HW5

Jadon Fowler jaf582 5778191

2022-12-06

Instructions

Download a copy of this markdown and the .xlsx file attached to this assignment. Change the `author:` tag above to have your name and NAU's ID. Fill in the file with your solution to the proposed problems. Knit your final document to PDF and submit it through BBLearn in the assignment `[HW] Homework 5: EDA and data visualization` by the end of the day on **Tuesday, November 22 (11:59:00 PM)**.

Note: to knit your homework to PDF, you need to have MikTex installed. You can download and install MikTex from here. If you don't want to install MikTex, you can knit to HTML, open the file in the browser and print the page to PDF. You can also submit the `.Rmd` file.

For all the problems, please import the useful libraries in this chunk of code:

Problem 1: Importing and tidying

Download the UNICEF dataset attached to this assignment on BBLearn. This dataset represents the estimates for under-five, infant, and neonatal mortality in several countries. Mortality rate is expressed as the number of under-five deaths per 1,000 live births.

For the column names, notice that `U5MR` stands for `Under-five mortality rate`; `IRM` stants for `Infant mortality rate` and `NMR` stands for `Neonatal mortality rate`. The columns that has the `___.Deaths.____` show the absolute number of deaths in that year.

Also notice that the the file has a .xlsx extension. Make sure you save the file as .csv format before starting the assignment.

Step 1: import the .csv file into a `raw.mortality` dataframe. Make sure your dataframe have only the necessary data (remove rows that do not contain observations). *Note:* DO NOT DELETE the rows from the CSV file. Fix the problem in the code, not in the source file!

```
raw.mortality <- tail(as.data.frame(read_csv("unicef_allindicators.csv", col_names = FALSE)), -7)
```

```
## Rows: 592 Columns: 399
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (399): X1, X2, X3, X4, X5, X6, X7, X8, X9, X10, X11, X12, X13, X14, X15,...
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
col.names <- list(
  "iso.code",
  "country",
  "uncertainty.bounds"
)

categories <- c("U5MR", "IMR", "NMR", "U5deaths", "Ideaths", "Ndeaths")
year.range <- c(1950, 2015)
for (category in categories) {
  for (year in seq(year.range[1], year.range[2])) {
    col.names[length(col.names) + 1] <- paste(category, year, sep=".")
  }
}

colnames(raw.mortality) <- col.names
dim(raw.mortality)
```

```
## [1] 585 399
```

Step 2: take a look at the structure. Is it a wide or long format? Justify your answer.

The data looks *long* because it has many unnecessary columns.

```
nrow(raw.mortality)
```

```
## [1] 585
```

```
#[1] 585
ncol(raw.mortality)
```

```
## [1] 399
```

```
#[1] 399
```

Step 3: Subset the dataset to the median estimate for each country (i.e., drop rows representing lower and upper uncertainty bounds). Name this reduced dataset as sub.mortality (do not replace your original, raw dataset). Drop the uncertainty bounds factor variable because it is not necessary anymore (all the observations are in the `Median` factor).

```
sub.mortality <- raw.mortality[raw.mortality$uncertainty.bounds == "Median",]
sub.mortality <- sub.mortality[,!(names(sub.mortality) %in% c("uncertainty.bounds", "iso.code"))]
nrow(sub.mortality)
```

```
## [1] 195
```

```
ncol(sub.mortality)
```

```
## [1] 397
```

Step 4: Make the data tidy. You can use any resources we've studied in class to produce the final dataset, which should be named `mortality` and have four variables:

- `country`: a factor variable with the country names;
- `type`: a factor variable with six categories: `U5MR`, `IMR`, `NMR`, `U5deaths`, `Ideaths`, and `Ndeaths`;
- `year`: an integer variable with the year;
- `value`: a numeric variable with the value for the given country, year, and type;

Don't worry about missing values, we will deal with them in the subsequent problems.

```
# pivot longer, turning the column names into (type, year) pairs
mortality <- as.data.frame(pivot_longer(
  sub.mortality,
  # every column but the first 3
  cols = as.vector(unlist(tail(col.names, -3))),
  # separate by literal '.'
  names_sep = "\\.",
  names_to = c("type", "year"),
  # transform the names into nice types
  names_transform = list(type = as.factor,
                         year = as.integer),
  values_to = "value",
  values_transform = list(value = as.numeric)
))
head(mortality)
```

```
##       country type year value
## 1 Afghanistan U5MR 1950    NA
## 2 Afghanistan U5MR 1951    NA
## 3 Afghanistan U5MR 1952    NA
## 4 Afghanistan U5MR 1953    NA
## 5 Afghanistan U5MR 1954    NA
## 6 Afghanistan U5MR 1955    NA
```

Problem 2: Exploratory Data Analysis

In this problem, we will inspect the dataframe we just created to increase our understanding of the data.

Step 1: Let's investigate the missing values. Apparently, there is no available data for particular types in some countries in a given year. Write a code to explore the missing data. You **MUST** comment your code to explain what you're doing and why. Then, answer the following questions:

Note: the answer to the questions must be supported by the exploration code. If you provide an answer, but your code doesn't show how you got to that conclusion, you do NOT earn full points! Make sure your code is commented to help us to understand your thoughts.

Note 2: Quantity is better than quality, but you don't have to show your failure attempts. Try a handful of different explorations, keep in your answer only the ones that helps you to support the answers to the questions.

```
#fail mortality$value == NA
#fail is.na(mortality$value)
#fail mortality[is.na(mortality$value)]
# sum every true value in the vector from is.na()
sum(is.na(mortality$value))
```

```
## [1] 19262
```

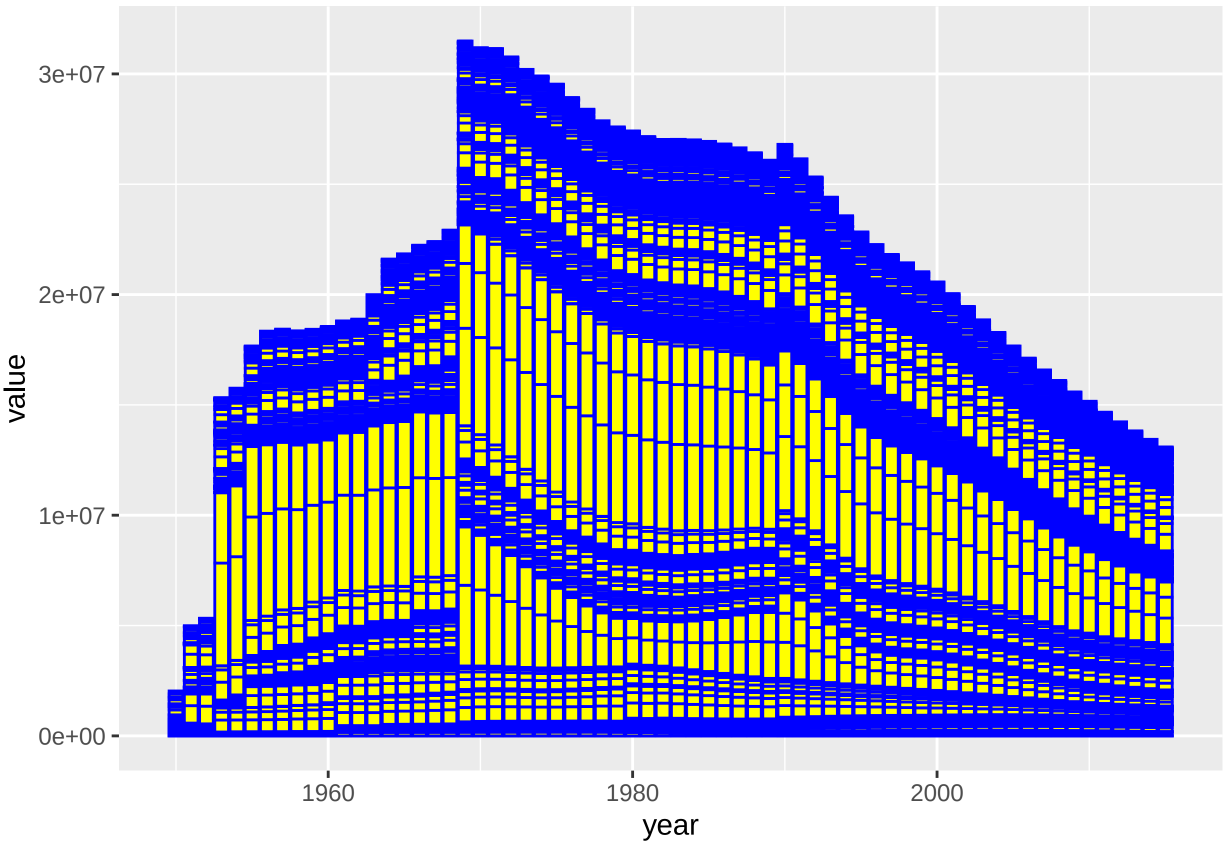
a. How many missing values are there?

