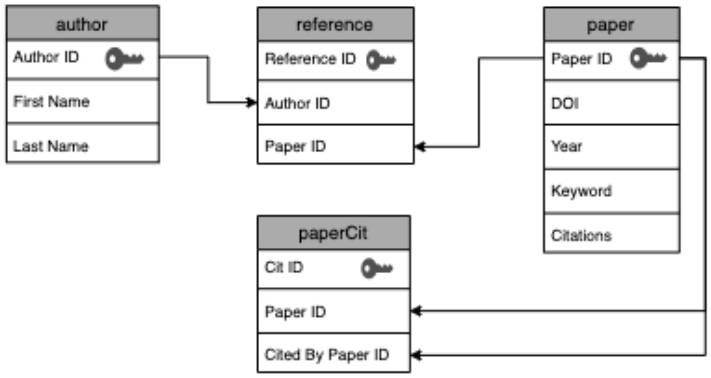
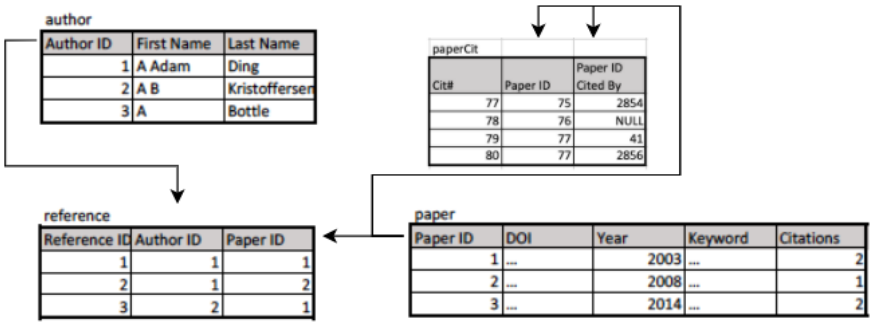
Problem 1

Tables:

Reference: reference ID (Primary key), Author ID (foreign key), Paper ID (foreign key)
Author: Author ID (primary key), First Name, Last Name
Paper: Paper ID (primary key), DOI, Year, Keyboard, Citation Count
paperCit: Citation ID (primary key), Paper ID (foreign key), CitedBy Paper ID (foreign key)





author

Author ID	First Name	Last Name
1	A Adam	Ding
2	A B	Kristoffersen
3	A	Bottle

paperCit

Cit#	Paper ID	Paper ID Cited By
77	75	2854
78	76	NULL
79	77	41
80	77	2856

reference

Reference ID	Author ID	Paper ID
1	1	1
2	1	2
3	2	1

paper

Paper ID	DOI	Year	Keyword	Citations
1	...	2003	...	2
2	...	2008	...	1
3	...	2014	...	2

