CODES\_cases\_deaths\_pm.R

user

2020-05-15

rm(list = ls(all.names = TRUE))

library(Amelia)

## Loading required package: Rcpp

## ##
## ## Amelia II: Multiple Imputation
## ## (Version 1.7.6, built: 2019-11-24)
## ## Copyright (C) 2005-2020 James Honaker, Gary King and Matthew Blackwell
## ## Refer to http://gking.harvard.edu/amelia/ for more information
## ##

library(bestNormalize)
library(openxlsx)
library(lubridate)

##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
## date

library(purrr)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:lubridate':
##
## intersect, setdiff, union

## The following objects are masked from 'package:stats':
##
## filter, lag

## The following objects are masked from 'package:base':
##
## intersect, setdiff, setequal, union

library(nlme)

##
## Attaching package: 'nlme'

## The following object is masked from 'package:dplyr':
##
## collapse

library(rsq)
library(lm.beta)
library(tidyverse)

## -- Attaching packages -------------------------------------------------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.0 v readr 1.3.1
## v tibble 2.1.3 v stringr 1.4.0
## v tidyr 1.0.2 v forcats 0.4.0

## -- Conflicts ----------------------------------------------------------------------------------- tidyverse\_conflicts() --
## x lubridate::as.difftime() masks base::as.difftime()
## x nlme::collapse() masks dplyr::collapse()
## x lubridate::date() masks base::date()
## x dplyr::filter() masks stats::filter()
## x lubridate::intersect() masks base::intersect()
## x dplyr::lag() masks stats::lag()
## x lubridate::setdiff() masks base::setdiff()
## x lubridate::union() masks base::union()

library(caret)

## Loading required package: lattice

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
## lift

library(leaps)
library(ggplot2)
library(reghelper)

##
## Attaching package: 'reghelper'

## The following object is masked from 'package:base':
##
## beta

library(ggpubr)

## Loading required package: magrittr

##
## Attaching package: 'magrittr'

## The following object is masked from 'package:tidyr':
##
## extract

## The following object is masked from 'package:purrr':
##
## set\_names

library(car)

## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
## recode

## The following object is masked from 'package:purrr':
##
## some

library(missMDA)

## Registered S3 methods overwritten by 'lme4':
## method from
## cooks.distance.influence.merMod car
## influence.merMod car
## dfbeta.influence.merMod car
## dfbetas.influence.merMod car

library(psych)

##
## Attaching package: 'psych'

## The following object is masked from 'package:car':
##
## logit

## The following object is masked from 'package:reghelper':
##
## ICC

## The following objects are masked from 'package:ggplot2':
##
## %+%, alpha

library(knitr)

setwd("C:/Users/user/Dropbox/BCG-COVID-19/Ver 3/TF\_SG")

CV19 <- read.xlsx("Table\_S1-9.xlsx")

DATA <- CV19[, which (colnames(CV19) %in% c(
"Country\_code",
"BCG.TB",
"cases\_per\_million\_UP",
"deaths\_per\_million\_UP",
#"Positive.cases.per.test",
"temp",
"BCG.type","Current.BCG.vaccination",
"prevalence\_overweight\_2012",
"Fraction\_urban\_population",
"Population\_ages\_65\_and\_above\_2018\_percent",
"TB.incidents.per.10.thosand.mean\_2000\_2018",
"TB.Incidents.100"))]

DATA\_COMPLETE <- na.omit(DATA)
dim(DATA\_COMPLETE)

## [1] 117 12

DATA\_COMPLETE$cpm\_N <- predict(orderNorm(DATA\_COMPLETE$cases\_per\_million\_UP))
hist(DATA\_COMPLETE$cpm\_N)



DATA\_COMPLETE$cpm\_Z <- scale(DATA\_COMPLETE$cases\_per\_million\_UP)[,1]
hist(DATA\_COMPLETE$cpm\_Z)



DATA\_COMPLETE$dpm\_N <- predict(orderNorm(DATA\_COMPLETE$deaths\_per\_million\_UP))

## Warning in orderNorm(DATA\_COMPLETE$deaths\_per\_million\_UP): Ties in data, Normal distribution not guaranteed

hist(DATA\_COMPLETE$dpm\_N)



DATA\_COMPLETE$dpm\_Z <- scale(DATA\_COMPLETE$deaths\_per\_million\_UP)[,1]
hist(DATA\_COMPLETE$dpm\_Z)



DATA\_COMPLETE$TB\_N <- predict(orderNorm(DATA\_COMPLETE$TB.incidents.per.10.thosand.mean\_2000\_2018))
hist(DATA\_COMPLETE$TB\_N)



DATA\_COMPLETE$TB\_Z <- scale(DATA\_COMPLETE$TB.incidents.per.10.thosand.mean\_2000\_2018)[,1]

DATA\_COMPLETE$Ob\_N <- predict(orderNorm(DATA\_COMPLETE$prevalence\_overweight\_2012))

## Warning in orderNorm(DATA\_COMPLETE$prevalence\_overweight\_2012): Ties in data, Normal distribution not guaranteed

hist(DATA\_COMPLETE$Ob\_N)



DATA\_COMPLETE$temp\_N <- predict(orderNorm(DATA\_COMPLETE$temp))

## Warning in orderNorm(DATA\_COMPLETE$temp): Ties in data, Normal distribution not guaranteed

hist(DATA\_COMPLETE$temp\_N)



DATA\_COMPLETE$frac\_N <- predict(orderNorm(DATA\_COMPLETE$Fraction\_urban\_population))
hist(DATA\_COMPLETE$frac\_N)



DATA\_COMPLETE$age\_N <- predict(orderNorm(DATA\_COMPLETE$Population\_ages\_65\_and\_above\_2018\_percent))
hist(DATA\_COMPLETE$age\_N)



#DATA\_COMPLETE$positives\_N <- predict(orderNorm(DATA\_COMPLETE$Positive.cases.per.test))
#hist(DATA\_COMPLETE$positives\_N)

DATA\_COMPLETE$Yy <- ifelse(DATA\_COMPLETE$BCG.TB == "Y\_y", 1, 0)
DATA\_COMPLETE$Yn <- ifelse(DATA\_COMPLETE$BCG.TB == "Y\_n", 1, 0)
DATA\_COMPLETE$Nn <- ifelse(DATA\_COMPLETE$BCG.TB == "N\_n", 1, 0)

