Technical Documentation--Data visualization team 2

- **Thechnical plan** :

  - Context :

  When an epidemic occurs, timely information on the extent of the spread and the number of people infected is an important prerequisite for the development of prevention programmes. If the number and extent of cases are not known in time, the immediate consequence is the spread and re-proliferation of the virus. I think one of the main reasons for today's situation is the lack of a realistic and intuitive understanding of the situation at the beginning of the year, which has led to some disregard for the development of the epidemic.

  Therefore, in the scenario of real-time surveillance of the epidemic, visualization is a very important element in addition to the realism and real-time nature of the data. And we see today that in some countries where data science is not well developed, the daily epidemic releases of the National Health Construction Commissions are still in the form of textual descriptions, which can result in untimely and incomprehensible information delivery. We also use data visualization as a weapon to predict inflection points and make predictions about the economies of countries affected by the epidemic, and humanity can finally overcome this disaster.

  - Goals :
    - -Using Heatmap to demonstrate the impact of Covid-19 on national economies.
    - -Predicting the inflection point of the epidemic

  - Exsting resources :
    - -Data source : WHO, World Bank, Data from national statistical offices
    - -Open source of solution: Tencent health and Johns Hopkins Coivd-19

  - Style guides:
    - -The language of the platform will be English, presented in the form of a map of global hotspot patterns.

  - Outline of topics:

  The number of confirmed Covid-19 diagnoses reflected in the chart does not reflect the true extent of the epidemic in the country.This relates to the relevance behind the data, where differences in health care and economic levels between countries can lead to differences in actual testing, with testing prevalence in developing countries being much lower than in developed countries, resulting in incomplete sources of outbreak data in developing countries and affecting the final results of data visualization.

- Tools and management :
  - Tools :
    Module 1 : data visualization, ggplot for data visualization in R
    Module 4 : Predictive modelling, Linear regression
  - Management :
    DSDM agile PM, using evolving solution to develop platform

- Deadline and final deliverables:
  - Deadline : 1st draft of platform 17th April
    final version of platform 19th April
  - Deliverables :
    kick-off : 13th April
    Iterative Development: 16th April
    integrated testing : 17th April
    final product : 19th April
    Technical documentation : 19th April

- **Design of the Workflow**

Data Type filtering

↓

Data WB cleansing & Mining

↓

Data Visualization

↓

Predictive result on the map

↓

Final analysis and report

- **Process explanation**

First, we need to download libraries, below are the libraries that we need to use in the process.

```
library(readxl)
library(ggplot2)
library(sf)
library(rnaturalearth)
library(ggmap)
library(flexdashboard)
```

readxl -The readxl package makes it easy to get data out of Excel and into R.
sf- Simple Features for R. Support for simple features, a standardized way to encode spatial vector data
rnaturalearth - a data package designed to provide map data that can be visualised using other R packages. Natural Earth is a public domain map dataset including vector country and other administrative boundaries.
flexdasborad -Use R Markdown to publish a group of related data visualizations as a dashboard. Support for a wide variety of components including htmlwidgets; base, lattice, and grid graphics; tabular data; gauges and value boxes; and text annotations.

After downloading the libraries, we have a first view of the data. We found that there are a lot of missing values, so the dataset need to be cleaned.

```
covid <- read_excel("team2/covid-fci-data.xlsx")
View(covid) #To viwe the data
```

For cleaning the data, we replace all the missing values by NA, and remove all of them.

```r
#finding the missing values
is.na(covid)
covid$LAT[covid$LAT==""] <- NA
mean(covid)

#storing the level 1 policy as policy1 and replacing the missing values with NA
policy1=as.factor(covid$`Level 1 policy measure`)
covid$`Level 1 policy measure`[covid$`Level 1 policy measure`==""] <- NA

income=covid$`Level of income`
date=as.factor(covid$`Entry date`)

#storing the level 1 policy as policy1 and replacing the missing values with NA
policy2=as.factor(covid$`Level 2 policy measure`)
covid$`Level 2 policy measure`[covid$`Level 2 policy measure`==""] <- NA

#remove the missing values
covid <- na.omit(covid)
```

After removing all the missing data, we got the clean data that we can use for visualization. Below is the result after cleaning.

