ImputeData

Shyju Kozhukkunnon

4/27/2022

Impute the secondary education expense

```
#install.packages("imputeTS")
install.packages("knitr")
```

```
##
##   There is a binary version available but the source version is later:
##       binary source needs_compilation
## knitr   1.38   1.39             FALSE
```

```
library(imputeTS)
sec_edu_ppp <- read.csv('../Data/Cleaned_Data/sec_exp_ppp_cleaned.csv')
#create a list of unique values.
countryUnique <- unique(sec_edu_ppp[c("Country.Code")])
df_sec_edu_ppp <- sec_edu_ppp[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
sec_edu_ppp_tmp <- sec_edu_ppp[sec_edu_ppp$Country.Code==ccode,]
  if ( nrow(unique(sec_edu_ppp_tmp[c("sec_exp_ppp")]))  > 2) {
  sec_edu_ppp_tmp$sec_exp_ppp <- na_interpolation(sec_edu_ppp_tmp$sec_exp_ppp)
  df_sec_edu_ppp <- rbind(df_sec_edu_ppp, sec_edu_ppp_tmp)
  }
}

names(df_sec_edu_ppp)[names(df_sec_edu_ppp)=="Country.Name"] <- "SECountryName"
names(df_sec_edu_ppp)[names(df_sec_edu_ppp)=="Country.Code"] <- "SECountryCode"
names(df_sec_edu_ppp)[names(df_sec_edu_ppp)=="Year"] <- "SEYear"

write.csv(df_sec_edu_ppp, '../Data/ImputedData/sec_exp_ppp_imputed.csv')
```

Missing data imputing for primary education data.

```
primary_edu_ppp <- read.csv('../Data/Cleaned_Data/pri_exp_ppp_cleaned.csv')
#create a list of unique values.
countryUnique <- unique(primary_edu_ppp[c("Country.Code")])
df_primary_edu_ppp <- primary_edu_ppp[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
```

```
pri_edu_ppp_tmp <- primary_edu_ppp[primary_edu_ppp$Country.Code==ccode,]
  if ( nrow(unique(pri_edu_ppp_tmp[c("pri_exp_ppp")]))  > 2) {
  pri_edu_ppp_tmp$pri_exp_ppp <- na_interpolation(pri_edu_ppp_tmp$pri_exp_ppp)
  df_primary_edu_ppp <- rbind(df_primary_edu_ppp, pri_edu_ppp_tmp)
  }
}

df_primary_edu_ppp <- subset(df_primary_edu_ppp, select = -c(Series, Series.Code))
names(df_primary_edu_ppp)[names(df_primary_edu_ppp)=="Country.Name"] <- "PECountryName"
names(df_primary_edu_ppp)[names(df_primary_edu_ppp)=="Country.Code"] <- "PECountryCode"
names(df_primary_edu_ppp)[names(df_primary_edu_ppp)=="Year"] <- "PEYear"

write.csv(df_primary_edu_ppp, '../Data/ImputedData/pri_exp_ppp_imputed.csv', row.names = FALSE)
```

Data imputing for GDI

```
country_GDI <- read.csv('../Data/Cleaned_Data/GDI_Cleaned.csv')
#create a list of unique values.
countryUnique <- unique(country_GDI[c("Country.Code")])
df_country_GDI <- country_GDI[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
country_GDI_tmp <- country_GDI[country_GDI$Country.Code==ccode,]
  if ( nrow(unique(country_GDI_tmp[c("Gender_EQ_Score")]))  > 2) {
  country_GDI_tmp$Gender_EQ_Score <- na_interpolation(country_GDI_tmp$Gender_EQ_Score)
  df_country_GDI <- rbind(df_country_GDI, country_GDI_tmp)
  }
}

df_country_GDI <- subset(df_country_GDI, select = -c(Indicator.Name, Indicator.Code, Attribute))
names(df_country_GDI)[names(df_country_GDI)=="Gender_EQ_Score"] <- "CountryGDI"
names(df_country_GDI)[names(df_country_GDI)=="Country.Name"] <- "GDICountryName"
names(df_country_GDI)[names(df_country_GDI)=="Country.Code"] <- "GDICountryCode"
names(df_country_GDI)[names(df_country_GDI)=="Year"] <- "GDIYear"

write.csv(df_country_GDI, '../Data/ImputedData/country_GDI_imputed.csv', row.names = FALSE)
```

