Megan Robertson & Ian Bachli Coding Exercise

#Ian Bachli Tidyverse Exercise  
#Loading these two packages makes coding easier and can save you time in the long run.   
  
library(tidyverse)

## ── Attaching packages ───────────────────────────────────────── tidyverse 1.2.1 ──

## ✔ ggplot2 3.2.1 ✔ purrr 0.3.2  
## ✔ tibble 2.1.3 ✔ dplyr 0.8.3  
## ✔ tidyr 1.0.0 ✔ stringr 1.4.0  
## ✔ readr 1.3.1 ✔ forcats 0.4.0

## ── Conflicts ──────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(skimr)

##   
## Attaching package: 'skimr'

## The following object is masked from 'package:stats':  
##   
## filter

library(dslabs)

#Ian Bachli Tidyverse Exercise  
#The glimpse function from dplyr helps us to look at the gapminder data.  
  
glimpse(gapminder)

## Observations: 10,545  
## Variables: 9  
## $ country <fct> Albania, Algeria, Angola, Antigua and Barbuda, …  
## $ year <int> 1960, 1960, 1960, 1960, 1960, 1960, 1960, 1960,…  
## $ infant\_mortality <dbl> 115.40, 148.20, 208.00, NA, 59.87, NA, NA, 20.3…  
## $ life\_expectancy <dbl> 62.87, 47.50, 35.98, 62.97, 65.39, 66.86, 65.66…  
## $ fertility <dbl> 6.19, 7.65, 7.32, 4.43, 3.11, 4.55, 4.82, 3.45,…  
## $ population <dbl> 1636054, 11124892, 5270844, 54681, 20619075, 18…  
## $ gdp <dbl> NA, 13828152297, NA, NA, 108322326649, NA, NA, …  
## $ continent <fct> Europe, Africa, Africa, Americas, Americas, Asi…  
## $ region <fct> Southern Europe, Northern Africa, Middle Africa…

#Glimpse shows 10,545 observations and 9 variables. The variables are listed in rows containing the variable name and class, as well as a few early observations from the set. Glimpse and the str function are similar.

#Ian Bachli Tidyverse Exercise  
#The Skim function from Skimr allows us to examine a summary of the data focusing on the variables.   
  
skim(gapminder)

## Skim summary statistics  
## n obs: 10545   
## n variables: 9   
##   
## ── Variable type:factor ──────────────────────────────────────────────────────────  
## variable missing complete n n\_unique  
## continent 0 10545 10545 5  
## country 0 10545 10545 185  
## region 0 10545 10545 22  
## top\_counts ordered  
## Afr: 2907, Asi: 2679, Eur: 2223, Ame: 2052 FALSE  
## Alb: 57, Alg: 57, Ang: 57, Ant: 57 FALSE  
## Wes: 1026, Eas: 912, Wes: 912, Car: 741 FALSE  
##   
## ── Variable type:integer ─────────────────────────────────────────────────────────  
## variable missing complete n mean sd p0 p25 p50 p75 p100  
## year 0 10545 10545 1988 16.45 1960 1974 1988 2002 2016  
## hist  
## ▇▇▇▇▇▇▇▇  
##   
## ── Variable type:numeric ─────────────────────────────────────────────────────────  
## variable missing complete n mean sd p0  
## fertility 187 10358 10545 4.08 2.03 0.84  
## gdp 2972 7573 10545 1.5e+11 7e+11 4e+07   
## infant\_mortality 1453 9092 10545 55.31 47.73 1.5   
## life\_expectancy 0 10545 10545 64.81 10.67 13.2   
## population 185 10360 10545 2.7e+07 1.1e+08 31238   
## p25 p50 p75 p100 hist  
## 2.2 3.75 6 9.22 ▅▇▃▃▅▆▂▁  
## 1.8e+09 7.8e+09 5.5e+10 1.2e+13 ▇▁▁▁▁▁▁▁  
## 16 41.5 85.1 276.9 ▇▃▂▂▁▁▁▁  
## 57.5 67.54 73 83.9 ▁▁▁▂▃▅▇▃  
## 1333486 5e+06 1.5e+07 1.4e+09 ▇▁▁▁▁▁▁▁

#Skim generates a summary of the data from gapminder, highlighting the variables in the set. Skim shows each variable and provides a short summary that is relevant to the data class. Skim also provides the total obsevations and missing values for each variable in the data set.