###Analysis###

#cases

cases\_lm <- lm(cpm\_N ~ Current.BCG.vaccination + TB\_Z +
Ob\_N + temp\_N + age\_N + frac\_N
, data = DATA\_COMPLETE)

summary(cases\_lm, corr=T)

##
## Call:
## lm(formula = cpm\_N ~ Current.BCG.vaccination + TB\_Z + Ob\_N +
## temp\_N + age\_N + frac\_N, data = DATA\_COMPLETE)
##
## Residuals:
## Min 1Q Median 3Q Max
## -1.53617 -0.34046 0.02929 0.36210 1.86071
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.445651 0.155347 2.869 0.004942 \*\*
## Current.BCG.vaccinationY -0.543140 0.176168 -3.083 0.002590 \*\*
## TB\_Z -0.150719 0.076175 -1.979 0.050362 .
## Ob\_N -0.006977 0.091360 -0.076 0.939264
## temp\_N -0.095208 0.073477 -1.296 0.197774
## age\_N 0.238983 0.082305 2.904 0.004458 \*\*
## frac\_N 0.360750 0.090349 3.993 0.000118 \*\*\*
## ---
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6155 on 110 degrees of freedom
## Multiple R-squared: 0.6399, Adjusted R-squared: 0.6202
## F-statistic: 32.57 on 6 and 110 DF, p-value: < 2.2e-16
##
## Correlation of Coefficients:
## (Intercept) Current.BCG.vaccinationY TB\_Z Ob\_N
## Current.BCG.vaccinationY -0.93
## TB\_Z 0.01 -0.01
## Ob\_N -0.04 0.05 0.33
## temp\_N 0.08 -0.08 -0.02 0.24
## age\_N -0.24 0.26 0.29 0.23
## frac\_N -0.13 0.14 0.07 -0.61
## temp\_N age\_N
## Current.BCG.vaccinationY
## TB\_Z
## Ob\_N
## temp\_N
## age\_N 0.45
## frac\_N -0.12 -0.26

shapiro.test(resid(cases\_lm)) #List of residuals

##
## Shapiro-Wilk normality test
##
## data: resid(cases\_lm)
## W = 0.98649, p-value = 0.2948

plot(density(resid(cases\_lm))) #A density plot



qqnorm(rstandard(cases\_lm))
qqline(rstandard(cases\_lm))



rsq.partial(cases\_lm, adj=TRUE)

## $adjustment
## [1] TRUE
##
## $variable
## [1] "Current.BCG.vaccination" "TB\_Z"
## [3] "Ob\_N" "temp\_N"
## [5] "age\_N" "frac\_N"
##
## $partial.rsq
## [1] 0.071171845 0.025587762 -0.009037409 0.006079646 0.062746007
## [6] 0.118648170

lm.beta(cases\_lm)

##
## Call:
## lm(formula = cpm\_N ~ Current.BCG.vaccination + TB\_Z + Ob\_N +
## temp\_N + age\_N + frac\_N, data = DATA\_COMPLETE)
##
## Standardized Coefficients::
## (Intercept) Current.BCG.vaccinationY TB\_Z
## 0.000000000 -0.209577498 -0.150895790
## Ob\_N temp\_N age\_N
## -0.006976627 -0.095203018 0.238983276
## frac\_N
## 0.360749632

Anova(cases\_lm)

## Anova Table (Type II tests)
##
## Response: cpm\_N
## Sum Sq Df F value Pr(>F)
## Current.BCG.vaccination 3.602 1 9.5054 0.0025900 \*\*
## TB\_Z 1.483 1 3.9148 0.0503616 .
## Ob\_N 0.002 1 0.0058 0.9392640
## temp\_N 0.636 1 1.6790 0.1977739
## age\_N 3.195 1 8.4311 0.0044585 \*\*
## frac\_N 6.041 1 15.9429 0.0001182 \*\*\*
## Residuals 41.679 110
## ---
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#selection of predictors

models\_cases <- regsubsets(cpm\_N ~ Current.BCG.vaccination + TB\_Z +
Ob\_N + temp\_N + age\_N + frac\_N
, data = DATA\_COMPLETE, nvmax = 9)

summary(models\_cases)

## Subset selection object
## Call: regsubsets.formula(cpm\_N ~ Current.BCG.vaccination + TB\_Z + Ob\_N +
## temp\_N + age\_N + frac\_N, data = DATA\_COMPLETE, nvmax = 9)
## 6 Variables (and intercept)
## Forced in Forced out
## Current.BCG.vaccinationY FALSE FALSE
## TB\_Z FALSE FALSE
## Ob\_N FALSE FALSE
## temp\_N FALSE FALSE
## age\_N FALSE FALSE
## frac\_N FALSE FALSE
## 1 subsets of each size up to 6
## Selection Algorithm: exhaustive
## Current.BCG.vaccinationY TB\_Z Ob\_N temp\_N age\_N frac\_N
## 1 ( 1 ) " " " " " " " " " " "\*"
## 2 ( 1 ) " " " " " " " " "\*" "\*"
## 3 ( 1 ) "\*" " " " " " " "\*" "\*"
## 4 ( 1 ) "\*" "\*" " " " " "\*" "\*"
## 5 ( 1 ) "\*" "\*" " " "\*" "\*" "\*"
## 6 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*"

res.sum <- summary(models\_cases)
data.frame(
 Adj.R2 = which.max(res.sum$adjr2),
 CP = which.min(res.sum$cp),
 BIC = which.min(res.sum$bic)
)

## Adj.R2 CP BIC
## 1 5 4 4

coef(models\_cases, 1:5)

## [[1]]
## (Intercept) frac\_N
## 4.857226e-17 6.686202e-01
##
## [[2]]
## (Intercept) age\_N frac\_N
## 4.857226e-17 4.159677e-01 4.733944e-01
##
## [[3]]
## (Intercept) Current.BCG.vaccinationY age\_N
## 0.4761734 -0.5803363 0.3331622
## frac\_N
## 0.4184520
##
## [[4]]
## (Intercept) Current.BCG.vaccinationY TB\_Z
## 0.4636196 -0.5650364 -0.1591164
## age\_N frac\_N
## 0.2838967 0.3598720
##
## [[5]]
## (Intercept) Current.BCG.vaccinationY TB\_Z
## 0.44515008 -0.54252901 -0.14880911
## temp\_N age\_N frac\_N
## -0.09387861 0.24045172 0.35657391

vcov(models\_cases, 5)