Data Imputing for GII

```
country_GII  <- read.csv('../Data/Cleaned_Data/GII_Cleaned.csv')
#create a list of unique values.
countryUnique <- unique(country_GII[c("Country.Code")])
df_country_GII <- country_GII[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
country_GII_tmp <- country_GII[country_GII$Country.Code==ccode,]
  if ( nrow(unique(country_GII_tmp[c("Gender_EQ_Score")]))  > 2) {
  country_GII_tmp$Gender_EQ_Score <- na_interpolation(country_GII_tmp$Gender_EQ_Score)
  df_country_GII <- rbind(df_country_GII, country_GII_tmp)
```

```
  }
}

df_country_GII <- subset(df_country_GII, select = -c(Indicator.Name, Indicator.Code, Attribute))
names(df_country_GII)[names(df_country_GII)=="Gender_EQ_Score"] <- "CountryGII"
names(df_country_GII)[names(df_country_GII)=="Country.Name"] <- "GIICountryName"
names(df_country_GII)[names(df_country_GII)=="Country.Code"] <- "GIICountryCode"
names(df_country_GII)[names(df_country_GII)=="Year"] <- "GIIYear"

write.csv(df_country_GII, '../Data/ImputedData/country_GII_imputed.csv', row.names = FALSE)
```

Data imputing for Rural Population Percentage

```
rural_pop_perc <- read.csv('../Data/Cleaned_Data/rural_pop_cleaned.csv')
#create a list of unique values.
countryUnique <- unique(rural_pop_perc[c("Country.Code")])
df_rural_pop_perc <- rural_pop_perc[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
rural_pop_perc_tmp <- rural_pop_perc[rural_pop_perc$Country.Code==ccode,]
  if ( nrow(unique(rural_pop_perc_tmp[c("Rur_perc")]))  > 2) {
  rural_pop_perc_tmp$Rur_perc <- na_interpolation(rural_pop_perc_tmp$Rur_perc)
  df_rural_pop_perc <- rbind(df_rural_pop_perc, rural_pop_perc_tmp)
  }
}

df_rural_pop_perc <- subset(df_rural_pop_perc, select = -c(Series.Name, Series.Code))
names(df_rural_pop_perc)[names(df_rural_pop_perc)=="Country.Name"] <- "RPPCountryName"
names(df_rural_pop_perc)[names(df_rural_pop_perc)=="Country.Code"] <- "RPPCountryCode"
names(df_rural_pop_perc)[names(df_rural_pop_perc)=="Year"] <- "RPPYear"

write.csv(df_rural_pop_perc, '../Data/ImputedData/rural_pop_imputed.csv', row.names = FALSE)
```

Data imputing for Rural population growth.

```
rural_pop_growth <- read.csv('../Data/Cleaned_Data/rural_pop_growth_cleaned.csv')
#create a list of unique values.
countryUnique <- unique(rural_pop_growth[c("Country.Code")])
df_rural_pop_growth <- rural_pop_growth[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
rural_pop_growth_tmp <- rural_pop_growth[rural_pop_growth$Country.Code==ccode,]
  if ( nrow(unique(rural_pop_growth_tmp[c("Rur_pop_growth")]))  > 2) {
  rural_pop_growth_tmp$Rur_pop_growth <- na_interpolation(rural_pop_growth_tmp$Rur_pop_growth)
  df_rural_pop_growth <- rbind(df_rural_pop_growth, rural_pop_growth_tmp)
  }
}

df_rural_pop_growth <- subset(df_rural_pop_growth, select = -c(Series.Name, Series.Code))
```

```
names(df_rural_pop_growth)[names(df_rural_pop_growth)=="Country.Name"] <- "RPGCountryName"
names(df_rural_pop_growth)[names(df_rural_pop_growth)=="Country.Code"] <- "RPGCountryCode"
names(df_rural_pop_growth)[names(df_rural_pop_growth)=="Year"] <- "RPGYear"

write.csv(df_rural_pop_growth, '../Data/ImputedData/rural_pop_growth_imputed.csv', row.names = FALSE)
```