#Ian Bachli Tidyverse Exercise  
#Extract only the African countries from the gapminder data set.   
  
africancountries <- filter(gapminder, continent == "Africa")  
  
#The africancountries object is used to store data for this exercise to distinguish itself from the previous object, africadata. Both objects contain the same data with 2907 observations and 9 variables.   
#TBy converting africancountries into a tibble, formatting the data into a clean view and prevents R from printing all of the data into the console if you view the object.  
  
africatibble <- tbl\_df(africancountries)  
africatibble

## # A tibble: 2,907 x 9  
## country year infant\_mortality life\_expectancy fertility population  
## <fct> <int> <dbl> <dbl> <dbl> <dbl>  
## 1 Algeria 1960 148. 47.5 7.65 11124892  
## 2 Angola 1960 208 36.0 7.32 5270844  
## 3 Benin 1960 187. 38.3 6.28 2431620  
## 4 Botswa… 1960 116. 50.3 6.62 524029  
## 5 Burkin… 1960 161. 35.2 6.29 4829291  
## 6 Burundi 1960 145. 40.6 6.95 2786740  
## 7 Camero… 1960 167. 43.5 5.65 5361367  
## 8 Cape V… 1960 NA 50.1 6.89 202316  
## 9 Centra… 1960 166. 37.4 5.84 1503501  
## 10 Chad 1960 NA 41.0 6.25 3002596  
## # … with 2,897 more rows, and 3 more variables: gdp <dbl>,  
## # continent <fct>, region <fct>

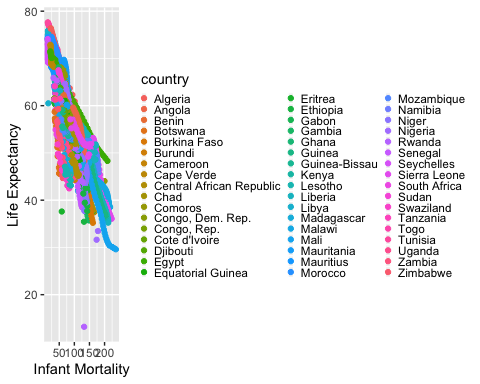
#Ian Bachli Tidyverse Exercise  
#Using only African countries select the following variables to keep: infant\_mortality, life\_expectancy, population, and country. Create a new object using the previously made africatibble and use the select function to choose the variables of interest.   
  
africa\_plot\_data <- select(africatibble, life\_expectancy, infant\_mortality, population, country)  
africa\_plot\_data

## # A tibble: 2,907 x 4  
## life\_expectancy infant\_mortality population country   
## <dbl> <dbl> <dbl> <fct>   
## 1 47.5 148. 11124892 Algeria   
## 2 36.0 208 5270844 Angola   
## 3 38.3 187. 2431620 Benin   
## 4 50.3 116. 524029 Botswana   
## 5 35.2 161. 4829291 Burkina Faso   
## 6 40.6 145. 2786740 Burundi   
## 7 43.5 167. 5361367 Cameroon   
## 8 50.1 NA 202316 Cape Verde   
## 9 37.4 166. 1503501 Central African Republic  
## 10 41.0 NA 3002596 Chad   
## # … with 2,897 more rows

#This results in a tibble with 2907 observations and 4 variables. You could also do this by selecting all variables that are not of interest and placing a - symbol in front of each of their names to subtract them.

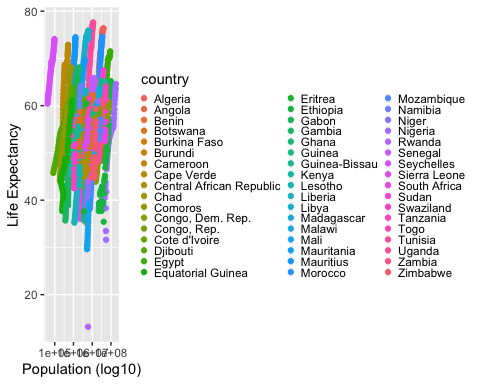
#Ian Bachli Tidyverse Exercise   
#Create two plots using ggplot for life expectancy as a function of infant mortality and population. Assign different colors for each country in the data set.   
#There are two different plotting functions within ggplot2: qplot (quick plot) and ggplot. qplot is streamlined and useful for any simple figures, while ggplot is ideal for more complex figures. qplot will be used for the first two figures and ggplot for the second.   
#Make a plot of life expectancy vs. infant mortality.  
#Using the qplot function input the desired variables starting with x then y, color defines the data point color, data assigns the africa\_plot\_data object, the labs function creates professional labels for the x and y axes, and the theme function sets formatting to the figure legend.   
  
