DA5020 - Week 10 SQLite and comparing dplyr to SQL Developed By Milad Tatari

2019-11-17

This week you are responsible for chapters 10, 11, 12 in the "Data Collection, Integration and Analysis" textbook. Review each chapter separately and work through all examples in the text BEFORE starting the assignment. You will use the schema you developed in homework 6 to store data in SQLite.

This week's assignment you use the relational schema you designed in week 6 and store data into the SQLite relational database system. Load the Unemployment and Educational data files into R studio. One file contains yearly unemployment rates from 1970 to 2015, for counties in the US. The other file contains aggregated data percentages on the highest level of education achieved for each census member. The levels of education are: "less than a high school diploma", "high school diploma awarded", "attended some college", "college graduate and beyond". The census tracks the information at the county level and uses a fips number to represent a specific county within a U.S. state. The fips number is a 5 digit number where the first two digits of the fips number represents a U.S. state, while the last three digits represent a specific county within that state.

Questions

1. Revisit the census schema you created for homework 6. After installing SQLite, implement the tables for your database design in SQLite and load the data into the correct tables using either SQL INSERT statements or CSV loads. Make sure the database design is normalized (at least 3NF) and has minimal redundancy. Make sure your SQLite tables have primary keys as well as foreign keys for relationships. (20 points)

```r
a <- read.csv("FipsEducationsDA5020v2.csv")
b <- read.csv("FipsUnemploymentDA5020.csv")
#install.packages("stringr")
library(stringr)
#install.packages("tidyr")
library(tidyr)
library(dplyr)

#every measurement for a year and fips is reapeted 4 times which is not good, so we use spread function
a.new <- a%>%
  spread(key=percent_measure,value=percent)

#Seperating the state and counties
a.sep <- a.new %>%
  separate(county_state, into = c("state","county"))
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 15721 rows
## [6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24,
## 25, ...].
```

```r
d <- select(a.sep, fips,county,state)
#making the FIPS data frame to make the table
fipsdf <- as_data_frame(d)%>%
  group_by(fips,county,state)  %>%
  summarize()
```

```
## Warning: `as_data_frame()` is deprecated, use `as_tibble()` (but mind the new semantics).
## This warning is displayed once per session.
```

```
# Renaming the a.sep colmns to amke it easier to work with.
colnames(a.sep)<- c("fips", "year","state","county","rural","description","PLUS4College","SOMEcollege",


#install.packages("RSQLite")

library("RSQLite")
# open a connection to SQLite and create the EDUEMPDB database
db<-dbConnect(SQLite(),dbname="EDUEMPDBmiladTA.sqlite")
summary(db)
```

```
##           Length            Class             Mode
##                1 SQLiteConnection               S4
```

```
dbListTables(db)
```

```
## character(0)
```

```
# In SQLite foreign key constraints are disabled by default, so they must be enabled for each database
dbSendQuery(conn = db, "pragma foreign_keys=on;")
```

```
## <SQLiteResult>
##   SQL  pragma foreign_keys=on;
##   ROWS Fetched: 0 [complete]
##        Changed: 0
```

```
# Use the unique function to remove redundancy
FIPS.DF<-unique(cbind.data.frame(as.integer(a.sep$fips),as.character(a.sep$county),as.character(a.sep$s
colnames(FIPS.DF)<- c("fipsID", "County","State")

# Create the FIPS table, specifying fipsID as the PRIMARY KEY
# Since we are specifying a primary ID, there is no need for autoincremented rowid that is automaticall
dbSendQuery(conn = db,  "CREATE TABLE FIPS (
            fipsID INTEGER PRIMARY KEY,
            County TEXT,
            State TEXT)
            WITHOUT ROWID")
```

```
## Warning: Closing open result set, pending rows
```

```
## <SQLiteResult>
##   SQL  CREATE TABLE FIPS (
##              fipsID INTEGER PRIMARY KEY,
##              County TEXT,
##             State TEXT)
##             WITHOUT ROWID
##   ROWS Fetched: 0 [complete]
##        Changed: 0
```

```
dbWriteTable(conn = db, name = "FIPS", value = FIPS.DF, row.names=FALSE, append = TRUE)
```

```
## Warning: Closing open result set, pending rows
```

```
dbListTables(db)
```

```
## [1] "FIPS"
```

```
#dbReadTable(db, "FIPS")
# Making the rural table
s1 <- select(a.sep, rural,description)
rural.DF <- as_data_frame(s1)%>%
  group_by(rural,description)  %>%
  summarize()
rural.DF <- unique(cbind.data.frame(as.integer(rural.DF$rural),as.character(rural.DF$description)))
colnames(rural.DF)<- c("ruraliD","description")
library(dplyr)
#rural.DF<-rural.DF[1:9,]


dbSendQuery(conn = db,  "CREATE TABLE RURAL(
            ruraliD INTEGER PRIMARY KEY,
            description TEXT)
            WITHOUT ROWID")
```

```
## <SQLiteResult>
##   SQL  CREATE TABLE RURAL(
##             ruraliD INTEGER PRIMARY KEY,
##             description TEXT)
##             WITHOUT ROWID
##   ROWS Fetched: 0 [complete]
##        Changed: 0
```

```
# insert the RURAL data frame into the Student table in the EDUEMPDB.sqlite database make sure you set
dbWriteTable(conn = db, name = "RURAL", value = rural.DF, row.names=FALSE, append = TRUE)
```

```
## Warning: Closing open result set, pending rows
```

```
dbListTables(db)
```

```
## [1] "FIPS"  "RURAL"
```

```
dbReadTable(db, "RURAL")
```

```
##    ruraliD
## 1        1
## 2        2
## 3        3
## 4        4
## 5        5
## 6        6
## 7        7
## 8        8
## 9        9
## 10      10
##                                                                            description
## 1                              Counties in metro areas of 1 million population or more
## 2                           Counties in metro areas of 250,000 to 1 million population
## 3                             Counties in metro areas of fewer than 250,000 population
## 4                         Urban population of 20,000 or more, adjacent to a metro area
## 5                     Urban population of 20,000 or more, not adjacent to a metro area
## 6                        Urban population of 2,500 to 19,999, adjacent to a metro area
## 7                    Urban population of 2,500 to 19,999, not adjacent to a metro area
## 8      Completely rural or less than 2,500 urban population, adjacent to a metro area
## 9  Completely rural or less than 2,500 urban population, not adjacent to a metro area
## 10                                                                                NULL
```

```
#Create the education table, specifying fips ID  and ruralID as foreign keys.
# In this table there is no column that can be used as a primary ID, so we will have to use and autoinc

Education.DF<-unique(cbind.data.frame(as.integer(a.sep$fips),as.character(a.sep$year),as.integer(a.sep$r

colnames(Education.DF)<- c("fipsID","YEAR","rural", "PLUS4College","SOMEcollege","DIPLOMA","LESSDiploma
f <- Education.DF%>%
  select(rural)%>%
  group_by(rural)%>%
  summarise()

dbSendQuery(conn = db,  "CREATE TABLE Education(
            fipsID INTEGER,
            YEAR INTEGER,
            rural INTEGER,
            PLUS4College REAL,
            SOMEcollege REAL,
            DIPLOMA REAL,
            LESSDiploma REAL,
            FOREIGN KEY(fipsID) REFERENCES FIPS(fipsID)
            FOREIGN KEY(rural) REFERENCES RURAL(ruraliD))")
```

```
## <SQLiteResult>
##   SQL  CREATE TABLE Education(
##               fipsID INTEGER,
##               YEAR INTEGER,
##               rural INTEGER,
##               PLUS4College REAL,
##               SOMEcollege REAL,
##               DIPLOMA REAL,
##               LESSDiploma REAL,
##             FOREIGN KEY(fipsID) REFERENCES FIPS(fipsID)
##             FOREIGN KEY(rural) REFERENCES RURAL(ruraliD))
##   ROWS Fetched: 0 [complete]
##        Changed: 0
```

```
dbWriteTable(conn = db, name = "Education", value = Education.DF,row.names = FALSE,append = TRUE)
```

```
## Warning: Closing open result set, pending rows
```

```
head(dbReadTable(db,"Education"))
```

```
##   fipsID YEAR rural PLUS4College SOMEcollege DIPLOMA LESSDiploma
## 1   1000 1970    10          7.8         7.5    25.9        58.7
## 2   1000 1980    10         12.2        12.5    31.8        43.5
## 3   1000 1990    10         15.7        21.7    29.4        33.1
## 4   1000 2000    10         19.0        25.9    30.4        24.7
## 5   1000 2015    10         23.5        29.7    31.0        15.7
## 6   1001 1970     2          6.4         7.7    31.1        54.8
```

```
EMPLOY.DF<-unique(cbind.data.frame(as.integer(b$fips),as.integer(b$year),as.double(b$percent_unemployed)

colnames(EMPLOY.DF)<- c("fips","year","unemployedRate")

dbSendQuery(conn = db,  "CREATE TABLE EMPLOYMENT (
            fips INTEGER,
            year INTEGER,
            unemployedRate REAL,
            FOREIGN KEY (fips) REFERENCES FIPS(fipsID))")
```

```
## <SQLiteResult>
##   SQL  CREATE TABLE EMPLOYMENT (
##             fips INTEGER,
##             year INTEGER,
##             unemployedRate REAL,
##             FOREIGN KEY (fips) REFERENCES FIPS(fipsID))
##   ROWS Fetched: 0 [complete]
##        Changed: 0
```

```
dbWriteTable(conn = db, name = "EMPLOYMENT", value = EMPLOY.DF,row.names = FALSE,append = TRUE)
```

```
## Warning: Closing open result set, pending rows
```

```
head(dbReadTable(db,"EMPLOYMENT"))
```

```
##   fips year unemployedRate
## 1 1000 2007            4.0
## 2 1000 2008            5.7
## 3 1000 2009           11.0
## 4 1000 2010           10.5
## 5 1000 2011            9.6
## 6 1000 2012            8.0
```

```
#We have created 4 tables as follows which have the minimum redundancy and are 3NF:
#FIPS: fipsID is the PrimaryKEY
#RURAL: ruraliD is the primary key. "NULL" is conisdered as 10 (integer)
#EDUCATION: fipsID and rural are the foreign keys
#EMPLOYMENT: fipsID is the foreign key
```