Data Imputing for death rate

Data Imputing for economy - GDP PC PPP

```
gdp_pc_ppp <- read.csv('../Data/Cleaned_Data/gdp_pc_ppp_cleaned.csv')
#create a list of unique values.
countryUnique <- unique(gdp_pc_ppp[c("Country.Code")])
df_gdp_pc_ppp <- gdp_pc_ppp[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
gdp_pc_ppp_tmp <- gdp_pc_ppp[gdp_pc_ppp$Country.Code==ccode,]
  if ( nrow(unique(gdp_pc_ppp_tmp[c("gdp_pc_ppp")]))  > 2) {
  gdp_pc_ppp_tmp$gdp_pc_ppp <- na_interpolation(gdp_pc_ppp_tmp$gdp_pc_ppp)
  df_gdp_pc_ppp <- rbind(df_gdp_pc_ppp, gdp_pc_ppp_tmp)
  }
}

df_gdp_pc_ppp <- subset(df_gdp_pc_ppp, select = -c(Series.Name, Series.Code))
names(df_gdp_pc_ppp)[names(df_gdp_pc_ppp)=="Country.Name"] <- "GDP_PC_CountryName"
names(df_gdp_pc_ppp)[names(df_gdp_pc_ppp)=="Country.Code"] <- "GDP_PC_CountryCode"

write.csv(df_gdp_pc_ppp, '../Data/ImputedData/gdp_pc_ppp_imputed.csv', row.names = FALSE)
```

Data Imputing for economy - GDP PC Growth

```
gdp_pc_growth <- read.csv('../Data/Cleaned_Data/gdp_pc_growth_cleaned.csv')
#create a list of unique values.
countryUnique <- unique(gdp_pc_growth[c("Country.Code")])
df_gdp_pc_growth <- gdp_pc_growth[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
gdp_pc_growth_tmp <- gdp_pc_growth[gdp_pc_growth$Country.Code==ccode,]
  if ( nrow(unique(gdp_pc_growth_tmp[c("gdp_pc_growth")]))  > 2) {
  gdp_pc_growth_tmp$gdp_pc_growth <- na_interpolation(gdp_pc_growth_tmp$gdp_pc_growth)
  df_gdp_pc_growth <- rbind(df_gdp_pc_growth, gdp_pc_growth_tmp)
  }
}

df_gdp_pc_growth <- subset(df_gdp_pc_growth, select = -c(Series.Name, Series.Code))
names(df_gdp_pc_growth)[names(df_gdp_pc_growth)=="Country.Name"] <- "GDP_GRWTH_CountryName"
names(df_gdp_pc_growth)[names(df_gdp_pc_growth)=="Country.Code"] <- "GDP_GRWTH_CountryCode"

write.csv(df_gdp_pc_growth, '../Data/ImputedData/gdp_pc_growth_imputed.csv', row.names = FALSE)
```

Data Imputing for economy - Poverty Percentage

```
poverty_pc <- read.csv('../Data/Cleaned_Data/poverty_pc_cleaned.csv')
#create a list of unique values.
countryUnique <- unique(poverty_pc[c("Country.Code")])
df_poverty_pc <- poverty_pc[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
poverty_pc_tmp <- poverty_pc[poverty_pc$Country.Code==ccode,]
  if ( nrow(unique(poverty_pc_tmp[c("poverty_perc")]))  > 2) {
  poverty_pc_tmp$poverty_perc <- na_interpolation(poverty_pc_tmp$poverty_perc)
  df_poverty_pc <- rbind(df_poverty_pc, poverty_pc_tmp)
  }
}

df_poverty_pc <- subset(df_poverty_pc, select = -c(Series.Name, Series.Code))
names(df_poverty_pc)[names(df_poverty_pc)=="Country.Name"] <- "Poverty_pct_CountryName"
names(df_poverty_pc)[names(df_poverty_pc)=="Country.Code"] <- "Poverty_pct_CountryCode"

write.csv(df_poverty_pc, '../Data/ImputedData/poverty_pct_imputed.csv', row.names = FALSE)
```

Imputing the data for Healthcare and Finance.

```
healthc_fin <- read.csv('../Data/Cleaned_Data/hcf_cleaned.csv')
#create a list of unique values.
countryUnique <- unique(healthc_fin[c("HcfCC")])
df_healthc_fin <- healthc_fin[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
healthc_fin_tmp <- healthc_fin[healthc_fin$HcfCC==ccode,]
  if ( nrow(unique(healthc_fin_tmp[c("HcfVal")]))  > 2) {
  healthc_fin_tmp$HcfVal <- na_interpolation(healthc_fin_tmp$HcfVal)
  df_healthc_fin <- rbind(df_healthc_fin, healthc_fin_tmp)
  }
}

write.csv(df_healthc_fin, '../Data/ImputedData/hcf_imputed.csv', row.names = FALSE)
```