## (Intercept) Current.BCG.vaccinationY TB\_Z
## (Intercept) 0.0238737392 -0.0251845522 0.0002909193
## Current.BCG.vaccinationY -0.0251845522 0.0306936428 -0.0003545718
## TB\_Z 0.0002909193 -0.0003545718 0.0051309063
## temp\_N 0.0009935842 -0.0012108044 -0.0005544932
## age\_N -0.0029703295 0.0036201475 0.0013131724
## frac\_N -0.0021759447 0.0026519370 0.0018470909
## temp\_N age\_N frac\_N
## (Intercept) 0.0009935842 -0.0029703295 -0.0021759447
## Current.BCG.vaccinationY -0.0012108044 0.0036201475 0.0026519370
## TB\_Z -0.0005544932 0.0013131724 0.0018470909
## temp\_N 0.0050502915 0.0023371623 0.0001774219
## age\_N 0.0023371623 0.0063470198 -0.0008829459
## frac\_N 0.0001774219 -0.0008829459 0.0051268816

cases\_lm\_4 <- lm(cpm\_N ~ Current.BCG.vaccination + TB\_Z +
age\_N + frac\_N, data = DATA\_COMPLETE)

summary(cases\_lm\_4, corr=T)

##
## Call:
## lm(formula = cpm\_N ~ Current.BCG.vaccination + TB\_Z + age\_N +
## frac\_N, data = DATA\_COMPLETE)
##
## Residuals:
## Min 1Q Median 3Q Max
## -1.58108 -0.29768 -0.01306 0.37725 1.88789
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.46362 0.15439 3.003 0.003298 \*\*
## Current.BCG.vaccinationY -0.56504 0.17494 -3.230 0.001626 \*\*
## TB\_Z -0.15912 0.07144 -2.227 0.027930 \*
## age\_N 0.28390 0.07280 3.899 0.000165 \*\*\*
## frac\_N 0.35987 0.07180 5.012 2.03e-06 \*\*\*
## ---
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6148 on 112 degrees of freedom
## Multiple R-squared: 0.6342, Adjusted R-squared: 0.6211
## F-statistic: 48.54 on 4 and 112 DF, p-value: < 2.2e-16
##
## Correlation of Coefficients:
## (Intercept) Current.BCG.vaccinationY TB\_Z age\_N
## Current.BCG.vaccinationY -0.93
## TB\_Z 0.04 -0.04
## age\_N -0.31 0.33 0.30
## frac\_N -0.20 0.22 0.37 -0.19

rsq.partial(cases\_lm\_4, adj=TRUE)

## $adjustment
## [1] TRUE
##
## $variable
## [1] "Current.BCG.vaccination" "TB\_Z"
## [3] "age\_N" "frac\_N"
##
## $partial.rsq
## [1] 0.07703581 0.03386326 0.11167513 0.17592928

lm.beta(cases\_lm\_4)

##
## Call:
## lm(formula = cpm\_N ~ Current.BCG.vaccination + TB\_Z + age\_N +
## frac\_N, data = DATA\_COMPLETE)
##
## Standardized Coefficients::
## (Intercept) Current.BCG.vaccinationY TB\_Z
## 0.0000000 -0.2180265 -0.1593033
## age\_N frac\_N
## 0.2838967 0.3598720

shapiro.test(resid(cases\_lm\_4)) #List of residuals

##
## Shapiro-Wilk normality test
##
## data: resid(cases\_lm\_4)
## W = 0.98451, p-value = 0.1989

#deaths

deaths\_lm <- lm(dpm\_N ~ Current.BCG.vaccination + TB\_Z +
Ob\_N + temp\_N + age\_N + frac\_N
, data = DATA\_COMPLETE)

summary(deaths\_lm, corr=T)

##
## Call:
## lm(formula = dpm\_N ~ Current.BCG.vaccination + TB\_Z + Ob\_N +
## temp\_N + age\_N + frac\_N, data = DATA\_COMPLETE)
##
## Residuals:
## Min 1Q Median 3Q Max
## -1.28595 -0.36669 0.00561 0.38020 1.33033
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.49607 0.14665 3.383 0.000995 \*\*\*
## Current.BCG.vaccinationY -0.59311 0.16630 -3.566 0.000537 \*\*\*
## TB\_Z -0.15443 0.07191 -2.148 0.033938 \*
## Ob\_N 0.10412 0.08624 1.207 0.229921
## temp\_N -0.08408 0.06936 -1.212 0.228025
## age\_N 0.30173 0.07769 3.884 0.000176 \*\*\*
## frac\_N 0.18664 0.08529 2.188 0.030759 \*
## ---
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5811 on 110 degrees of freedom
## Multiple R-squared: 0.6635, Adjusted R-squared: 0.6451
## F-statistic: 36.14 on 6 and 110 DF, p-value: < 2.2e-16
##
## Correlation of Coefficients:
## (Intercept) Current.BCG.vaccinationY TB\_Z Ob\_N
## Current.BCG.vaccinationY -0.93
## TB\_Z 0.01 -0.01
## Ob\_N -0.04 0.05 0.33
## temp\_N 0.08 -0.08 -0.02 0.24
## age\_N -0.24 0.26 0.29 0.23
## frac\_N -0.13 0.14 0.07 -0.61
## temp\_N age\_N
## Current.BCG.vaccinationY
## TB\_Z
## Ob\_N
## temp\_N
## age\_N 0.45
## frac\_N -0.12 -0.26

rsq.partial(deaths\_lm, adj=TRUE)

## $adjustment
## [1] TRUE
##
## $variable
## [1] "Current.BCG.vaccination" "TB\_Z"
## [3] "Ob\_N" "temp\_N"
## [5] "age\_N" "frac\_N"
##
## $partial.rsq
## [1] 0.095500562 0.031517714 0.004104512 0.004211754 0.112580244 0.033007285

lm.beta(deaths\_lm)

##
## Call:
## lm(formula = dpm\_N ~ Current.BCG.vaccination + TB\_Z + Ob\_N +
## temp\_N + age\_N + frac\_N, data = DATA\_COMPLETE)
##
## Standardized Coefficients::
## (Intercept) Current.BCG.vaccinationY TB\_Z
## 0.00000000 -0.23435647 -0.15832818
## Ob\_N temp\_N age\_N
## 0.10661181 -0.08609724 0.30897920
## frac\_N
## 0.19112454

Anova(deaths\_lm)

