Imputation of hydro logical data using R language

Student’s Name

Institution

Course Code

Date

**Methodology**

Multiple imputations are normally used to solve the problem of incomplete data. Recently, two approaches were identified as the best alternative techniques which can be used to solve the problem of incomplete dataset (Carvalho, Almeida, Assad, & Nakai, 2017, p. 573). These methods are joints modeling and multiple imputations by chained equation. However, in this study, MICE and linear regression were used to complete the study. The MIC algorithm was implemented as S-Function. Therefore, for each incomplete variable the user has an option of choosing a set of predictors which can be used for the purpose of imputation. However, to complete the study, software, packages and data were applied uniquely to obtain, install and analyze the data for the study to be completed successfully.

**Software**

In order to complete the study various techniques were used to collect and analyze the data. Several software or applications were installed in a desktop computer and used for gathering and analyzing the data as well. First, the software (*R statistic 3.6)* for managing the rainfall data and conduct imputation were downloaded and installed in the desktop computer. The software R statistic 3.6 ( <https://www.r-project.org/>) were used because of its effectiveness and detailed result it provides. To use the software specific commands were issued as illustrated below. And to develop the script and manage the data directories, R Studio 1.2 (<https://www.rstudio.com/>) was used as an application. This ensured that the data is obtained and analyzed as required. It is important to point that; I started by installing packages which are helpful for the management and imputation of data. The tidyverse package was installed. The tidyverse package is a tool used for restructuring and manipulating datasets. Tidyverse was used to introduce the missing data. Then I installed the MICE (Gelman and Hill, 2011) package, which allowed for the construction of missing data plots, and the MICE (van Buuren and Groothuis-Oudshoorn, 2011) package, which allowed for missing data imputation

install. Packages("tidyverse")  
install. Packages("mi")  
install. Packages("mice")

These packages were loaded into the current session by using the library command

library("tidyverse")  
library("mi")  
library("mice")

**Data**

The data used for the computation of the rain were obtained by downloading using the below codes from the two rainfall stations and then turning the station numbers into vectors.

SE=read.csv("Climates[2702].txt" ,sep=" ")  
Fitz=read.csv("ClimStaFitz[2701].txt" ,sep=" ")  
STN=c(SE$StationNo,Fitz$StationNo)

I set the working directory to my current folder so I could download the data files, using the below code

setwd("E:/Research Implementation")

To download the data, I created a loop. The loop went through each station list, and then inserted it into the longpaddock website URL to download that station’s data. This downloaded one data file for each station in the list from 1960-2019.

for(i in 1:length(STN)){  
 stnno <- STN[i]  
 FileOut <- paste('data/Patched\_',stnno,'.txt',sep='') #give the download a file name - here the only thing changing is the station number to set  
 URL <- paste('https://legacy.longpaddock.qld.gov.au/cgi-bin/silo/PatchedPointDataset.php?format=Standard&station=',stnno,'&start=19600101&finish=20181231&username=CQUNPEDDOJU&password=CQU3M530',sep='') ##where you're downloading the file from'  
 download. File(URL, FileOut, method='auto') #R command to download the file  
#}

Once the download was completed, I set up a data frame ‘dist’ in which I was going to save accuracy data for the two imputation methods I used. This data frame was to be saved as an excel file later.

dist=data.frame(STN=numeric(),  
 PMM=numeric(),  
 LM=numeric(),  
 MEAN=numeric(),  
 MI=numeric()  
 )

**Imputation**

In order to obtained and conduct appropriate analysis, the imputation was done using different techniques.