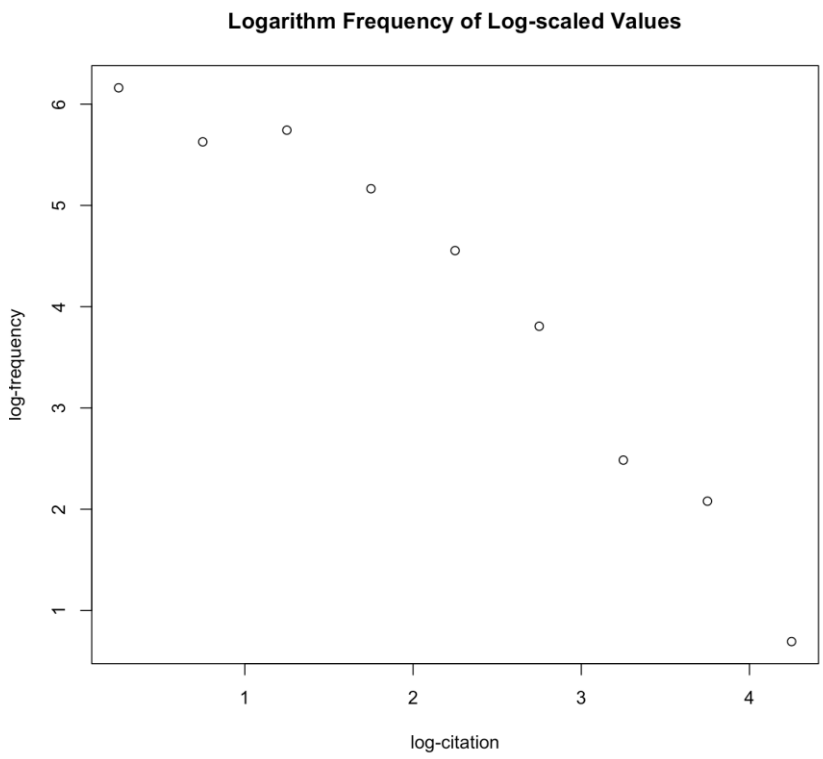
Problem 2

```
> dbListTables(db.stat)
[1] "authorNames" "paperCit"    "paperInfo"   "refTable"
>
> head(dbReadTable(db.stat, "paperInfo"))
  Paper_numb     DOI year                                                              title citCounts
1          1 10.1214 2012                   Rerandomization to improve covariate balance in experiments         0
2          2 10.1214 2012                   Realized {L}aplace transforms for pure-jump semimartingales         0
3          3 10.1214 2012                                       Degrees of freedom in lasso problems         0
4          4 10.1214 2012 Noisy matrix decomposition via convex relaxation: optimal rates in high dimensions         0
5          5 10.1214 2012               Nonparametric regression with nonparametrically generated covariates         0
6          6 10.1214 2012                        Bayesian empirical likelihood for quantile regression         0
> head(dbReadTable(db.stat, "refTable"))
  refID AuthorID Paper_numb
1     1        1       1887
2     2        1       3019
3     3        2       1696
4     4        3       1100
5     5        4       1203
6     6        4       1593
> head(dbReadTable(db.stat, "paperCit"))
  refID3 Paper_numb CitedBy
1      1          1      NA
2      2          2      NA
3      3          3      NA
4      4          4      NA
5      5          5      NA
6      6          6      NA
> head(dbReadTable(db.stat, "authorNames"))
   Fname         Lname AuthorID
1 A_Adam          Ding        1
2    A_B Kristoffersen        2
3      A        Bottle        3
4    A_C       Davison        4
5      A    Chatterjee        5
6    A_D      Tsodikov        6
>
```

Problem 3

1 Antonio Lijoi	37 J_N_K Rao	74 Peihua Qiu
2 Arnaud Doucet	38 J_S Marron	75 Pengfei Li
3 Aurore Delaigle	39 Jae_Kwang Kim	76 Peter_B Gilbert
4 Bing Li	40 James_R Robins	77 Peter Buhlmann
5 Bradley Efron	41 Jane-Ling Wang	78 Peter_D Hoff
6 Bruce_G Lindsay	42 Jason_P Fine	79 Peter_E Jupp
7 Chih-Ling Tsai	43 Jean_D Opsomer	80 Peter Hall
8 Christian_P Robert	44 Jeffrey_D Hart	81 Peter Radchenko
9 Daniel_J Nordman	45 Jens_Perch Nielsen	82 Peter_Z_G Qian
10 David Dunson	46 Ji Zhu	83 Piotr Fryzlewicz
11 David Ruppert	47 Jiahua Chen	84 Qiwei Yao
12 David Siegmund	48 Jiancheng Jiang	85 R_Dennis Cook
13 Donglin Zeng	49 Jianhua_Z Huang	86 Raymond_J Carroll
14 Dylan_S Small	50 Jianqing Fan	87 Robert_J Tibshirani
15 Efstathios Paparoditis	51 Jianwen Cai	88 Ross_L Prentice
16 Els Goetghebeur	52 Jing Qin	89 Runze Li
17 Elvezio Ronchetti	53 John_D Kalbfleisch	90 Shiqing Ling
18 Enno Mammen	54 Jonathan_E Taylor	91 Song_Xi Chen
19 F_Jay Breidt	55 Joseph_G Ibrahim	92 Soumendra_N Lahiri
20 Faming Liang	56 Judith Rousseau	93 Stephen_G Walker
21 Fang Yao	57 Jun Zhu	94 Stephen_M_S Lee
22 Gareth_M James	58 Lan Wang	95 Stijn Vansteelandt
23 George Casella	59 Larry Wasserman	96 Subhashis Ghosal
24 George Michailidis	60 Liping Zhu	97 Susan_A Murphy
25 Gerda Claeskens	61 Liugen Xue	98 T_Tony Cai
26 Guido Consonni	62 Lixing Zhu	99 Tapabrata Maiti
27 Haibo Zhou	63 M_J Bayarri	100 Theo Gasser
28 Hannu Oja	64 Malay Ghosh	101 Thomas_S Richardson
29 Hans-Georg Muller	65 Marc_G Genton	102 Tyler_J VanderWeele
30 Hansheng Wang	66 Mark_J_van_der Laan	103 Willa_W Chen
31 Heping Zhang	67 Michael_G Akritas	104 Xiao-Hua Zhou
32 Holger Dette	68 Michael_L Stein	105 Xiaofeng Shao
33 Hongtu Zhu	69 Michael_R Kosorok	106 Yanyuan Ma
34 Hui Zou	70 Ming Yuan	107 Yi Lin
35 Huixia_Judy Wang	71 Naisyin Wang	108 Yingcun Xia
36 Igor Prunster	72 Natalie Neumeyer	109 Yoav Benjamini
	73 Nicolai Meinshausen	110 Yongdai Kim

Problem 4



The plot follows a linear trend.

