

COVID-19に対する社会的経済的レジリエンス： Rの学習(2)

メタデータ	言語: ja 出版者: 静岡大学人文社会科学部 公開日: 2020-11-13 キーワード (Ja): キーワード (En): 作成者: 遠山, 弘徳 メールアドレス: 所属:
URL	https://doi.org/10.14945/00027752

資料

COVID-19に対する社会的経済的レジリエンス – Rの学習(2)

遠山 弘 徳

I. パズル

以下の図1は、2020年5月1日時点でのCOVID-19による死者数トップ100カ国について、y軸に死亡者数、x軸に確認感染者数(いずれも対数表示)をとった散布図です。あわせて2017年更新のThe BCG World AtlasにもとづきBCG接種の有無を示しています。これは対数表示ですから、アメリカ(USA)、イタリア(ITA)、フランス(FRA)、スペイン(ESP)、イギリス(GBR)といった国の感染者数、死亡者数がいかに爆発的に増加しているかが分かります。

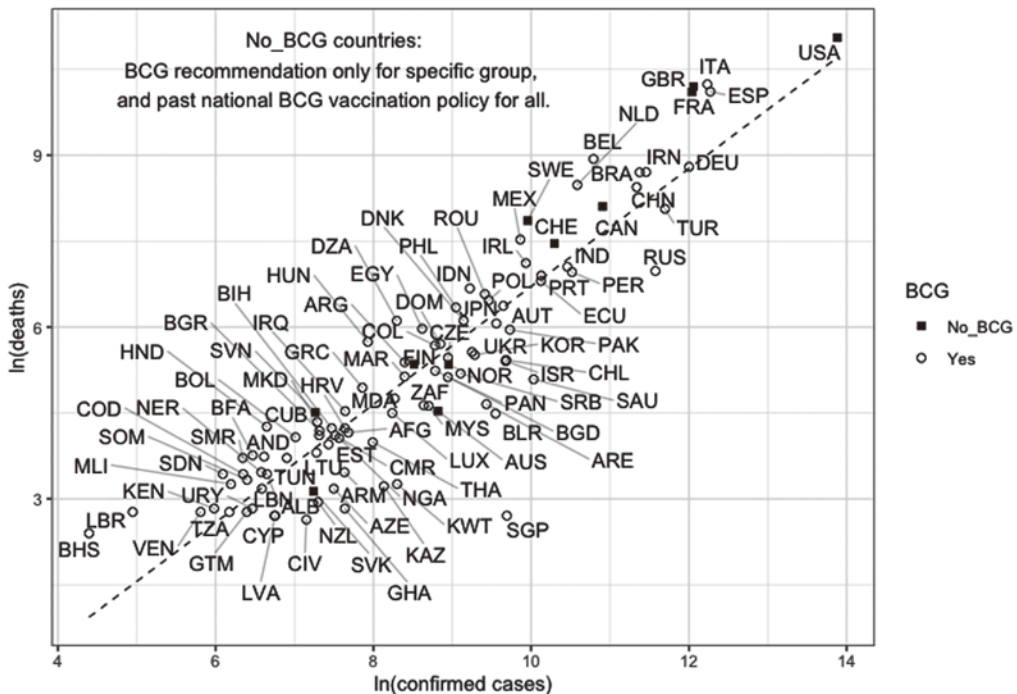


図1 死亡者数トップ100カ国の死亡者数・確認感染者数の散布図

注. 人口100万人以下の国・地域は除外してある。点線の直線は回帰線。No_BCG国は特定グループにのみBCG接種が推奨されている国と過去に全国的なBCG接種を実施していた国である。

この図1を見ると、第1に、確認感染者数および死者数の増加に大きな相違が見られることに気づきます。さらに、第2に、感染者数の増加と死者数の増加は一当然のことですが一相関しています。それでも、確認感染者から予測される死者数(回帰線)を基準に考えると、感染者数に対して大幅に死者数を生み出している国(回帰線を上回る国)とそれを大幅に下回っている国(回帰線を下回っている国)が存在することがわかります。たとえば、ベルギー(BEL)、イタリア(ITA)、スペイン(ESP)、フランス(FRA)、イギリス(GBR)は感染者数の増加に比べて死者が多く、他方、ロシア(RUS)、トルコ(TUR)、シンガポール(SGP)等は感染者数の上昇に比べ死者が下回っています。

このように確認感染者数と死者数の増加、および確認感染者数に対する死者数をみると各国で大きな相違が観察されます。感染者数の確認—PCR検査の違い—に問題があるものの、これはパズルです(e.g., Bayer and Kuhn, 2020)。こうした違いを説明するために、たとえばBCG仮説も提起されています(Miller, et.al., 2020)。図1にはBCG接種の有無も示されていますが、全国的な接種の有無が確認感染者数・死者数の増加の違いを生み出しているようには見えません(ここではBCGの種類は考慮されていません)。あるいは両親と同居する生産年齢人口比が対感染者死亡率と関連していることが示されています(Bayer and Kuhn, 2020)。

各国の相違を説明するために、いろいろな要因をあげることができるかもしれませんが、しかし、ここでは単一の要因からアプローチするのではなく、さまざまな要因を考慮し、このパズルにアプローチしたいと思います。つまり、さまざまな指標を総合し、各国の違いを探ることにします。言わば、COVID-19に対する社会経済的レジリエンス(resilience)を尺度する指標を作成する試みです。

II. 社会経済的レジリエンスの比較

COVID-19に対する社会経済的抵抗力(resilience)が高ければ、感染者数の増加は抑えられるかもしれません。さらに、感染者が増加しても、死者は抑えられるかもしれません。他面、そうした社会経済的抵抗力が弱ければ、感染者数、死者数の増加を食い止めることはとても難しくなるかもしれません。こうした社会経済的レジリエンスは社会経済のさまざまな要因の複合と言えるでしょう。したがってそれは、さまざまな、複数の指標の情報を総合することによって表現されます。このため、さまざまな指標の情報を要約する、主成分分析という手法を使って社会経済的レジリエンス指標を作成することにします。

II. 1. 次元の縮小—主成分分析

最初に、簡単に主成分分析を説明しておきましょう(主成分分析の詳細については、たとえば中山, 2011を参照)。図2に示されている表を見て下さい。このテーブルは63行×12列からなります。列が変数を示し、行が各国の観察値を示しています。第1列のiso2c変数は国名の2文字コード、第2列の変数countryは国名です。実際のデータはそれ以下の列になります。したがって実際の観察値が入った変数は10列ということになります。