2. Write SQL expressions to answer the following queries: (40 points)

- 2.0 In the year 1970, what is the population percent that did not earn a high school diploma for the Nantucket county in Massachusetts? What about the year 2015?

```
dbGetQuery(db, "Select LESSDiploma FROM Education LEFT JOIN FIPS ON  Education.fipsID = FIPS.fipsID WHEI
```

```
##   LESSDiploma
## 1        33.7
```

```
# returns 33.7

dbGetQuery(db, "Select LESSDiploma FROM Education LEFT JOIN FIPS ON  Education.fipsID = FIPS.fipsID WHEI
```

```
##   LESSDiploma
## 1         5.2
```

```
#returns 5.2
```

- 2.1 What is the average population percentage that did not earn a high school diploma for the counties in Alabama for the year 2015?

```
dbGetQuery(db, "SELECT AVG(LESSDiploma) FROM Education LEFT JOIN FIPS ON  Education.fipsID = FIPS.fipsI
```

```
##   AVG(LESSDiploma)
## 1         19.75882
```

- 2.2 What is the average percentage of college graduates for the counties in the state of Massachusetts for the year 2015?

```
dbGetQuery(db, "SELECT AVG(PLUS4College) FROM Education LEFT JOIN FIPS ON  Education.fipsID = FIPS.fips
```

```
##   AVG(PLUS4College)
## 1          38.52667
```

```
#It is 38.52%
```

- 2.3 Determine the average percentage of the population that did not earn a high school diploma for the counties in Alabama for each year within the dataset. The result should return the calendar year and the average percentage drop out rate for that year.

```
dbGetQuery(db, paste("SELECT YEAR, AVG(LESSDiploma)", "FROM Education LEFT JOIN FIPS ON  Education.fips
```

```
##   YEAR AVG(LESSDiploma)
## 1 1970         65.15882
## 2 1980         50.62059
## 3 1990         40.10000
## 4 2000         30.26471
## 5 2015         19.75882
```

- 2.4 What is the most common rural_urban code for the U.S. counties?

```
dbGetQuery(db, paste("SELECT rural, COUNT(rural)", "FROM Education","GROUP BY rural","ORDER BY COUNT(ru
```

```
##    rural COUNT(rural)
## 1      6         2961
## 2      7         2165
## 3      1         2153
## 4      9         2091
## 5      2         1890
## 6      3         1779
## 7      8         1097
## 8      4         1070
## 9      5          460
## 10    10          255
```

```
#the most common rural code is 6
```

- 2.5 Which counties have not been coded with a rural urban code? Return a result that contains two fields: County, State for the counties that has not been assigned a rural urban code. Do not return duplicate values in the result. Order the result alphabetically by state.

```
#rural code number 10 has not been assigned any rural ID
dbGetQuery(db, paste("SELECT county, state", "FROM Education LEFT JOIN FIPS ON  Education.fipsID = FIPS
```

```
##           County State
## 1         Alaska    AK
## 2        Alabama    AL
## 3       Arkansas    AR
## 4        Arizona    AZ
## 5     California    CA
## 6       Colorado    CO
## 7    Connecticut    CT
## 8       District    DC
## 9       Delaware    DE
## 10       Florida    FL
## 11       Georgia    GA
## 12        Hawaii    HI
## 13          Iowa    IA
## 14         Idaho    ID
## 15      Illinois    IL
## 16       Indiana    IN
## 17        Kansas    KS
## 18      Kentucky    KY
## 19     Louisiana    LA
## 20 Massachusetts    MA
## 21      Maryland    MD
## 22         Maine    ME
## 23      Michigan    MI
## 24     Minnesota    MN
## 25      Missouri    MO
## 26   Mississippi    MS
## 27       Montana    MT
## 28         North    NC
## 29      Nebraska    NE
## 30           New    NH
## 31        Nevada    NV
## 32          Ohio    OH
## 33      Oklahoma    OK
## 34        Oregon    OR
## 35  Pennsylvania    PA
## 36         Rhode    RI
## 37         South    SC
## 38     Tennessee    TN
## 39         Texas    TX
## 40          Utah    UT
## 41      Virginia    VA
## 42       Vermont    VT
## 43    Washington    WA
## 44     Wisconsin    WI
## 45          West    WV
## 46       Wyoming    WY
```

