Data Tidy Notes

Chen Chen

11/16/2017

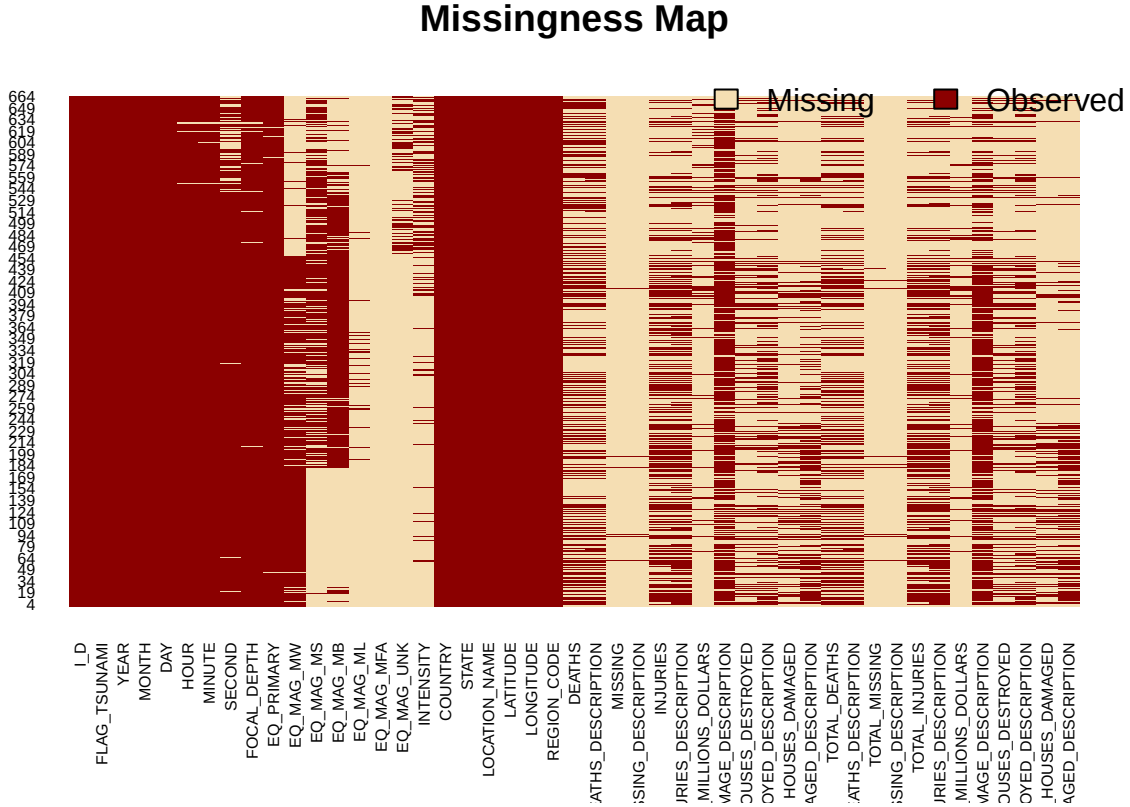
To tidy the data and inspect the missing data, I used three libraries loaded: Amelia, tidyr, and dplyr.

```
library(Amelia)
library(tidyr)
library(dplyr)
```

I have three data files to clean up for this analysis. The first one is the significant earthquake database from NOAA. And the other two are the population data and gdp data for the past 57 years, downloaded from the worldbank.

I first loaded the earthquake data and plotted the missing values :

```
eq_data <- read.table(file = 'results.tsv', sep = '\t',
                      header = TRUE, stringsAsFactors = F)

missmap(eq_data,x.cex=0.5,y.cex=0.5,rank.order = F)
```



Looking at the missing field map, I noticed that the data is missing hour, minute and second for a few earthquakes. I converted them to 0, in order to generate a date variable later in the data frame. Then I re-organized the data frame according to the country

```
eq_data[c("HOUR","MINUTE","SECOND")][is.na(eq_data[c("HOUR","MINUTE","SECOND")])] <- 0

# Convert Date from character to date in R
```

