WORKING ON CRUTEM LESSON 2

Victor Nyabuti Ong'era

2022-10-18

Lesson 1 code

```
path1 = "C:/Users/VICTOR_NYABUTI/Climate/data/"
values = as.matrix(read.table(paste(path1,"GL.csv", sep=""), sep = ",",dec =
                                "."))
missing_values = which(values==-9.999)
values[1,1]
```

```
##   V1
## 1850
```

```
values[missing_values] =  NA
n_column = length(values[1,])
n_rows   = length(values[,1])
values = values[, -n_column]
even_row = seq(2,n_rows, 2)
odd_row = seq(1,n_rows, 2)
temp = values[even_row,]
perc = values[odd_row,]
temp  = temp[-(1:7),]
temp  = temp[-length(temp[,1]),]
perc  = perc[-(1:7),]
perc  = perc[-length(perc[,1]),]
colnames(temp) = c("Year", "January" , "February", "March", "April", "May",
                   "June", "July","Aug", "Sep","Oct", "Nov", "December")
colnames(temp)  =   c("Year", "January" , "February", "March", "April", "May",
                      "June", "July","Aug", "Sep","Oct", "Nov", "December")
temp["Year"]
```

```
## [1] NA
```

```
path2 = "C:/Users/VICTOR_NYABUTI/Climate/output/"
write.table(temp,paste(path2,"Temparature_anomally.csv", sep = ""), sep = ",",
            col.names = TRUE, row.names = FALSE, quote =FALSE )
```

We last exported our data into a table.

Now we continue with the second part of the analysis. We have temperatures.we will find temperature means. Remember how to calculate mean. Its simply the
dividing the sum of all values in a data set by the number of values.

Lets make all things ready for this calculation.We are going to divide the total sum of values in data set by the number of the values. In our data we can see the number of the values is 166. Simply see the end of the table. The number of rows is 165.

To calulate the number of years,we will subtract the first year from the last year

First year

```
start_year = temp[1,1]
start_year
```

```
## Year
## 1857
```

That translates to create a variable *start_year* the year is in cell 1 of the table i.e row1 col1

end year. first select column we are intrested at is 1 **(temp[,1])**

```
(temp[,1])
```

```
##   [1] 1857 1858 1859 1860 1861 1862 1863 1864 1865 1866 1867 1868 1869 1870 1871
##  [16] 1872 1873 1874 1875 1876 1877 1878 1879 1880 1881 1882 1883 1884 1885 1886
##  [31] 1887 1888 1889 1890 1891 1892 1893 1894 1895 1896 1897 1898 1899 1900 1901
##  [46] 1902 1903 1904 1905 1906 1907 1908 1909 1910 1911 1912 1913 1914 1915 1916
##  [61] 1917 1918 1919 1920 1921 1922 1923 1924 1925 1926 1927 1928 1929 1930 1931
##  [76] 1932 1933 1934 1935 1936 1937 1938 1939 1940 1941 1942 1943 1944 1945 1946
##  [91] 1947 1948 1949 1950 1951 1952 1953 1954 1955 1956 1957 1958 1959 1960 1961
## [106] 1962 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976
## [121] 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991
## [136] 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006
## [151] 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021
```

check the length

```
length(temp[,1])
```

```
## [1] 165
```

Then we put this together and save it as a variable *end_year*

```
end_year = temp[length(temp[,1]),1]
```

so whats the number of years. last -first.

```
n_year <- end_year-start_year+1
```

why do we add 1 at the end

Note We add 1 to the start year. This is because we want to capture the value of first year. Think about it if you have years 1, 2 and 3 the difference between them is 3-1 = 2. But the data you are sonsindering for the mean was recorded for the whole period of 3 years therefore makes sense if you add 1.

