William Norfolk-Coding Exercise

William Norfolk

8/24/2019

#Load dslabs library and gapminder data set.  
library(dslabs)  
  
data(gapminder)  
  
#The gapminder data set is a collection of health and income outcomes for 184 countries from 1960 to 2016.

#Gather information on gapminder data set with R functions: help, str, summary, and class.   
help(gapminder)

## starting httpd help server ... done

#The help function displays available help pages (in the help tab) about the selected package.  
  
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 ...

#Str or the structure function will display the basic structural information for an object in the console. Using the str function we can see that gapminder consists of nine variables: country, infant\_mortality, life\_expectancy, fertility, population, gdp, continent, and region.   
  
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

#The sumary function will display statistical summaries of the selected data in the console for each of the variables within the data set. The type of data shown will depend on the class (i.e. numeric, integer, character, etc.) of the individual variables within a data set.   
  
class(gapminder)

## [1] "data.frame"

#The class function will display the type of object that is represented by a given name. For the example here, gapminder is of the data.frame class.

#Assign only the African countries to a new object titled africadata.  
  
africadata <- gapminder$continent == "Africa"  
  
#To assign a new object use the <- to the right of the chosen object name. All African countries can be accessed in gapminder through the continent variable. The accessor $ is used to access specific variables within the data frame. To determine if a country is in Africa or not, africadata is assigned as a logical vector using == to determine if the continent variable is listed as "Africa" or other.   
  
str(africadata)

## logi [1:10545] FALSE TRUE TRUE FALSE FALSE FALSE ...

#The logical vector africadata will output information as TRUE or FALSE depending on the continent variable of each country. In this example, 10545 different data points are maeasured. Countries in Africa receive the output TRUE whereas all other continents receive a FALSE.   
  
summary(africadata)

## Mode FALSE TRUE   
## logical 7638 2907

#The summary shows results of the logical vector. Of the total 10545 measured countries, 2907 of them are located in Africa. These values can be verified by reviewing the previous summary(gapminder) output in the console and subtracting the African data points from the total measurements under the continent variable.

#Create an object containing infant\_mortality and life\_expectancy for africadata.  
  
#Create and object for infant\_mortality for African countries by accessing gapminder with $ and applying the logical vector africadata.  
  
infant\_mortality\_africa <- gapminder$infant\_mortality[africadata]  
  
#Repeat the same process to create an object fot life\_expectancy for African countries.  
  
life\_expectancy\_africa <- gapminder$life\_expectancy[africadata]  
  
#Combine the two new objects into a data frame to create a single object containing both variables.   
  
infant\_vs\_life\_africa <- data.frame(infantmortality = infant\_mortality\_africa, lifeexpectancy = life\_expectancy\_africa)  
  
str(infant\_vs\_life\_africa)

## 'data.frame': 2907 obs. of 2 variables:  
## $ infantmortality: num 148 208 187 116 161 ...  
## $ lifeexpectancy : num 47.5 36 38.3 50.3 35.2 ...

#The result is a data frame with 2907 observations and two variables: infantmortality and lifeexpectancy.  
  
summary(infant\_vs\_life\_africa)

## infantmortality lifeexpectancy   
## 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

#Summary gives a quick glance of min, max, median, and mean for both variables. Note there are 226 NA records indicated for infantmortality.

#Create an object containing population and life\_expectancy for africadata.  
  
#Repeat the process above for the population variable.  
  
population\_africa <- gapminder$population[africadata]  
  
#Combine the population and life\_expectancy data into a data frame.  
  
pop\_vs\_life\_africa <- data.frame(population = population\_africa, lifeexpectancy = life\_expectancy\_africa)  
  
str(pop\_vs\_life\_africa)

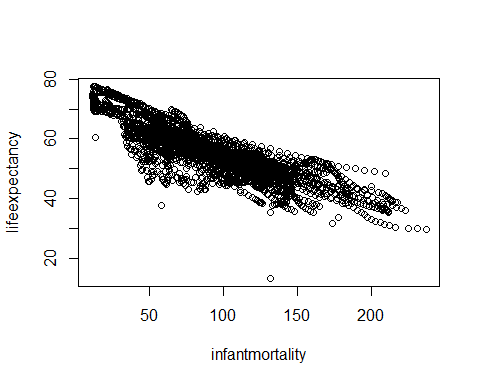
## 'data.frame': 2907 obs. of 2 variables:  
## $ population : num 11124892 5270844 2431620 524029 4829291 ...  
## $ lifeexpectancy: num 47.5 36 38.3 50.3 35.2 ...