19,262 missing values

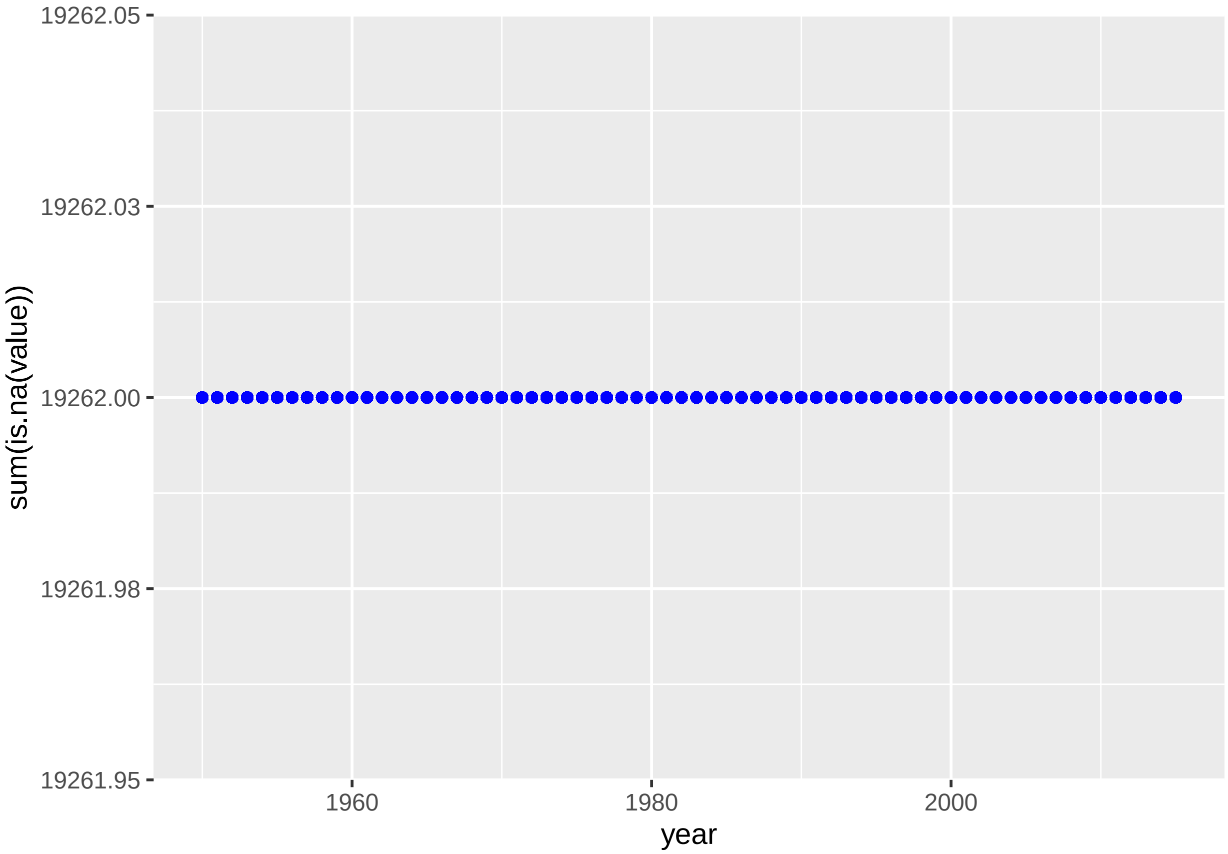
b. Which year range have more missing data (you can define what "range" means based on your exploration)? Why do you think the pattern you see is (or it is not) reasonable? Research a little bit to compare our exploration results to the knowledge from the real world (UNICEF history, neonatal care, mortality rates records, etc). Write a short paragraph (3-8 sentences) to explain your data and why it makes sense (or why it doesn't!).

```
# exploration for b - d
# trying to make a graph of NA values
ggplot(data=mortality, aes(x=year, y=value)) +
  geom_bar(stat="identity", color="blue", fill="yellow")
```

```
## Warning: Removed 19262 rows containing missing values (position_stack).
```



```
ggplot(data=mortality, aes(x=year, y=sum(is.na(value)))) +
  geom_point(stat="identity", color="blue", fill="yellow")
```

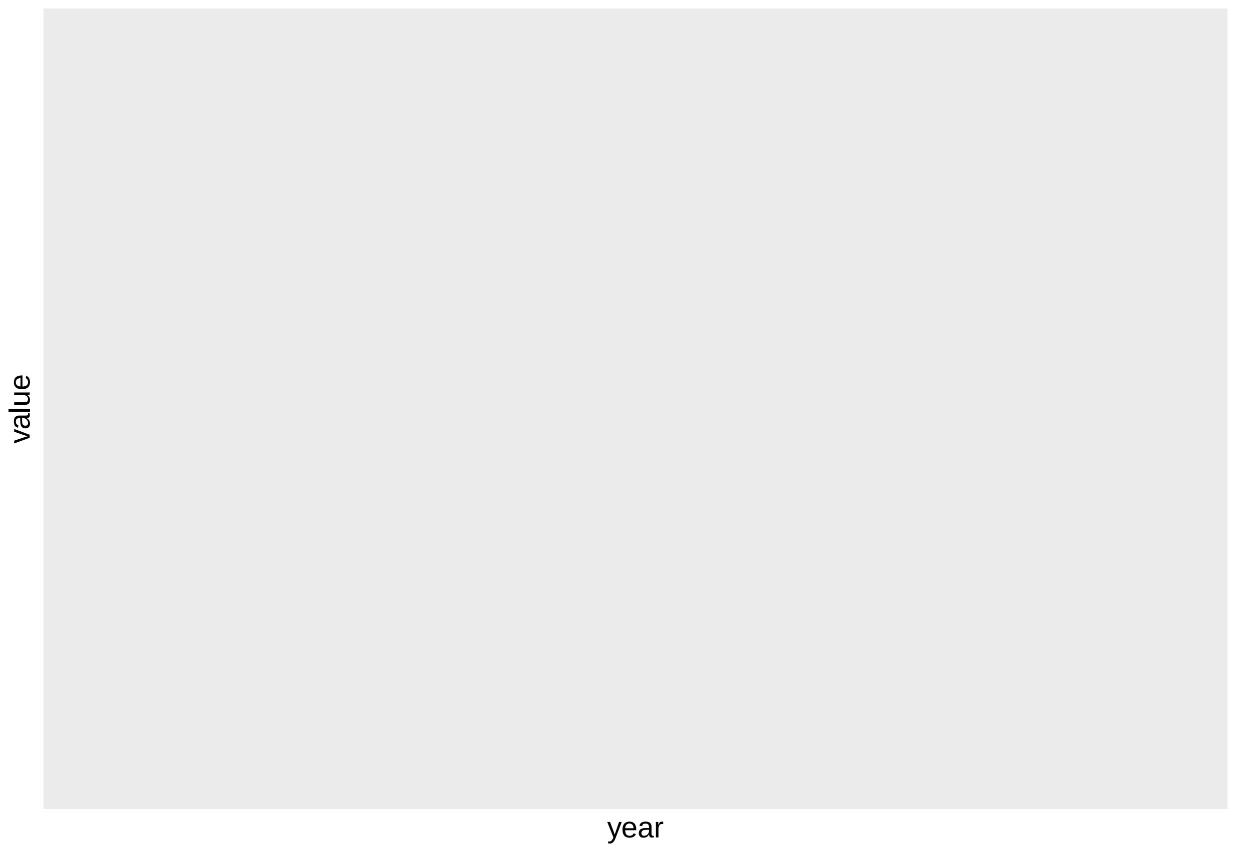


```
head(mortality[is.na(mortality$value),])
```

```
##       country type year value
## 1 Afghanistan U5MR 1950    NA
## 2 Afghanistan U5MR 1951    NA
## 3 Afghanistan U5MR 1952    NA
## 4 Afghanistan U5MR 1953    NA
## 5 Afghanistan U5MR 1954    NA
## 6 Afghanistan U5MR 1955    NA
```

