Jisu Kim (jk5nc), Timothy Kim (tsk8va), Olivia Ryu (hr2ad)
STAT 3280-001
Professor Tianxi Li
1 May 2020

Homework 4

Problem 1

Author

Column	**Type**	**Allowable Values**
Au_id (pk)	Smallint (unsigned)	
Au_fname	Varchar(20)	
Au_lname	Varchar(20)	

Paper

Column	**Type**	**Allowable Values**
P_id (pk)	Smallint (unsigned)	
DOI	Varchar(20)	
Year	Year	
Title	Varchar(50)	
Citation_counts	Smallint (unsigned)	

Authorship

Column	**Type**	**Allowable Values**
Au_id (pk, fk)	Smallint (unsigned)	
P_id (pk, fk)	Smallint (unsigned)	

Citations

Column	**Type**	**Allowable Values**
P_id (pk, fk)	Smallint (unsigned)	
Citing (pk, fk)	Smallint (unsigned)	

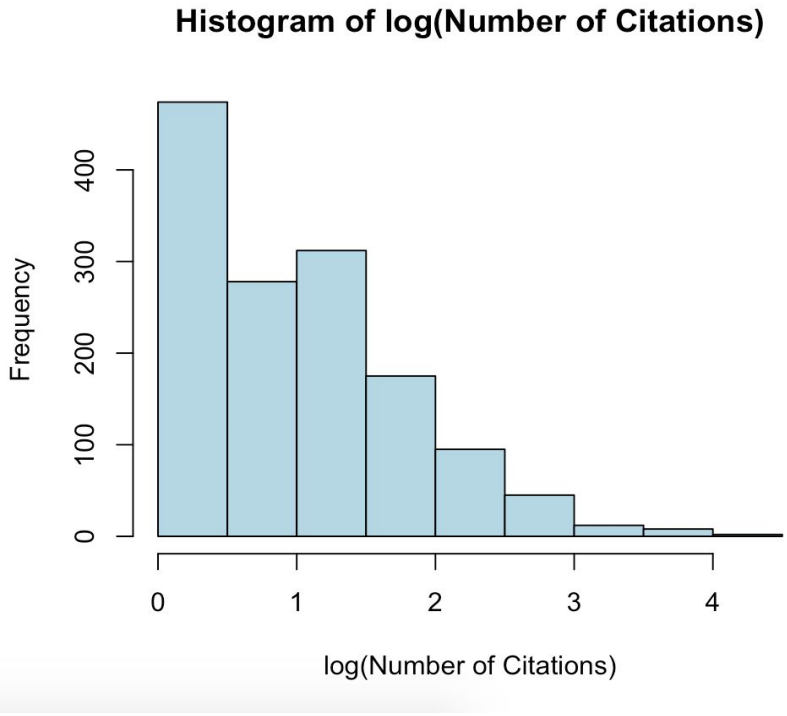
Problem 2

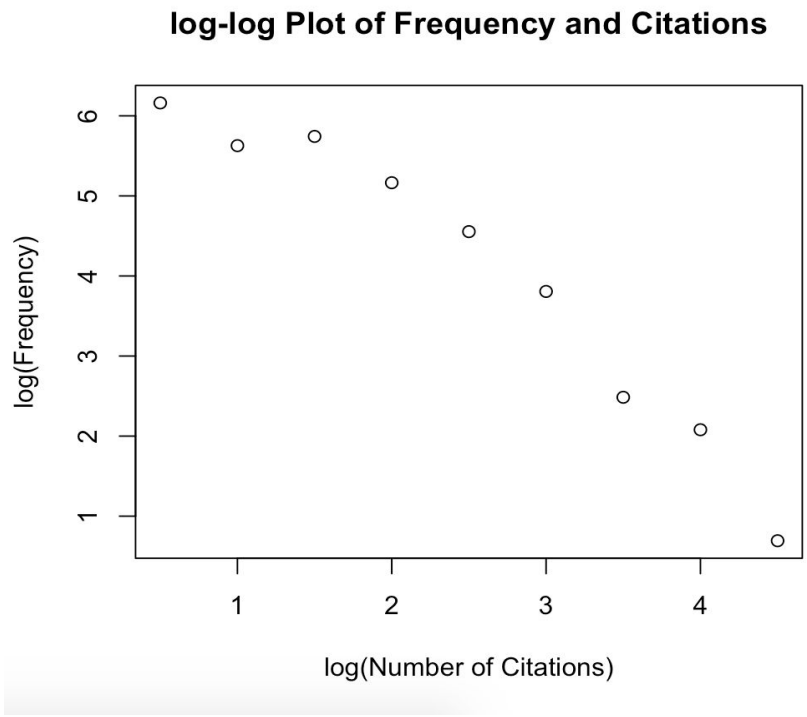
```
install.packages('RSQLite')
library(RSQLite)

## Create an empty database
db.stat = dbConnect(SQLite(), dbname="stat.sqlite")

## Populate tables with data
# Converting source data into data.frames
authors <- read.table("authorList.txt", stringsAsFactors=FALSE)
new.authors <- data.frame(1:dim(authors)[1],
                          sapply(strsplit(authors$V1, " "), head, 1),
                          sapply(strsplit(authors$V1, " "), tail, 1))
colnames(new.authors) <- c("Au_id", "Au_fname", "Au_lname")

papers <- read.delim("paperListnew.txt", header=TRUE, sep=",")
new.papers <- data.frame(1:dim(papers)[1], papers)
colnames(new.papers) <- c("P_id", "DOI", "Year", "Title", "Citation_counts")

A2P <- read.table("authorPaperBiadj.txt")
A2P.coords <- data.frame(which(A2P==1, arr.ind = TRUE))
colnames(A2P.coords) <- c("Au_id", "P_id")

cit.adj <- read.table("paperCitAdj.txt")
cit.adj.coords <- data.frame(which(cit.adj==1, arr.ind = TRUE))
colnames(cit.adj.coords) <- c("P_id", "Citing")

# Write the data into database
dbWriteTable(conn = db.stat, name = "Author", new.authors, overwrite=T,
row.names = FALSE)
dbWriteTable(conn = db.stat, name = "Paper", new.papers, overwrite=T,
row.names = FALSE)
dbWriteTable(conn = db.stat, name = "Authorship", A2P.coords, overwrite=T,
row.names = FALSE)
dbWriteTable(conn = db.stat, name = "Citations", cit.adj.coords, overwrite=T,
row.names = FALSE)

# Check the content of the database
dbReadTable(db.stat, "Author")
dbReadTable(db.stat, "Paper")
dbReadTable(db.stat, "Authorship")
dbReadTable(db.stat, "Citations")
```

Problem 3

```
dbGetQuery(db.stat, "select Au_fname, Au_lname from Author where Au_id IN
(select Au_id from Authorship where P_id IN (select P_id from Paper where (DOI
like '%10.1214%') or (DOI like '%10.1093%') or (DOI like '%10.1046%') or (DOI
like '%10.1111%') or (DOI like '%10.1080%') or (DOI like '%10.1198%')) group
by Au_id having count(Au_id)>=4)")
```

Problem 4





Since the log-frequency vs. log-citation shows a linear trend, the citation system most likely follows a power-law distribution.

```
Freq = dbGetQuery(db.stat, "select P_id, count(P_id) as numCited from
Citations where P_id in (select P_id from Paper where Year < 2010) group
by P_id")
plot1 <- hist(log(Freq$numCited), col="lightblue")
frequencies <- plot1$counts
breaks <- plot1$breaks
citations <- breaks[-1]

plot(citations, log(frequencies))
```