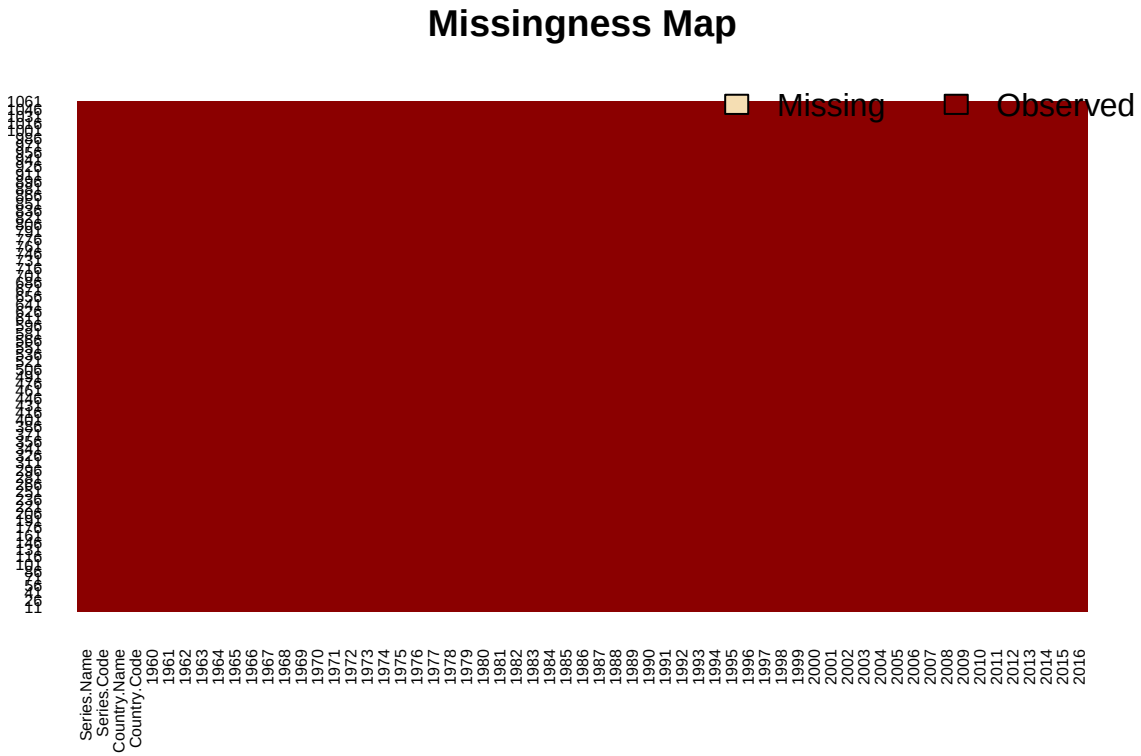
```r
eq_date <- eq_data %>% unite(DATE,YEAR:DAY,sep="-",remove=T)
eq_time <- eq_date %>% unite(E_TIME,HOUR:SECOND,sep=":",remove=T)
eq_origin <- eq_time %>% unite(ORIGIN,DATE:E_TIME,sep=" ",remove=T)
eq_origin$ORIGIN <- as.POSIXct(eq_origin$ORIGIN,tz="GMT")

# reorder the columns and arrange the data frame by country
country_eq <- eq_origin %>%
  select(COUNTRY:LONGITUDE,FOCAL_DEPTH,REGION_CODE,ORIGIN,everything()) %>%
  arrange(COUNTRY)

head(country_eq,3)
```

```
##       COUNTRY STATE                                LOCATION_NAME LATITUDE
## 1 AFGHANISTAN       AFGHANISTAN:  ROSTAQ; TAJIKISTAN:  DUSHANBE   37.075
## 2 AFGHANISTAN        AFGHANISTAN-TAJIKISTAN:  YAR HUSAIN, ASTOR   36.479
## 3 AFGHANISTAN                  AFGHANISTAN:  BADAKHSHAN, TAKHAR   37.106
##   LONGITUDE FOCAL_DEPTH REGION_CODE              ORIGIN  I_D FLAG_TSUNAMI
## 1    70.089          33          40 1998-02-04 14:33:21 5485
## 2    71.086         236          40 1998-02-20 12:18:06 5486
## 3    70.110          33          40 1998-05-30 06:22:28 5495
##   EQ_PRIMARY EQ_MAG_MW EQ_MAG_MS EQ_MAG_MB EQ_MAG_ML EQ_MAG_MFA EQ_MAG_UNK
## 1        5.9       5.9       6.1       5.6        NA         NA         NA
## 2        6.4       6.4       5.7       5.8        NA         NA         NA
## 3        6.6       6.6       6.9       5.9        NA         NA         NA
##   INTENSITY DEATHS DEATHS_DESCRIPTION MISSING MISSING_DESCRIPTION INJURIES
## 1        NA   2323                  4      NA                  NA      818
## 2        NA      1                  1      NA                  NA       11
## 3        NA   4700                  4      NA                  NA       NA
##   INJURIES_DESCRIPTION DAMAGE_MILLIONS_DOLLARS DAMAGE_DESCRIPTION
## 1                    3                     NA                  4
## 2                    1                     NA                  2
## 3                    4                     10                  3
##   HOUSES_DESTROYED HOUSES_DESTROYED_DESCRIPTION HOUSES_DAMAGED
## 1             8094                            4             NA
## 2               35                            1             NA
## 3               NA                            4             NA
##   HOUSES_DAMAGED_DESCRIPTION TOTAL_DEATHS TOTAL_DEATHS_DESCRIPTION
## 1                         NA         2323                       NA
## 2                         NA            1                       NA
## 3                         NA         4700                        4
##   TOTAL_MISSING TOTAL_MISSING_DESCRIPTION TOTAL_INJURIES
## 1            NA                        NA            818
## 2             1                        NA             11
## 3            NA                        NA             NA
##   TOTAL_INJURIES_DESCRIPTION TOTAL_DAMAGE_MILLIONS_DOLLARS
## 1                          3                            NA
## 2                          1                            NA
## 3                          4                            10
##   TOTAL_DAMAGE_DESCRIPTION TOTAL_HOUSES_DESTROYED
## 1                        4                   8094
## 2                        2                     35
## 3                        3                     NA
##   TOTAL_HOUSES_DESTROYED_DESCRIPTION TOTAL_HOUSES_DAMAGED
## 1                                  4                   NA
```

```
## 2                                    1                    NA
## 3                                    4                    NA
##   TOTAL_HOUSES_DAMAGED_DESCRIPTION
## 1                               NA
## 2                               NA
## 3                               NA
```