#The result is a data frame with 2907 observations and two variables: population and lifeexpectancy.  
  
summary(pop\_vs\_life\_africa)

## population lifeexpectancy   
## 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

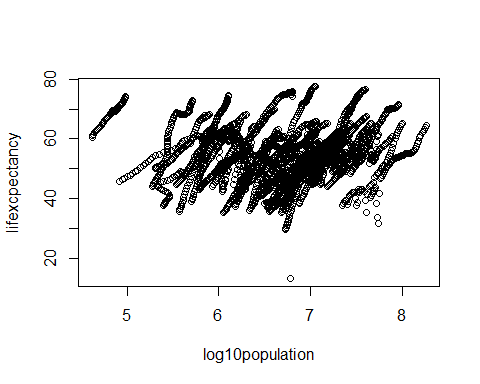
#Standard statistical values are given in the summary of the object. Note 51 NA values were recorded for the population variable.

#Plot of infant mortality verses life expectancy for all African countries.  
  
plot(infant\_vs\_life\_africa)



#The plot illistrates a distinct negative correlation between the two variables. As expected, countries with increased infant mortality show decreased life expectancy.

#Plot of population verses life expectancy for all African countries.   
  
#Convert previous population object into a new object at log scale.  
  
logten\_pop\_africa <- log10(population\_africa)  
  
#Input the new log population data into a data frame with life expectancy.   
  
logtenpop\_vs\_life\_africa <- data.frame( log10population = logten\_pop\_africa, lifexcpectancy = life\_expectancy\_africa)  
  
plot(logtenpop\_vs\_life\_africa)



#The plot shows a distinct scattering of "streaks" of data. The individual streaks typically move in a positive trend illistrating a positve correlation between increased population size and increased life expectancy. The "streaks" seen in the data likely represent different contries within the continent of Africa measured over time.

#Determine which years have missing data for infant mortality.  
  
#Create an index object for the NA values within the infant\_mortality\_africa object using the is.na function. This produces a logical vector that tells us which entries are NA.   
  
na\_index <- is.na(infant\_mortality\_africa)  
  
#Apply the na\_index to the gapminder year data.  
  
gapminder$year[na\_index]

## [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 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1975 1976  
## [239] 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976  
## [253] 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976  
## [267] 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976  
## [281] 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976 1976  
## [295] 1976 1976 1976 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977  
## [309] 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977  
## [323] 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977  
## [337] 1977 1977 1977 1977 1977 1977 1977 1977 1978 1978 1978 1978 1978 1978  
## [351] 1978 1978 1978 1978 1978 1978 1978 1978 1978 1978 1978 1978 1978 1978  
## [365] 1978 1978 1978 1979 1979 1979 1979 1979 1979 1979 1979 1979 1979 1979  
## [379] 1979 1979 1979 1979 1979 1979 1979 1979 1979 1979 1979 1980 1980 1980  
## [393] 1980 1980 1980 1980 1980 1980 1981 1981 1981 1991 1991 1991 1991 1991  
## [407] 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991  
## [421] 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991  
## [435] 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991  
## [449] 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991  
## [463] 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 1991  
## [477] 1991 1991 1991 1991 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992  
## [491] 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992  
## [505] 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992  
## [519] 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992 1992  
## [533] 1992 1992 1992 1992 1992 1992 1993 1993 1993 1993 1993 1993 1993 1993  
## [547] 1993 1993 1993 1993 1993 1993 1993 1993 1993 1993 1993 1993 1993 1993  
## [561] 1993 1993 1993 1993 1993 1993 1993 1993 1993 1993 1993 1993 1993 1993  
## [575] 1993 1993 1993 1993 1993 1994 1994 1994 1994 1994 1994 1994 1994 1994  
## [589] 1994 1994 1994 1994 1994 1994 1994 1994 1994 1994 1994 1995 1995 1995  
## [603] 1995 1995 1995 1995 1995 1995 1995 1995 1995 1995 1995 1995 1995 1995  
## [617] 1995 1995 1996 1996 1996 1996 1996 1996 1996 1997 1997 2006 2006 2006  
## [631] 2006 2006 2006 2006 2006 2006 2006 2006 2006 2006 2006 2006 2006 2006  
## [645] 2006 2006 2006 2006 2006 2006 2006 2006 2007 2007 2007 2007 2007 2007  
## [659] 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007  
## [673] 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007  
## [687] 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007  
## [701] 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 2007  
## [715] 2007 2007 2007 2007 2007 2007 2007 2007 2008 2008 2008 2008 2008 2008  
## [729] 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008  
## [743] 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008  
## [757] 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008  
## [771] 2008 2008 2008 2008 2008 2008 2008 2008 2009 2009 2009 2009 2009 2009  
## [785] 2009 2009 2009 2009 2009 2009 2009 2009 2009 2009 2009 2009 2009 2009  
## [799] 2009 2009 2009 2009 2009 2009 2009 2009 2009 2009 2009 2009 2010 2010  
## [813] 2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 2010  
## [827] 2010 2010 2010 2010 2010 2011 2011 2011 2011 2011 2011 2011 2011 2011  
## [841] 2011 2011 2011 2011 2011 2011 2011 2012 2012 2012 2012 2012 2013