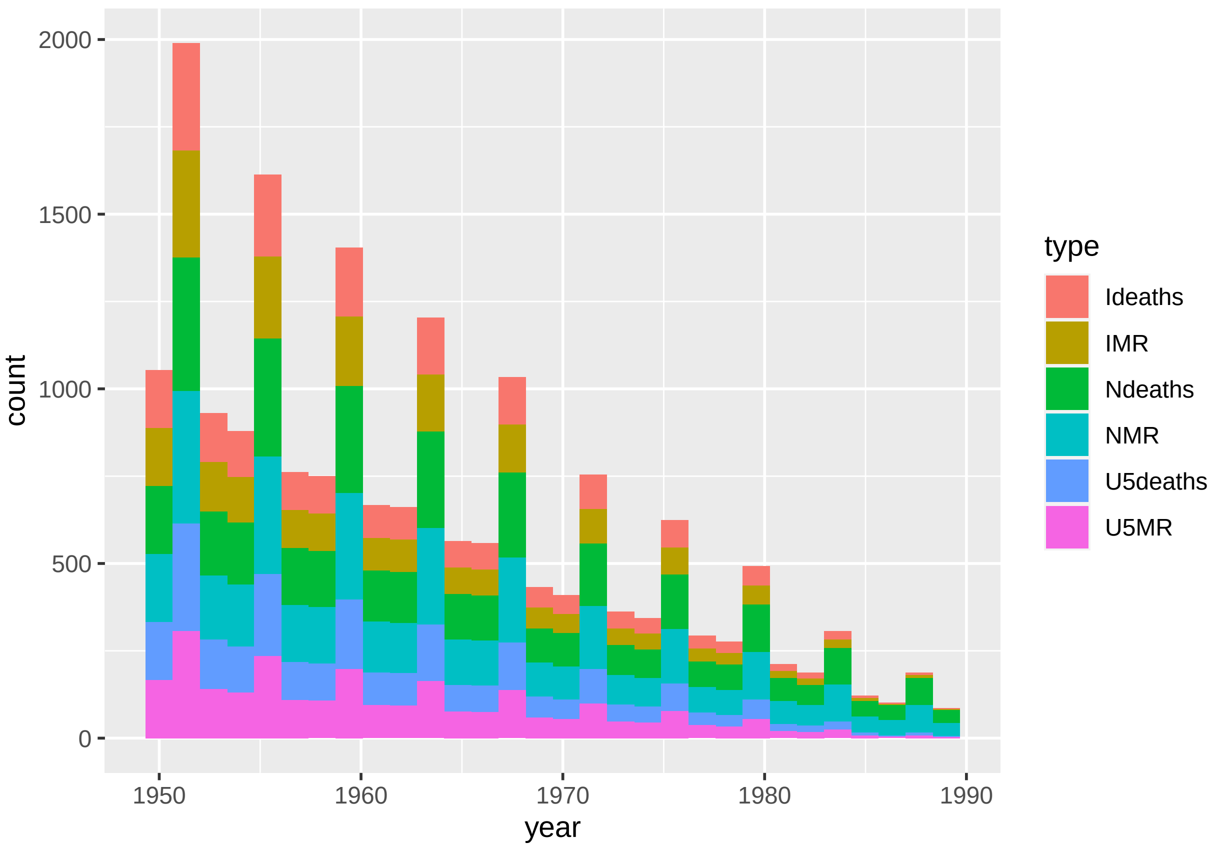
```
ggplot(data=mortality[is.na(mortality$value),], aes(x=year, y=value, fill=type)) +
  geom_bar(stat="identity", color="blue", fill="yellow")
```

```
## Warning: Removed 19262 rows containing missing values (position_stack).
```

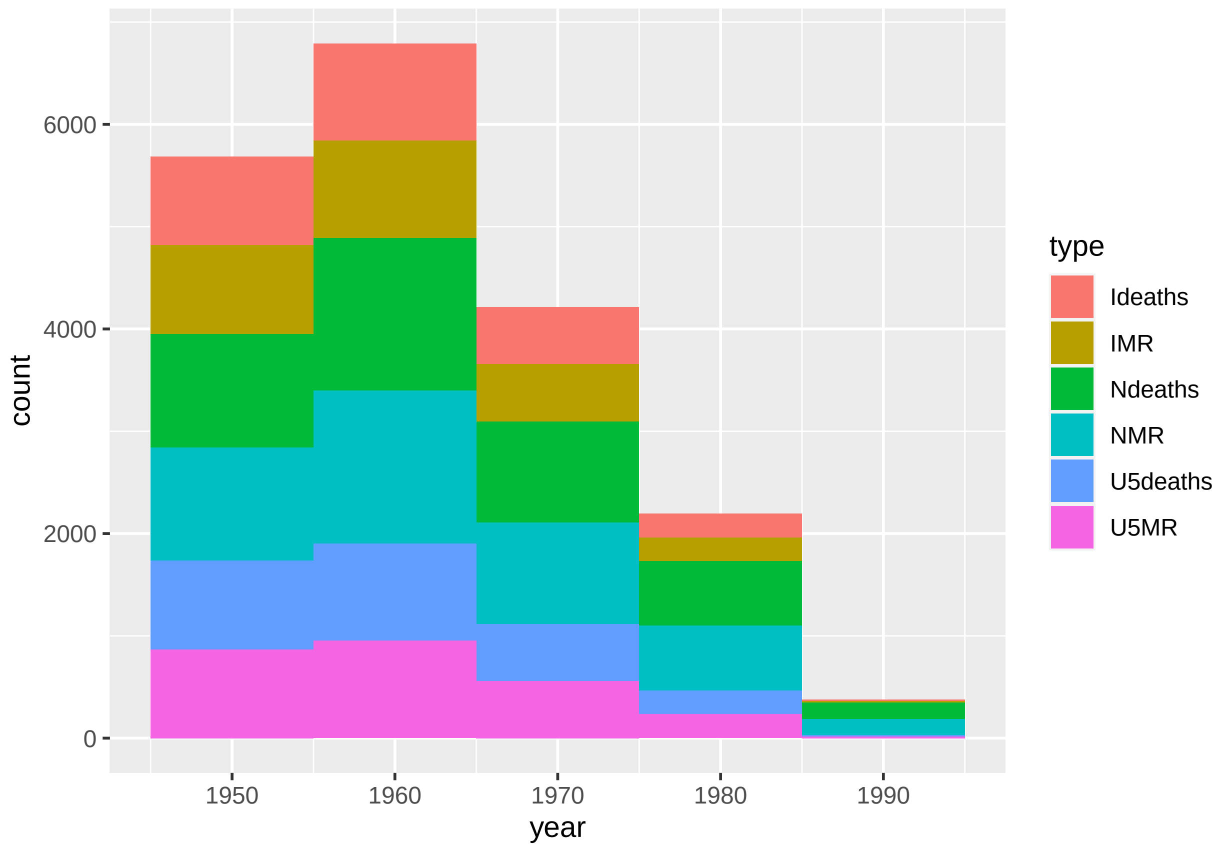


```
ggplot(data=mortality[is.na(mortality$value),], aes(x=year, fill=type)) +
  geom_histogram()
```

