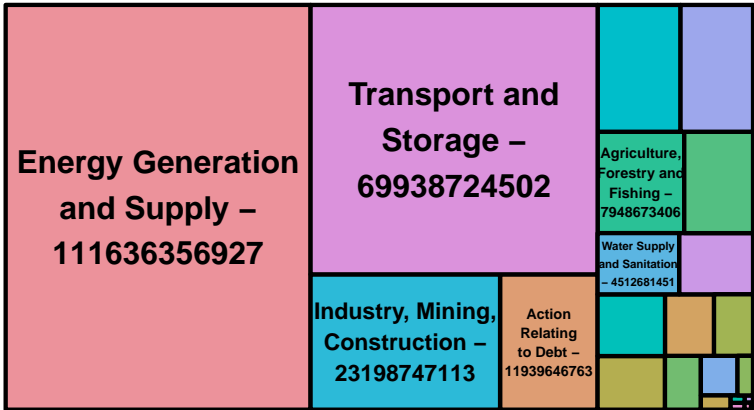
African Recipients of Chinese Financing

2000 – 2014   **Financing Received (USD)**
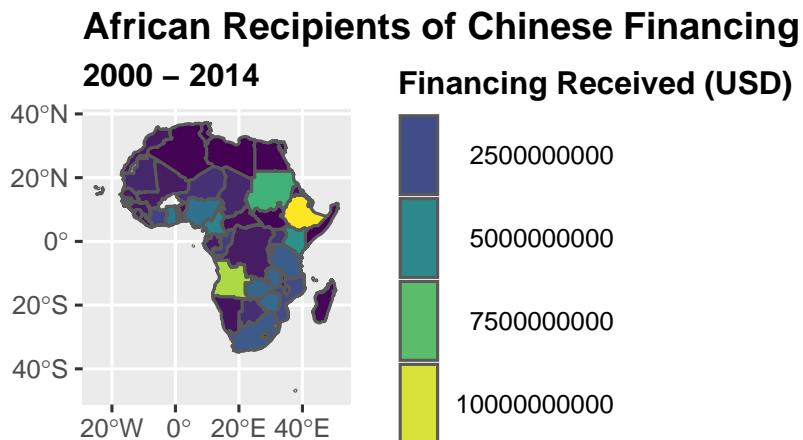
Treemap

Here I used a Treemap to visualize spending by sector since it is a
great way to organize proportions of categorical data when you have
large datasets in comparison to a pie-chart.

However, I had a difficult time with changing certain parameters
such as forcing my text to fit into boxes (abbreviate wasn't working
for me) and changing the color scheme to where it didn't duplicate
colors. Also when I knit to a PDF the treemap changes its structures
and doesn't display some sectors.

```
chinaprojects %>%
  group_by(ad_sector_names) %>%
  summarise(sumtotal = sum(total_commitments, na.rm=TRUE)) %>%
  mutate(adtotal = paste(ad_sector_names,sumtotal, sep = " - ")) %>%
  treemap(
    index = "adtotal",
    vSize = "sumtotal",
    type = "index",
    title = "Spending by sector")
```

Spending by sector

Energy Generation and Supply – 111636356927

Transport and Storage – 69938724502

Industry, Mining, Construction – 23198747113

Action Relating to Debt – 11939646763

Agriculture, Forestry and Fishing – 7948673406

Water Supply and Sanitation – 4512681451

Sources

Original publication
    Pre-merged data