Next, I cleaned the world GDP data.

```
world_gdp <- read.csv(file='world_gdp_Data.csv',sep=",",
                      col.names=c("Series.Name","Series.Code",
                      "Country.Name","Country.Code",1960:2016),
                      stringsAsFactors = F,check.names = F)
# plot the missing values
missmap(world_gdp,x.cex=0.5,y.cex=0.5,rank.order=F)
```




```
# remove last 5 rows and gather the data
tail(world_gdp,6)
```

```
##                                        Series.Name    Series.Code
## 1056                         GDP (constant 2010 US$) NY.GDP.MKTP.KD
## 1057
## 1058
## 1059
## 1060 Data from database: World Development Indicators
## 1061                           Last Updated: 10/30/2017
##      Country.Name Country.Code            1960            1961
## 1056     Zimbabwe          ZWE 3349805801.0933 3561384803.7092
## 1057
## 1058
## 1059
```

```
## 1060
## 1061
##                  1962             1963             1964             1965
## 1056 3612471831.90166 3838047018.29173 3795591622.24677 3981976828.04014
## 1057
## 1058
## 1059
## 1060
## 1061
##                  1966             1967             1968             1969
## 1056 4042627512.75241 4380874517.76023 4467183658.60973 5022375778.86567
## 1057
## 1058
## 1059
## 1060
## 1061
##                  1970             1971             1972            1973
## 1056 6155682449.07748 6704620764.88613 7263100570.93597 7452283618.9859
## 1057
## 1058
## 1059
## 1060
## 1061
##                  1974           1975             1976             1977
## 1056 7946008856.76807 7792553687.169 7828776508.44197 7291667388.52174
## 1057
## 1058
## 1059
## 1060
## 1061
##                  1978             1979            1980             1981
## 1056 7094287604.46678 7328188778.12031 8384963717.6939 9435216047.66681
## 1057
## 1058
## 1059
## 1060
## 1061
##                  1982             1983             1984             1985
## 1056 9683767674.54837 9837284972.16778 9649652522.95076 10319761812.1881
## 1057
## 1058
## 1059
## 1060
## 1061
##                  1986             1987             1988             1989
## 1056 10536376618.6647 10657622624.4262 11462526198.6773 12058550789.5555
## 1057
## 1058
## 1059
## 1060
## 1061
##                  1990             1991             1992             1993
## 1056 12901268994.4586 13614939079.7948 12387474741.0302 12517723898.2569
## 1057
```