## Anova Table (Type II tests)
##
## Response: dpm\_N
## Sum Sq Df F value Pr(>F)
## Current.BCG.vaccination 4.295 1 12.7198 0.0005370 \*\*\*
## TB\_Z 1.557 1 4.6123 0.0339381 \*
## Ob\_N 0.492 1 1.4575 0.2299213
## temp\_N 0.496 1 1.4695 0.2280247
## age\_N 5.092 1 15.0817 0.0001761 \*\*\*
## frac\_N 1.617 1 4.7889 0.0307588 \*
## Residuals 37.141 110
## ---
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#residual checks

shapiro.test(resid(deaths\_lm)) #List of residuals

##
## Shapiro-Wilk normality test
##
## data: resid(deaths\_lm)
## W = 0.99124, p-value = 0.6662

plot(density(resid(deaths\_lm))) #A density plot



#selection of predictors

models\_deaths <- regsubsets(dpm\_N ~ Current.BCG.vaccination + TB\_Z +
Ob\_N + temp\_N + age\_N + frac\_N, data = DATA\_COMPLETE, nvmax = 6)

summary(models\_deaths)

## Subset selection object
## Call: regsubsets.formula(dpm\_N ~ Current.BCG.vaccination + TB\_Z + Ob\_N +
## temp\_N + age\_N + frac\_N, data = DATA\_COMPLETE, nvmax = 6)
## 6 Variables (and intercept)
## Forced in Forced out
## Current.BCG.vaccinationY FALSE FALSE
## TB\_Z FALSE FALSE
## Ob\_N FALSE FALSE
## temp\_N FALSE FALSE
## age\_N FALSE FALSE
## frac\_N FALSE FALSE
## 1 subsets of each size up to 6
## Selection Algorithm: exhaustive
## Current.BCG.vaccinationY TB\_Z Ob\_N temp\_N age\_N frac\_N
## 1 ( 1 ) " " " " " " " " "\*" " "
## 2 ( 1 ) " " " " " " " " "\*" "\*"
## 3 ( 1 ) "\*" " " "\*" " " "\*" " "
## 4 ( 1 ) "\*" "\*" " " " " "\*" "\*"
## 5 ( 1 ) "\*" "\*" " " "\*" "\*" "\*"
## 6 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*"

res.sum <- summary(models\_deaths)
data.frame(
 Adj.R2 = which.max(res.sum$adjr2),
 CP = which.min(res.sum$cp),
 BIC = which.min(res.sum$bic)
)

## Adj.R2 CP BIC
## 1 6 5 4

coef(models\_deaths, 1:5)

## [[1]]
## (Intercept) age\_N
## 0.009425929 0.661056752
##
## [[2]]
## (Intercept) age\_N frac\_N
## 0.009425929 0.480227154 0.385293850
##
## [[3]]
## (Intercept) Current.BCG.vaccinationY Ob\_N
## 0.5542458 -0.6640019 0.3075202
## age\_N
## 0.4340699
##
## [[4]]
## (Intercept) Current.BCG.vaccinationY TB\_Z
## 0.5239991 -0.6271361 -0.1943396
## age\_N frac\_N
## 0.3279063 0.2526038
##
## [[5]]
## (Intercept) Current.BCG.vaccinationY TB\_Z
## 0.5035547 -0.6022221 -0.1829301
## temp\_N age\_N frac\_N
## -0.1039167 0.2798160 0.2489531

vcov(models\_deaths, 5)

## (Intercept) Current.BCG.vaccinationY TB\_Z
## (Intercept) 0.0215548805 -0.0227383741 0.0002626623
## Current.BCG.vaccinationY -0.0227383741 0.0277123661 -0.0003201322
## TB\_Z 0.0002626623 -0.0003201322 0.0046325409
## temp\_N 0.0008970772 -0.0010931989 -0.0005006352
## age\_N -0.0026818211 0.0032685222 0.0011856238
## frac\_N -0.0019645950 0.0023943541 0.0016676828
## temp\_N age\_N frac\_N
## (Intercept) 0.0008970772 -0.0026818211 -0.0019645950
## Current.BCG.vaccinationY -0.0010931989 0.0032685222 0.0023943541
## TB\_Z -0.0005006352 0.0011856238 0.0016676828
## temp\_N 0.0045597562 0.0021101535 0.0001601889
## age\_N 0.0021101535 0.0057305332 -0.0007971853
## frac\_N 0.0001601889 -0.0007971853 0.0046289070

deaths\_lm\_4 <- lm(dpm\_N ~ Current.BCG.vaccination + TB\_Z +
age\_N + frac\_N, data = DATA\_COMPLETE)

summary(deaths\_lm\_4, corr=T)

##
## Call:
## lm(formula = dpm\_N ~ Current.BCG.vaccination + TB\_Z + age\_N +
## frac\_N, data = DATA\_COMPLETE)
##
## Residuals:
## Min 1Q Median 3Q Max
## -1.43602 -0.36232 0.02462 0.37141 1.41178
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.52400 0.14710 3.562 0.000542 \*\*\*
## Current.BCG.vaccinationY -0.62714 0.16669 -3.762 0.000270 \*\*\*
## TB\_Z -0.19434 0.06807 -2.855 0.005129 \*\*
## age\_N 0.32791 0.06937 4.727 6.67e-06 \*\*\*
## frac\_N 0.25260 0.06841 3.693 0.000345 \*\*\*
## ---
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5858 on 112 degrees of freedom
## Multiple R-squared: 0.6517, Adjusted R-squared: 0.6393
## F-statistic: 52.4 on 4 and 112 DF, p-value: < 2.2e-16
##
## Correlation of Coefficients:
## (Intercept) Current.BCG.vaccinationY TB\_Z age\_N
## Current.BCG.vaccinationY -0.93
## TB\_Z 0.04 -0.04
## age\_N -0.31 0.33 0.30
## frac\_N -0.20 0.22 0.37 -0.19

rsq.partial(deaths\_lm\_4, adj=TRUE)

## $adjustment
## [1] TRUE
##
## $variable
## [1] "Current.BCG.vaccination" "TB\_Z"
## [3] "age\_N" "frac\_N"
##
## $partial.rsq
## [1] 0.1042750 0.0595172 0.1588777 0.1005694

lm.beta(deaths\_lm\_4)