#The result is a list of years where infant mortality data is missing. Repedative years in the list represent years where infant mortality data is missing from multiple countries within the continent of Africa.   
  
#The sum function check the total recorded Na values within the na\_index to ensure it is consistent the previous summary of infant mortality.  
  
sum(na\_index)

## [1] 226

#Create a new object by extracting the data from the year 2000 from africadata.   
#Create an index to select for the year 2000.  
  
year2000 <- gapminder$year == "2000"  
  
#Combine the new index with africadata to select for Africa and the year 2000 combined.  
  
africa2000 <- africadata & year2000  
  
#Apply this index to all variables in the gapminder data set.   
  
year2000 <- gapminder$year[africa2000]   
region2000 <- gapminder$region[africa2000]  
country2000 <- gapminder$country[africa2000]  
infantmort2000 <- gapminder$infant\_mortality[africa2000]  
lifeexpect2000 <- gapminder$life\_expectancy[africa2000]  
fertility2000 <- gapminder$fertility[africa2000]  
gdp2000 <- gapminder$gdp[africa2000]  
continent2000 <- gapminder$continent[africa2000]  
population2000 <- gapminder$population[africa2000]  
  
#Create a data frame using the new object names to select for African countries and the year 2000.   
  
africa2000df <- data.frame(year = year2000, country = country2000, population = population2000, region = region2000, infant\_mortality = infantmort2000, life\_expectancy = lifeexpect2000, fertility = fertility2000, gdp = gdp2000, continent = continent2000)  
  
str(africa2000df)

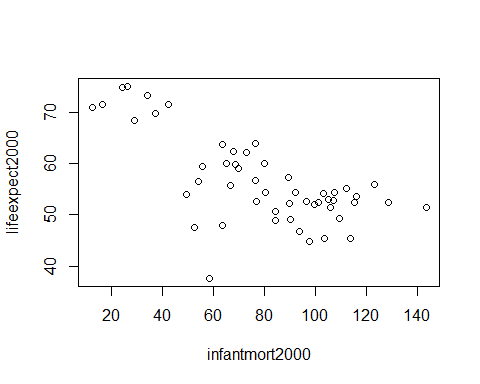
## 'data.frame': 51 obs. of 9 variables:  
## $ year : int 2000 2000 2000 2000 2000 2000 2000 2000 2000 2000 ...  
## $ country : Factor w/ 185 levels "Albania","Algeria",..: 2 3 18 22 26 27 29 31 32 33 ...  
## $ population : num 31183658 15058638 6949366 1736579 11607944 ...  
## $ region : Factor w/ 22 levels "Australia and New Zealand",..: 11 10 20 17 20 5 10 20 10 10 ...  
## $ 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 ...  
## $ 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 ...