- 2.6 What is the minimal percentage of college graduates for the counties in the state of Mississippi for the year 2010?

```
#year 2010 does not exist, I consider 2015
dbGetQuery(db, paste("SELECT county,PLUS4College FROM Education LEFT JOIN FIPS ON  Education.fipsID = F
```

```
##          County PLUS4College
## 1     Issaquena          7.2
## 2        Greene          8.2
## 3         Perry          8.3
## 4  Tallahatchie          8.5
## 5        Benton         10.6
## 6     Chickasaw         10.7
## 7      Walthall         11.0
## 8       Calhoun         11.2
## 9         Scott         11.2
## 10   Tishomingo         11.2
## 11       Kemper         11.5
## 12        Leake         11.8
## 13      Noxubee         11.9
## 14     Prentiss         11.9
## 15       George         12.0
## 16       Amite         12.1
## 17   Humphreys         12.1
## 18    Lawrence         12.1
## 19      Holmes         12.3
## 20   Yalobusha         12.4
## 21     Simpson         12.7
## 22     Quitman         12.9
## 23      Tippah         12.9
## 24      Marion         13.0
## 25    Pontotoc         13.0
## 26       Smith         13.0
## 27    Itawamba         13.2
## 28       Yazoo         13.3
## 29      Jasper         13.4
## 30    Marshall         13.4
## 31       Stone         13.4
## 32     Choctaw         13.5
## 33     Neshoba         13.5
## 34     Carroll         13.6
## 35       Wayne         13.6
## 36       Union         13.8
## 37      Clarke         13.9
## 38       Pearl         13.9
## 39      Copiah         14.1
## 40   Jefferson         14.2
## 41   Sunflower         14.4
## 42   Wilkinson         14.5
## 43      Monroe         14.6
## 44      Attala         14.9
## 45      Panola         14.9
## 46     Lincoln         15.2
## 47   Covington         15.3
## 48   Claiborne         15.6
## 49     Grenada         15.6
## 50  Montgomery         15.8
## 51      Newton         15.8
## 52      Alcorn         16.2
## 53        Pike         16.3
## 54   Jefferson         16.5
## 55     Winston         16.5
## 56        Tate         16.7
## 57     Webster         17.2
## 58      Tunica         17.3
## 59     Leflore         17.5
## 60     Coahoma         17.6
## 61       Adams         17.8
## 62        Clay         17.8
## 63    Franklin         17.9
## 64       Jones         18.4
## 65  Lauderdale         18.7
## 66  Washington         18.8
## 67     Sharkey         20.0
## 68     Jackson         20.1
## 69 Mississippi         20.7
## 70      Hancock         21.0
## 71      Bolivar         21.1
## 72     Harrison         21.3
## 73          Lee         21.8
## 74      Lowndes         21.8
## 75       DeSoto         22.4
## 76       Warren         24.4
## 77      Forrest         26.7
## 78        Hinds         27.7
## 79       Rankin         29.0
## 80        Lamar         35.9
## 81    Lafayette         38.3
## 82    Oktibbeha         43.0
## 83      Madison         46.0
```

```
#Minimum percent is 7.2 for "Issaquena" county
```

- 2.7 Which state contains the most number of counties that have not been provided a rural urban code?

```
dbGetQuery(db, paste("SELECT state, COUNT(county)", "FROM Education LEFT JOIN FIPS ON  Education.fipsID
```

```
##    State COUNT(county)
## 1     AK             5
## 2     AL             5
## 3     AR             5
## 4     AZ             5
## 5     CA             5
## 6     CO             5
## 7     CT             5
## 8     DC             5
## 9     DE             5
## 10    FL             5
## 11    GA             5
## 12    HI             5
## 13    IA             5
## 14    ID             5
## 15    IL             5
## 16    IN             5
## 17    KS             5
## 18    KY             5
## 19    LA             5
## 20    MA             5
## 21    MD             5
## 22    ME             5
## 23    MI             5
## 24    MN             5
## 25    MO             5
## 26    MS             5
## 27    MT             5
## 28    NC             5
## 29    ND             5
## 30    NE             5
## 31    NH             5
## 32    NJ              5
## 33    NM              5
## 34    NV              5
## 35    NY              5
## 36    OH              5
## 37    OK              5
## 38    OR              5
## 39    PA              5
## 40    RI              5
## 41    SC              5
## 42    SD              5
## 43    TN              5
## 44    TX              5
## 45    UT              5
## 46    VA              5
## 47    VT              5
## 48    WA              5
## 49    WI              5
## 50    WV              5
## 51    WY              5
```

```
#In all states, there are 5 counties that have not been assigned a rural ID
```

- 2.8 In the year 2015, which fip counties, U.S. states contain a higher percentage of unemployed citizens than the percentage of college graduates? List the county name and the state name. Order the result alphabetically by state.

```
dbGetQuery(db, "SELECT county, state, PLUS4College, unemployedRate FROM Education
           LEFT JOIN Fips ON Education.fipsID = Fips.fipsID
           LEFT JOIN EMPLOYMENT
           ON Education.fipsID = EMPLOYMENT.fips
           AND Education.YEAR = EMPLOYMENT.year
           WHERE Education.YEAR='2015'
           AND unemployedRate>PLUS4College")
```

```
##        County State PLUS4College unemployedRate
## 1     Conecuh    AL          8.2            9.2
## 2      Greene    AL         10.9           11.0
## 3      Wilcox    AL         12.5           14.7
## 4      Bethel    AK         11.6           14.4
## 5    Kusilvak    AK          5.0           23.2
## 6   Northwest    AK         10.6           15.5
## 7       Yukon    AK         11.2           18.0
## 8      Apache    AZ         10.8           13.4
## 9        Yuma    AZ         14.4           21.8
## 10     Colusa    CA         14.6           15.3
## 11   Imperial    CA         14.1           24.0
## 12     Hendry    FL          9.8           10.3
## 13       Clay    GA          7.8           11.3
## 14      Macon    GA          7.9            8.9
## 15    Webster    GA          7.6            8.9
## 16    Wheeler    GA          5.6           10.7
## 17 Alexander    IL          8.0            8.6
## 18      Clay    KY          9.6            9.7
## 19   Elliott    KY          7.5           10.0
## 20       Lee    KY          7.9            8.5
## 21    Leslie    KY          8.6           10.8
## 22  McCreary    KY          7.0            8.3
## 23  Magoffin    KY          8.5           14.7
## 24    Martin    KY          6.5            9.6
## 25      East    LA          8.8           13.9
## 26      West    LA         11.1           13.3
## 27 Humphreys    MS         12.1           12.9
## 28 Issaquena    MS          7.2           16.9
## 29      Luna    NM         12.1           17.6
## 30   Tyrrell    NC          8.0            9.4
## 31    Monroe    OH          9.9           10.0
## 32  Marlboro    SC          8.5           10.1
## 33    Oglala    SD         11.4           11.6
## 34    Morgan    TN          6.4            7.6
## 35     Scott    TN          9.0            9.6
## 36     Duval    TX          8.1            8.2
## 37    Loving    TX          1.9            5.1
## 38    Newton    TX          6.4            7.5
## 39     Starr    TX          9.1           13.6
## 40   Willacy    TX          8.3           13.1
## 41    Zavala    TX          9.0           11.1
## 42  Buchanan    VA          9.9           10.8
## 43     Boone    WV          8.8            9.6
## 44   Calhoun    WV         10.4           12.5
## 45      Clay    WV          9.8           11.2
## 46   Lincoln    WV          9.5            9.7
## 47     Logan    WV          8.1           11.4
## 48  McDowell    WV          5.1           13.0
## 49     Mingo    WV         10.2           13.1
## 50     Roane    WV         11.3           11.5
## 51   Wyoming    WV          7.9            9.7
```

- 2.9 Return the county, U.S. state and year that contains the highest percentage of college graduates in this dataset?

```
dbGetQuery(db, paste("SELECT county,state,YEAR, MAX(PLUS4College)", "FROM Education LEFT JOIN FIPS ON
```

```
##   County State YEAR MAX(PLUS4College)
## 1  Falls    VA 2015              78.8
```

```
#MAX happens in Falls county, VA state in 2015
```

3. Compare your SQL SELECT statements to your dplyr statements written to answer the same questions. Do you have a preference between the two methods? State your reasons for your preference. (10 points)

```
#RSQLite is a database management system, But if you use Exploratory and/or modern R, most likely you a

# Based on my experience the total number of codes are more or less same. It is a little bit harder to
```