Now the averages

First we create an empty matrix. The matrix is stored by the variable *annual_mean_1*

```
annual_mean_1 <- matrix(ncol=2,nrow=n_year)
```

so we tell R to create a matrix. number of columns ncol 2 and nuumber of rows nrow 165 i,e same as the value saved by the variable *n_year(number of years)*

To make sure its empty

```
head(annual_mean_1)
```

```
##      [,1] [,2]
## [1,]   NA   NA
## [2,]   NA   NA
```

```
## [3,]   NA   NA
## [4,]   NA   NA
## [5,]   NA   NA
## [6,]   NA   NA
```

Lets give the columns titles

```
colnames(annual_mean_1) <- c("Year","Annual_mean")
```

used the function **colnames**, For the columns in our variable *annnual_mean_1* name them year and annual mean. We are using *C-* Concatenate, concatenation, or concat that means combining a string, text, or other data in a series without any gaps. In this example we have combined a string

Check the matrix can u see the column titles?

```
head(annual_mean_1)
```

```
##      Year Annual_mean
## [1,]   NA          NA
## [2,]   NA          NA
## [3,]   NA          NA
## [4,]   NA          NA
## [5,]   NA          NA
## [6,]   NA          NA
```

Now lets fill this up with the values. years first then Annual means.For years, we know that they are between the *start_year* and *end_year*

```
annual_mean_1[,1] <- start_year:end_year
```

So for matrix *annual_mean_1* in coolumn 1 insert values between the *start_year* and *end_year*.

Quick check on our matrix

```
tail.matrix(annual_mean_1)
```

```
##        Year Annual_mean
## [160,] 2016          NA
## [161,] 2017          NA
## [162,] 2018          NA
## [163,] 2019          NA
## [164,] 2020          NA
## [165,] 2021          NA
```

Using the for loop. A loop repeats the same thing over and over again. Good for repetetive tasks we dont wanna do.or loop consists of three parts: The keyword **For that** starts the loop, **the condition** being tested, and **the End** For keyword that terminates the

loop. This is our cord

our code

for(i in 1:n_year){ annual_mean_1[i,2] = mean(temp[i,2:13],na.rm=FALSE)

defination The **for that** part has an alphabet *i* and a range *1:n_year* The alphabet can be any letter The range part must be an interger.

The condition has the variable *annul_mean_1*. So here we say that in the variable *annul_mean_1* in theindexes that are located in row *i* and col *2* insert mean temperatures.

The function **mean**, should calculate the means of values located in rows *i* and columns **2:13** of the temp variable

```
for(i in 1:n_year){
  annual_mean_1[i,2] <- mean(temp[i,2:13],na.rm=FALSE)
}
```

lets check our matrix

```
head.matrix(annual_mean_1)
```

```
##      Year Annual_mean
## [1,] 1857    33.33333
## [2,] 1858    30.50000
## [3,] 1859    29.58333
## [4,] 1860    25.00000
## [5,] 1861    16.66667
## [6,] 1862    18.16667
```

The means are present.

we can export this amazing data

```
write.table(annual_mean_1,paste(path2,"Yearly_average_temperature_anomaly.csv",
                                sep=""),sep=";",col.names=TRUE,row.names=FALSE,
            quote=FALSE)
```