##
## Call:
## lm(formula = dpm\_N ~ Current.BCG.vaccination + TB\_Z + age\_N +
## frac\_N, data = DATA\_COMPLETE)
##
## Standardized Coefficients::
## (Intercept) Current.BCG.vaccinationY TB\_Z
## 0.0000000 -0.2478030 -0.1992428
## age\_N frac\_N
## 0.3357853 0.2586734

shapiro.test(resid(deaths\_lm\_4)) #List of residuals

##
## Shapiro-Wilk normality test
##
## data: resid(deaths\_lm\_4)
## W = 0.99389, p-value = 0.8919

###strains###

DATA\_str <- CV19[, which (colnames(CV19) %in% c(
"Country\_code",
#"BCG.TB",
"cases\_per\_million\_UP",
"deaths\_per\_million\_UP",
#"Positive.cases.per.test",
#"temp",
"BCG.type","Current.BCG.vaccination",
#"prevalence\_overweight\_2012",
#"Fraction\_urban\_population",
#"Population\_ages\_65\_and\_above\_2018\_percent",
#"TB.incidents.per.10.thosand.mean\_2000\_2018",
"Strains"#,
#"TB.Incidents.100"
))]

DATA\_str <- subset(DATA\_str, Strains=="G3+" | Strains=="G3-")
#DATA\_str <- subset(DATA\_str, Current.BCG.vaccination=="Y")

DATA\_str\_COMPLETE <- na.omit(DATA\_str)

DATA\_str\_COMPLETE$cpm\_N <- predict(orderNorm(DATA\_str\_COMPLETE$cases\_per\_million\_UP))
hist(DATA\_str\_COMPLETE$cpm\_N)



DATA\_str\_COMPLETE$cpm\_Z <- scale(DATA\_str\_COMPLETE$cases\_per\_million\_UP)[,1]
hist(DATA\_str\_COMPLETE$cpm\_Z)



DATA\_str\_COMPLETE$dpm\_N <- predict(orderNorm(DATA\_str\_COMPLETE$deaths\_per\_million\_UP))

## Warning in orderNorm(DATA\_str\_COMPLETE$deaths\_per\_million\_UP): Ties in data, Normal distribution not guaranteed

hist(DATA\_str\_COMPLETE$dpm\_N)



DATA\_str\_COMPLETE$dpm\_Z <- scale(DATA\_str\_COMPLETE$deaths\_per\_million\_UP)[,1]
hist(DATA\_str\_COMPLETE$dpm\_Z)



#DATA\_str\_COMPLETE$TB\_N <- predict(orderNorm(DATA\_str\_COMPLETE$TB.incidents.per.10.thosand.mean\_2000\_2018))
#hist(DATA\_str\_COMPLETE$TB\_N)

#DATA\_str\_COMPLETE$TB\_Z <- scale(DATA\_str\_COMPLETE$TB.incidents.per.10.thosand.mean\_2000\_2018)[,1]

kruskal.test(Strains~Current.BCG.vaccination,data=DATA\_str\_COMPLETE)

##
## Kruskal-Wallis rank sum test
##
## data: Strains by Current.BCG.vaccination
## Kruskal-Wallis chi-squared = 10.027, df = 1, p-value = 0.001543

chisq.test(DATA\_str\_COMPLETE$Strains,DATA\_str\_COMPLETE$Current.BCG.vaccination, correct=FALSE)

##
## Pearson's Chi-squared test
##
## data: DATA\_str\_COMPLETE$Strains and DATA\_str\_COMPLETE$Current.BCG.vaccination
## X-squared = 10.174, df = 1, p-value = 0.001424

t.test(cpm\_N ~ Strains, data = DATA\_str\_COMPLETE)

##
## Welch Two Sample t-test
##
## data: cpm\_N by Strains
## t = -2.9291, df = 62.303, p-value = 0.004744
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -1.0520075 -0.1986029
## sample estimates:
## mean in group G3- mean in group G3+
## -0.4168702 0.2084351

t.test(dpm\_N ~ Strains, data = DATA\_str\_COMPLETE)

##
## Welch Two Sample t-test
##
## data: dpm\_N by Strains
## t = -2.7384, df = 66.126, p-value = 0.007925
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.9706255 -0.1520921
## sample estimates:
## mean in group G3- mean in group G3+
## -0.3712996 0.1900592

#####
#PCA#

cases\_PCA\_lm <- lm(cpm\_N ~ Current.BCG.vaccination + TB\_Z +
RC1 +
RC2 + RC3
, data = CV19)

summary(cases\_PCA\_lm, corr=T)

##
## Call:
## lm(formula = cpm\_N ~ Current.BCG.vaccination + TB\_Z + RC1 + RC2 +
## RC3, data = CV19)
##
## Residuals:
## Min 1Q Median 3Q Max
## -1.60339 -0.30866 0.01859 0.38078 1.45323
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.32803 0.14498 2.263 0.02524 \*
## Current.BCG.vaccinationY -0.39143 0.16246 -2.409 0.01732 \*
## TB\_Z -0.10935 0.06144 -1.780 0.07733 .
## RC1 0.62938 0.06910 9.108 9.46e-16 \*\*\*
## RC2 0.13998 0.05059 2.767 0.00645 \*\*
## RC3 0.11289 0.05063 2.230 0.02742 \*
## ---
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5937 on 136 degrees of freedom
## Multiple R-squared: 0.6593, Adjusted R-squared: 0.6468
## F-statistic: 52.64 on 5 and 136 DF, p-value: < 2.2e-16
##
## Correlation of Coefficients:
## (Intercept) Current.BCG.vaccinationY TB\_Z RC1 RC2
## Current.BCG.vaccinationY -0.94
## TB\_Z -0.01 0.01
## RC1 -0.44 0.47 0.51
## RC2 -0.05 0.06 0.14 0.10
## RC3 -0.14 0.15 0.06 0.10 0.02

shapiro.test(resid(cases\_PCA\_lm)) #List of residuals

##
## Shapiro-Wilk normality test
##
## data: resid(cases\_PCA\_lm)
## W = 0.98416, p-value = 0.101

plot(density(resid(cases\_PCA\_lm))) #A density plot



qqnorm(rstandard(cases\_PCA\_lm))
qqline(rstandard(cases\_PCA\_lm))



rsq.partial(cases\_PCA\_lm, adj=TRUE)

## $adjustment
## [1] TRUE
##
## $variable
## [1] "Current.BCG.vaccination" "TB\_Z"
## [3] "RC1" "RC2"
## [5] "RC3"
##
## $partial.rsq
## [1] 0.03388490 0.01557782 0.37431387 0.04632776 0.02816803

lm.beta(cases\_PCA\_lm)