3.0 In the year 1970, what is the population percent that did not earn a high school diploma for the Nantucket county in Massachusetts? What about the year 2015?

```
# Percent not attaining a high school diploma in MA and Nantucket county in 1970 and 2015
#Filter works on the rows
#select works on the columns (variables)
#group_by gathers all the same parameters in column and make them ready for other analysis by summarize
a <- read.csv("FipsEducationsDA5020v2.csv")
b <- read.csv("FipsUnemploymentDA5020.csv")
#install.packages("stringr")
library(stringr)
#install.packages("tidyr")
library(tidyr)
library(dplyr)

#every measurement for a year and fips is reapeted 4 times which is not good, so we use spread function
a.new <- a%>%
  spread(key=percent_measure,value=percent)

#Seperating the state and counties
a.sep <- a.new %>%
  separate(county_state, into = c("state","county"))
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 15721 rows
## [6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24,
## 25, ...].
```

```
filter(a.sep, state=="MA",county=="Nantucket",year=="1970") %>%
  select(`percent_less than_hs_diploma`) %>%
head() ##33.7%
```

```
##   percent_less than_hs_diploma
## 1                         33.7
```

```
filter(a.sep, state=="MA",county=="Nantucket",year=="2015") %>%
  select(`percent_less than_hs_diploma`) %>%
head() #5.2%
```

```
##   percent_less than_hs_diploma
## 1                          5.2
```

- 3.1 What is the average population percentage that did not earn a high school diploma for the counties in Alabama for the year 2015?

```
s<- filter (a.sep, state=="AL",year== "2015") %>%
  select(`percent_less than_hs_diploma`)

head(mean(s$`percent_less than_hs_diploma`))
```

```
## [1] 19.75882
```

- 3.2 What is the average percentage of college graduates for the counties in the state of Massachusetts for the year 2015?

```
x<- filter (a.sep, state=="MA",year== "2015") %>%
  select(percent_four_plus_years_college)

head(mean(x$percent_four_plus_years_college))
```

```
## [1] 38.52667
```

- 3.3 Determine the average percentage of the population that did not earn a high school diploma for the counties in Alabama for each year within the dataset. The result should return the calendar year and the average percentage drop out rate for that year.

```
filter (a.sep, state=="AL") %>%
  select(year,`percent_less than_hs_diploma`) %>%
  group_by(year) %>%
  summarise(avg.not.hs.diploma=mean(`percent_less than_hs_diploma`)) %>%
  head()
```

```
## # A tibble: 5 x 2
##    year avg.not.hs.diploma
##   <int>              <dbl>
## 1  1970               65.2
## 2  1980               50.6
## 3  1990               40.1
## 4  2000               30.3
## 5  2015               19.8
```

- 3.4 What is the most common rural_urban code for the U.S. counties?

```
a.sep %>%
count(rural_urban_cont_code) %>%
arrange(desc(n))
```

```
## # A tibble: 10 x 2
##    rural_urban_cont_code     n
##    <fct>                 <int>
##  1 6                      2961
##  2 7                      2165
##  3 1                      2153
##  4 9                      2091
##  5 2                      1890
##  6 3                      1779
##  7 8                      1097
##  8 4                      1070
##  9 5                       460
## 10 NULL                    255
```

- 3.5 Which counties have not been coded with a rural urban code? Return a result that contains two fields: County, State for the counties that has not been assigned a rural urban code. Do not return duplicate values in the result. Order the result alphabetically by state.

```
#whenever the name of county is exactly the name of state, rural urban code is NULL. for 5 years it has
q <- a.sep %>%
filter (rural_urban_cont_code=="NULL")%>%
select(state,county,rural_urban_cont_code) %>%
group_by(state,county,rural_urban_cont_code) %>%
  summarise()
q <- q[order(q$state),]#making in alphabetical order
```

- 3.6 What is the minimal percentage of college graduates for the counties in the state of Mississippi for the year 2010?

```
#There is no data for year 2010, I calculate it for 2015
a.sep %>%
filter (state=="MS",year== "2015") %>%
  select(county,percent_four_plus_years_college) %>%
  arrange(desc(percent_four_plus_years_college)) %>%
  tail()
```

```
##          county percent_four_plus_years_college
## 78    Chickasaw                            10.7
## 79       Benton                            10.6
## 80 Tallahatchie                             8.5
## 81        Perry                             8.3
## 82       Greene                             8.2
## 83    Issaquena                             7.2
```

```
a.sep %>%
filter (state=="MS",year== "2015") %>%
  select(county,percent_four_plus_years_college) %>%
  summarise(min(percent_four_plus_years_college))
```

```
##   min(percent_four_plus_years_college)
## 1                                  7.2
```

```
#the mimimum percentage belongs to Issaquena which is 7.2 %
```

- 2.7 Which state contains the most number of counties that have not been provided a rural urban code?

```
v <- b %>%
  filter(year=="2015")
mean(v$percent_unemployed) #average is 5.528102
```

```
## [1] 5.528102
```

```
d <- select(a.sep, fips,county,state)
fips <- as_data_frame(d)%>%
  group_by(fips,county,state)  %>%
  summarize()
z <- inner_join(v,fips, by="fips")
desc.2015 <- z%>%
filter(percent_unemployed>5.528102) %>%
  arrange(desc(percent_unemployed)) %>%
  select(state,county,percent_unemployed)
```

- 3.8 In the year 2015, which fip counties, U.S. states contain a higher percentage of unemployed citizens than the percentage of college graduates? List the county name and the state name. Order the result alphabetically by state.

```
n <- filter(a.sep,year=="2015")
m <- filter(b,year=="2015") %>%
  select(fips,percent_unemployed)
l<- merge(n,m,by="fips")
k <- l %>%
  filter(percent_unemployed>percent_four_plus_years_college) %>%
  select(state,county,percent_unemployed,percent_four_plus_years_college)

k <- k[order(k$state),]#making in alphabetical order
```

- 3.9 Return the county, U.S. state and year that contains the highest percentage of college graduates in this dataset?

```
a.sep %>%
  select(county,year,state,percent_four_plus_years_college) %>%
  arrange(desc(percent_four_plus_years_college)) %>%
  head()
```