qplot(infant\_mortality, life\_expectancy, color = country, data = africa\_plot\_data) + labs(y = "Life Expectancy", x = "Infant Mortality") + theme(legend.key.size = unit(0.2, "cm"), legend.key.width = unit(0.1, "cm"))

## Warning: Removed 226 rows containing missing values (geom\_point).



#The scatterplot shows the same negative correlation as seen before in the coding exercise (Megan Roberston's work) with the addition of a color coded legend to illistrate different countries.   
#The warning of the removal of 226 rows is from the NA variables as before, and is nothing to be concerned about.   
#Make a plot of life expectancy vs. population. Set the population size to a log scale.  
  
qplot(population, life\_expectancy, color = country, data = africa\_plot\_data) + labs(y = "Life Expectancy", x = "Population (log10)") + scale\_x\_log10() + theme(legend.key.size = unit(0.2, "cm"), legend.key.width = unit(0.1, "cm"))

## Warning: Removed 51 rows containing missing values (geom\_point).



#The resulting scatter plot shows the color coded countries with the original streaks from the previous data. Adding color makes it easier to see and interpret the data.  
# We can see that as a opulation within a country increases, the life expectancy increases also.   
# Another warning of the removal of 51 rows is consistent with the measure of NA values for population, as expected.

#First, let’s load our data package: “dslabs”. Use the function library.

library(dslabs)

#Next, we’ll examine what the gapminder dataset is about, using the help() function.

help("gapminder")

#According the Help window, Gapminder is a dataset with health and income outcomes for 184 countries from 1960 to 2016.

#Next, let’s look at the data itself. Using Str and Summary functions will provide a general overview.

str(gapminder)

## 'data.frame': 10545 obs. of 9 variables:  
## $ country : Factor w/ 185 levels "Albania","Algeria",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ year : int 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 ...  
## $ infant\_mortality: num 115.4 148.2 208 NA 59.9 ...  
## $ life\_expectancy : num 62.9 47.5 36 63 65.4 ...  
## $ fertility : num 6.19 7.65 7.32 4.43 3.11 4.55 4.82 3.45 2.7 5.57 ...  
## $ population : num 1636054 11124892 5270844 54681 20619075 ...  
## $ gdp : num NA 1.38e+10 NA NA 1.08e+11 ...  
## $ continent : Factor w/ 5 levels "Africa","Americas",..: 4 1 1 2 2 3 2 5 4 3 ...  
## $ region : Factor w/ 22 levels "Australia and New Zealand",..: 19 11 10 2 15 21 2 1 22 21 ...

summary(gapminder)

## country year infant\_mortality  
## Albania : 57 Min. :1960 Min. : 1.50   
## Algeria : 57 1st Qu.:1974 1st Qu.: 16.00   
## Angola : 57 Median :1988 Median : 41.50   
## Antigua and Barbuda: 57 Mean :1988 Mean : 55.31   
## Argentina : 57 3rd Qu.:2002 3rd Qu.: 85.10   
## Armenia : 57 Max. :2016 Max. :276.90   
## (Other) :10203 NA's :1453   
## life\_expectancy fertility population gdp   
## Min. :13.20 Min. :0.840 Min. :3.124e+04 Min. :4.040e+07   
## 1st Qu.:57.50 1st Qu.:2.200 1st Qu.:1.333e+06 1st Qu.:1.846e+09   
## Median :67.54 Median :3.750 Median :5.009e+06 Median :7.794e+09   
## Mean :64.81 Mean :4.084 Mean :2.701e+07 Mean :1.480e+11   
## 3rd Qu.:73.00 3rd Qu.:6.000 3rd Qu.:1.523e+07 3rd Qu.:5.540e+10   
## Max. :83.90 Max. :9.220 Max. :1.376e+09 Max. :1.174e+13   
## NA's :187 NA's :185 NA's :2972   
## continent region   
## Africa :2907 Western Asia :1026   
## Americas:2052 Eastern Africa : 912   
## Asia :2679 Western Africa : 912   
## Europe :2223 Caribbean : 741   
## Oceania : 684 South America : 684   
## Southern Europe: 684   
## (Other) :5586

class(gapminder)

## [1] "data.frame"

#You can see from the output below that the structure of Gapminder has 9 different variables ranging from country, year, life expectancy, etc, with a total of 10,545 observations. The summary function shows briefly the min, max, median, mean, etc for the first 6 countries. Finally, Gapminder is a data frame.