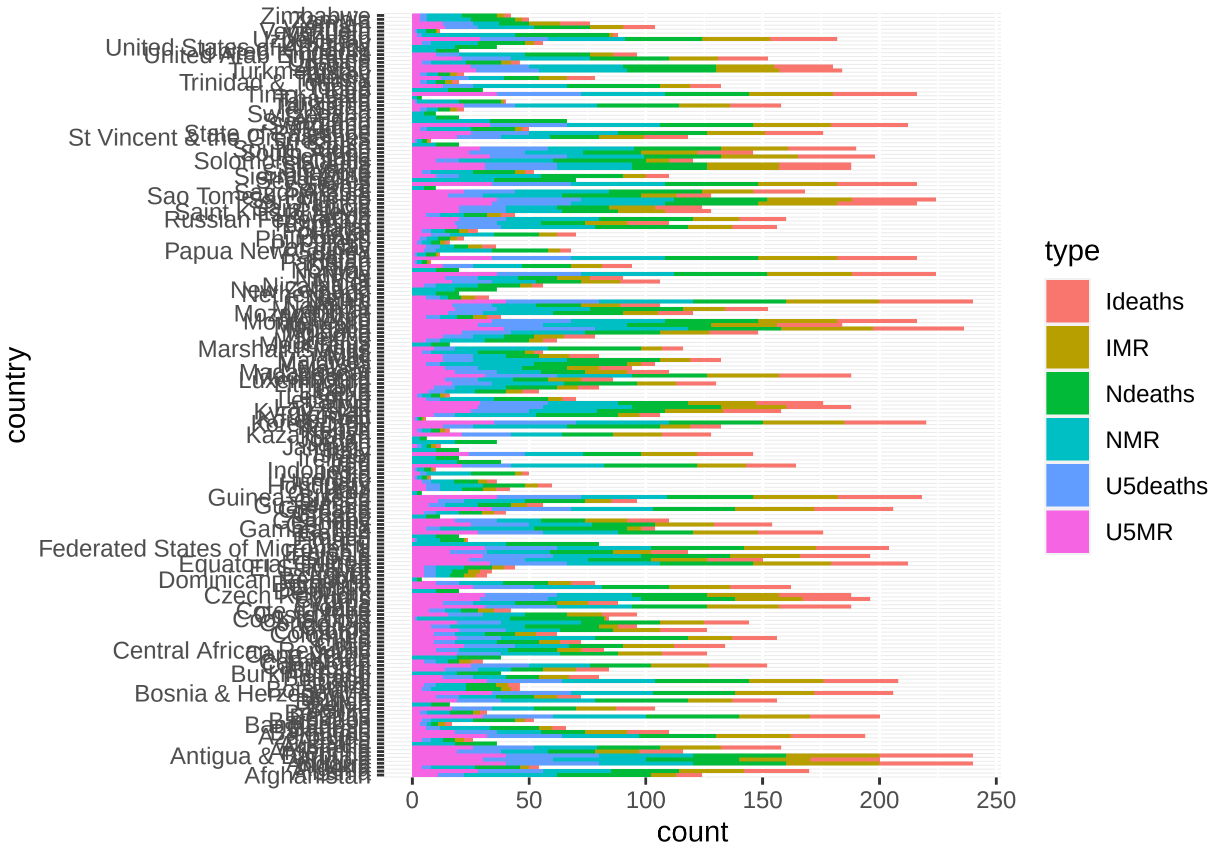
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
# plot bins of decades of the missing data - shows the 1960s have the most NAs
ggplot(data=mortality[is.na(mortality$value),], aes(x=year, fill=type)) +
  geom_histogram(binwidth=10)
```



```
ggplot(data=mortality[is.na(mortality$value),], aes(y=country, fill=type)) +
  geom_bar()
```



```
missing.mortality <- mortality[is.na(mortality$value),]
sum(missing.mortality$year)
```

```
## [1] 37807071
```

```
count(missing.mortality)
```

```
##       n
## 1 19262
```

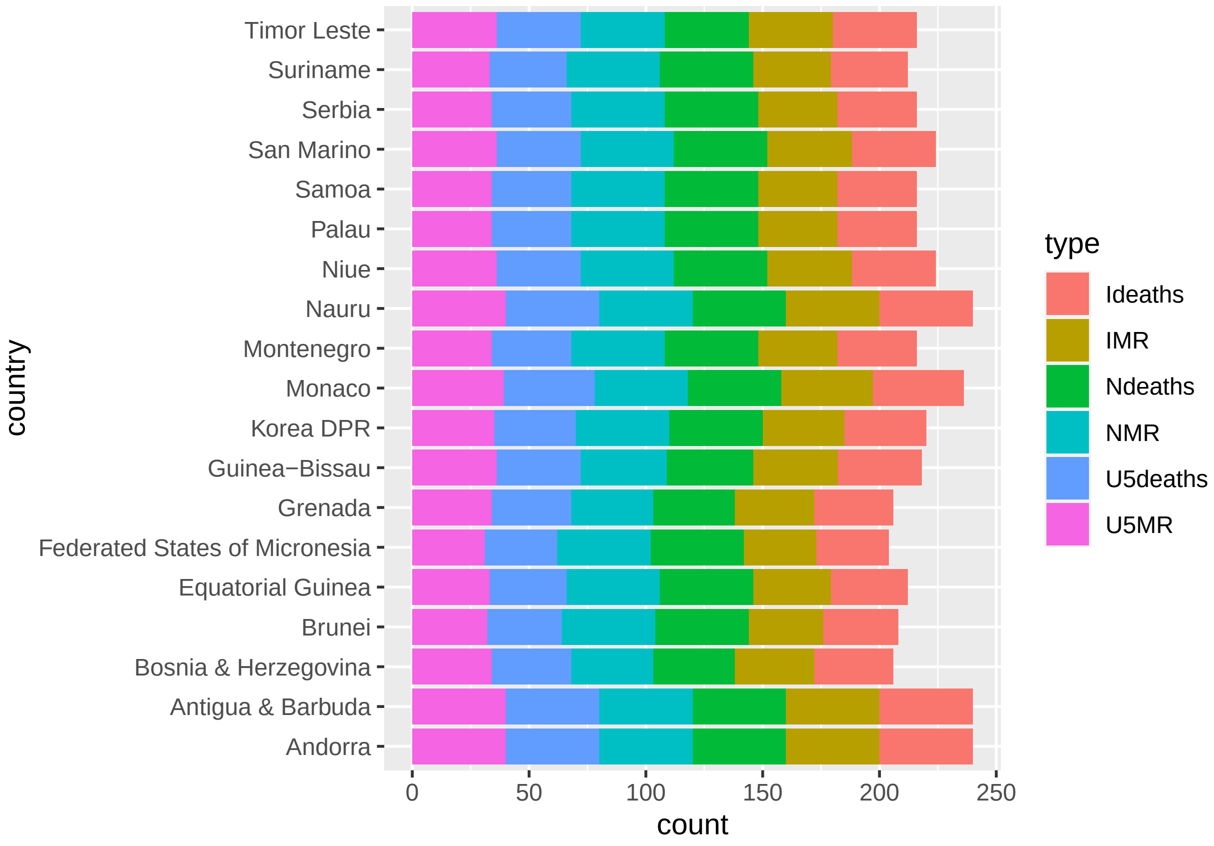
```
head(table(missing.mortality$country))
```