readme.md × dashboard.r × .gitignore × covid ×

Filter

	Iso 3 Code	Level of income	Region	Country	Date (at or prior to)	Entry date	Level 1 policy measure	Level 2 policy measure	LAT	LONG
1	CHN	Upper middle income	EAP	China	2020-02-01	2020-03-21	Financial Institutions	Operational continuity	31.8257	117.2264
2	CAN	High income	Other G20	Canada	2020-02-03	2020-03-21	Financial Markets	Market functioning	53.9333	-116.5765
3	THA	Upper middle income	EAP	Thailand	2020-02-05	2020-03-21	Liquidity/funding	Policy rate	15.0000	101.0000
4	BLR	Upper middle income	ECA	Belarus	2020-02-19	2020-03-31	Liquidity/funding	Liquidity (incl FX)/ELA	53.7098	27.9534
5	BLR	Upper middle income	ECA	Belarus	2020-02-19	2020-03-31	Financial Markets	Market functioning	53.7098	27.9534
6	BLR	Upper middle income	ECA	Belarus	2020-02-19	2020-03-31	Liquidity/funding	Policy rate	53.7098	27.9534
7	CHN	Upper middle income	EAP	China	2020-02-20	2020-03-21	Liquidity/funding	Policy rate	31.8257	117.2264
8	IDN	Lower middle income	EAP	Indonesia	2020-02-20	2020-03-21	Liquidity/funding	Policy rate	-0.7893	113.9213
9	JPN	High income	Other G20	Japan	2020-02-20	2020-03-21	Financial Institutions	Support borrowers	36.0000	138.0000
10	JPN	High income	Other G20	Japan	2020-02-25	2020-03-21	Financial Institutions	Operational continuity	36.0000	138.0000
11	ITA	High income	Other G20	Italy	2020-03-01	2020-03-21	Financial Institutions	Operational continuity	43.0000	12.0000

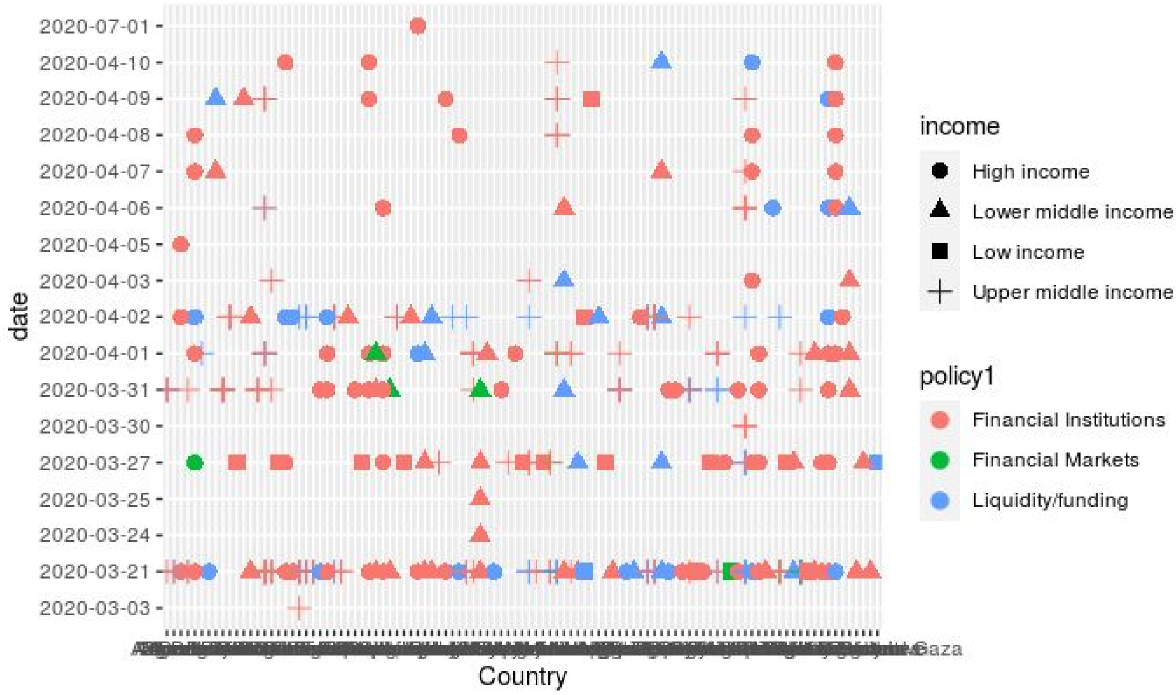
Showing 1 to 13 of 648 entries, 10 total columns