Appendix

Problems 1 and 2

```
##########Get paper DoI Info

paperList<-read.csv("paperList.txt",header=T)
doisplit<-strsplit(as.character(paperList$DOI),split = "/")
firstDoI<-sapply(doisplit,'[[',1)
paperList$DOI<-as.factor(firstDoI)
Paper_numb<-1:nrow(paperList)
paperInfo<-data.frame(Paper_numb,paperList)


########## Author first and last names
authors<-read.table("authorList.txt")
splitnames<-strsplit(as.character(authors$V1),split= " ")
Lname<-sapply(splitnames,tail,1)
firstnames<-sapply(splitnames,head,n=-1)
Fname<-sapply( firstnames, paste0, collapse="_")
AuthorID<-1:length(Fname)
authorNames<-data.frame(AuthorID,Fname,Lname)

########## Authorship
biadjauthorship<-read.table("authorPaperBiadj.txt",header=F)
authorship<-data.frame(AuthorID,biadjauthorship)

########## Papers written by authors
numbs<-list()
for (i in 1:nrow(authorship)){
  numbs[[i]]<-which(apply(authorship[i,2:ncol(authorship)], 2, function(x) any(grepl(1, x))))
}
df00<-data.frame(AuthorID,Paper_numb=unlist(lapply(numbs,paste0,collapse=" ")))
library(tidyr)
df01<-separate_rows(df00,Paper_numb,sep=" ")
refID<-1:nrow(df01)
refTable<-data.frame(refID,df01)


########## Citation papers table ref again
citadj<-read.table("paperCitAdj.txt",header=F)
citations<-data.frame(Paper_numb,citadj)
```

```
numbs3<-list()      ############ THIS CODE FOR FINDING WHICH PAPER WAS CITED BY WHICH PAPER
for (i in 1:nrow(citations)){
  numbs3[[i]]<-which(apply(citations[i,2:ncol(citations)], 2, function(x) any(grepl(1, x))))
}
df002<-data.frame(Paper_numb,CitedBy=unlist(lapply(numbs3,paste0,collapse=" ")))
df003<-separate_rows(df002,CitedBy,sep=" ")
refID3<-1:nrow(df003)
refTable3<-data.frame(refID3,df003)
refTable3$CitedBy<-as.numeric(refTable3$CitedBy)


write.csv(refTable,file="refTable.csv",row.names=F)
write.csv(paperInfo,file="paperInfo.csv",row.names = F)
write.csv(authorNames,file="authorNames.csv",row.names = F)
write.csv(refTable3,file="paperCit.csv",row.names=F)

setwd('~/Desktop')

## read csv file
authorNames = read.csv('authorNames.csv')
#authorship = read.csv('authorship.csv')
#citations = read.csv("citations.csv")
paperInfo = read.csv("paperInfo.csv")
refTable = read.csv("refTable.csv")
paperCit = read.csv('paperCit.csv')
library(RSQLite)
## first create an empty database
db.stat = dbConnect(SQLite(), dbname="stat.sqlite")


## write the csv data into database
dbWriteTable(conn = db.stat, name = "authorNames", authorNames, overwrite=T, row.names = FALSE)
## check the content of the database
dbReadTable(db.stat, "authorNames")

dbWriteTable(conn = db.stat, name = "paperInfo", paperInfo, overwrite=T, row.names = FALSE)
## check the content of the database
dbReadTable(db.stat, "paperInfo")

dbWriteTable(conn = db.stat, name = "refTable", refTable, overwrite=T, row.names = FALSE)
## check the content of the database
dbReadTable(db.stat, "refTable")
```

```
dbWriteTable(conn = db.stat, name = "paperCit", paperCit, overwrite=T, row.names = FALSE)
## check the content of the database
dbReadTable(db.stat, "paperCit")


dbListTables(db.stat)
```