```
##
##       Afghanistan             Albania             Algeria             Andorra
##               124                 170                  54                 240
##            Angola Antigua & Barbuda
##               200                 240
```

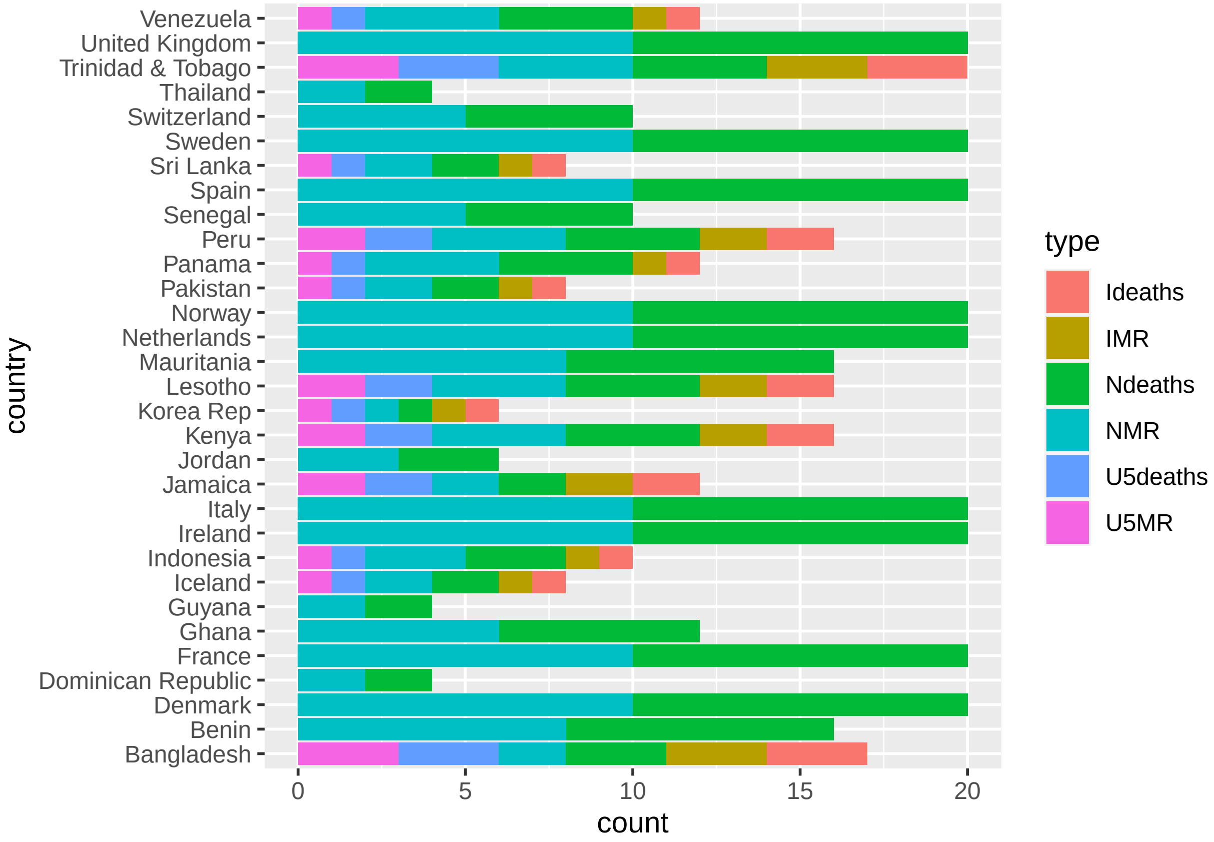
```
# count the frequency of countries by the amount of data they're missing
country.counts <- as.data.frame(table(missing.mortality$country))
colnames(country.counts)
```

```
## [1] "Var1" "Freq"
```

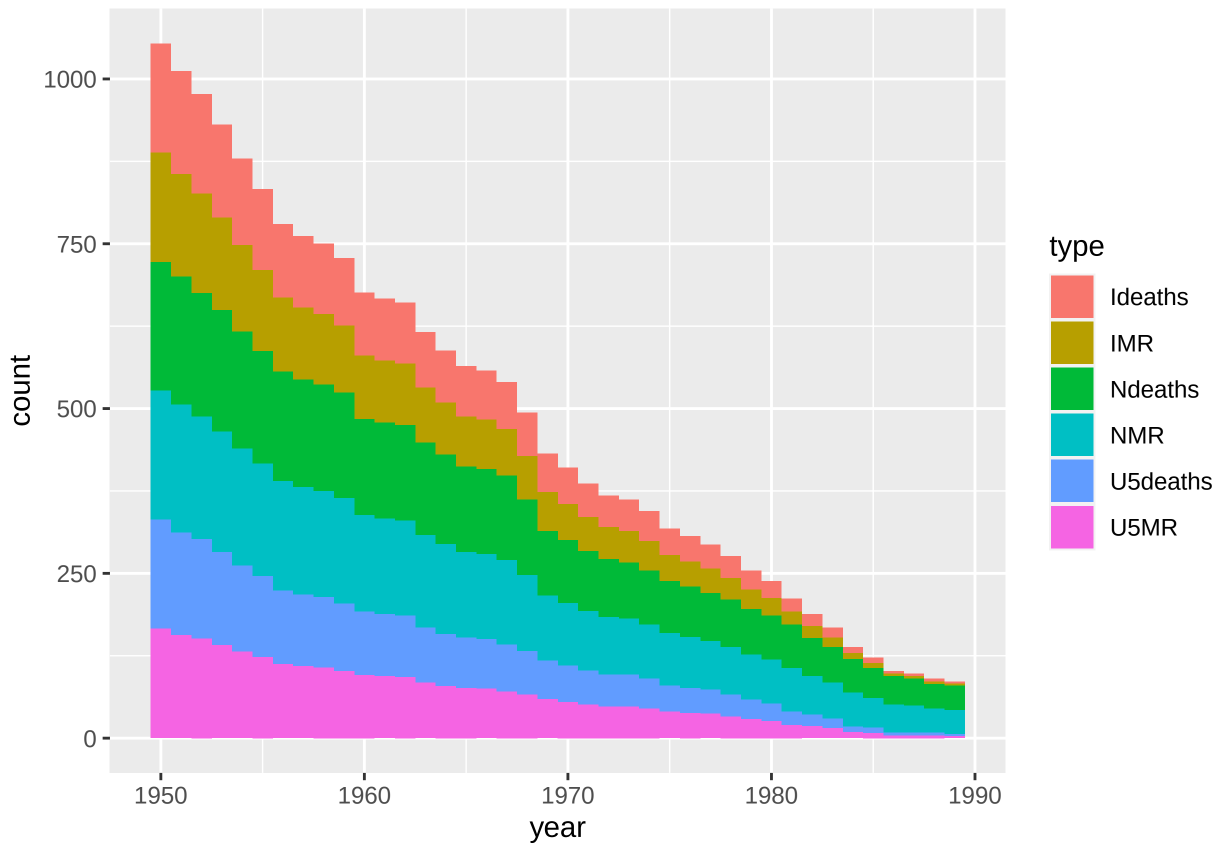
```
#ggplot(data=filter(country.counts, country.counts$Freq > 150), aes(y=Var1, fill=type)) +
#  geom_bar()
# get the countries with over 200 missing values
many.missing.countries <- as.data.frame(filter(country.counts, country.counts$Freq > 200))
# get the original mortality df entries that are missing data and are in the above list
many.missing.mortality <- missing.mortality[missing.mortality$country %in% many.missing.countries$Var1,]
# plot count of missing values
ggplot(data=many.missing.mortality, aes(y=country, fill=type)) +
  geom_bar()
```