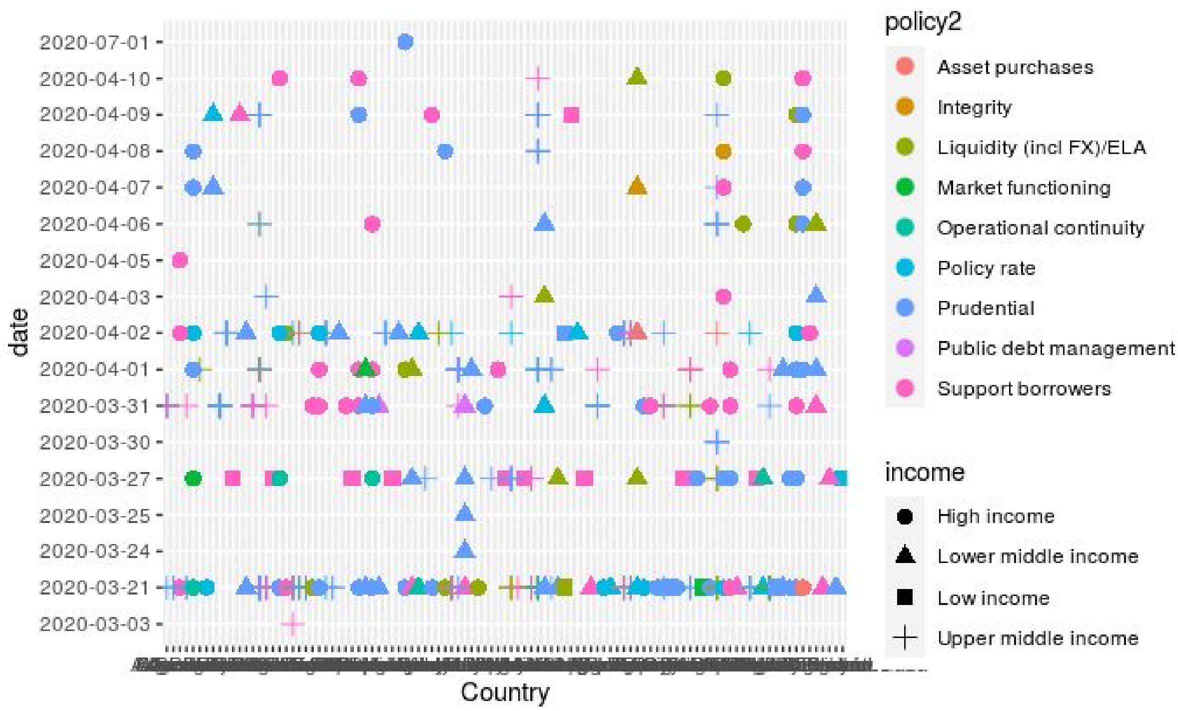
Plotting the data, by using ggplot, we can have a view of every country.

```r
#ploting the cointries with respect to policy1 and income
ggplot(covid, aes(Country,date, color = policy1)) +
  geom_point(aes(shape = income), size = 3)
#ploting the cointries with respect to policy1 and income
ggplot(covid, aes(Country,date, color = policy2)) +
  geom_point(aes(shape = income), size = 3)
```

Belows are the Scatter which we got from the last operation.