Lets do some ploting

To plot we will utilise ggplot2 package.Load it

```
library(ggplot2)
```

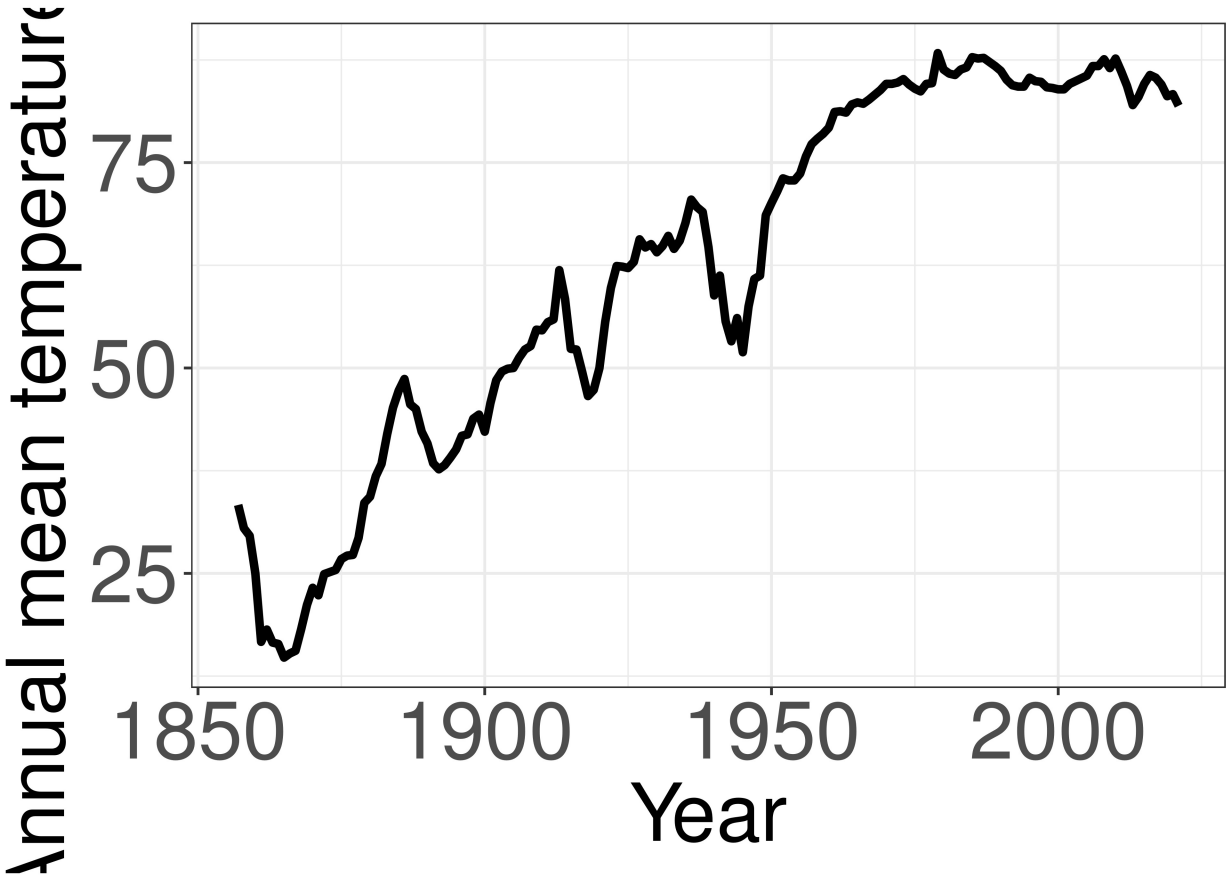
convert the matrix into a data frame. ggplot2 woorks with dataframes

```
annual_mean <- as.data.frame(annual_mean_1)
```

Means that save as a dataframe our variable *annual_mean_1* to anew variable *annual_mean*. In the vocabulary of ggpplot2 notice the location of +sign

Lets plot a blank grey paper page

```
library(ggplot2)
ggplot()+ #I create a grey paper
  geom_line(data=annual_mean,aes(x=Year,y=Annual_mean),size=1.5)+ #I add a line.
  #The data are included in the dataframe annual_mean. On the X-axis I put the
  #Years while on the Y-axis I put the annual mean
  xlab("Year")+ #I change the title of the X-axis
  ylab("Annual mean temperature [?C]")+ #I change the title of the Y-axis
  theme_bw()+ #I change the background from gray that is the default to white
  theme(axis.title.x = element_text(size=30), # I increase the size of the
        #X-axis title (You have to set these numbers manually)
        axis.title.y = element_text(size=30, angle=90), # I increase the size
        #of the Y-axis title and I rotate it of an angle equal to 90?
        axis.text.x = element_text(size=30),
        axis.text.y = element_text(size=30))
```




```
ggsave(file="01_Annual_mean_temperature_anomaly.png",path=path2,dpi=500,
       width=40,height=40,units="cm") #I save the plot in a file
```