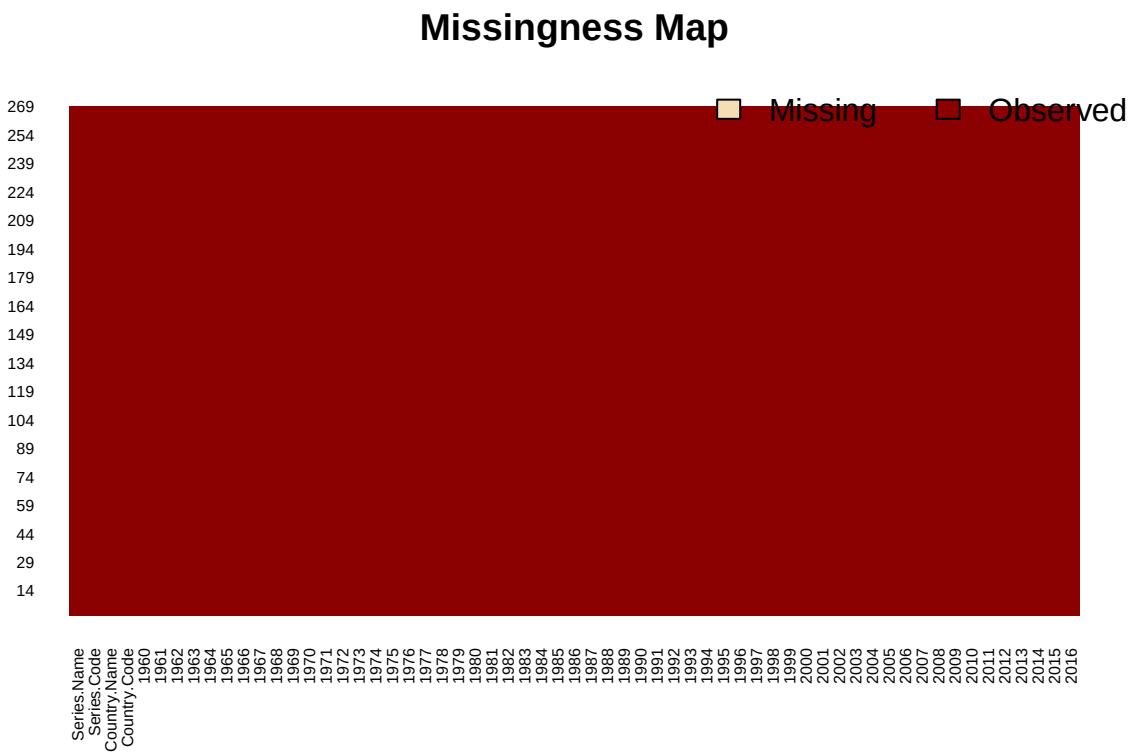
```
## 1058
## 1059
## 1060
## 1061
##                   1994              1995              1996              1997
## 1056 13673760578.2194 13695368641.2577 15114304254.0546 15519457413.2946
## 1057
## 1058
## 1059
## 1060
## 1061
##                   1998              1999              2000              2001
## 1056 15967226648.0827 15836643331.6042 15352170381.7708 15573182540.8437
## 1057
## 1058
## 1059
## 1060
## 1061
##                   2002              2003              2004              2005
## 1056 14188100038.8483 11776821863.2026 11092878422.5658 10459354836.3551
## 1057
## 1058
## 1059
## 1060
## 1061
##                   2006             2007              2008             2009
## 1056 10097304786.3907 9728417219.9754 8009508376.58441 8707074504.40407
## 1057
## 1058
## 1059
## 1060
## 1061
##             2010              2011             2012              2013
## 1056 10052045200 11693792186.8512 13285220264.301 13985398474.7183
## 1057
## 1058
## 1059
## 1060
## 1061
##                   2014              2015              2016
## 1056 14372132601.1641 14576782196.1365 14677933169.4928
## 1057
## 1058
## 1059
## 1060
## 1061
```

```
world_gdp_data <- head(world_gdp,-5)
gdp_tidy <- world_gdp_data %>% gather("Year","Value",5:61)

# rearrange the data by country
gdp_order <- gdp_tidy %>% select(Country.Name,Country.Code,everything()) %>%
  arrange(Country.Name)
```

```
# spread the GDP columns into
gdp_spread <- gdp_order %>% unite(temp,Series.Name,Series.Code,sep="|") %>%
  spread(key=temp,value = Value)
```