##
## Call:
## lm(formula = cpm\_N ~ Current.BCG.vaccination + TB\_Z + RC1 + RC2 +
## RC3, data = CV19)
##
## Standardized Coefficients::
## (Intercept) Current.BCG.vaccinationY TB\_Z
## 0.0000000 -0.1448638 -0.1094597
## RC1 RC2 RC3
## 0.6299940 0.1401170 0.1130005

kruskal.test(RC1 ~ Current.BCG.vaccination, data = CV19)

##
## Kruskal-Wallis rank sum test
##
## data: RC1 by Current.BCG.vaccination
## Kruskal-Wallis chi-squared = 44.852, df = 1, p-value = 2.125e-11

kruskal.test(RC2 ~ Current.BCG.vaccination, data = CV19)

##
## Kruskal-Wallis rank sum test
##
## data: RC2 by Current.BCG.vaccination
## Kruskal-Wallis chi-squared = 0.81961, df = 1, p-value = 0.3653

kruskal.test(RC3 ~ Current.BCG.vaccination, data = CV19)

##
## Kruskal-Wallis rank sum test
##
## data: RC3 by Current.BCG.vaccination
## Kruskal-Wallis chi-squared = 1.5591, df = 1, p-value = 0.2118

deaths\_PCA\_lm <- lm(dpm\_N ~ Current.BCG.vaccination + TB\_Z +
RC1 +
RC2 + RC3
, data = CV19)

summary(deaths\_PCA\_lm, corr=T)

##
## Call:
## lm(formula = dpm\_N ~ Current.BCG.vaccination + TB\_Z + RC1 + RC2 +
## RC3, data = CV19)
##
## Residuals:
## Min 1Q Median 3Q Max
## -1.58135 -0.30657 -0.02918 0.42908 1.34421
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.43391 0.14113 3.074 0.00255 \*\*
## Current.BCG.vaccinationY -0.50567 0.15815 -3.197 0.00173 \*\*
## TB\_Z -0.18550 0.05981 -3.102 0.00234 \*\*
## RC1 0.50709 0.06727 7.538 6.05e-12 \*\*\*
## RC2 0.07907 0.04925 1.605 0.11072
## RC3 0.21158 0.04929 4.292 3.34e-05 \*\*\*
## ---
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.578 on 136 degrees of freedom
## Multiple R-squared: 0.6603, Adjusted R-squared: 0.6478
## F-statistic: 52.86 on 5 and 136 DF, p-value: < 2.2e-16
##
## Correlation of Coefficients:
## (Intercept) Current.BCG.vaccinationY TB\_Z RC1 RC2
## Current.BCG.vaccinationY -0.94
## TB\_Z -0.01 0.01
## RC1 -0.44 0.47 0.51
## RC2 -0.05 0.06 0.14 0.10
## RC3 -0.14 0.15 0.06 0.10 0.02

shapiro.test(resid(deaths\_PCA\_lm)) #List of residuals

##
## Shapiro-Wilk normality test
##
## data: resid(deaths\_PCA\_lm)
## W = 0.99262, p-value = 0.6737

plot(density(resid(deaths\_PCA\_lm))) #A density plot



rsq.partial(deaths\_PCA\_lm, adj=TRUE)

## $adjustment
## [1] TRUE
##
## $variable
## [1] "Current.BCG.vaccination" "TB\_Z"
## [3] "RC1" "RC2"
## [5] "RC3"
##
## $partial.rsq
## [1] 0.06307656 0.05919085 0.28952080 0.01138243 0.11283562

lm.beta(deaths\_PCA\_lm)

##
## Call:
## lm(formula = dpm\_N ~ Current.BCG.vaccination + TB\_Z + RC1 + RC2 +
## RC3, data = CV19)
##
## Standardized Coefficients::
## (Intercept) Current.BCG.vaccinationY TB\_Z
## 0.00000000 -0.19197587 -0.19047304
## RC1 RC2 RC3
## 0.52068799 0.08118825 0.21724907

####PLOTS####

c = ggplot(DATA\_COMPLETE, aes(x=Current.BCG.vaccination, y=cpm\_N, fill=Current.BCG.vaccination))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
#c = c + geom\_point()
c = c + scale\_fill\_manual(name="BCG policy", breaks = c("N", "Y"),
 values=c("#00BA38","#F8766D"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 cases per 1M pop.") + xlab("")
c = c + ggtitle("Cases by BCG policy")
c = c + theme(legend.position="bottom")
#c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_cases\_boxplot\_Figure\_1a.png", plot=c, dpi=300, width=5, height=5)

c = ggplot(DATA\_COMPLETE, aes(x=TB.Incidents.100, y=cpm\_N, fill=TB.Incidents.100))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
#c = c + geom\_point()
c = c + scale\_fill\_manual(name="TB incidence 100", breaks = c("n", "y"),
 values=c("orange","purple"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 cases per 1M pop.") + xlab("")
c = c + ggtitle("Cases by TB incidence (100)")
c = c + theme(legend.position="bottom")
#c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_cases\_boxplot\_Figure\_1e.png", plot=c, dpi=300, width=5, height=5)

c = ggplot(DATA\_COMPLETE, aes(x=Current.BCG.vaccination, y=dpm\_N, fill=Current.BCG.vaccination))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
#c = c + geom\_point()
c = c + scale\_fill\_manual(name="BCG policy", breaks = c("N", "Y"),
 values=c("#00BA38","#F8766D"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 deaths per 1M pop.") + xlab("")
c = c + ggtitle("Deaths by BCG policy")
c = c + theme(legend.position="bottom")
#c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_deaths\_boxplot\_Figure\_1b.png", plot=c, dpi=300, width=5, height=5)

c = ggplot(DATA\_COMPLETE, aes(x=TB.Incidents.100, y=dpm\_N, fill=TB.Incidents.100))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
#c = c + geom\_point()
c = c + scale\_fill\_manual(name="TB incidence 100", breaks = c("n", "y"),
 values=c("orange","purple"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 deaths per 1M pop.") + xlab("")
c = c + ggtitle("Deaths by TB incidence (100)")
c = c + theme(legend.position="bottom")
#c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_deaths\_boxplot\_Figure\_1f.png", plot=c, dpi=300, width=5, height=5)