for(i in 1:length(STN)){  
 stnno <- STN[i]  
 filen <- paste('data/Patched\_',stnno,'.txt',sep='') #give the download a file name - here the only thing changing is the station number to set  
 FileOut<-filen  
 # reading data from Patch file  
 XL <- as.integer(grep("(yyyymmdd)", readLines(FileOut))) #lines to skip before the actual data  
 ColNames <- read.table(FileOut,header=FALSE,nrows=1,skip=(XL-2),colClasses = "character") #read column names  
 Units <- read.table(FileOut,header=FALSE,nrows=1,skip=(XL-1),colClasses = "character") #read units  
 Data <- data.frame(read.table(FileOut,header=FALSE,skip=XL)) #read data  
 colnames(Data) <- ColNames  
 Data$DateUse <- paste(substr(Data$Date,1,4),substr(Data$Date,5,6),substr(Data$Date,7,8),sep='-') #this is how R sees dates  
 Data <- Data[Data$DateUse>='1961-01-01' & Data$DateUse<='2018-12-31',]  
   
 # aggregating rainfall data  
 Mon <- strftime(Data$DateUse, "%m")  
 Year <- strftime(Data$DateUse, "%Y")  
 Rain <- Data$Rain #selected the variable of interest here - check colnames(Data)  
 RainData <- data.frame(Mon, Year, Rain) #select the variable you would like to process  
 RainMonthTotal <- aggregate(Rain ~ Mon + Year, RainData, FUN = sum) #monthly totala  
 RainYearTotal <- aggregate(Rain ~ Year, RainData, FUN = sum) #Yearly totala  
 RainYearMax <- aggregate(Rain ~ Year, RainData, FUN = max) #Yearly daily Max  
 RainDayMean <- aggregate(Rain ~ Day, Data, FUN = mean) #mean of the days of the year  
  
 # specify which variables should have missing data and % of missing data  
 c\_names = c("Rain")  
 prc\_missing = 0.20  
 #  
   
 RainMonthTotal$Mon <- as.numeric(as.character(RainMonthTotal$Mon))  
 RainMonthTotal$Year <- as.numeric(as.character(RainMonthTotal$Year))  
 RainMonthTotalMiss <- data.frame(id=1:nrow(RainMonthTotal),RainMonthTotal)  
   
 mdf <- missing\_data.frame(RainMonthTotalMiss)  
 pdf(paste("output/",STN[i],"\_NOMISS.pdf",sep=""))  
 image(mdf)  
 dev.off()  
   
 #   
 RainMonthTotalMiss <- RainMonthTotalMiss %>%  
 gather(var, value, -id) %>% # reshape data  
 mutate(r = runif(nrow(.)), # simulate a random number from 0 to 1 for each row  
 value = ifelse(var %in% c\_names & r <= prc\_missing, NA, value)) %>% # if it's one of the variables you specified and the random number is less than your threshold update to NA  
 select(-r) %>% # remove random number  
 spread(var, value) # reshape back to original format  
 #  
   
 RainMonthTotalMiss <- RainMonthTotalMiss[,c('id','Mon','Year','Rain')]  
   
 #viewing missing pattern  
   
 mdf <- missing\_data.frame(RainMonthTotalMiss)  
 pdf(paste("output/",STN[i],"\_MISS.pdf",sep=""))  
 image(mdf)  
 dev.off()  
   
 #Impute missing  
 set.seed(10)  
 init=mice(RainMonthTotalMiss,maxit = 5)  
 meth=init$method  
 predM=init$predictorMatrix  
   
 cln=RainMonthTotalMiss  
   
 predM[, c("id")]=0  
 meth[c("Rain")]="pmm" #predictive mean matching  
 imputed=mice(RainMonthTotalMiss, method=meth,predictorMatrix = predM,m=5)  
 imputed=complete(imputed)  
  
 RainMonthTotalMiss$RainImputed=imputed$Rain  
 RainMonthTotalMiss$RainOriginal=RainMonthTotal$Rain  
   
 #regression imputation  
   
 lm.imp.1=lm(Rain~Mon +Year,data=RainMonthTotalMiss)  
 pred.1=predict(lm.imp.1,RainMonthTotalMiss)  
 RainMonthTotalMiss$lmP=impute(RainMonthTotalMiss$Rain,pred.1)  
   