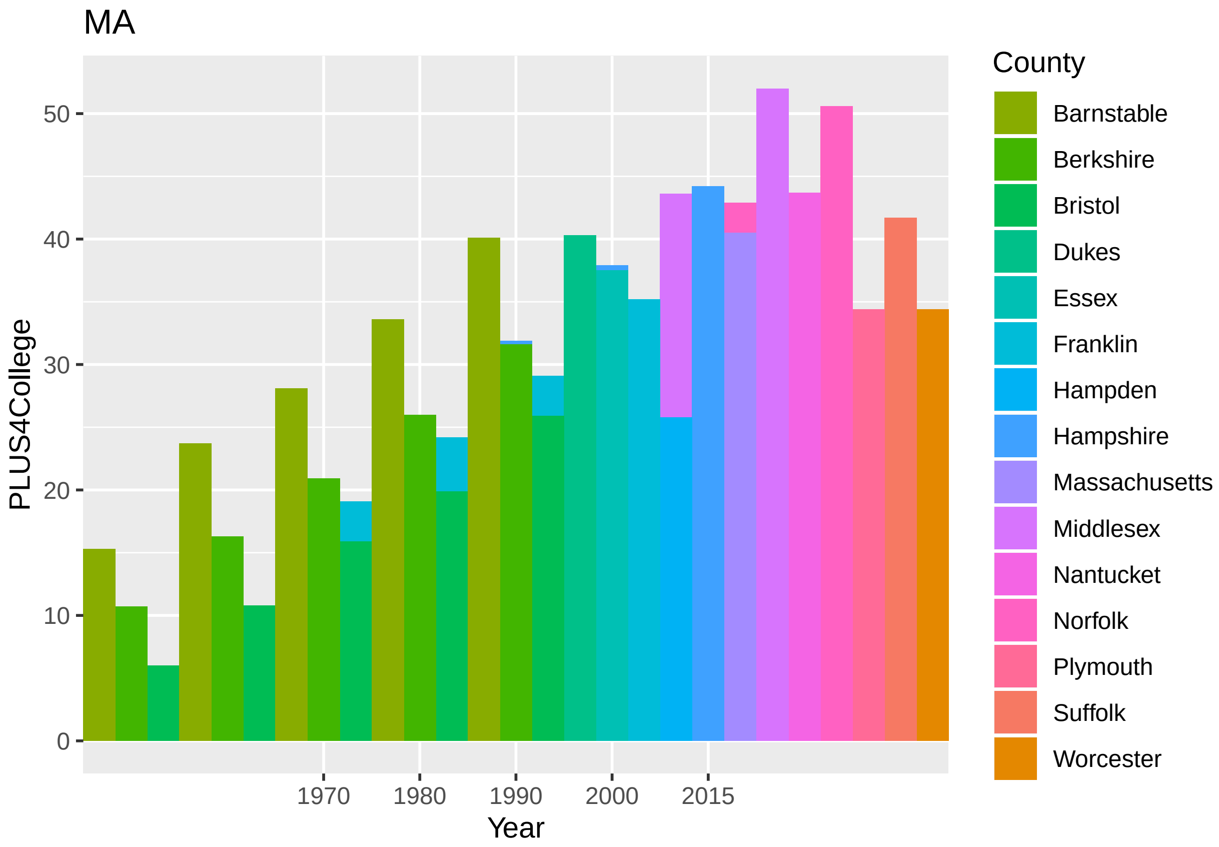
```
##       county year state percent_four_plus_years_college
## 1      Falls 2015    VA                            78.8
## 2  Arlington 2015    VA                            72.9
## 3        Los 2015    NM                            64.2
## 4      Falls 2000    VA                            63.7
## 5 Alexandria 2015    VA                            61.4
## 6     Howard 2015    MD                            60.6
```

```
# The highest percentage goes to county "Falls" and state "VA" in 2015
```

4. Write a R function named get_state_county_education_data_dplyr(edf, state), it accepts a data frame containing education data and a state's abbreviation for arguments and produces a chart that shows the change in education across time for each county in that state. Use dplyr to extract the data. Write a few R statements that call the function with different state values. (5 points)

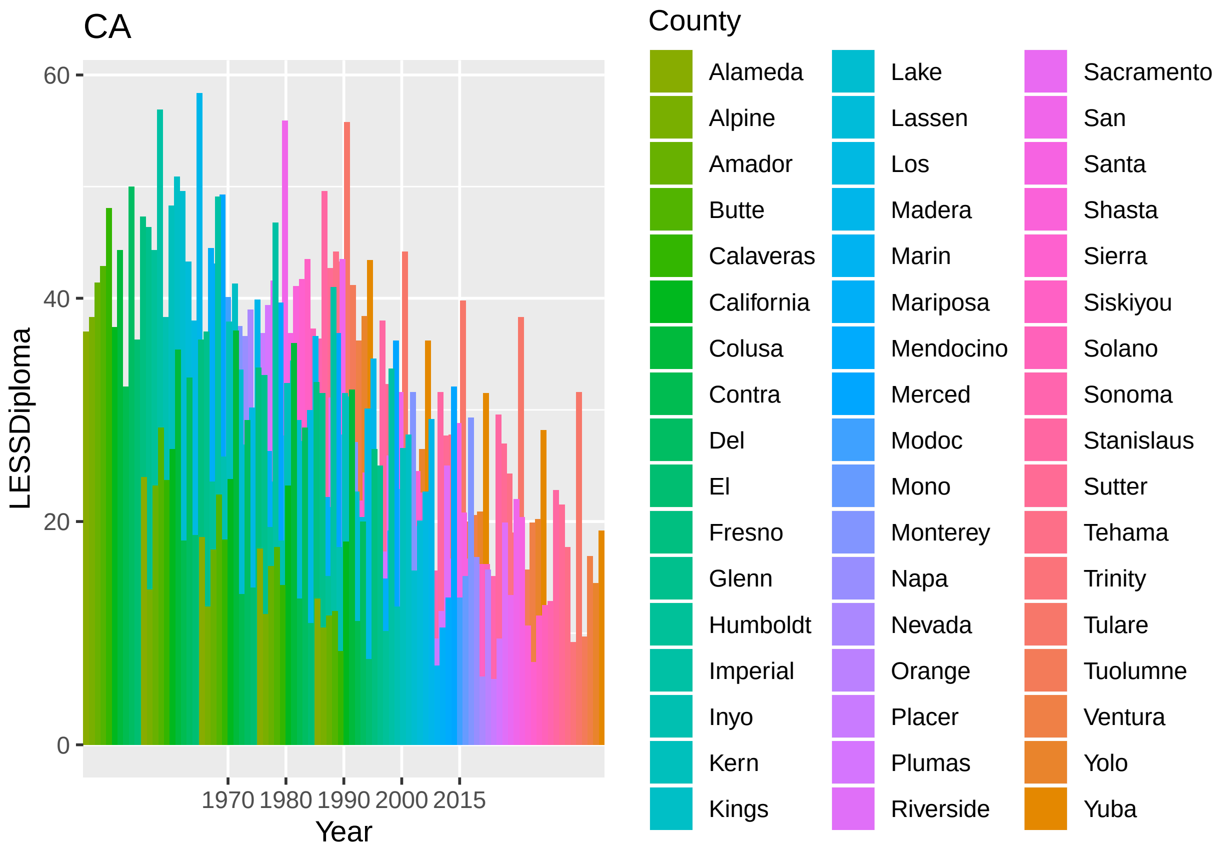
```
library("ggplot2")
get_state_county_education_data_dplyr <- function(EDF, STATEID, EL){
  stateEDUData <- filter(EDF, State==STATEID)

  ggplot(stateEDUData, aes(fill=County, y=stateEDUData[,EL], x=YEAR)) +
  geom_bar(width=5, position="dodge", stat="identity", show.legend = T) +
  scale_fill_hue(h = c(100, 400)) +
  xlab('Year') + ylab(EL) + ggtitle(STATEID)
}
EDUCATION <- left_join(Education.DF,FIPS.DF)
#Joining 2 dataframes
get_state_county_education_data_dplyr(EDUCATION, "MA", "PLUS4College")
```

```
## Warning: position_dodge requires non-overlapping x intervals
```



```
#The number of people with the 4 year college degree is increasing from 1970 to 2015 in almost all coun
get_state_county_education_data_dplyr(EDUCATION, "CA", "LESSDiploma")
```