Imputing the data for LEAB (life expectancy at birth)

```
life_exp <- read.csv('../Data/Cleaned_Data/leab_cleaned.csv')
#create a list of unique values.
countryUnique <- unique(life_exp[c("LeabCC")])
df_life_exp <- life_exp[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
life_exp_tmp <- life_exp[life_exp$LeabCC==ccode,]
```

```
  if ( nrow(unique(life_exp_tmp[c("LeabVal")]))  > 2) {
  life_exp_tmp$LeabVal <- na_interpolation(life_exp_tmp$LeabVal)
  df_life_exp <- rbind(df_life_exp, life_exp_tmp)
  }
}

write.csv(df_life_exp, '../Data/ImputedData/leab_imputed.csv', row.names = FALSE)
```

Imputing for death rate.

```
death_rate <- read.csv('../Data/Cleaned_Data/death_rate_cleaned.csv')
death_rate <- subset(death_rate, select = -c(Series.Name, Series.Code))
names(death_rate)[names(death_rate)=="Year"] <- "DRYear"
names(death_rate)[names(death_rate)=="Country.Name"] <- "DRCountryName"
names(death_rate)[names(death_rate)=="Country.Code"] <- "DRCountryCode"

countryUnique <- unique(death_rate[c("DRCountryCode")])
df_death_rate <- death_rate[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
death_rate_tmp <- death_rate[death_rate$DRCountryCode==ccode,]
  if ( nrow(unique(death_rate_tmp[c("death_per_1000")]))  > 2) {
  death_rate_tmp$death_per_1000 <- na_interpolation(death_rate_tmp$death_per_1000)
  df_death_rate <- rbind(df_death_rate, death_rate_tmp)
  }
}

write.csv(df_death_rate, '../Data/ImputedData/death_rate_imputed.csv', row.names = FALSE)
```

Next two measures have data only every 5 years at best. This needs to be made to yearly and imputed.

Key vaccinations measure.

```
key_vaccn <- read.csv('../Data/Cleaned_Data/kvc_cleaned.csv')
death_rate_impt <- read.csv('../Data/ImputedData/death_rate_imputed.csv')

df_combine <- merge(death_rate_impt, key_vaccn,  by.x = c("DRCountryCode","DRYear"), by.y = c("KvcCC","
df_combine <- subset(df_combine, select = -c(X.x, X.y, KvcCountry,death_per_1000))

names(df_combine)[names(df_combine)=="DRCountryName"] <- "KvcCountry"
names(df_combine)[names(df_combine)=="DRCountryCode"] <- "KvcCC"
names(df_combine)[names(df_combine)=="DRYear"] <- "KvcYear"

#Rows added, now impute.
countryUnique <- unique(df_combine[c("KvcCC")])
df_kvc <- df_combine[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
```

```
{
ccode = countryUnique[i,]
df_combine_tmp <- df_combine[df_combine$KvcCC==ccode,]
  if ( nrow(unique(df_combine_tmp[c("KvcVal")]))  > 2) {
  df_combine_tmp$KvcVal <- na_interpolation(df_combine_tmp$KvcVal)
  df_kvc <- rbind(df_kvc, df_combine_tmp)
  }
}

write.csv(df_kvc, '../Data/ImputedData/kvc_imputed.csv', row.names = FALSE)
```