The below dot plot show the level of income of various countries with respect to different policies they implemented in time to fight against COVID-19

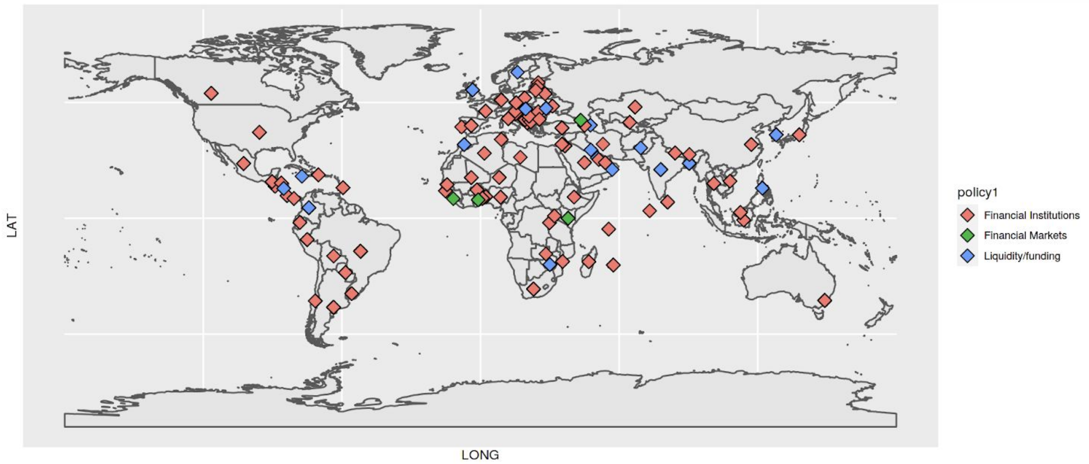


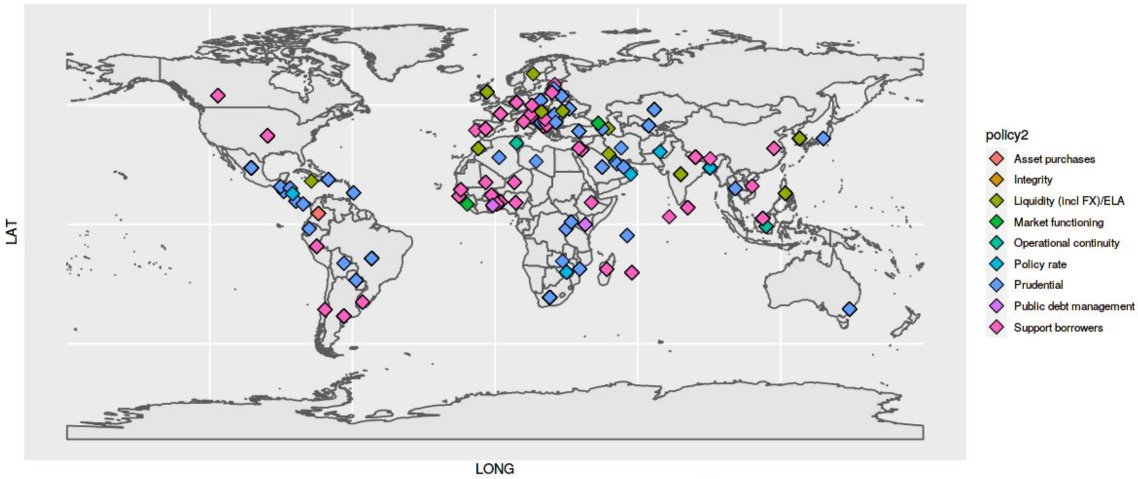


By using the libraries rnaturalearth and rnaturalearthdata we can have a global view of the data.

```
library(rnaturalearth)
library(rnaturalearthdata)

world <- ne_countries(scale = "medium", returnclass = "sf")
class(world)
ggplot(data = world) +
  geom_sf() +
  geom_point(data = covid, aes(x = LONG, y = LAT,fill = policy1), size = 4,
             shape = 23)
```

Belows are the global view of the data.





By using libraries shiny and shinydashborad we can create our dashboard.