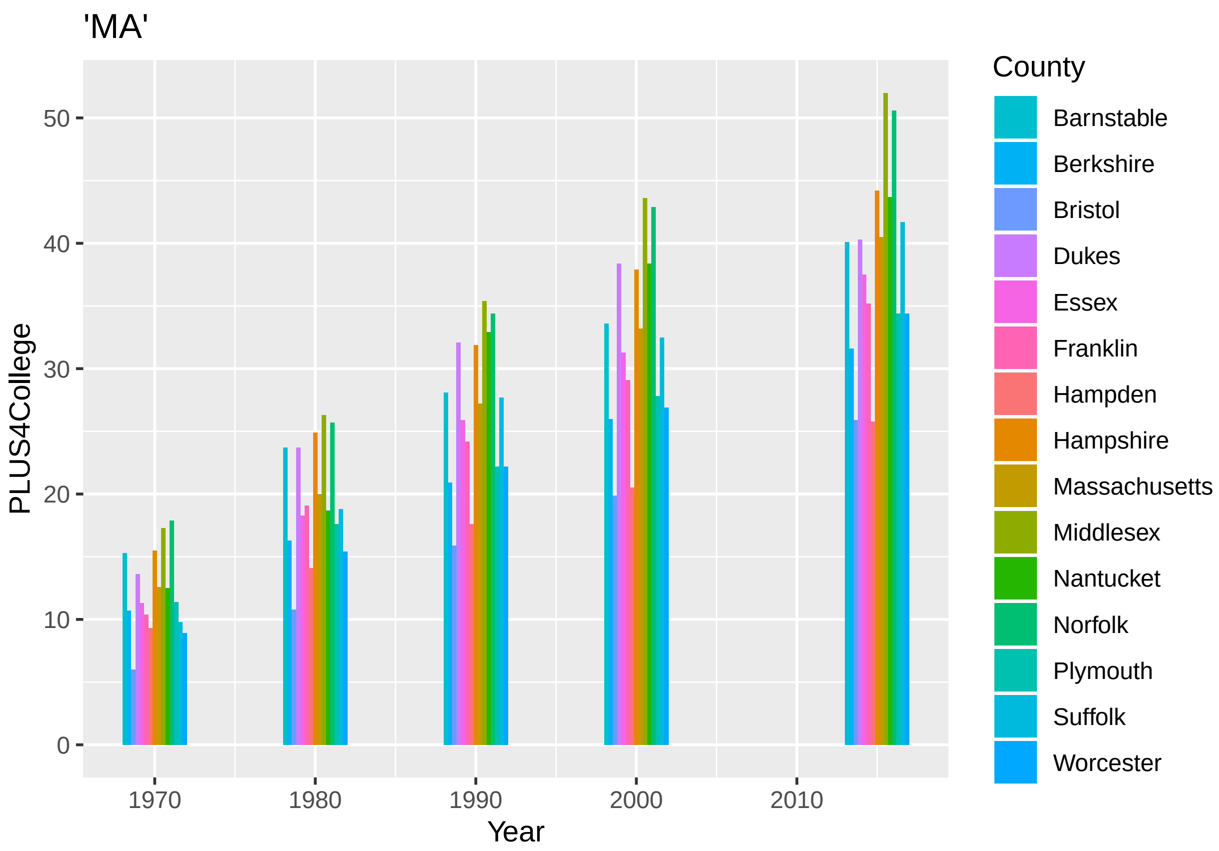
```
## Warning: position_dodge requires non-overlapping x intervals
```



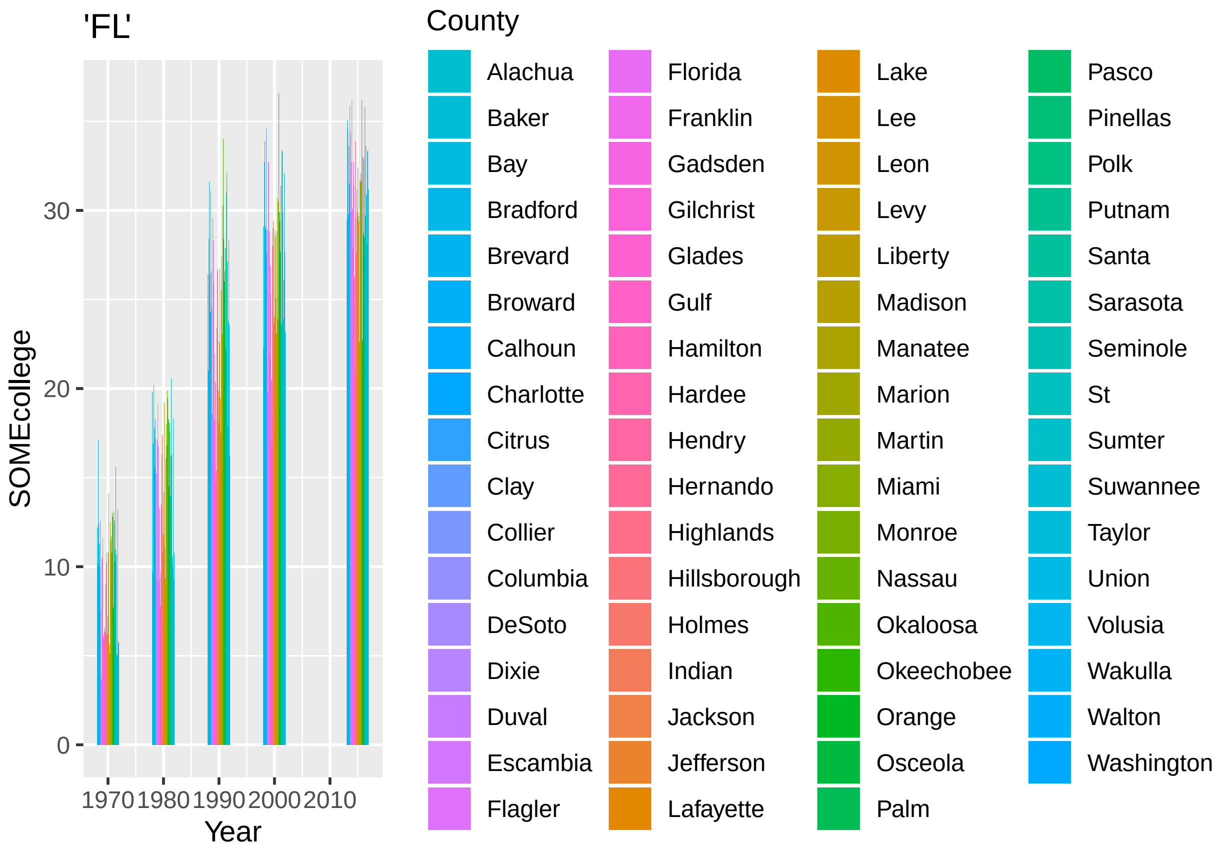
```
# we see that the number of students with less diploma degree is decreasing from 1970 to 2015 in Califo
```

5. Write a R function named get_state_county_education_data_sql(edSQL, state), it accepts a SQL database connection containing education data and a state's abbreviation for arguments and produces a chart that shows the change in education across time for each county in that state. Use SQL SELECT to extract the data from the database. Write a few R statements that call the function with different state values. (10 points)

```
get_state_county_education_data_sql <- function(edSQL, STATEID, EL){
 Db.conn <- dbGetQuery(edSQL, paste("SELECT county, state, year, ", EL,
           " FROM Education LEFT JOIN FIPS ON  Education.fipsID = FIPS.fipsID WHERE state=", STATEID, s

  ggplot(Db.conn, aes(fill=County, y=Db.conn[,EL], x=YEAR)) +
  geom_bar(width=4, position="dodge", stat="identity", show.legend = T) +
  scale_fill_hue(h = c(200, 600)) +
  xlab('Year') + ylab(EL) + ggtitle(STATEID)
}

get_state_county_education_data_sql(db, "'MA'", "PLUS4College")
```