#plots Figures 2

d2 = ggplot(DATA\_COMPLETE, aes(x=TB\_Z, y=cpm\_N, color=BCG.type)) +geom\_point()
d2 = d2 + stat\_smooth(method="lm",size = 1, colour="black")
d2 = d2 + scale\_color\_discrete( name="BCG type", labels = c("A: Currently administering", "B: Used to administer", "C: Never administered"))
d2 = d2 + theme(legend.text=element\_text(size=14))
d2 = d2 + ylab("Standardized COVID-19 cases per 1M pop.") + xlab("TB incidents per 100K (scaled)")
d2 = d2 + ggtitle("COVID-19 cases by TB incidents")
d2 = d2 + geom\_vline(xintercept=-0.25, linetype="dotted",size = 1)
d2 = d2 + theme(text = element\_text(size = 14))
d2

## `geom\_smooth()` using formula 'y ~ x'



ggsave(file="scatterCOVID19\_cases\_TB\_Figure\_2a.png", plot=d2, dpi=300, width=7, height=5)

## `geom\_smooth()` using formula 'y ~ x'

d2 = ggplot(DATA\_COMPLETE, aes(x=TB\_Z, y=dpm\_N, color=BCG.type)) +geom\_point()
d2 = d2 + stat\_smooth(method="lm",size = 1, colour="black")
d2 = d2 + scale\_color\_discrete( name="BCG type", labels = c("A: Currently administering", "B: Used to administer", "C: Never administered"))
d2 = d2 + theme(legend.text=element\_text(size=14))
d2 = d2 + ylab("Standardized COVID-19 deaths per 1M pop.") + xlab("TB incidents per 100K (scaled)")
d2 = d2 + ggtitle("COVID-19 deaths by TB incidents")
d2 = d2 + geom\_vline(xintercept=-0.25, linetype="dotted",size = 1)
d2 = d2 + theme(text = element\_text(size = 14))
d2

## `geom\_smooth()` using formula 'y ~ x'



ggsave(file="scatterCOVID19\_deaths\_TB\_Figure\_2d.png", plot=d2, dpi=300, width=7, height=5)

## `geom\_smooth()` using formula 'y ~ x'

DATA\_COMPLETE\_TB <- subset(DATA\_COMPLETE, TB.Incidents.100=="n")

a<-aggregate(TB\_Z~Current.BCG.vaccination, DATA\_COMPLETE\_TB, FUN=mean)
SEM<-(aggregate(TB\_Z~Current.BCG.vaccination, DATA\_COMPLETE\_TB, FUN=sd)[,2])/sqrt(length(DATA\_COMPLETE\_TB[,1]))
Tub<-cbind.data.frame(a,SEM)

dat\_temp <- subset(DATA\_COMPLETE, temp>0 & temp<15)

a<-aggregate(temp~Current.BCG.vaccination, dat\_temp, FUN=mean)
SEM<-(aggregate(temp~Current.BCG.vaccination, dat\_temp, FUN=sd)[,2])#/sqrt(length(dat\_temp[,1]))
temp<-cbind.data.frame(a,SEM)

c = ggplot(DATA\_COMPLETE\_TB, aes(x=Current.BCG.vaccination, y=cpm\_N, fill=Current.BCG.vaccination))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
#c = c + geom\_point()
c = c + scale\_fill\_manual(name="BCG policy", breaks = c("N", "Y"),
 values=c("#00BA38","#F8766D"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 cases per 1M pop.") + xlab("")
c = c + ggtitle("Cases by BCG policy (filtered TB<100)")
c = c + theme(legend.position="bottom")
c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_cases\_boxplot\_Figure\_2b.png", plot=c, dpi=300, width=5, height=5)

c = ggplot(DATA\_COMPLETE\_TB, aes(x=Current.BCG.vaccination, y=dpm\_N, fill=Current.BCG.vaccination))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
#c = c + geom\_point()
c = c + scale\_fill\_manual(name="BCG policy", breaks = c("N", "Y"),
 values=c("#00BA38","#F8766D"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 deaths per 1M pop.") + xlab("")
c = c + ggtitle("Deaths by BCG policy (filtered TB<100)")
c = c + theme(legend.position="bottom")
c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_deaths\_boxplot\_Figure\_2e.png", plot=c, dpi=300, width=5, height=5)

c = ggplot(DATA\_str\_COMPLETE, aes(x=Strains, y=cpm\_N, fill=Strains))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
c = c + scale\_fill\_manual(name="Strain", breaks = c("G3-", "G3+"),
 values=c("blue","yellow"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 cases per 1M pop.") + xlab("")
c = c + ggtitle("COVID-19 cases by BCG vaccine Strains")
c = c + theme(legend.position="bottom")
c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_cases\_strains\_Figure\_1c.png", plot=c, dpi=300, width=5, height=5)

c = ggplot(DATA\_str\_COMPLETE, aes(x=Strains, y=dpm\_N, fill=Strains))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
c = c + scale\_fill\_manual(name="Strain", breaks = c("G3-", "G3+"),
 values=c("blue","yellow"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 deaths per 1M pop.") + xlab("")
c = c + ggtitle("COVID-19 deaths by BCG vaccine Strains")
c = c + theme(legend.position="bottom")
c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_deaths\_strains\_Figure\_1d.png", plot=c, dpi=300, width=5, height=5)

DATA\_str\_fig <- CV19[, which (colnames(CV19) %in% c(
"Country\_code",
#"BCG.TB",
"cases\_per\_million\_UP",
"deaths\_per\_million\_UP",
#"Positive.cases.per.test",
#"temp",
"BCG.type","Current.BCG.vaccination",
#"prevalence\_overweight\_2012",
#"Fraction\_urban\_population",
#"Population\_ages\_65\_and\_above\_2018\_percent",
"TB.incidents.per.10.thosand.mean\_2000\_2018",
"Strains",
"TB.Incidents.100"
))]

DATA\_str\_fig <- subset(DATA\_str\_fig, Strains=="G3+" | Strains=="G3-")
DATA\_str\_fig <- subset(DATA\_str\_fig, TB.Incidents.100=="n")

DATA\_str\_COMPLETE\_fig <- na.omit(DATA\_str\_fig)