▲	iso2c	country	pop65	life	urban	inb	trade	fdi
1	AF	Afghanistan	2.468788	63.10675	24.70450	NA	NA	0.4033717
2	AR	Argentina	10.689757	75.98587	91.43575	6751500	28.49716	1.7788220
3	AU	Australia	14.679350	82.31098	85.65287	7373250	42.00828	3.8693603
4	BD	Bangladesh	5.057620	71.33675	33.92450	NA	42.49143	1.2097720
5	BE	Belgium	18.084764	81.04573	97.85263	7995625	159.89627	1.7830268
6	BF	Burkina Faso	2.425823	59.59213	27.25100	185750	60.11925	2.5638978
7	BO	Bolivia	6.702951	69.99712	68.19638	909000	71.50913	2.5494868
8	BR	Brazil	7.919718	74.83875	85.61813	6177000	25.53028	3.6935653
9	CA	Canada	15.832934	81.79299	81.24362	18114125	64.42658	2.8297300
10	CD	Congo, Dem. Rep.	3.015472	58.98163	42.47750	NA	71.27717	4.5594774
11	CI	Cote d'Ivoire	2.834043	55.67200	49.24500	1024875	74.76492	1.5966130
12	CL	Chile	10.482954	79.54888	87.33825	4529125	62.32597	6.8914077
13	CM	Cameroon	2.761119	57.29888	54.27375	NA	50.15858	2.0488976
14	CN	China	9.344788	75.74787	54.86200	58301250	43.00661	2.3471847
15	CO	Colombia	7.585840	76.40762	79.57600	2853750	37.55939	4.2173472
16	CU	Cuba	13.933748	78.55100	76.87475	3511250	36.07752	NA
17	CZ	Czech Republic	17.713713	78.56372	73.44625	8638625	150.35501	3.6163579
18	DE	Germany	21.108992	80.77165	77.21213	33773375	86.07521	1.9236858
19	DO	Dominican Republic	6.622138	73.34317	78.96200	5691283	53.03605	3.3769520

図2 サンプルデータ

データの出所：世界銀行WDI.

いま、各国を1つの変数pop65—全人口に占める65歳以上人口比率—で表現してみましょう(図3)。これは、1つの変数—65歳以上人口—を基準に各国を比較することになります。とても簡単に各国の特徴を理解できると思います。たとえば、日本、イタリア、ドイツといった国は高齢化率が高く、逆に、タイのような新興国は比較的高齢化が進んでいない、と特徴づけることができます。

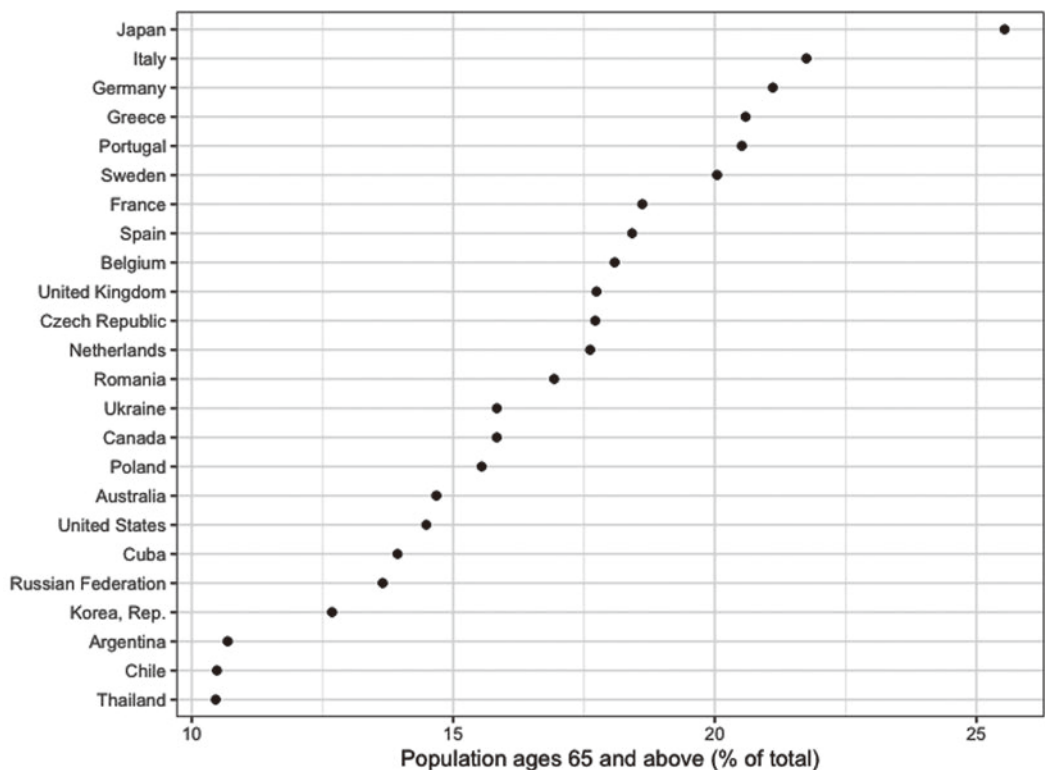


図3 1変数による各国の表現

注. サンプル経済中、総人口に占める65歳以上人口が10%を超える経済についてのみ表示.

さらに、各国を最初の2つの変数で表現してみましょう。つまり、2つの変数—pop65とlife変数—によって各国の特徴を理解しようとする試みです。y軸は平均寿命、x軸は図3と同じく65歳以上人口比率を示しています。

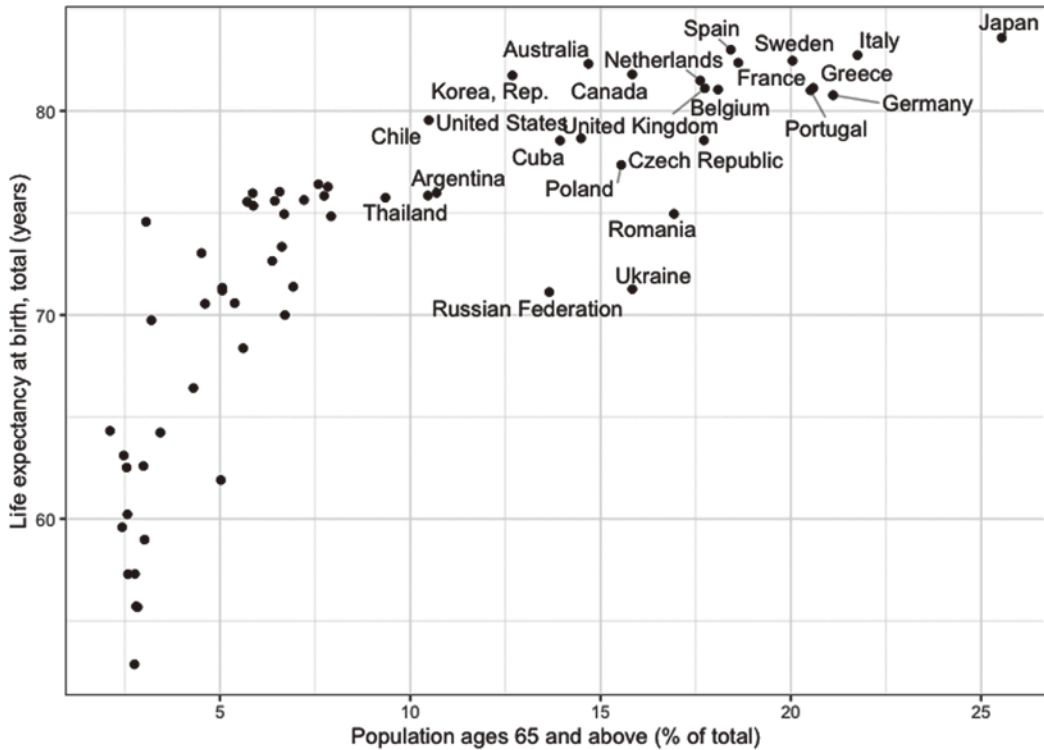


図4 2変数による各国の表現

注. 65歳以上人口比率が10%を超える国についてのみ国名を表示.