#Now let’s only look at data from Africa. To do this, because gapminder is a dataframe, we can conditionally index the row “contient” to only have “Africa”. Once again, let’s look at str and summary of this new dataset.

africandata <- gapminder[gapminder$continent == "Africa", ]  
str(africandata)

## 'data.frame': 2907 obs. of 9 variables:  
## $ country : Factor w/ 185 levels "Albania","Algeria",..: 2 3 18 22 26 27 29 31 32 33 ...  
## $ year : int 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 ...  
## $ infant\_mortality: num 148 208 187 116 161 ...  
## $ life\_expectancy : num 47.5 36 38.3 50.3 35.2 ...  
## $ fertility : num 7.65 7.32 6.28 6.62 6.29 6.95 5.65 6.89 5.84 6.25 ...  
## $ population : num 11124892 5270844 2431620 524029 4829291 ...  
## $ gdp : num 1.38e+10 NA 6.22e+08 1.24e+08 5.97e+08 ...  
## $ continent : Factor w/ 5 levels "Africa","Americas",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ region : Factor w/ 22 levels "Australia and New Zealand",..: 11 10 20 17 20 5 10 20 10 10 ...

summary(africandata)

## country year infant\_mortality life\_expectancy  
## Algeria : 57 Min. :1960 Min. : 11.40 Min. :13.20   
## Angola : 57 1st Qu.:1974 1st Qu.: 62.20 1st Qu.:48.23   
## Benin : 57 Median :1988 Median : 93.40 Median :53.98   
## Botswana : 57 Mean :1988 Mean : 95.12 Mean :54.38   
## Burkina Faso: 57 3rd Qu.:2002 3rd Qu.:124.70 3rd Qu.:60.10   
## Burundi : 57 Max. :2016 Max. :237.40 Max. :77.60   
## (Other) :2565 NA's :226   
## fertility population gdp continent   
## Min. :1.500 Min. : 41538 Min. :4.659e+07 Africa :2907   
## 1st Qu.:5.160 1st Qu.: 1605232 1st Qu.:8.373e+08 Americas: 0   
## Median :6.160 Median : 5570982 Median :2.448e+09 Asia : 0   
## Mean :5.851 Mean : 12235961 Mean :9.346e+09 Europe : 0   
## 3rd Qu.:6.860 3rd Qu.: 13888152 3rd Qu.:6.552e+09 Oceania : 0   
## Max. :8.450 Max. :182201962 Max. :1.935e+11   
## NA's :51 NA's :51 NA's :637   
## region   
## Eastern Africa :912   
## Western Africa :912   
## Middle Africa :456   
## Northern Africa :342   
## Southern Africa :285   
## Australia and New Zealand: 0   
## (Other) : 0

#2907 observations are associated with Africa. To check our work, looking back at the above Gapminder dataframe, observations 2 and 3 are indeed African countries.

#Let’s look at different health outcomes in these countries. First we will create new variables from africandata with infant mortality and life expectancy using indexing again. We select only the relevant rows from africandata to put into a object. Finally we create a new data frame called “africaninfantlife” with the two new variables.

infantmort <- africandata$infant\_mortality  
lifeexp <- africandata$life\_expectancy  
africaninfantlife <- data.frame("infant mortality" = infantmort, "life expectancy" = lifeexp)  
str(africaninfantlife)

## 'data.frame': 2907 obs. of 2 variables:  
## $ infant.mortality: num 148 208 187 116 161 ...  
## $ life.expectancy : num 47.5 36 38.3 50.3 35.2 ...

summary(africaninfantlife)

## infant.mortality life.expectancy  
## Min. : 11.40 Min. :13.20   
## 1st Qu.: 62.20 1st Qu.:48.23   
## Median : 93.40 Median :53.98   
## Mean : 95.12 Mean :54.38   
## 3rd Qu.:124.70 3rd Qu.:60.10   
## Max. :237.40 Max. :77.60   
## NA's :226

#A data frame has been created with 2907 observations with the 2 variables of infant mortality and life expectancy.