DATA\_str\_COMPLETE\_fig$cpm\_N <- predict(orderNorm(DATA\_str\_COMPLETE\_fig$cases\_per\_million\_UP))
DATA\_str\_COMPLETE\_fig$dpm\_N <- predict(orderNorm(DATA\_str\_COMPLETE\_fig$deaths\_per\_million\_UP))

c = ggplot(DATA\_str\_COMPLETE\_fig, aes(x=Strains, y=cpm\_N, fill=Strains))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
c = c + scale\_fill\_manual(name="Strain", breaks = c("G3-", "G3+"),
 values=c("blue","yellow"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 cases per 1M pop.") + xlab("")
c = c + ggtitle("BCG vaccine Strains (filtered TB<100)")
c = c + theme(legend.position="bottom")
c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_cases\_strains\_Figure\_2c.png", plot=c, dpi=300, width=5, height=5)

c = ggplot(DATA\_str\_COMPLETE\_fig, aes(x=Strains, y=dpm\_N, fill=Strains))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
c = c + scale\_fill\_manual(name="Strain", breaks = c("G3-", "G3+"),
 values=c("blue","yellow"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 deaths per 1M pop.") + xlab("")
c = c + ggtitle("BCG vaccine Strains (filtered TB<100)")
c = c + theme(legend.position="bottom")
c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_deaths\_strains\_Figure\_2f.png", plot=c, dpi=300, width=5, height=5)

DATA\_str\_fig65 <- CV19[, which (colnames(CV19) %in% c(
"Country\_code",
#"BCG.TB",
"cases\_per\_million\_UP",
"deaths\_per\_million\_UP",
#"Positive.cases.per.test",
#"temp",
"BCG.type","Current.BCG.vaccination",
#"prevalence\_overweight\_2012",
#"Fraction\_urban\_population",
"Population\_ages\_65\_and\_above\_2018\_percent",
#"TB.incidents.per.10.thosand.mean\_2000\_2018",
"Strains"#,
#"TB.Incidents.100"
))]

DATA\_str\_fig65 <- subset(DATA\_str\_fig65, Strains=="G3+" | Strains=="G3-")
DATA\_str\_fig65 <- subset(DATA\_str\_fig65, Population\_ages\_65\_and\_above\_2018\_percent>15)

DATA\_str\_COMPLETE\_fig65 <- na.omit(DATA\_str\_fig65)

DATA\_str\_COMPLETE\_fig65$cpm\_N <- predict(orderNorm(DATA\_str\_COMPLETE\_fig65$cases\_per\_million\_UP))
DATA\_str\_COMPLETE\_fig65$dpm\_N <- predict(orderNorm(DATA\_str\_COMPLETE\_fig65$deaths\_per\_million\_UP))

c = ggplot(DATA\_str\_COMPLETE\_fig65, aes(x=Strains, y=cpm\_N, fill=Strains))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
c = c + scale\_fill\_manual(name="Strain", breaks = c("G3-", "G3+"),
 values=c("blue","yellow"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 cases per 1M pop.") + xlab("")
c = c + ggtitle("BCG vaccine Strains (>65 years > 15%)")
c = c + theme(legend.position="bottom")
c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_cases\_strains\_age\_Figure\_3e.png", plot=c, dpi=300, width=5, height=5)

c = ggplot(DATA\_str\_COMPLETE\_fig65, aes(x=Strains, y=dpm\_N, fill=Strains))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
c = c + scale\_fill\_manual(name="Strain", breaks = c("G3-", "G3+"),
 values=c("blue","yellow"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 deaths per 1M pop.") + xlab("")
c = c + ggtitle("BCG vaccine Strains (65 years > 15%)")
c = c + theme(legend.position="bottom")
c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_deaths\_strains\_age\_Figure\_3f.png", plot=c, dpi=300, width=5, height=5)

DATA\_str\_fig\_frac <- CV19[, which (colnames(CV19) %in% c(
"Country\_code",
#"BCG.TB",
"cases\_per\_million\_UP",
"deaths\_per\_million\_UP",
#"Positive.cases.per.test",
#"temp",
"BCG.type","Current.BCG.vaccination",
#"prevalence\_overweight\_2012",
"Fraction\_urban\_population",
#"Population\_ages\_65\_and\_above\_2018\_percent",
#"TB.incidents.per.10.thosand.mean\_2000\_2018",
"Strains"#,
#"TB.Incidents.100"
))]

DATA\_str\_fig\_frac <- subset(DATA\_str\_fig\_frac, Strains=="G3+" | Strains=="G3-")
DATA\_str\_fig\_frac$Fraction\_urban\_population\_100 <- DATA\_str\_fig\_frac$Fraction\_urban\_population\*100
DATA\_str\_fig\_frac <- subset(DATA\_str\_fig\_frac, Fraction\_urban\_population\_100>75)

DATA\_str\_COMPLETE\_fig\_frac <- na.omit(DATA\_str\_fig\_frac)

DATA\_str\_COMPLETE\_fig\_frac$cpm\_N <- predict(orderNorm(DATA\_str\_COMPLETE\_fig\_frac$cases\_per\_million\_UP))
DATA\_str\_COMPLETE\_fig\_frac$dpm\_N <- predict(orderNorm(DATA\_str\_COMPLETE\_fig\_frac$deaths\_per\_million\_UP))

c = ggplot(DATA\_str\_COMPLETE\_fig\_frac, aes(x=Strains, y=cpm\_N, fill=Strains))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
c = c + scale\_fill\_manual(name="Strain", breaks = c("G3-", "G3+"),
 values=c("blue","yellow"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 cases per 1M pop.") + xlab("")
c = c + ggtitle("BCG vaccine Strains (Urbanization>75%)")
c = c + theme(legend.position="bottom")
c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_cases\_strains\_Figure\_frac\_4c.png", plot=c, dpi=300, width=5, height=5)

c = ggplot(DATA\_str\_COMPLETE\_fig\_frac, aes(x=Strains, y=dpm\_N, fill=Strains))
c = c + geom\_boxplot()
c = c + theme(axis.title.x=element\_blank(),
 axis.text.x=element\_blank(),
 axis.ticks.x=element\_blank(),
 axis.text.y = element\_text( size = 14))
c = c + geom\_jitter(shape=17, position=position\_jitter(0.1))
c = c + scale\_fill\_manual(name="Strain", breaks = c("G3-", "G3+"),
 values=c("blue","yellow"))
c = c + theme(legend.text=element\_text(size=14))
c = c + ylab("Standardized COVID-19 deaths per 1M pop.") + xlab("")
c = c + ggtitle("BCG vaccine Strains (Urbanization>75%)")
c = c + theme(legend.position="bottom")
c = c + stat\_compare\_means(method = "t.test", label.x = 1.3, label.y = 2.7, size=5)
c = c + theme(text = element\_text(size = 14))
c



ggsave(file="COVID19\_deaths\_strains\_frac\_Figure\_4f.png", plot=c, dpi=300, width=5, height=5)