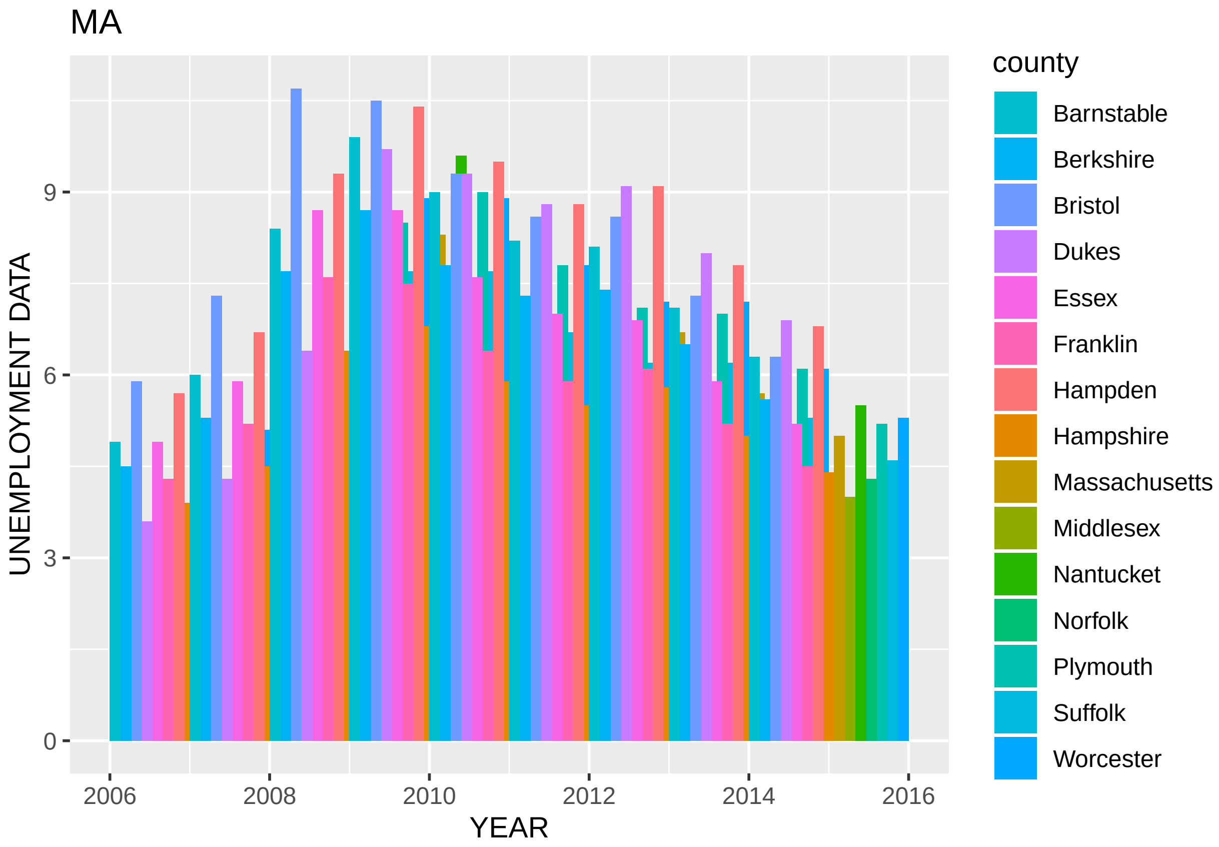
```
get_state_county_education_data_sql(db, "'FL'", 'SOMEcollege')
```



6. Write a R function named get_state_county_unemployment_data_dplyr(udf, state), it accepts a data frame containing unemployment data and state's abbreviation and produces a chart that shows the change in unemployment across time for each county in that state. Use dplyr to extract the data. Write a few R statements that call the function with different state values. (5 points)

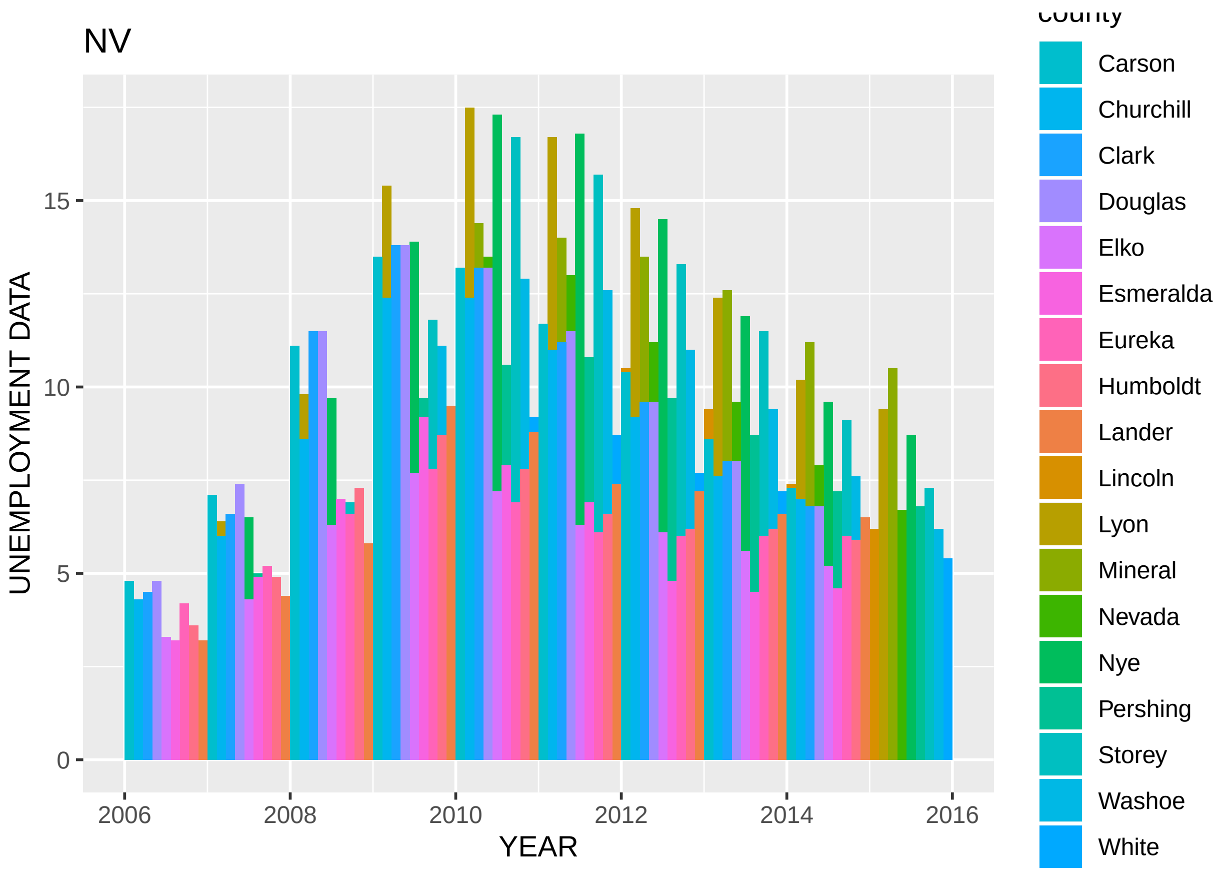
```
UNEM.ST <- left_join(EMPLOY.DF, fips, by='fips')

get_state_county_unemployment_data_dplyr <- function(udf, STATEID){
  stateEMPData <- filter(udf, state==STATEID)

  ggplot(stateEMPData, aes(fill=county, y=unemployedRate, x=year)) +
  geom_bar(width=2, position="dodge", stat="identity", show.legend = T) +
  scale_fill_hue(h = c(200, 600)) +
  xlab("YEAR") + ylab("UNEMPLOYMENT DATA") + ggtitle(STATEID)
}

get_state_county_unemployment_data_dplyr(UNEM.ST, "MA")
```

```
## Warning: position_dodge requires non-overlapping x intervals
```



```r
#unemployment rate is first increasing and then decreasing as the recession is passed

get_state_county_unemployment_data_dplyr(UNEM.ST, "NV")
```

```
## Warning: position_dodge requires non-overlapping x intervals
```