この図4をみると、図3と同じように、各国を2つの変数で比較し、各国の特徴を理解することができます。たとえば、高齢化率が高い国は寿命も長いといった特徴を見出すことができます。

それでは3つの変数の場合はどうでしょうか。次の図5を見てください。今度は、65歳以上人口比率、平均寿命および都市人口比率の3つの変数によって各国を表現しています。図3、4に比べ、一目見ただけでは各国の特徴をとらえることは難しくなります。

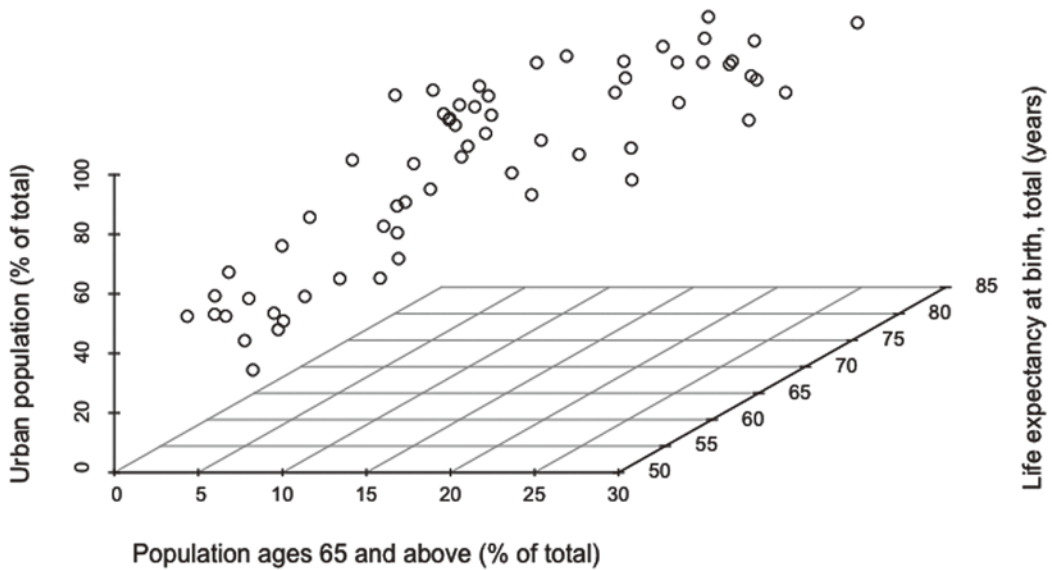


図5 3変数による各国の表現：3次元プロット

注. x軸は65歳以上人口比率, y軸は平均寿命, z軸は都市人口比率を示す. 空間中の○は各国を示す.

2次元から3次元の空間へと移っただけでも、理解が難しくなるのに、さらに変数が増え、4次元、5次元…(グラフでは表現できませんが)へと次元が増えていくと、もはや各国を比較し、その特徴を捉えることは不可能となります。そこで4次元、5次元…の空間の特徴(情報量)をもっとも良く要約したものがあれば、とても便利です。そうした要約されたものが主成分と呼ばれます。主成分分析とは要約したものを探す手法だと言えます。

技術的に言えば、主成分分析のアルゴリズムは、データの分散一ぱらつきが各国の特徴・違いを表現します—がもっとも大きいベクトルを探し出します。このベクトルはデータの中でより多くの情報を含んでおり、第1主成分と呼ばれます。次に第1成分と直交する方向からふたたび分散の大きいベクトル—次に情報量の多い第2主成分—を探し出します。

II. 2. レジリエンス指標作成のためのデータ

それではCOVID-19に対する社会経済の抵抗力を考えてみましょう。ざっと考えただけでも、大きくは4種類の指標グループを考えることができます。

第1に、医療支出、医療従事者数、病床数、医師・看護師の数といった社会の医療インフラです。第2に、国内において人々が相互に接触する可能性の程度です。これには人口密度、都市化などをあげることができます。第3に、対外的な接触の程度です。ここではインバウンドや貿易の大きさといった経済要因を考えることができます。第4に、高齢者比率、平均寿命といった

COVID-19に対する脆弱性に関わる指標です。

もちろん、これ以外にも清潔さを好む国民性など文化的要因、ロックダウン政策の実行可能性—政府への信頼—なども考えることができるでしょう。たとえば、Bayer and Kuhn(2020)は世代間の結びつき—家族内の同居—を主要な説明要因とし、死亡者・感染者比率の国際比較を行っています。しかし、ここでは学習目的を考え、こうした要因は省略しておきます。

それではこうした多くの指標を使ってCOVID-19に対する各国の社会経済的な抵抗力を総合的に示す要約指標を作成しましょう。このために最初に、WDIパッケージを使って世界銀行のデータベースから以下のデータをダウンロードします。対象サンプル国は死者トップ100の国(ただし、人口1000万人未満は除外)、期間は2011~2018年です。

ここで利用する指標は以下のとおりです。指標の説明の後の記号はWDIにおいて与えられた変数名です(たとえば、SH.MED.BEDS.ZS)。WDIの変数名は長いので本資料においては、簡単な変数名に変更しています(たとえば、beds, surgical等、斜字体で表示)

医療インフラ

- 1000人あたり病床数：SH.MED.BEDS.ZS *beds*
- 10万人あたりの外科医数：SH.MED.SAOP.P5 *surgical*
- 国内一般政府保健支出(% of GDP)：SH.XPD.GHED.GD.ZS *d_health*
- 国内一般政府保健支出(% of 一般政府支出)：SH.XPD.GHED.GE.ZS *g_health*

国内的な接触程度

- 都市人口比率(% of 総人口)：SP.URB.TOTL.IN.ZS *urban*

国際的な接触程度

- インバウンド訪問者数：UIS.MSEP.56 *inb*
- 貿易額(% of GDP)：NE.TRD.GNFS.ZS *trade*
- 対外直接投資(% of GDP)：BX.KLT.DINV.WD.GD.ZS *fdi*

脆弱性

- 65歳以上人口(% of 総人口)：SP.POP.65UP.TO.ZS *pop65*
- 出生時の平均寿命, 総数(年)：SP.DYN.LE00.IN *life*

それではこうした指標をダウンロードすることにしましょう。主な手順は以下の通りです。

1. ジョン・ホプキンス大学CSSEからCOVID-19のデータを取得し、死者数トップ100の国を抽出
2. 世界銀行WDIから死者数トップ100の国について、上の指標をダウンロード
3. ダウンロードした世銀データの結合とデータの整理

(1) COVID-19のデータの取得と死者数トップ100の国を抽出

最初に、COVID-19のデータをジョン・ホプキンス大学CSSEのGitHubサイトからダウンロードする必要がありますが、データを整理された形でダウンロード可能にする便利なパッケージtidycovid19が開発されています。このパッケージを利用してダウンロードします。

このパッケージは開発者Joachim Gassen氏のGitHubサイトから直接取得する必要があります。GitHub上のRパッケージをインストールするためには、remotesパッケージを使います。そこでまずコンソール上に次のように入力し、remotesをインストールします。

```
install.packages("remotes")
```

次に、remotesを使ってtidycovid19パッケージをインストールします。次のように入力してください。

```
remotes::install_tidycovid19
```

次に、スクリプトに移り、tidycovid19を利用するために次のように入力します。

```
library(tidycovid19)
```

これでtidycovid19が利用可能となりました。それでは同パッケージのdownload_jhu_csse_covid19_data()関数を使ってデータをダウンロードし、それをオブジェクト covidに容れます。

```
covid <- download_jhu_csse_covid19_data()
```