#Now the same thing for population and life expectancy.

pop <- africandata$population  
lifeexp <- africandata$life\_expectancy  
africanpoplife <- data.frame("population" = pop, "life expectancy" = lifeexp)  
str(africanpoplife)

## 'data.frame': 2907 obs. of 2 variables:  
## $ population : num 11124892 5270844 2431620 524029 4829291 ...  
## $ life.expectancy: num 47.5 36 38.3 50.3 35.2 ...

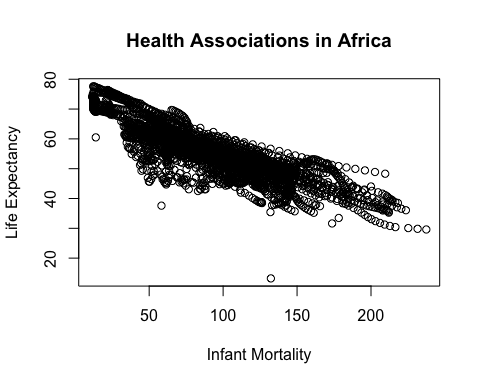
summary(africanpoplife)

## population life.expectancy  
## Min. : 41538 Min. :13.20   
## 1st Qu.: 1605232 1st Qu.:48.23   
## Median : 5570982 Median :53.98   
## Mean : 12235961 Mean :54.38   
## 3rd Qu.: 13888152 3rd Qu.:60.10   
## Max. :182201962 Max. :77.60   
## NA's :51

#Again another data frame has been created with 2907 observations and the 2 variables of population and life expectany.

#Let’s plot! We’ll use the basic plot function (use xlab/ylab to label our x/y axis, main to title our plot, and change the output to points using type) to look at infant mortality vs life expectancy.

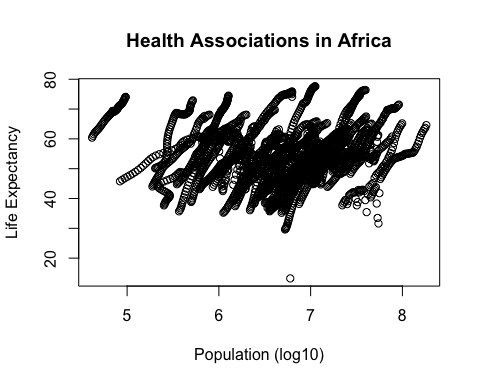
plot(africaninfantlife, xlab = "Infant Mortality", ylab = "Life Expectancy", main = "Health Associations in Africa", type = "p")



#From this plot, you can see a negative correlation between infant mortality and life expectancy.

#Let’s look at population vs life expectancy. To improve the readability, the x-axis is converted to a log 10 scale to improve readability.

plot(log10(pop), lifeexp, xlab = "Population (log10)", ylab = "Life Expectancy", main = "Health Associations in Africa", type = "p")



#There is a positive correlation between population size and life expectancy, which can be expected. #In both plots there are data “streaks”, which correspond to a country’s progression through the years observed.

#Instead of all the years, we will look at the year with the most data. First we need to find what years have missing data for infant mortality. To do this, let’s use is.na and which observations are missing in infant mortality.

missing <- which(is.na(infantmort))  
print(missing)

## [1] 8 10 15 17 18 20 23 24 37 43 53 59 61 62  
## [15] 63 66 68 69 70 71 75 80 86 87 88 89 94 104  
## [29] 110 112 113 114 117 119 120 121 122 126 131 137 138 139  
## [43] 145 155 161 163 164 165 168 170 171 172 173 177 182 188  
## [57] 189 190 196 206 212 214 215 216 219 221 222 223 224 228  
## [71] 233 240 241 247 257 263 265 266 267 270 272 273 274 275  
## [85] 279 291 292 298 308 314 316 317 318 321 323 324 326 330  
## [99] 342 343 349 359 365 367 368 369 372 374 375 377 381 400  
## [113] 410 416 418 419 420 423 425 426 428 432 451 461 470 474  
## [127] 476 479 483 502 525 527 530 534 553 563 576 578 581 585  
## [141] 604 614 627 629 632 636 655 665 678 680 683 687 706 716  
## [155] 729 731 734 738 767 780 782 785 789 818 833 836 869 884  
## [169] 887 920 935 971 986 1037 1088 2857 2858 2859 2860 2861 2862 2863  
## [183] 2864 2865 2866 2867 2868 2869 2870 2871 2872 2873 2874 2875 2876 2877  
## [197] 2878 2879 2880 2881 2882 2883 2884 2885 2886 2887 2888 2889 2890 2891  
## [211] 2892 2893 2894 2895 2896 2897 2898 2899 2900 2901 2902 2903 2904 2905  
## [225] 2906 2907