```
# look at the countries with the least amount of missing data
no.missing.countries <- as.data.frame(filter(country.counts, country.counts$Freq <= 20))
no.missing.mortality <- missing.mortality[missing.mortality$country %in% no.missing.countries$Var1,]
ggplot(data=no.missing.mortality, aes(y=country, fill=type)) +
  geom_bar()
```



```
# good graph showing missing data by year by type
ggplot(data=mortality[is.na(mortality$value),], aes(x=year, fill=type)) +
  geom_histogram(binwidth=1)
```



The 1950s have the greatest amount of missing data. I think this data makes sense because there are many missing values in earliest decades and few missing values in the later decades. As time goes on, we are logging numbers more carefully. The last graph looks as I expected it to.

c. Is there any relationship between the amount of missing data and the type? Briefly explain.

Some types have very few missing entries (U5deaths, U5MR) in the last couple decades, and some types (NMR, Ndeaths) continue to have many missing values. Some stats may be reported on more accurately than others.

d. Is there any relationship between the amount of missing data and the country? Briefly explain.

```
unique(no.missing.mortality$country)
```

```
##  [1] "Benin"              "Bangladesh"         "Switzerland"
##  [4] "Denmark"            "Dominican Republic" "Spain"
##  [7] "France"             "United Kingdom"     "Ghana"
## [10] "Guyana"             "Indonesia"          "Ireland"
## [13] "Iceland"            "Italy"              "Jamaica"
## [16] "Jordan"             "Kenya"              "Korea Rep"
## [19] "Sri Lanka"          "Lesotho"            "Mauritania"
## [22] "Netherlands"        "Norway"             "Pakistan"
## [25] "Panama"             "Peru"               "Senegal"
## [28] "Sweden"             "Thailand"           "Trinidad & Tobago"
## [31] "Venezuela"
```

```
length(unique(no.missing.mortality$country)) # 31 countries with <= 20 missing data points
```

```
## [1] 31
```

```
unique(many.missing.mortality$country)
```

```
##  [1] "Andorra"                        "Antigua & Barbuda"
##  [3] "Bosnia & Herzegovina"           "Brunei"
##  [5] "Federated States of Micronesia" "Guinea-Bissau"
##  [7] "Equatorial Guinea"              "Grenada"
##  [9] "Monaco"                         "Montenegro"
## [11] "Niue"                           "Nauru"
## [13] "Palau"                          "Korea DPR"
## [15] "San Marino"                     "Serbia"
## [17] "Suriname"                       "Timor Leste"
## [19] "Samoa"
```

```
length(unique(many.missing.mortality$country)) # 19 countries with > 200 missing data points
```

```
## [1] 19
```

Wealthier countries look like they're more likely to have mostly complete data. In the list of countries missing a lot of data, there are many islands and very small countries.

Problem 3: Data visualization

In this final problem, let's focus on data from the 90s (because they are complete)!

Step 1: Filter out all the observations before 1990. Name the subset dataframe as `mortality90`. Show that we don't have any missing data in the new dataframe.

```
# filter year 1990 && not NA
head(mortality90 <- mortality[mortality$year >= 1990,])
```

```
##        country type year value
## 41 Afghanistan U5MR 1990 181.0
## 42 Afghanistan U5MR 1991 174.2
## 43 Afghanistan U5MR 1992 167.8
## 44 Afghanistan U5MR 1993 162.0
## 45 Afghanistan U5MR 1994 156.8
## 46 Afghanistan U5MR 1995 152.3
```

```
head(mortality90 <- mortality90[!is.na(mortality90$value),])
```

```
##        country type year value
## 41 Afghanistan U5MR 1990 181.0
## 42 Afghanistan U5MR 1991 174.2
## 43 Afghanistan U5MR 1992 167.8
## 44 Afghanistan U5MR 1993 162.0
## 45 Afghanistan U5MR 1994 156.8
## 46 Afghanistan U5MR 1995 152.3
```

Step 2: Let's investigate this data a little.

a. Write an R code that shows how many under-five, infant, and neonatal deaths occurred in total in the world in the years 1990, 1995, 2000, 2005, and 2015. Which age range had the largest number of death?

The 1990s had the most amount of deaths, and the amount has gone down ever since.

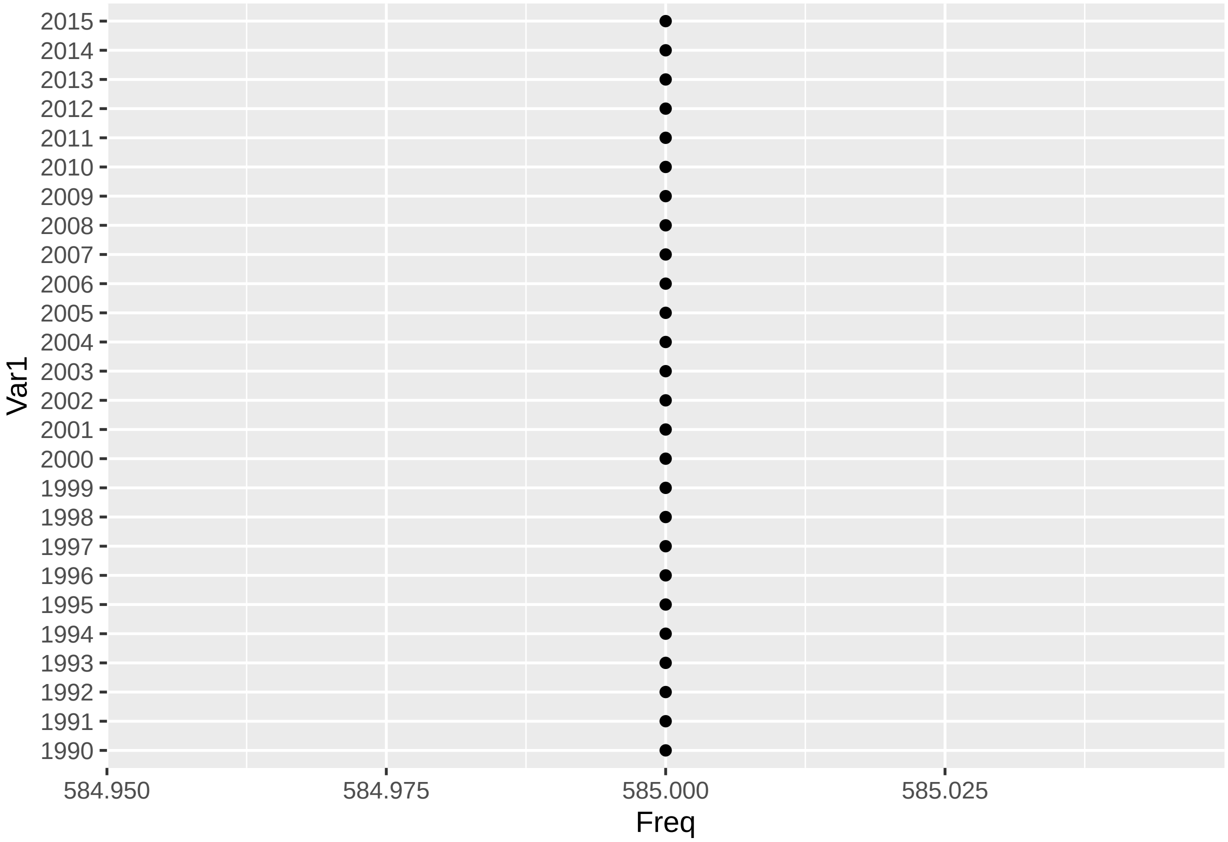
```
select.cols <- c("U5deaths", "Ideaths", "Ndeaths")
head(selected.types <- mortality90[mortality90$type %in% select.cols,])
```