これでデータが取得できました。head(covid)やView(covid)を使ってcovidの中を確認してみてください。次に死者数ランキングにもとづき、トップ100カ国を抽出します。

```

top100 <- covid %>% # パイプ以下の作業結果をオブジェクトtop100に容れる
group_by(country,iso3c) %>% # countryごと, iso3cごとにグループ化
summarise(total_death=max(deaths)) %>% # 死者数の最大値を抽出し, total_deatに容れる
arrange(-total_death) %>% # 最大死者数total_deathを基準に降順に並び替える
head(100) # トップ100を抽出

```

ジョン・ホプキンス大学のデータではアメリカの国名は“US”と表示されています。後で世銀のデータを利用するために、“United States”に置き換えておきます。データフレームtop100の第1行, 第1列目には“US”が入っています。そこで次のように入力し, 入れ代えます。

```
top100[1, 1] <- "United States"
```

世銀WDIから死者数トップ100の国の指標をダウンロードしますが, そのさい, top100の1列目に入っている国名を利用します。しかし, RパッケージWDIを使って世銀の指標をダウンロードするさい, 国名は2文字(もしくは3文字)の国コード表記を利用しなければなりません。そこでRパッケージcountrycodeを使って, top100の1列目の通常为国名表記を2文字の国コードに変更し, その列だけを抽出し, iso2に入れます。

```
iso2 <- countrycode(top100[[1]],origin='country.name',destination='iso2c')
```

これで死者数トップ100の国についてのみ世銀WDIから指標をダウンロードする準備ができました。いくつかに分けてダウンロードし, 適当な名前をつけたオブジェクトに容れておきます。データの期間はすべて期間は2011年~2018年です。

最初に, 医療インフラ関係の指標をダウンロードし, それをinfraに容れます。

```

infra <- WDI(country = iso2, indicator = c("SH.MED.BEDS.ZS", "SH.MED.SAOPP5", "SH.XPD.
GHED.GDZS", "SH.XPD.GHED.GE.ZS", "SP.POP.TOTL"), start = 2011, end = 2018, extra = FALSE,
cache = NULL) %>%
filter(SP.POP.TOTL >= 10000000)

```

最後の行でfilter(SP.POP.TOTL >= 10000000)を利用し, 人口1000万以上の国だけを抽出してい

ます。つまり、小国は除外します。

次に、国内の人々の相互の接触度に関連する指標をダウンロードし、dcontactに容れます。

```
dcontact <- WDI(country = iso2, indicator = c("SP.URB.TOTL.IN.ZS","SP.POP:TOTL" ), start
= 2011,end = 2018, extra = FALSE, cache = NULL) %>%
filter(SP.POP.TOTL >= 10000000)
```

次に、国際的な、人々の相互接触の程度を代理する指標をダウンロードし、excontactに容れます。

```
excontact <- WDI(country = iso2, indicator = c("UIS.MSEP.56","NE.TRD.GNFS.ZS","ST.INT.
ARVL","BX.KLT.DINV.WD.GD.ZS","SP.POP:TOTL"), start = 2011,end = 2018, extra = FALSE,
cache = NULL) %>%
filter(SP.POP.TOTL >= 10000000)
```

最後に、高齢化からみた社会の脆弱性に関連すると思われる指標をダウンロードし、agedに容れます

```
aged<-WDI(country = iso2, indicator = c("SP.POP.65UP.TO.ZS", "SP.DYN.IMRT", "SP.DYN.
LE00.IN","SP.POP:TOTL"), start = 2011,end = 2018, extra = FALSE, cache = NULL) %>%
filter(SP.POP.TOTL >= 10000000)
```

以上、分析に必要な指標のダウンロードは終わりです。次に、期間における個々の指標の変動・ばらつきを均すために、それぞれのデータフレームに格納された指標を期間全体について平均します。

```
infra<-infra %>% #パイプ%>%以後の操作を行った上でinfraに容れます。
group_by(iso2c,country) %>%
summarise(
  beds = mean(SH.MED.BEDS.ZS,na.rm = T), #世銀WDIの指標の平均をとります。
  surgical=mean(SH.MED.SAOP.P5,na.rm = T),
  d_health=mean(SH.XPD.GHED.GD.ZS,na.rm = T),
  g_health=mean(SH.XPD.GHED.GE.ZS,na.rm = T))
```

`group_by(iso2c,country)`によって`iso2,country`(結果は同じですので、この場合、1つでもかまいません)を基準にグループ化します。その上で、`summarise(beds = mean(SH.MED.BEDS.ZS,na.rm = T))`によって平均をとります。平均をとる関数`mean`(指標名)を利用します。平均をとるさい、`WDI`の変数名は長いので、変更しています。たとえば、`SH.MED.BEDS.ZS`の平均は`beds`という新しい変数名に変更しています。また、平均をとる際に、`na.rm=TRUE(Tでも可)`を入れています。これは欠損値`NA`がある場合、「無視してください」という意味です。これを指定せずに欠損値をいれたまま計算しようとすると、集約関数は通常の欠損値規則—すなわち入力に欠損値があれば、出力も欠損値—にしたがいがい、多数の欠損値を出してしまいます。

同じように、データフレーム`dcontact`, `excontact`, `aged`の中の指標についても`iso2c,country`変数を基準にグループ化し、指標の平均をとります。

```
dcontact <- dcontact %>%
  group_by(iso2c,country) %>%
  summarise(urban=mean(SP.URB.TOTL.IN.ZS,na.rm = T))
```

```
excontact <- excontact %>%
  group_by(iso2c,country) %>%
  summarise(inb=mean(UIS.MSEP.56,na.rm = T),
            trade=mean(NE.TRD.GNFS.ZS,na.rm = T),
            fdi=mean(BX.KLT.DINV.WD.GD.ZS,na.rm = T),
            inb=mean(ST.INT.ARVL,na.rm = T))
```

```
aged <- aged %>%
  group_by(iso2c,country) %>%
  summarise(pop65 = mean(SP.POP.65UP.TO.ZS,na.rm = T),
            life=mean(SP.DYN.LE00.IN,na.rm = T))
```

以上の4つのデータフレームを結合することにししましょう。そのために、`left_join()`を利用します。書式は次のようになります。

```
left_join(結合するデータフレーム名, 結合するさいのキー変数)
```

どのデータフレームにも同じiso2cとcountryが入っていますので、データフレームagedに左から順に、iso2c変数country変数を基準に結合しています。

```
soc_res <- aged %>%  
  left_join(dcontact,by=c("iso2c","country"))%>%  
  left_join(excontact,by=c("iso2c","country")) %>%  
  left_join(infra,c("iso2c","country")) %>%
```

この結合したデータセットはsoc_resという名前をつけたデータフレームに容れてあります。このsoc_resを主成分分析に利用するために、いったんcsvファイルとして保存しておきましょう。csv形式での保存のためのコードは次のようになります。

```
write.csv(データフレーム名, "保存csvファイル名")
```

保存するデータフレーム(データセット)の名前はsoc_resです。また、保存するcsvの名前としてsoc_res.csvとしておきます。

```
write.csv(soc_res, "soc_res.csv")
```

II. 3. レジリエンス指標の作成

それではII. 2で作成したデータセットに主成分分析を適用してみましょう。最初に、作成したデータセット(csvファイル)を読み込みます。読み込みのためにはread.csv()関数を使います。この関数のもっとも基本的な書式は

```
read.csv("読み込むcsvファイル名")
```