#There are 226 observations with missing data in infant mortality for our African dataset. Let’s convert these to year data to see what years are missing.

yeardata <- gapminder$year[missing]  
print(yeardata)

## [1] 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960  
## [15] 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960  
## [29] 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960  
## [43] 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 1961  
## [57] 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961  
## [71] 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961  
## [85] 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961 1961  
## [99] 1961 1961 1961 1961 1961 1961 1961 1961 1962 1962 1962 1962 1962 1962  
## [113] 1962 1962 1962 1962 1962 1962 1962 1962 1962 1962 1962 1962 1962 1962  
## [127] 1962 1962 1962 1962 1962 1962 1962 1962 1962 1963 1963 1963 1963 1963  
## [141] 1963 1963 1963 1963 1963 1963 1963 1963 1963 1963 1963 1963 1963 1963  
## [155] 1963 1963 1963 1963 1964 1964 1964 1964 1964 1964 1964 1964 1964 1964  
## [169] 1964 1964 1965 1965 1965 1965 1965 1975 1975 1975 1975 1975 1975 1975  
## [183] 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975  
## [197] 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975  
## [211] 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975  
## [225] 1975 1975

#These 226 observations are now matched with the years.

#We will look at year 2000 data. Let’s create another datas frame with just year 2000 data selected from our africandata.

yeary2k <- africandata[africandata$year == 2000, ]  
str(yeary2k)

## 'data.frame': 51 obs. of 9 variables:  
## $ country : Factor w/ 185 levels "Albania","Algeria",..: 2 3 18 22 26 27 29 31 32 33 ...  
## $ year : int 2000 2000 2000 2000 2000 2000 2000 2000 2000 2000 ...  
## $ infant\_mortality: num 33.9 128.3 89.3 52.4 96.2 ...  
## $ life\_expectancy : num 73.3 52.3 57.2 47.6 52.6 46.7 54.3 68.4 45.3 51.5 ...  
## $ fertility : num 2.51 6.84 5.98 3.41 6.59 7.06 5.62 3.7 5.45 7.35 ...  
## $ population : num 31183658 15058638 6949366 1736579 11607944 ...  
## $ gdp : num 5.48e+10 9.13e+09 2.25e+09 5.63e+09 2.61e+09 ...  
## $ continent : Factor w/ 5 levels "Africa","Americas",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ region : Factor w/ 22 levels "Australia and New Zealand",..: 11 10 20 17 20 5 10 20 10 10 ...

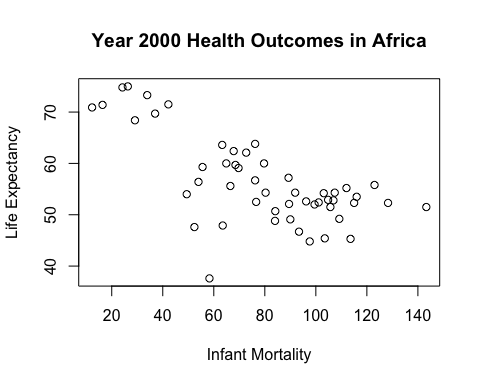
summary(yeary2k)