```
##         country     type year  value
## 239 Afghanistan U5deaths 1990 100437
## 240 Afghanistan U5deaths 1991 101417
## 241 Afghanistan U5deaths 1992 104899
## 242 Afghanistan U5deaths 1993 109625
## 243 Afghanistan U5deaths 1994 113758
## 244 Afghanistan U5deaths 1995 116169
```

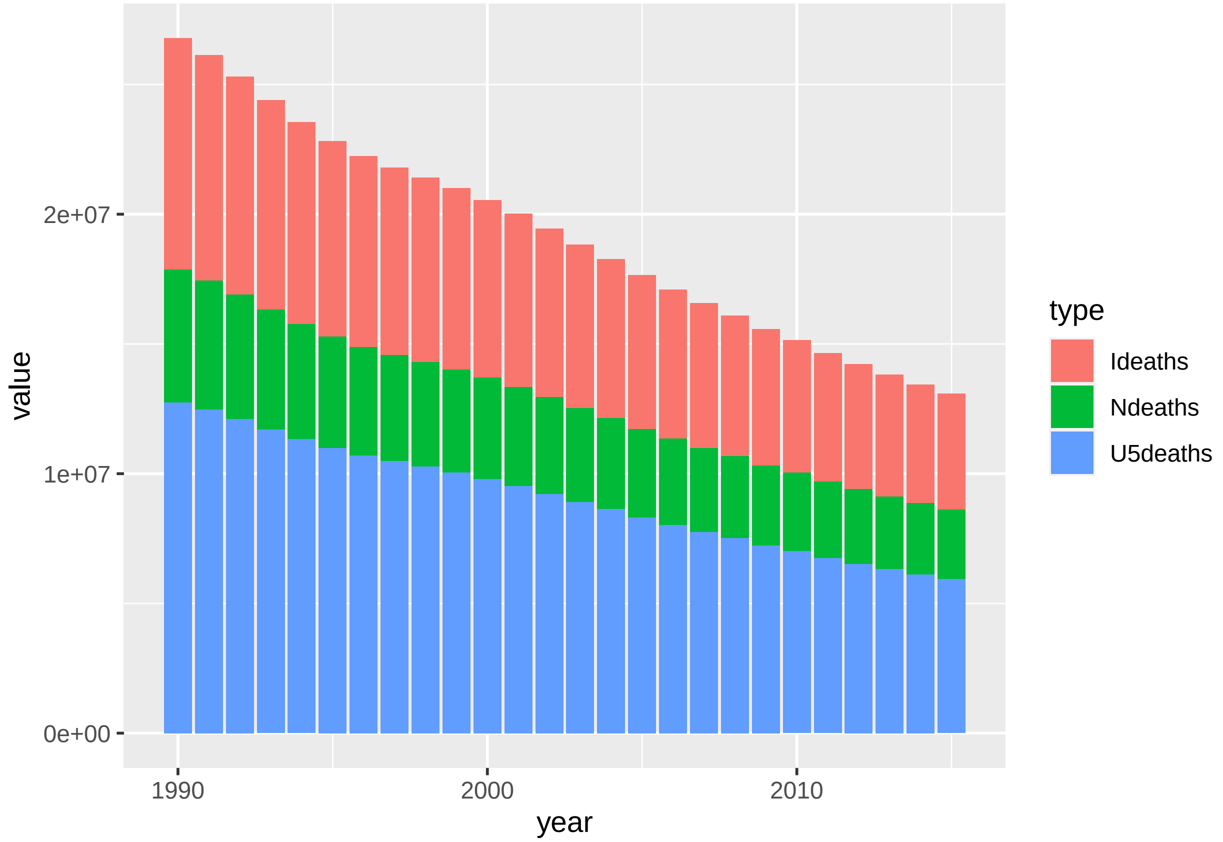
```
head(year.freq <- as.data.frame(table(selected.types$year)))
```

```
##   Var1 Freq
## 1 1990  585
## 2 1991  585
## 3 1992  585
## 4 1993  585
## 5 1994  585
## 6 1995  585
```

```
ggplot() +
  geom_point(data=year.freq, aes(y=Var1,x=Freq))
```



```
#select.years <- c(1990, 1995, 2000, 2005, 2015)
ggplot(data=selected.types, aes(x=year, y=value, fill=type)) +
  geom_col()
```



b. Write an R code that shows which countries have the largest mortality rates for neonatal? You can decide how many countries make the list of "largest mortality rates" based on your explorations. Provide an explanation about your reasoning.

```
ndeaths <- mortality90[mortality90$type == "NMR",]
sum(is.na(ndeaths$value))
```

```
## [1] 0
```

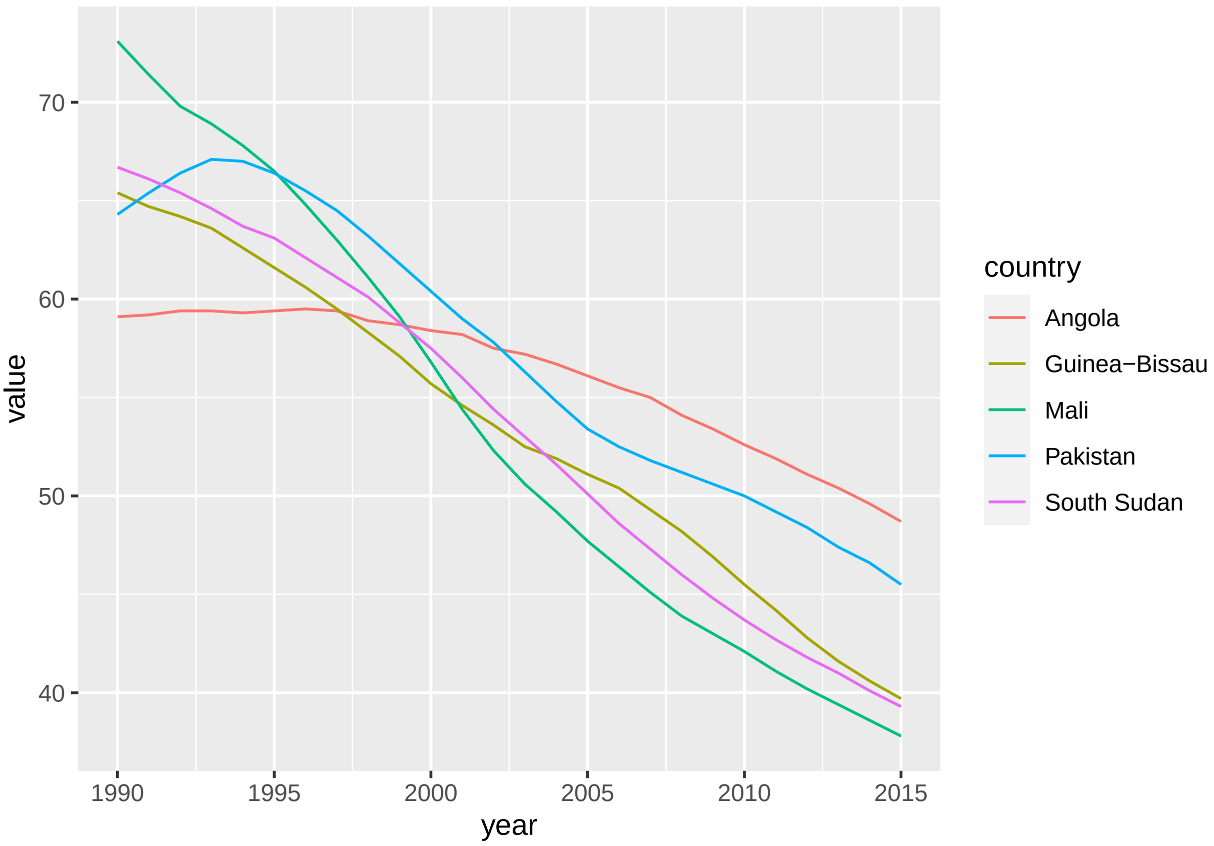
```
head(ndeaths)
```

```
##         country type year value
## 173 Afghanistan  NMR 1990  52.8
## 174 Afghanistan  NMR 1991  51.9
## 175 Afghanistan  NMR 1992  50.9
## 176 Afghanistan  NMR 1993  49.9
## 177 Afghanistan  NMR 1994  49.1
## 178 Afghanistan  NMR 1995  48.2
```