です。それではこの関数を使ってII. 2で作成したsoc_res.csvを読み込みます。

```
pca_soc <- read.csv("soc_res.csv", row.names=1)
```

ここではダウンロードしたデータセットをpca_socというデータフレームに容れます。また、

row.names=1の引数は行の名前に「csvファイルの1列めを利用しなさい」という命令です。このcsvファイルの1列めにはiso2国コードが入っていますので、行の名前が国コードで表現されます。

本資料では主成分分析にあたってはグラフ化に優れたFactoMineRというパッケージを利用します。これはさまざまな多変量解析を行う上でとても便利なパッケージです。それでは次のように入力し、インストールします。

```
install.packages("FactoMineR")
```

インストールを終えたら、library()で呼び出します。

```
library(FactoMineR)
```


FactoMineRパッケージを利用した主成分分析の書式は次のとおりです。

```
PCA(データフレームの変数)
```

引数を指定することでより詳細な設定を行うことができますが、ここではもっとも簡単な形を採用します。利用されるデータフレームpca_socの1列めの変数countryは国名を格納しているだけです。実際の観察値は2列めからです。そこでsoc[,2:10]で対象とする変数列を指示します。

```
res.soc <- PCA(pca_soc[,2:10])
```

データフレームにおいて特定の行や列を指定する方法は、データフレーム[行,列]です。ここでは行は何も指定していませんが、これは「すべての行」を対象にすることを意味します。また、列のコロン:は「～から～まで」という意味です。ここでは「2:10」となっていますので、「2列から10列まで」を指示しています。

また、主成分分析の結果を代入演算子<-を使ってres.socというオブジェクトに格納しています。実行すると、自動的に、RStudioの右下の画面がPlotsタブに代わり、図6のようなグラフが現れます。Plot画面の左上のをクリックすると、さらに、作成された他の図も見ることができます。

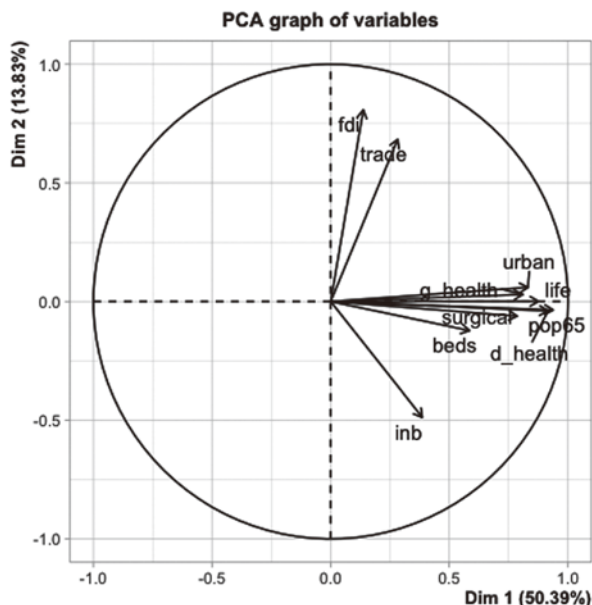


図6 主成分と変数の関連

この図においてx軸はDim1, y軸はDim2と表示されていますが、それぞれ第1主成分と第2主成分を表現しています。そしてカッコの中に50.39%, 13.83%と表記されています。これはすべての変数が提供する情報量のうち50.39%を第1主成分が説明し、13.83%を第2主成分が説明することを示しています。したがって2つの主成分によって全体の情報量の64.22%が説明されることになります。言い換えれば、全体の情報の64.22%がこの2つの主成分に縮約されていると言えます。

わたしたちは複雑な変数の空間(この例では10次元の空間)を理解できませんが、その空間が持つ情報量をほとんど失うことなく—というのも64.22%を保持していますから—2次元に縮約することができました。これでこのデータセットで表現された世界を容易に理解することができるようになりました。

主成分分析の重要なポイントは、データ全体の情報量の64.22%を縮約した、この2つの主成分が何を意味しているのか、それを解釈することにあります。主成分の解釈に役立つのが、上の図6です。この図に描かれた矢印はその長さや方向によって各主成分とオリジナルな変数の相関を示しています。これを見ると、第1主成分は高齢化率(pop65)、平均寿命(life)といった社会的脆弱性を表現する変数、国内において人々の相互の接触程度を表現する都市化(urban)と高い相関を示しています。したがって第1主成分はCOVID-19に対する潜在的な国内リスクを表現していると解釈できるようです。

他方で第1主成分は対GDP比率保健支出(d_health), 政府支出に占める保健支出(g_health), 外科医数(surgical), 病床数(beds)とも高い相関を示しています。これらは医療インフラの程度を示していると考えられますから, この点に注目すると, 第1主成分は医療インフラを表現していると解釈することもできます。したがってこの第1主成分は, 潜在的な国内リスクと医療インフラの強さを表現していると解釈できます。

他方, 第2主成分は対GDP貿易額(trade), GDPに対する対外直接投資, 純流入(fdi), および国際ツーリズム訪問者数(inb)と高い相関を示しています。より詳細にみると, プラスの側はfdi,tradeと相関しています。他方, マイナスの側はinbと相関しています。したがって第2主成分のプラス側はモノ・サービスのグローバリゼーションおよび生産のグローバリゼーションをつうじた潜在的な接触リスク, マイナスの側はヒトのグローバリゼーションをつうじた潜在的な接触リスクを表現していると言えます。したがってこの第2主成分はCOVID-19に対する対外的な潜在的リスクを表現していると解釈できます。

図6の視覚的な表現にもとづいて主成分を解釈しましたが, dimdesc()関数を使うと, 主成分と各変数の相関およびp-値を得ることができ, 解釈を補強することができます。次のように入力し, 実行してみてください。これにより, 図7の結果が出力されます。

```
dimdesc(res.pca)
```

```
> dimdesc(res.pca)
$Dim.1
$quanti
      correlation      p.value
pop65    0.9371466 4.802600e-30
d_health  0.9159601 2.843511e-26
life      0.8757777 2.807791e-21
urban     0.8312348 1.871183e-17
g_health  0.8091640 5.952954e-16
surgical  0.7858909 1.451161e-14
beds      0.5856520 3.706382e-07
inb       0.3864974 1.607178e-03
trade     0.2823535 2.378907e-02

attr("class")
[1] "condes" "list "

$Dim.2
$quanti
      correlation      p.value
fdi     0.8071487 7.983421e-16
trade   0.6833443 4.914530e-10
inb     -0.4880143 4.303464e-05
```

図7 dimdesc()を使ったオブジェクトres.pcaの結果表示。

以上のように、主成分とオリジナルな変数との相関にもとづき、第1主成分はCOVID-19に対する潜在的な国内リスクと医療インフラの強さ、第2主成分はCOVID-19に対する対外的な潜在的リスクを表現すると解釈しました。ここでは主成分分析によって得た要約指標—つまり2つの主成分—を社会経済的レジリエンス指標と呼ぶことにします。

次に、こうした社会経済的レジリエンス指標—2つの主成分—にもとづき各国を評価してみましょう。このためには、各国の主成分上の位置・座標(主成分得点と呼ばれます)を計算する必要があります。RパッケージFactoMineRではこれを自動的に計算します。さらにこのパッケージは、2つの主成分を軸とする座標平面上に各国(の得点)を自動的にプロットしてくれます。その結果が次の図8です。