 # mean imputation  
 meanrain = mean(RainMonthTotalMiss$Rain,na.rm=TRUE)  
   
 for (e in 1:nrow(RainMonthTotalMiss)){  
 if(is.na(RainMonthTotalMiss$Rain[e])){  
 RainMonthTotalMiss$RainMeanImputed[e]=meanrain  
 } else{  
 RainMonthTotalMiss$RainMeanImputed[e]=RainMonthTotalMiss$Rain[e]  
 }  
 }  
   
 # mi packagae  
 imputations <-mi(mdf, n.iter = 2, n.chains = 1, max.minutes = 20)  
 impdf <-mi::complete(imputations, m = 1)  
 RainMonthTotalMiss$RainMiImputed = impdf$Rain  
   
 RainMissPMM=cln  
 RainMissLM=cln  
 RainMissMEAN=cln  
 RainMissMI=cln  
   
 RainMissPMM$Imputed=RainMonthTotalMiss$RainImputed  
 RainMissLM$Imputed=RainMonthTotalMiss$lmP  
 RainMissMEAN$Imputed=RainMonthTotalMiss$RainMeanImputed  
 RainMissMI$Imputed=RainMonthTotalMiss$RainMiImputed  
   
   
 #calc average differences  
 dat=subset(RainMonthTotalMiss,is.na(RainMonthTotalMiss$Rain))  
 dat$abspmm=abs(dat$RainImputed-dat$RainOriginal)  
 dat$abslm=abs(dat$RainOriginal-dat$lmP)  
 dat$absmean=abs(dat$RainMeanImputed-dat$RainOriginal)  
 dat$absmi=abs(dat$RainOriginal-dat$RainMiImputed)  
   
 vec=c(STN[i],mean(dat$abspmm),mean(dat$abslm),mean(dat$absmean),mean(dat$absmi))  
 dist[nrow(dist)+1,]=vec  
   
 write.csv(RainMissPMM,paste("output/",STN[i],"\_PMM.csv"))  
 write.csv(RainMissLM,paste("output/",STN[i],"\_LM.csv"))  
 write.csv(RainMissMI,paste("output/",STN[i],"\_MI.csv"))  
 write.csv(RainMissMEAN,paste("output/",STN[i],"\_MEAN.csv"))  
 write.csv(RainMonthTotalMiss,paste("output/",STN[i],"\_ALL.csv"))  
}

**Predictive mean matching (PMM)**

The first method used to conduct imputation was called predictive mean matching. The predictive mean matching is the use of different variables to account for the distribution of the original variables in order to generate the values which can match the skewed variables. The predictive mean matching test was conducted using MICE package with the listed codes below. It is also important to point that the last two lines of the codes are regarded as the imputed values into the new columns of the main dataset which are used for the comparison of the datasets.

#Impute missing  
 set.seed(10)  
 init=mice(RainMonthTotalMiss,maxit = 5)  
 meth=init$method  
 predM=init$predictorMatrix  
   
 cln=RainMonthTotalMiss  
   
 predM[, c("id")]=0  
 meth[c("Rain")]="pmm" #predictive mean matching  
 imputed=mice(RainMonthTotalMiss, method=meth,predictorMatrix = predM,m=5)  
 imputed=complete(imputed)  
  
 RainMonthTotalMiss$RainImputed=imputed$Rain  
 RainMonthTotalMiss$RainOriginal=RainMonthTotal$Rain

This technique was helpful in obtaining the rain imputation data which were then analyzed to understand the trend of the data performance.