Problem 3

```
#code for extracting the author names from the database with who published at least 4 out of the 6 in the list of DOIs
DOI1 <- dbGetQuery(db.stat, "SELECT Paper_numb FROM paperInfo WHERE DOI == 10.1214")

#adds the list to the db.stat so conditional statements are allowed
dbWriteTable(conn = db.stat, name = "DOI1", DOI1, overwrite=T, row.names = FALSE)

DOI2 <- dbGetQuery(db.stat, "SELECT Paper_numb FROM paperInfo WHERE DOI == 10.1093")

dbWriteTable(conn = db.stat, name = "DOI2", DOI2, overwrite=T, row.names = FALSE)

DOI3 <- dbGetQuery(db.stat, "SELECT Paper_numb FROM paperInfo WHERE DOI == 10.1046")

dbWriteTable(conn = db.stat, name = "DOI3", DOI3, overwrite=T, row.names = FALSE)

DOI4 <- dbGetQuery(db.stat, "SELECT Paper_numb FROM paperInfo WHERE DOI == 10.1111")
dbWriteTable(conn = db.stat, name = "DOI4", DOI4, overwrite=T, row.names = FALSE)

DOI5 <- dbGetQuery(db.stat, "SELECT Paper_numb FROM paperInfo WHERE DOI == 10.1080")
dbWriteTable(conn = db.stat, name = "DOI5", DOI5, overwrite=T, row.names = FALSE)

DOI6 <- dbGetQuery(db.stat, "SELECT Paper_numb FROM paperInfo WHERE DOI == 10.1198")
dbWriteTable(conn = db.stat, name = "DOI6", DOI6, overwrite=T, row.names = FALSE)

#code for extracting the author ID from who have written the paper with the corresponding DOI
ars_ID1 <- dbGetQuery(db.stat, "
        SELECT AuthorID
        FROM refTable
```

```
        WHERE Paper_numb IN DOI1")

ars_ID2 <- dbGetQuery(db.stat, "
        SELECT AuthorID
            FROM refTable
                WHERE Paper_numb IN DOI2")

ars_ID3 <- dbGetQuery(db.stat, "
        SELECT AuthorID
                FROM refTable
                WHERE Paper_numb IN DOI3")

ars_ID4 <- dbGetQuery(db.stat, "
        SELECT AuthorID
                FROM refTable
                WHERE Paper_numb IN DOI4")

ars_ID5 <- dbGetQuery(db.stat, "
        SELECT AuthorID
                FROM refTable
                WHERE Paper_numb IN DOI5")

ars_ID6 <- dbGetQuery(db.stat, "
        SELECT AuthorID
                FROM refTable
                WHERE Paper_numb IN DOI6")

#empty vector to store the IDs that have published with at least 4 of 6 DOIs listed
name_ID <- vector()

#for every ID in the list of the ids, add 1 to counter if the author published a paper with the DOI
for (i in 1:3607){
  ctr = 0
  if (i %in% ars_ID1$AuthorID){
    ctr = ctr+1
  }
  if (i %in% ars_ID2$AuthorID){
    ctr = ctr+1
  }
  if (i %in% ars_ID3$AuthorID){
    ctr = ctr+1
  }
  if (i %in% ars_ID4$AuthorID){
    ctr = ctr+1
```

```
  }
  if (i %in% ars_ID5$AuthorID){
    ctr = ctr+1
  }
  if (i %in% ars_ID6$AuthorID){
    ctr = ctr+1
  }
  if (ctr >=4){
    print(i)
    name_ID <- c(name_ID, i)
  }
  else{next}
}

#make the name_ID into a dataframe
name_ID <- data.frame(name_ID)

#add name_ID to the db.stat
dbWriteTable(conn = db.stat, name = "name_ID", name_ID, overwrite=T, row.names = FALSE)

#conditional statement to extract first and last name of author
names <- dbGetQuery(db.stat, "SELECT Fname, Lname FROM authorNames WHERE
AuthorID IN name_ID")

names
```

Problem 4

```
before2010 <- dbGetQuery(db.stat, "SELECT Paper_numb, DOI,year, title,citCounts FROM
paperInfo WHERE year < 2010")
dbWriteTable(conn = db.stat, name = "before2010", before2010, overwrite=T, row.names =
FALSE)
df=as.data.frame(dbReadTable(db.stat, "before2010"))
#head(df)

library(ggplot2)
#x=hist(df$citCounts,plot=F)$mids
#y=hist(df$citCounts,plot=F)$counts
#plot(log(x),log(y))

freqcounts=hist(log(df$citCounts),plot=F)$counts
logcts=hist(log(df$citCounts),plot=F)$mids
```

```
plot(logcts,log(freqcounts),main='Logarithm Frequency of Log-scaled Values',xlab = 'log-
citation',ylab = 'log-frequency')
```