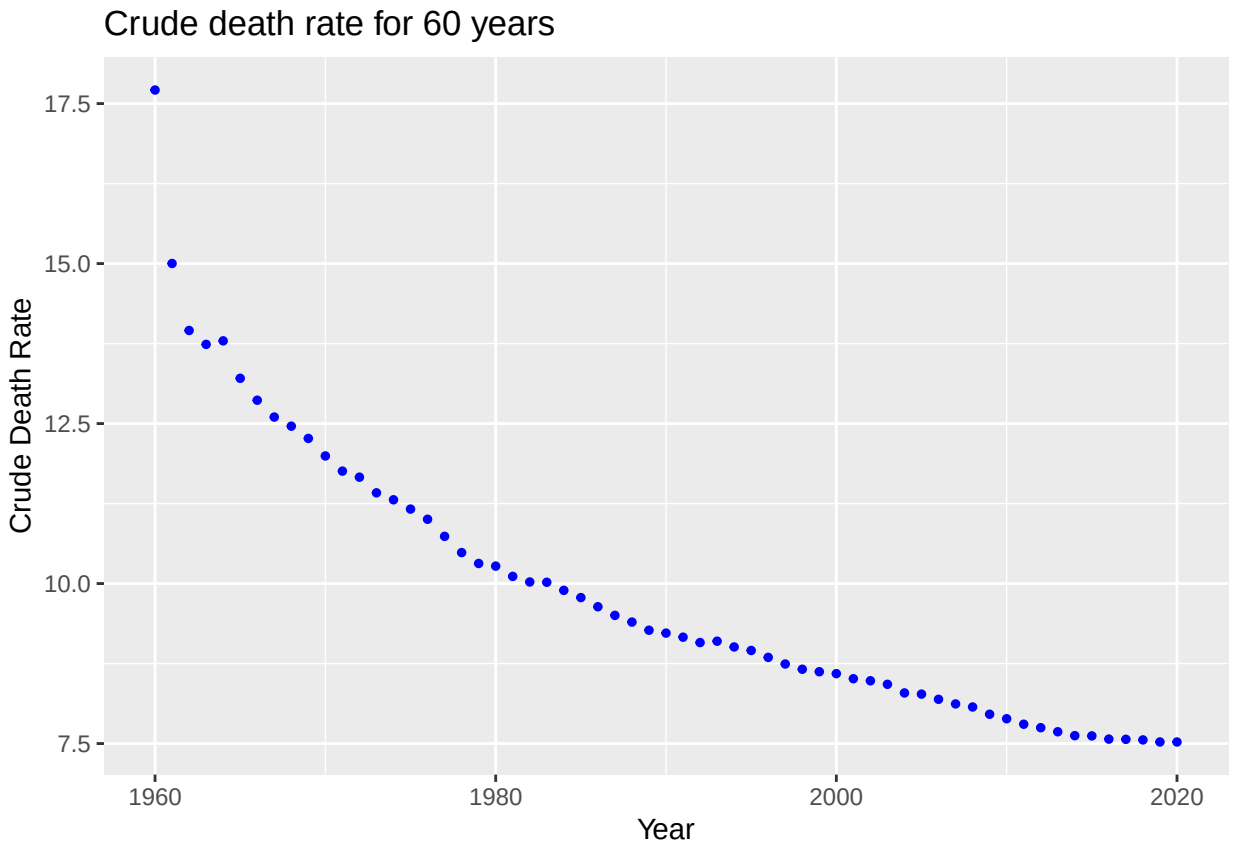
Median Age

```
medn_age <- read.csv('../Data/Cleaned_Data/med_age_cleaned.csv')
death_rate_impt <- read.csv('../Data/ImputedData/death_rate_imputed.csv')

df_combine <- merge(death_rate_impt, medn_age,  by.x = c("DRCountryCode","DRYear"), by.y = c("MedAgeCC"
df_combine <- subset(df_combine, select = -c(X.x, X.y, MedAgeCountry,death_per_1000))

names(df_combine)[names(df_combine)=="DRCountryName"] <- "MedAgeCountry"
names(df_combine)[names(df_combine)=="DRCountryCode"] <- "MedAgeCC"
names(df_combine)[names(df_combine)=="DRYear"] <- "MedAgeYear"

#Rows added, now impute.
countryUnique <- unique(df_combine[c("MedAgeCC")])
df_med_age <- df_combine[0,]
rowsize = nrow(countryUnique)

for ( i in 1:rowsize)
{
ccode = countryUnique[i,]
df_combine_tmp <- df_combine[df_combine$MedAgeCC==ccode,]
  if ( nrow(unique(df_combine_tmp[c("MedAgeVal")]))  > 2) {
  df_combine_tmp$MedAgeVal <- na_interpolation(df_combine_tmp$MedAgeVal)
  df_med_age <- rbind(df_med_age, df_combine_tmp)
  }
}

write.csv(df_med_age, '../Data/ImputedData/med_age_imputed.csv', row.names = FALSE)
```

```
require(ggplot2)
```

```
## Loading required package: ggplot2
```

```
df_death_rate <- read.csv('../Data/ImputedData/death_rate_imputed.csv')
#df_death_rate <- subset(df_death_rate, select = -c(Series.Name, Series.Code))
df_death_rate_WLD <- df_death_rate[df_death_rate$DRCountryCode=='WLD',]

ggplot(df_death_rate_WLD, aes(DRYear, death_per_1000)) +
           geom_point(na.rm=TRUE, color="blue", size=1) +
           ggtitle("Crude death rate for 60 years") +
           xlab("Year") + ylab("Crude Death Rate")
```


Crude death rate for 60 years
Crude Death Rate
17.5
15.0
12.5
10.0
7.5
1960
1980
2000
2020
Year