図8は第1主成分と第2主成分を座標平面とし、各国を同平面上にプロットしています。これにより各国を比較したり、その特徴を明らかにすることができます。10次元の世界(ここでの例は10の変数)ではとても無理なことでしたが、今では2次元の世界へと縮約され、容易に評価・比較することができます。

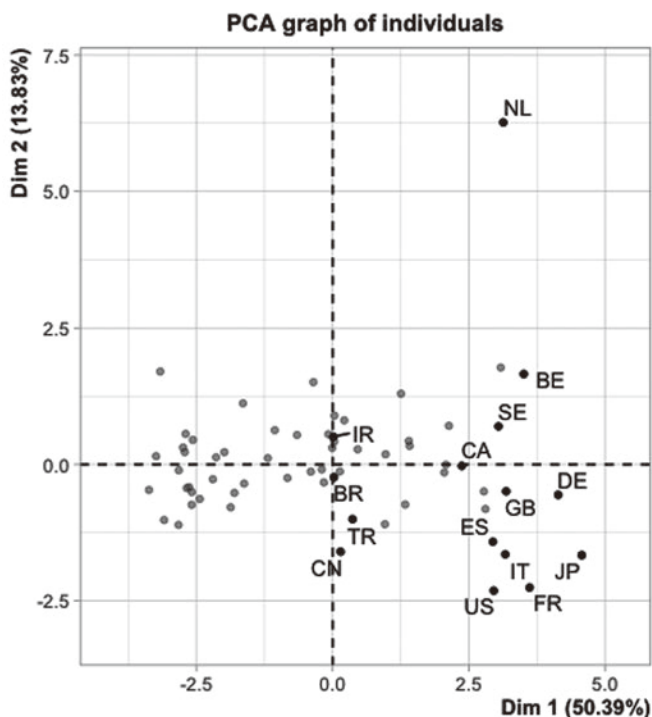


図8 社会経済的レジリエンス

注. 日本と死者数トップ14カ国の合計15カ国のみ国名(iso2コード)を表示. US(アメリカ合衆国), IT(イタリア), GB(イギリス), ES(スペイン), FR(フランス), BE(ベルギー), BR(ブラジル), DE(ドイツ), IR(イラン), NL(オランダ), CN(中国), CA(カナダ), TR(トルコ), SE(スウェーデン)

図8には死者数トップ14カ国と日本のみ国名(2文字国コード表記)を表示しています。容易に理解されるように、第4象限に位置する国の多くがCOVID-19の犠牲者を多く出していることが理解できます。上述の主成分の解釈にもとづくと、この第4象限に位置する経済は、第1主成分の観点から評価すれば、潜在的な国内リスクと医療インフラの強さ、第2主成分からみれば、対外的な潜在的リスクを抱える国です。したがって、そうした国々は、医療インフラが強いにもかかわらず、国内的にも対外的にもリスクを抱えており、そのかぎりではCOVID-19に対する社会的経済的レジリエンスが脆弱な国である、と言えます。

IV. COVID-19と社会経済的レジリエンスの関連

最後に、ここで作成した社会経済的レジリエンス指標とCOVID-19の関連を見ることします。ここでは単純に、死亡者数と社会経済的レジリエンス指標の関連の分析—相関と回帰—を行います。

レジリエンス指標—すなわち主成分分析の結果得られる各国の主成分得点—はオブジェクト `res.soc` の中にあります。以下のように入力し、出力させることができます。`$`は「~の中の」という意味です。したがって `res.soc$ind` でデータフレーム `res.soc` 中の `ind` を指定しています。さらに、`$` でつなげることによって `ind` 中の `coord` (主成分得点) を指定します。

```
res.soc$ind$coord
```

コンソール画面に、上のコードを入力すると、第5主成分(`dim1`から`dim5`)が出力されます。今、必要なのは第2主成分までです。そこでデータフレームの行列指定[行,列]を使って、`res.pcaindcoord[,1:2]`とします。そうすると、第2主成分(`dim2`)まで出力されます。ただし、これによって出力されたデータは行列形式ですので、次のように`as.data.frame()`関数を使ってデータフレームに変換し、`reslience`に格納します。

```
reslience <- as.data.frame(res.pca$ind$coord[,1:2])
```

このレジリエンス指標を各国の感染者数と死亡者数データが入ったデータフレーム `top100` と結合します。`iso2`コードを基準に結合するために、さらに、パイプを使って新たな変数を作成しておきます。つまり、`reslience`の行の名前が国コード(`iso2`)になっていますが、これを変数—この例では`iso`という変数名にしています—にします。

```
reslience <- mutate(reslience, iso=row.names(reslience))
```

select()関数を使ってデータフレーム top100のうちcountry, iso3c, total_death, total_confirmed変数を抽出します。

```
top100 <- top100 %>%  
  select(country, iso3c, total_death, total_confirmed)
```

次に、2つのデータフレームの結合の基準となる、Rパッケージ countrycodeを利用し、top100の中の変数countryを2文字の国コードに変換し、データフレーム isoに容れます。

```
iso <- as.data.frame(countrycode(top100[[2]],origin='iso3c',destination='iso2c'))
```

列の長さが同じですので、ここではcbind()関数を使って2つのデータフレーム—top100とiso—を結合します。

```
top100 <- cbind(top100, iso)
```

データフレームのisoの2文字表記国コードが入っている変数名が長くなっていますので、rename()関数を使ってtop100の5列目top100[[5]]の列名をisoという名前の変数名に書き換えます。

```
top100 <- rename(top100, iso= colnames(top100[5]))
```

renameのコードは

```
rename (データフレーム名, 新たな変数名=古い変数名)
```

です。ここではcolnames()関数を使ってデータフレーム top100の5列目—[5]—の列名を指示しています。

最後に、iso変数を基準にデータフレーム top100とデータフレーム reslienceを結合します。

```
top100 <- top100 %>%
  left_join(resilience,by="iso")
```