```
country.freq <- as.data.frame(table(ndeaths$country))
ndeaths.by.country <- aggregate(x=ndeaths$value, by=list(ndeaths$country), FUN=mean)
sorted.deaths <- ndeaths.by.country[order(ndeaths.by.country$x),]
many.deaths <- sorted.deaths[sorted.deaths$x > 50,]
head(select.countries <- many.deaths$Group.1)
```

```
## [1] "Guinea-Bissau" "South Sudan"   "Mali"          "Angola"
## [5] "Pakistan"
```

```
ndeaths.top <- ndeaths[ndeaths$country %in% select.countries,]
ggplot(data=ndeaths.top, aes(x=year, y=value, color=country)) +
  geom_line()
```



Countries with an average neonatal mortality rate > 50: [1] “Guinea-Bissau” “South Sudan” “Mali” “Angola” “Pakistan”
The last graph shows the rate per year for each country.

Step 3: Choose one scenario from the data. Write a question about the data and answer your question with a communication plot. Make this plot nice for other people to see. Consider: color scheme, theme, labels, shape and size of elements, etc. You plot must include at least three variables, but doesn’t have to include all the data. You will be graded based on how neat your plot is and how well it demonstrate the answer to your question.

```
select.cols <- c("U5deaths", "Ideaths", "Ndeaths")
head(selected.types <- mortality90[mortality90$type %in% select.cols,])
```

```
##         country     type year  value
## 239 Afghanistan U5deaths 1990 100437
## 240 Afghanistan U5deaths 1991 101417
## 241 Afghanistan U5deaths 1992 104899
## 242 Afghanistan U5deaths 1993 109625
## 243 Afghanistan U5deaths 1994 113758
## 244 Afghanistan U5deaths 1995 116169
```

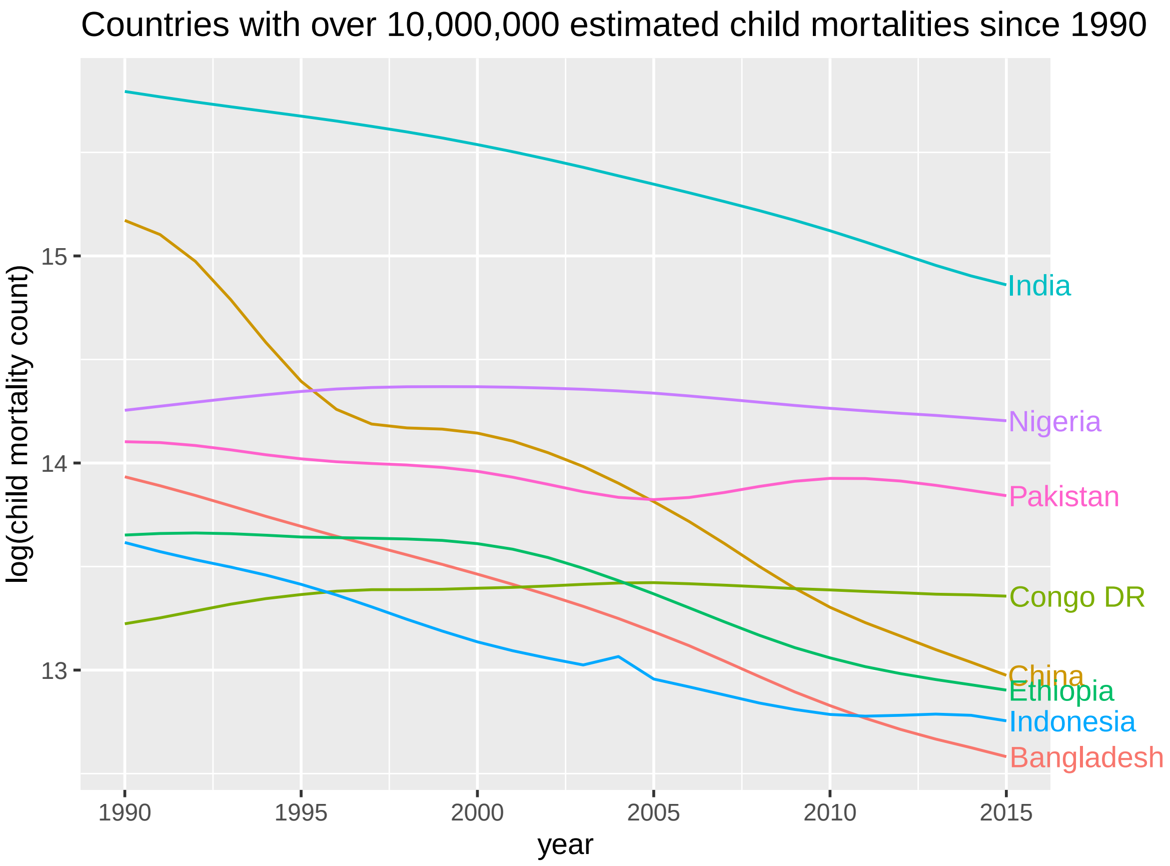
```
deaths.by.country <- aggregate(x=selected.types$value, by=list(selected.types$country), FUN=sum)
many.deaths <- deaths.by.country[deaths.by.country$x > 10000000,]

select.countries <- many.deaths$Group.1

deaths.top <- mortality90[mortality90$country %in% select.countries,]
head(deaths.per.year <- aggregate(x=deaths.top$value, by=list(deaths.top$country, deaths.top$year), FUN=
```

```
##      Group.1 Group.2         x
## 1 Bangladesh    1990 1125236.8
## 2      China    1990 3880691.6
## 3   Congo DR    1990  553295.9
## 4   Ethiopia    1990  849308.1
## 5      India    1990 7233061.5
## 6  Indonesia    1990  819255.2
```

```
head(last.deaths.per.year <- tapply(deaths.per.year$x, deaths.per.year$Group.1, last))
```

```
## Bangladesh      China   Congo DR   Ethiopia      India  Indonesia
##   291273.6   431484.4   632351.9   401675.3  2843267.3   346314.5
```

```
last.year <- max(mortality$year)
ggplot() +
  geom_line(data=deaths.per.year, aes(x=Group.2, y=log(x), color=Group.1)) +
  geom_text(aes(label = names(last.deaths.per.year), x = last.year, colour = names(last.deaths.per.year)
  # Allow labels to bleed past the canvas boundaries
  coord_cartesian(clip = 'off') +
  # Remove legend & adjust margins to give more space for labels
  # Remember, the margins are t-r-b-l
  theme(legend.position = 'none',
        plot.margin = margin(0.1, 2.6, 0.1, 0.1, "cm")) +
  ggtitle("Countries with over 10,000,000 estimated child mortalities since 1990") +
  ylab("log(child mortality count)") +
  xlab("year")
```



Which countries experiencing the worst child mortality have not improved over the last 30 years?

India, China, Ethiopia, Indonesia, and Bangladesh have greatly improved their total child mortality over the last 30 years. Nigeria, Pakistan, and Congo DR have not improved.