Then I cleaned the population data similarly.

```
popul <- read.csv(file="world_population_Data.csv",sep=",",
                        col.names=c("Series.Name","Series.Code",
                                    "Country.Name","Country.Code",1960:2016),
                  stringsAsFactors = F,check.names = F)

missmap(popul,x.cex=0.5,y.cex=0.5,rank.order=F)
```




```
world_popu <- head(popul,-5)
popu_tidy <- world_popu %>% gather("Year","Population",5:61)
popu_order <- popu_tidy %>% arrange(Country.Name) %>%
  select(Country.Name,Country.Code,Year,everything())

# join the population and gdp data frames together
gdp_population <- left_join(gdp_spread,popu_order)
```

```
## Joining, by = c("Country.Name", "Country.Code", "Year")
```

```
colnames(gdp_population)
```

```
##  [1] "Country.Name"
##  [2] "Country.Code"
##  [3] "Year"
##  [4] "GDP (constant 2010 US$)|NY.GDP.MKTP.KD"
##  [5] "GDP (current US$)|NY.GDP.MKTP.CD"
##  [6] "GDP per capita (constant 2010 US$)|NY.GDP.PCAP.KD"
##  [7] "GDP per capita (current US$)|NY.GDP.PCAP.CD"
```

```
##  [8] "Series.Name"
##  [9] "Series.Code"
## [10] "Population"
```

```
head(gdp_population)
```

```
##   Country.Name Country.Code Year GDP (constant 2010 US$)|NY.GDP.MKTP.KD
## 1  Afghanistan          AFG 1960                                       ..
## 2  Afghanistan          AFG 1961                                       ..
## 3  Afghanistan          AFG 1962                                       ..
## 4  Afghanistan          AFG 1963                                       ..
## 5  Afghanistan          AFG 1964                                       ..
## 6  Afghanistan          AFG 1965                                       ..
##   GDP (current US$)|NY.GDP.MKTP.CD
## 1                537777811.111111
## 2                548888895.555556
## 3                546666677.777778
## 4                751111191.111111
## 5                800000044.444444
## 6                1006666637.77778
##   GDP per capita (constant 2010 US$)|NY.GDP.PCAP.KD
## 1                                                ..
## 2                                                ..
## 3                                                ..
## 4                                                ..
## 5                                                ..
## 6                                                ..
##   GDP per capita (current US$)|NY.GDP.PCAP.CD       Series.Name
## 1                           59.7773265083934 Population, total
## 2                           59.8781528089471 Population, total
## 3                           58.4928738323479 Population, total
## 4                           78.7827580362892 Population, total
## 5                           82.2084438594401 Population, total
## 6                           101.290471274167 Population, total
##   Series.Code Population
## 1 SP.POP.TOTL    8996351
## 2 SP.POP.TOTL    9166764
## 3 SP.POP.TOTL    9345868
## 4 SP.POP.TOTL    9533954
## 5 SP.POP.TOTL    9731361
## 6 SP.POP.TOTL    9938414
```