これで分析のためのデータフレームが出来上がりました。それでは死者数と社会的レジリエンスの関連を散布図—以下の図9—に描いてみます。

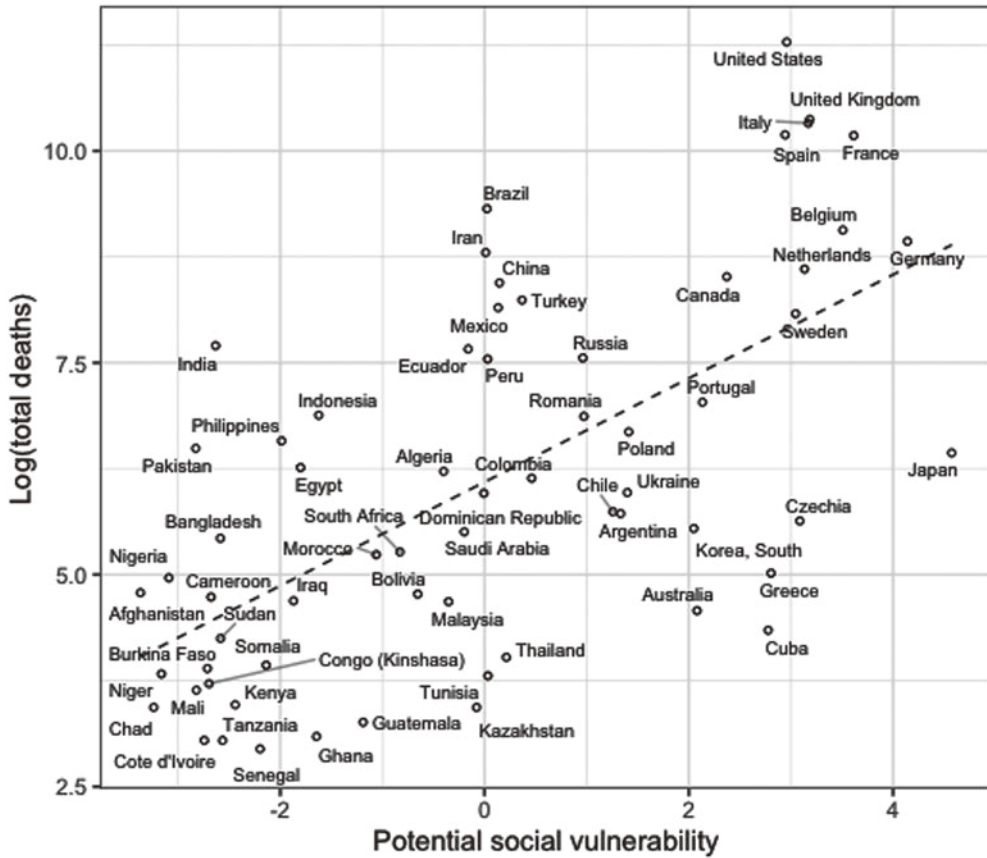


図9 COVID-19の死者数と社会経済的レジリエンス
 注. y軸の死者数は対数表示, x軸は主成分分析から得た各国の第1主成分の得点.

図9には回帰線も描かれていますが、容易に理解されるように、社会的脆弱性が高い国ほど、高い死亡数を生み出しています。この効果を確認するために回帰分析を行っておきましょう。

```
res.top100 <- lm(log(top100$total_death)~top100$Dim.1)
```

この分析では、死亡者数を社会的レジリエンス指標に回帰させ、その結果をres.top100に格納しています。summary()関数によって結果を表示してみましょう。

```
summary(res.top100)
```

この結果はコンソール画面に次のように出力されます。

```
> summary(res.top100)

Call:
lm(formula = log(top100$total_death) ~ top100$Dim.1)

Residuals:
    Min       1Q   Median       3Q      Max
-3.4474 -1.1477 -0.2471  1.3125  3.3820

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   6.0927     0.2126  28.66 < 2e-16 ***
top100$Dim.1  0.6117     0.0947   6.46 1.84e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.701 on 62 degrees of freedom
(36 observations deleted due to missingness)
Multiple R-squared:  0.4023,    Adjusted R-squared:  0.3926
F-statistic: 41.73 on 1 and 62 DF,  p-value: 1.835e-08
```

図10 潜在的な社会的脆弱性が死亡者数に与える効果

単回帰ですが³、社会的レジリエンス指標が³、死亡者数に有意な正の効果を与えていることが確認されます。したがって社会的レジリエンスが弱くなると、死亡者数の増加を招くことが理解されます。

【参考文献一覧】

Bayer, C. and Kuhn, M. (2020) Intergenerational Ties and Case Fatality Rates: A Cross-Country Analysis, IZA DP No. 13114.

Fukui, M., Kawaguchi, K., and Matsuura, H. (2020) Does TB Vaccination Reduce COVID-19 Infection? No Evidence from a Regression Discontinuity Analysis, mimeo.

Miller, A., Reandelar, M.J., Fasciglione, K., Roumenva, V., Li, Y., and Otazu, G.H. (2020) Correlation between universal BCG vaccination policy and reduced morbidity and mortality for COVID-19: an epidemiological study, mimeo.

Ilan Noy, Nguyen Doan, Benno Ferrarini and Donghyun Park (2020) Measuring the economic risk of Covid-19, *COVID Economics*, 3, pp. 103-118.

中山慶一郎(2011)「研究ノート 探索的多変量解析によるデータ解析—主成分分析法」関西大学社会学部紀要, 113, pp. 45-58.

中山慶一郎(2014)「研究ノート 多因子分析による探索的多変量データ分析」関西大学社会学部紀要, 118, pp. 53-78.

【スクリプト一覧】

```
# =====
# 図1を描くためのスクリプト
# =====
# 利用するパッケージのインストール.
install.packages("ggrepel")      # グラフのテキストの重なりを修正するパッケージ.
install.packages("remotes")     # GitHubサイトからパッケージを読み込むためのパッケージ.
remotes::install_tidycovid19

# 利用するパッケージの読み込み.
library(tidyverse)
library(ggrepel)
library(tidycovid19)
library(WDI)
library(readr)
library(countrycode)

# パッケージtidycovid19を利用して、ジョンホプキンス大学CSSEよりCOVID-19のデータを取得.
covid<-download_jhu_csse_covid19_data()

# 死者数の多い順に各国を並べ、トップ100を抽出.
top100<-covid %>%
  group_by(country,iso3c) %>%
  summarise(total_death=max(deaths)) %>%
```

```
arrange(-total_death) %>%
head(100)

top100[1, 1] <- "United States"

# 世銀WDIから人口データを取得し、人口100万人未満の国を除外する。
iso2<-countrycode(top100[[1]],origin='country.name',destination='iso2c') # トップ100の国名を
iso2に容れる。
pop<-WDI(country = iso2, indicator = "SP.POP.TOTL", start = 2018, end = 2018, extra = FALSE,
cache = NULL)
pop<-pop %>%
  filter(SP.POP.TOTL >= 1000000) # 人口100万未満の国を除外。

# "The BCG world atlas"(http://www.bcgatlas.org/index.php)にもととづき変数BCGを作成。
non_bcg<-c("AUS", "FIN", "NOR", "SWE", "GBR", "GER", "AUS", "FRA", "CHE", "SVN", "SVK", "USA",
"CAN")
# BCGの全国的な接種を行っていない国の国コードをnon_bcgベクトルに容れる。

top100 <- top100 %>%
  mutate(BCG=ifelse(iso3c %in% non_bcg, "No_BCG", "Yes")) # 変数BCGを作成し、top100に追
  加する。
# ifelse関数を利用し、iso3cの国コードがnon_bcgの中の国コードのどれかと一致する場合は "No_
BCG", 一致しない場合は "Yes" を持つ、変数BCGを作成。

# パッケージggplot2を使って図1を描く。
fig1 <- ggplot(top100,aes(log(total_confirmed), log(total_death),label=iso3c))+ #log()関数を利用し、
対数表示。
  scale_shape_manual(values = c(15, 1))+
  geom_point(aes(shape=BCG),size=1.8)+
  geom_text_repel(segment.alpha = 0.2)+ # テキストの重なりをおさえる。 segment.alpahで線
  の濃度を指定。
  labs(x="ln(confirmed cases)", y= "ln(deaths)")+
```

```

stat_smooth(method=lm, se=F, lty="dashed", lwd=0.4, col=1) +
theme_bw()+
annotate("text",x=7.5,y=10.5, label="No_BCG countries:\nBCG recommendation only for specific
group, \nand past national BCG vaccination policy for all.")