## country year infant\_mortality life\_expectancy  
## Algeria : 1 Min. :2000 Min. : 12.30 Min. :37.60   
## Angola : 1 1st Qu.:2000 1st Qu.: 60.80 1st Qu.:51.75   
## Benin : 1 Median :2000 Median : 80.30 Median :54.30   
## Botswana : 1 Mean :2000 Mean : 78.93 Mean :56.36   
## Burkina Faso: 1 3rd Qu.:2000 3rd Qu.:103.30 3rd Qu.:60.00   
## Burundi : 1 Max. :2000 Max. :143.30 Max. :75.00   
## (Other) :45   
## fertility population gdp continent   
## Min. :1.990 Min. : 81154 Min. :2.019e+08 Africa :51   
## 1st Qu.:4.150 1st Qu.: 2304687 1st Qu.:1.274e+09 Americas: 0   
## Median :5.550 Median : 8799165 Median :3.238e+09 Asia : 0   
## Mean :5.156 Mean : 15659800 Mean :1.155e+10 Europe : 0   
## 3rd Qu.:5.960 3rd Qu.: 17391242 3rd Qu.:8.654e+09 Oceania : 0   
## Max. :7.730 Max. :122876723 Max. :1.329e+11   
##   
## region   
## Eastern Africa :16   
## Western Africa :16   
## Middle Africa : 8   
## Northern Africa : 6   
## Southern Africa : 5   
## Australia and New Zealand: 0   
## (Other) : 0

#Ok we have 51 observations with 9 variables!

#Let’s plot! We don’t need to do intermediate variables, we can just select what variables we want out of our new year2k data frame. Again, let’s label the axes and title

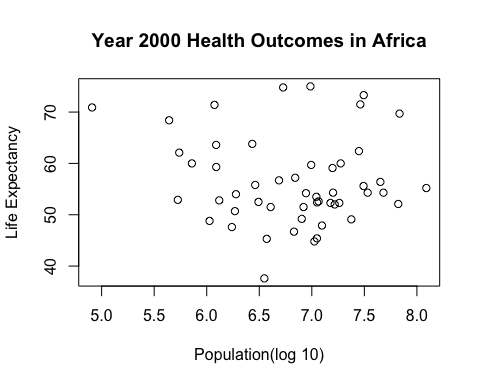
plot(yeary2k$infant\_mortality, yeary2k$life\_expectancy, xlab = "Infant Mortality", ylab = "Life Expectancy", main = "Year 2000 Health Outcomes in Africa", type = "p")



#There is still a negative correlation between infant mortality and life expectancy.

#Let’s look at population and life expectancy. Again, population is converted into log10.

plot(log10(yeary2k$population), yeary2k$life\_expectancy, xlab = "Population(log 10)", ylab = "Life Expectancy", main = "Year 2000 Health Outcomes in Africa", type = "p")



#It appears there is no correlation between population and life expectancy. Let’s run some statistics! We will use the lm function which is used to fit linear models. Infant mortality and population are the predictors and life expectancy is the outcome.

fit1 <- lm(yeary2k$life\_expectancy ~ yeary2k$infant\_mortality)  
fit2 <- lm(yeary2k$life\_expectancy ~ yeary2k$population)

#Let’s run a summary on these linear models.

summary(fit1)

##   
## Call:  
## lm(formula = yeary2k$life\_expectancy ~ yeary2k$infant\_mortality)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -22.6651 -3.7087 0.9914 4.0408 8.6817   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 71.29331 2.42611 29.386 < 2e-16 \*\*\*  
## yeary2k$infant\_mortality -0.18916 0.02869 -6.594 2.83e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6.221 on 49 degrees of freedom  
## Multiple R-squared: 0.4701, Adjusted R-squared: 0.4593   
## F-statistic: 43.48 on 1 and 49 DF, p-value: 2.826e-08

summary(fit2)

##   
## Call:  
## lm(formula = yeary2k$life\_expectancy ~ yeary2k$population)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -18.429 -4.602 -2.568 3.800 18.802   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.593e+01 1.468e+00 38.097 <2e-16 \*\*\*  
## yeary2k$population 2.756e-08 5.459e-08 0.505 0.616   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 8.524 on 49 degrees of freedom  
## Multiple R-squared: 0.005176, Adjusted R-squared: -0.01513   
## F-statistic: 0.2549 on 1 and 49 DF, p-value: 0.6159

#The p-value of Fit1 is 2.826e-08. This means that the correlation between higher infant mortality seems to be associated with lower life expectancy, which makes sense as shown on the plot.

#The p-value of Fit2 is 0.6159. There is in line with the graph as there seems to be no correlation between population size and life expectancy.

#However, we are just looking at one year so population size changes are lost in this dataset, compared to the previous data of all years.