**Linear regression**

The linear regression techniques were used to determine the relationship between interested variables. According to Khalifeloo, Mohammad, & Heydari (2015) linear regression analysis is one of the widely used statistical methods in different science to determine the relationship between two or more variables. As stated by Cisty & Celar (2015) the dependent variables are known as response and the independent variables are regarded as explanatory variables. However, the linear regression techniques assume that there is a linear relationship which exists between dependent variable and predictor. In order to determine the linear regression, I first define the regression model of the datasets which mostly regarded as dependent variable Y and independent variable X. However, in the case of this study, the variables were identified as Day, Month and Year. These were used to create workable data for the study. It was also used the inbuilt R code and therefore, there no need for a package to be used. The data used for the inbuilt is therefore, illustrated below:

#impute f  
impute<-function(a,a.impute){  
 ifelse(is.na(a),a.impute,a)  
}  
  
 lm.imp.1=lm(Rain~Mon +Year,data=RainMonthTotalMiss)  
 pred.1=predict(lm.imp.1,RainMonthTotalMiss)  
 RainMonthTotalMiss$lmP=impute(RainMonthTotalMiss$Rain,pred.1)  
   
 RainMissPMM=cln  
 RainMissLM=cln  
   
 RainMissPMM$Imputed=RainMonthTotalMiss$RainImputed  
 RainMissLM$Imputed=RainMonthTotalMiss$lmP

The third method replaced missing values with the mean of the rainfall dataset for that station

meanrain = mean(RainMonthTotalMiss$Rain,na.rm=TRUE)  
  
 for (i in 1:nrow(RainMonthTotalMiss)){  
 if(is.na(RainMonthTotalMiss$Rain[i])){  
 RainMonthTotalMiss$RainMeanImputed[i]=meanrain  
 } else{  
 RainMonthTotalMiss$RainMeanImputed[i]=RainMonthTotalMiss$Rain[i]  
 }  
 }

**Mean imputation**

The predictive mean was conducted using SPSS to get the accurate answers. As illustrated in the diagram 1 below. The man for Rain Imputed was obtained to be 79.05 and standard deviation to be 109.036. The mean for rain original was also obtained to be 80.76 and standard deviation to 108.399. However, the mean for rain mean imputed was obtained to be 80.282 and standard deviation was 99.0665.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Descriptive Statistics** | | | | | |
|  | N | Minimum | Maximum | Mean | Std. Deviation |
| RainImputed | 696 | .000000000000000 | 685.300000000000000 | 79.052298850574870 | 109.036393213767800 |
| RainOriginal | 696 | .000000000000000 | 685.300000000000000 | 80.762356321839020 | 108.399323803511210 |
| lmP | 696 | .000000000000000 | 685.300000000000000 | 79.870724337925680 | 99.625685829049870 |
| RainMeanImputed | 696 | .000000000000000 | 685.300000000000000 | 80.282459312839100 | 99.066543001347810 |
| RainMiImputed | 696 | -176.056583746439000 | 685.300000000000000 | 73.109241653545140 | 110.075488450424870 |
| Valid N (listwise) | 696 |  |  |  |  |

# Bibliography

Carvalho, J. R., Almeida, J. E., Assad, D. E., & Nakai, M. A. 2017. Model for Multiple Imputation to Estimate Daily Rainfall Data and Filling of Faults. *Revista Brasileira de Meteorologia* , 575-583.

Cisty, M., & Celar, L. 2015. Using R in Water Resources Education. *International Journal for Innovation Education and Research* *, 2* (3), 2-38.

Gelman, A. and Hill, J. 2011. “Opening Windows to the Black Box.” *Journal of Statistical Software*, *40*.

Khalifeloo, M. H., Mohammad, M., & Heydari, M. 2015. Multiple Imputation For Hydrological Missing Data By Using A Regression Method (Klang River Basin). *International Journal of Research in Engineering and Technology* , 2-38.

van Buuren, S. and Groothuis-Oudshoorn, K. 2011. mice: Multivariate Imputation by Chained Equations in R. Journal of Statistical Software, 45(3), 1-67. URL <https://www.jstatsoft.org/v45/i03/>.

Wickham, H. 2017. tidyverse: Easily Install and Load the ‘Tidyverse’. R package version 1.2.1. <https://CRAN.R-project.org/package=tidyverse>