```

fig1 # 図1 を出力.

```

# =====
# 図2 データセット(データフレーム)の作成スクリプト
# =====
# 世銀WDIから医療インフラ関連の指標をダウンロード.
infra<-WDI(country = iso2, indicator = c("SH.MED.BEDS.ZS", "SH.MED.SAOPP5", "SH.XPD.GHED.
GD.ZS", "SH.XPD.GHED.GE.ZS", "SP.POP.TOTL"), start = 2011, end = 2018, extra = FALSE, cache
= NULL) %>% # exclude counries wiht pop under one million
  filter(SP.POP.TOTL >= 10000000)

# 世銀WDIから国内の接触程度に関連する指標をダウンロード.
dcontact<-WDI(country = iso2, indicator = c("SP.URB.TOTL.IN.ZS", "SP.URB.LCTY.UR.ZS", "SP.
URB.LCTY.UR.ZS", "IN.TRANSPORT.URBNRD.DENSIT", "SP.POP.LAND.ZS", "SP.POP.TOTL" ),
start = 2011, end = 2018, extra = FALSE, cache = NULL) %>% # exclude counries wiht pop under
one million
  filter(SP.POP.TOTL >= 10000000)

# 世銀WDIから対外的接触程度に関連する指標をダウンロード.
excontact<-WDI(country = iso2, indicator = c("UIS.MSEP.56", "NE.TRD.GNFS.ZS", "ST.INT.ARVL",
"BX.KLT.DINV.WD.GD.ZS", "SP.POP.TOTL"), start = 2011, end = 2018, extra = FALSE, cache = NULL)
%>% # exclude counries wiht pop under one million
  filter(SP.POP.TOTL >= 10000000)

# 世銀WDIから社会の脆弱性の程度に関連する指標をダウンロード.
aged<-WDI(country = iso2, indicator = c("SP.POP.65UP.TO.ZS", "SP.DYN.IMRT", "IN.POV.INF.
MORTRATE", "SP.DYN.LE00.IN", "SP.POP.TOTL"), start = 2011, end = 2018, extra = FALSE, cache

```



```
= NULL) %>% # exclude counries wiht pop under one million
  filter(SP.POP.TOTL >= 10000000)

# iso2c,countryに応じてグループ化し，各変数の期間全体に関して平均をとる.
aged<-aged%>%
  select(-IN.POV.INF.MORTRATE) %>%
  group_by(iso2c,country)%>%
  summarise(pop65 = mean(SP.POP.65UP.TO.ZS,na.rm = T),
            life=mean(SP.DYN.LE00.IN,na.rm = T))

dcontact<-dcontact %>%
  select(-IN.TRANSPORT.URBNRD.DENSIT)%>%
  group_by(iso2c,country) %>%
  summarise(urban=mean(SP.URB.TOTL.IN.ZS))

excontact<-excontact %>%
  group_by(iso2c,country) %>%
  summarise(inb=mean(UIS.MSEP.56),
            trade=mean(NE.TRD.GNFS.ZS),
            fdi=mean(BX.KLT.DINV.WD.GD.ZS),
            inb=mean(ST.INT.ARVL))

infra<-infra %>%
  group_by(iso2c,country)%>%
  summarise(beds = mean(SH.MED.BEDS.ZS,na.rm = T),
            surgent=mean(SH.MED.SAOP.P5,na.rm = T),
            d_health=mean(SH.XPD.GHED.GD.ZS,na.rm = T),
            g_health=mean(SH.XPD.GHED.GE.ZS,na.rm = T))

# 4つのデータセットを結合し，soc_resオブジェクトに容れる.
soc_res<-aged %>%
  left_join(dcontact,by=c("iso2c","country"))%>%
```

```

left_join(excontact,by=c("iso2c","country")) %>%
left_join(infra,by=c("iso2c","country"))

# データフレーム soc_res の中身を表示.
View(soc_res)

# データを csv ファイルとして保存.
write_csv(soc_res,"soc_res.csv")

# =====
# 図3を作成するためのスクリプト
# =====
# データフレーム soc_res の変数 pop65 について 10% 以上の観察値だけを抽出.
over10pop<-soc_res %>%
  filter(pop65>=10)

# ggplot2 を使って グラフ を作成.
fig3_pop65<-ggplot(over10pop,aes(reorder(country,desc(-pop65)),pop65))+
  geom_point()+
  coord_flip()+
  labs(
    y="Population ages 65 and above (% of total)",
    x=""
  )+
  theme_bw()
fig3_pop65 # 図3 を出力.

# =====
# 図4を作成するためのスクリプト
# =====
fig4_life <- ggplot(soc_res,aes(pop65,life))+
  geom_point()+

```

```
geom_text_repel(aes(label=country),data=over10pop,segment.alpha = 0.3)+
labs(
  x="Population ages 65 and above (% of total)",
  y="Life expectancy at birth, total (years)"
)+
theme_bw()
```

geom_text_repel()を使ってデータ点に国ラベルをつける。aes(label=country)で指示、そのさい、65歳以上人口比率10%を超える国だけを表示するために、図3で作ったover10popを利用する。これはdata=over10popで指示する。

```
fig4_life      # 図4 を出力.
```

```
# =====
```

```
# 図5 を作成するためのスクリプト
```

```
# =====
```

```
install.packages("scatterplot3d") # 3次元プロットを描くためのパッケージをインストール.
library(scatterplot3d)
```

scatterplot3d()関数を使って3次元グラフを描く。基本的な書式はscatterplot3d(x軸に利用する変数, y軸に利用する変数, z軸に利用する変数)である。

```
fig5_3d <- scatterplot3d(soc_res$pop65,soc_res$life,soc_res$urban,xlab = "Population ages 65 and
above (% of total)",ylab = "Life expectancy at birth, total (years)",zlab = "Urban population (% of
total)",box=F,pch=1)
```

```
fig5_3d
```

```
# =====
```

```
# 図8 を作成するためのスクリプト
```

```
# =====
```

```
# 主成分分析を実行し、その結果をres.pcaに容れる.
```

```
res.pca<-PCA(pca_soc[,2:11])
```

主成分分析の結果をグラフ化するplot.PCA()関数を利用。基本的な書式はplot.PCA(主成分分析の結果を容れたデータフレーム, x軸y軸の指定, 個別の国を表示させる指定, 表示させる国名を選択)。

```
plot.PCA(res.pca,axes = c(1, 2),choix = "ind",cex=0.9,select = c("US","IT","GB","ES","FR","BE",
"BR","DE","IR","NL","CN","CA","TR","SE","JP"))
```

```
# axes= x軸y軸の指定.
```

```
# choix = "ind" "ind" によって個々の国を表示させることを指示.
```

```
# select = 表示国を指定.
```

```
# =====
```

```
# 図9を作成するためのスクリプト
```

```
# =====
```

```
reslience.fig <-ggplot(top100,aes(Dim.1,log(total_death)))+
  geom_point(shape=21,size=1)+
  geom_text_repel(aes(label=country),segment.alpha = 0.3,cex=2.5)+
  stat_smooth(method=lm, se=F,lty="dashed",lwd=0.4, col=1) +
  labs(
    x = "Potential social vulnerability",
    y = "Log(total deaths)"
  )+
  theme_bw()
```

```
reslience.fig      # 図9 の出力.
```

【パッケージ一覧】

- tidycovid19
 - ジョンホプキンス大学CSEEより新型コロナウイルスデータをtidy形式でダウンロードするパッケージ.
- WDI
 - 世界銀行WDI指標をダウンロードするためのパッケージ.

- countrycode
 - 国名, iso国コードを変換するためのパッケージ.
- FactoMineR
 - 多変量解析のさまざまな分析手法を実行するパッケージ.
- ggrepel
 - グラフ上のテキストの重なりを修正するパッケージ.