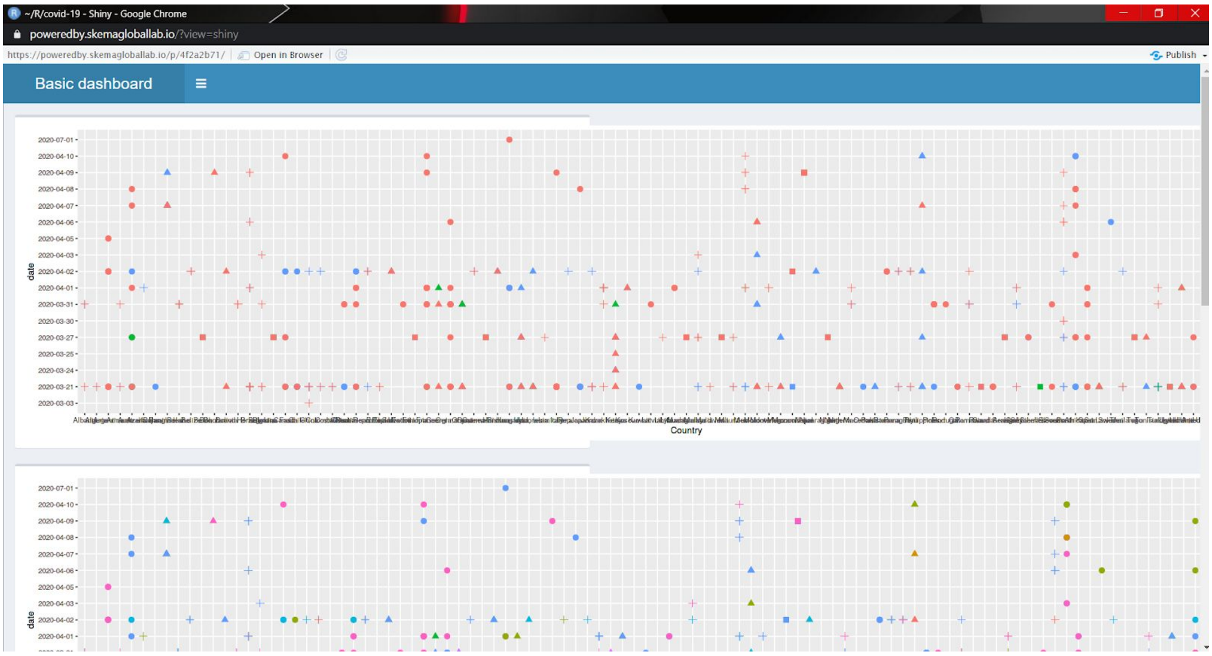
```
## app.R ##
library(shiny)
library(shinydashboard)


ui <- dashboardPage(
  dashboardHeader(title = "Basic dashboard"),
  dashboardSidebar(),
  dashboardBody(
```

Putting 4 boxes in rows.

```
    # Boxes need to be put in a row (or column)
    fluidRow(
      box(plotOutput("plot1", height = 400,width = 1750))
    ),
    fluidRow(
      box(plotOutput("plot2", height = 400,width = 1750))
    ),
    fluidRow(
      box(plotOutput("plot3", height = 400,width = 1000))
    ),
    fluidRow(
      box(plotOutput("plot4", height = 400,width = 1000))
    ),
  )
)
```

```
server <- function(input, output,session) {
  output$plot1 <-  renderPlot({
    ggplot(covid, aes(Country,date, color = policy1)) +
      geom_point(aes(shape = income), size = 3)
  })
  output$plot2 <-  renderPlot({
    ggplot(covid, aes(Country,date, color = policy2)) +
      geom_point(aes(shape = income), size = 3)
  })
  output$plot3 <-  renderPlot({
    ggplot(data = world) +
      geom_sf() +
      geom_point(data = covid, aes(x = LONG, y = LAT,fill =policy1), size = 4,
                 shape = 23)
  })
  output$plot4 <-  renderPlot({
    ggplot(data = world) +
      geom_sf() +
      geom_point(data = covid, aes(x = LONG, y = LAT,fill =policy2), size = 4,
                 shape = 23)
  })
}
```

The below screenshot represents the dashboard created in shiny environment of various plots shown in this documentation



- **Conclusion**

First of all, we would like to thank SKEMA quantum AI Lab for giving us a chance to truly understand covid-19 with the method of data scientific analysis. Through this project, we realize the high efficiency and strong readability of data visualization for solving practical problems, but at the same time, we also recognize that data visualization is not a “Silver Bullet”, Because of the source and authenticity of the data itself, the results of data visualization will be fundamentally different. At the same time, the practical significance behind each number can not be easily displayed by several pictures. As a powerful tool, data visualization can be used to help us understand reality, but it still needs us to be human to make the final decision, in some cases, it is impossible to exclude some countries from making the data "Say what their citizens want to hear" for political reasons.