#The resulting data frame has 51 observations and 9 vairables.   
  
summary(africa2000df)

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

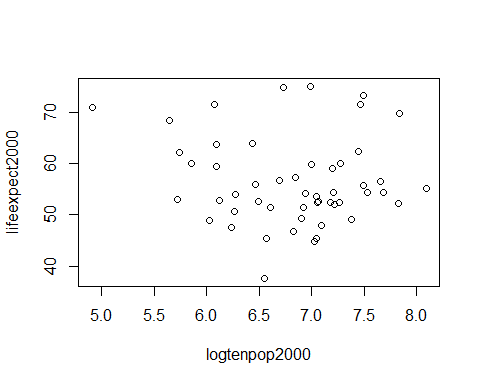
#Specific statistical values are given for each of the 9 variables within the object.

#Recreate the plot for infant mortality verses life expectancy.  
  
plot(infantmort2000, lifeexpect2000)



#The resulting figure shows a negative correlation between infant mortality and life expectancy.

#Recreate the plot for population verses life expectancy.   
  
#Redefine the population data for Africa in 2000 to a log 10 scale.   
  
logtenpop2000 <- log10(population2000)  
  
#Plot log 10 population verses life expectancy.  
  
plot(logtenpop2000, lifeexpect2000)



#The resulting figure shows no distinct correlation between population and life expectancy.

#Create a linear regression model for life expectancy as the outcome (y variable), and infant mortality as the predictor (x variable).Ensure the y variable is the first variable entered to the lm() function.  
  
fit1 <- lm(lifeexpect2000 ~ infantmort2000)  
   
summary(fit1)

##   
## Call:  
## lm(formula = lifeexpect2000 ~ infantmort2000)  
##   
## 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 \*\*\*  
## infantmort2000 -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

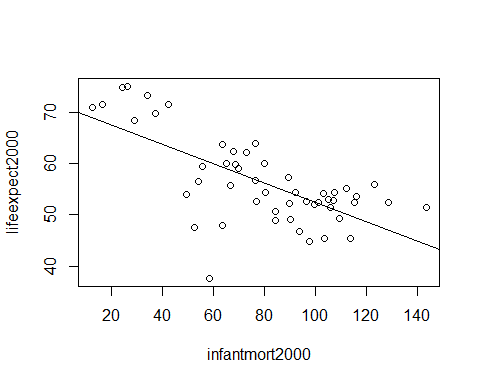
#Summary prints the fit results to the console and provides a number of important statistical data points, in particular the p-value. The p-value from this example is 2.826e-08. The very low p-value suggests that infant mortality is strongly related to life expectancy.

#Create a linear regression model for life expectancy as the outcome (y variable), and population size as the predictor (x variable).  
  
fit2 <- lm(lifeexpect2000 ~ logtenpop2000)  
  
summary(fit2)

##   
## Call:  
## lm(formula = lifeexpect2000 ~ logtenpop2000)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -19.113 -4.809 -1.554 3.907 18.863   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 65.324 12.520 5.217 3.65e-06 \*\*\*  
## logtenpop2000 -1.315 1.829 -0.719 0.476   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 8.502 on 49 degrees of freedom  
## Multiple R-squared: 0.01044, Adjusted R-squared: -0.009755   
## F-statistic: 0.517 on 1 and 49 DF, p-value: 0.4755

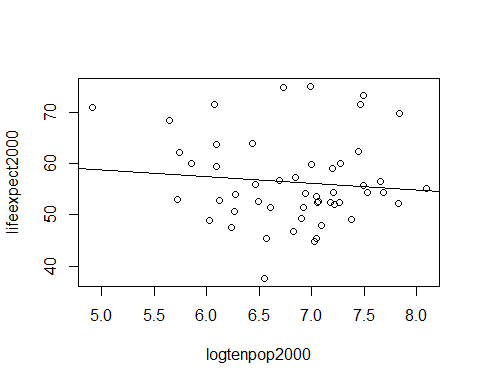
#The p-value from this example is 0.4755. The high p-value suggests that poplation size is not related to life expectancy.

#Fit line to scatterplot for life expectancy verses infant mortality.  
  
#Replot the scatterplot.  
  
plot(infantmort2000, lifeexpect2000)  
  
#Use the abline function to fit a line to the plot according to the new linear model object.   
  
abline(fit1)



#Here we see a strong negative correlation with the fit line which is consistent with the p-value.

#Fit line to scatterplot for life expectancy verses population size.  
  
plot(logtenpop2000, lifeexpect2000)  
abline(fit2)



#Here we see no distinct correlation with the fit line which is consistent with the p-value.