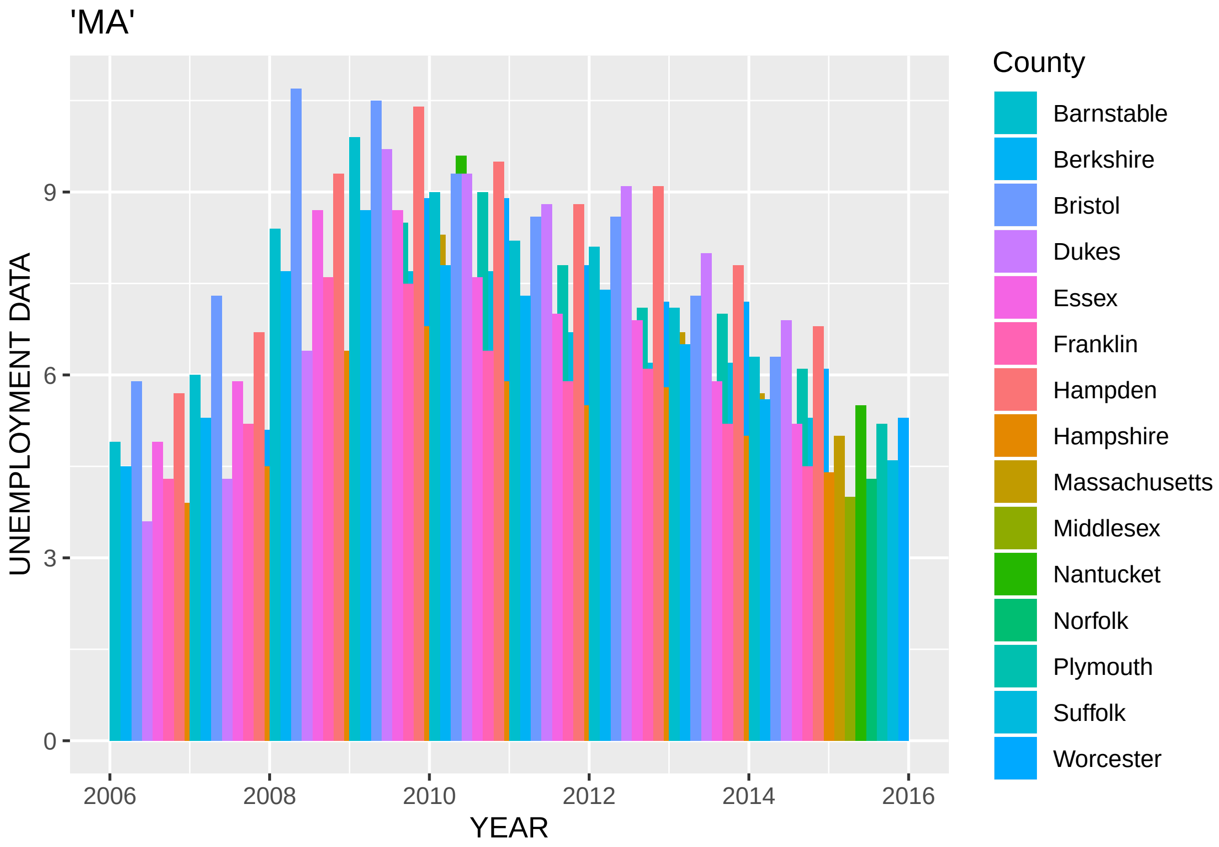


```
#In nevada, the pattern is the same
```

7. Write a R function named get_state_county_unemployment_data_sql(udfSQL, state), it accepts a SQL database oject containing unemployment data and state's abbreviation and produces a chart that shows the change in education across time for each county in that state. Use SQL SELECT to extract the data. Write a few R statements that call the function with different state values. (10 points)

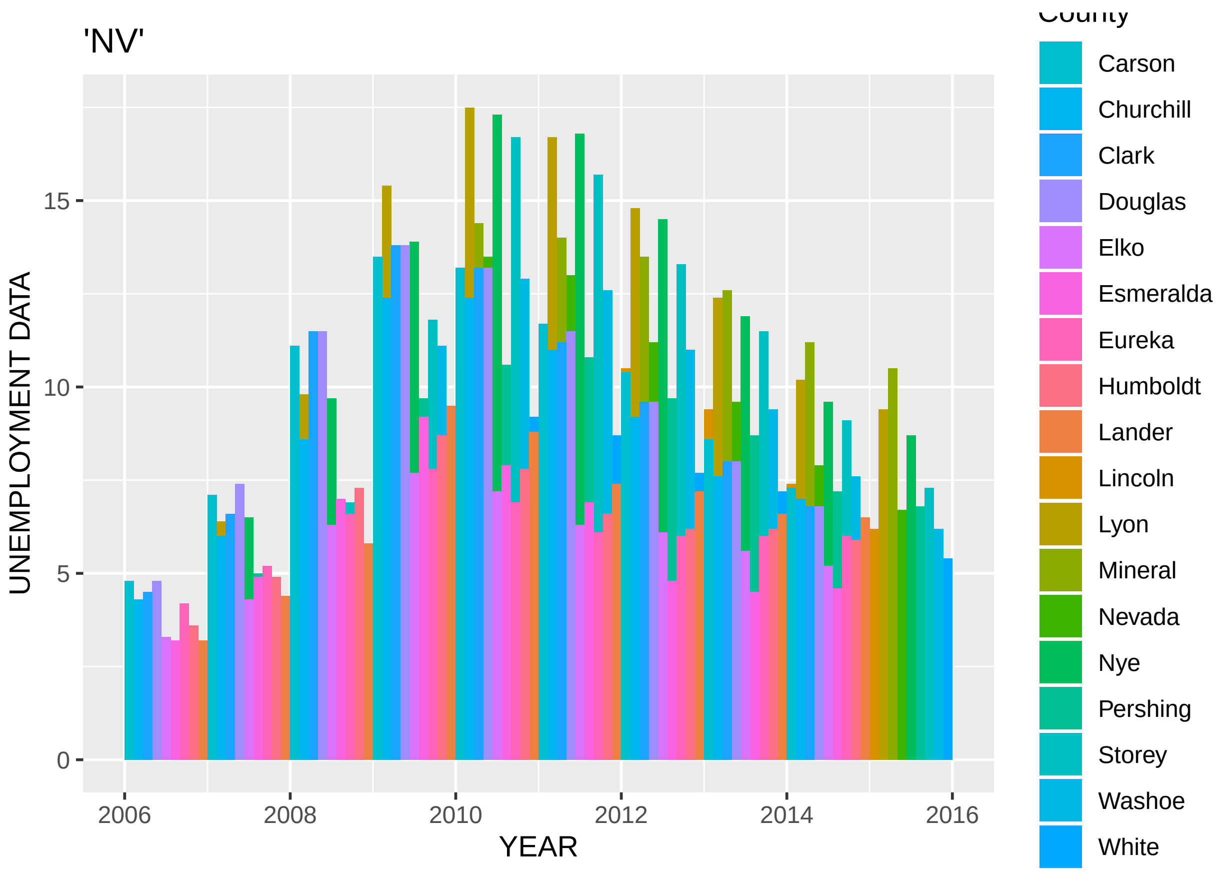
```
get_state_county_unemployment_data_sql <- function(udfSQL, STATEID){
  Db.conn <- dbGetQuery(udfSQL, paste("SELECT county, state, year, unemployedRate",
           " FROM EMPLOYMENT LEFT JOIN FIPS ON EMPLOYMENT.fips = FIPS.fipsID
           WHERE state=", STATEID, sep=""))

  ggplot(Db.conn, aes(fill=County, y=unemployedRate, x=year)) +
  geom_bar(width=2, position="dodge", stat="identity", show.legend = T) +
  scale_fill_hue(h = c(200, 600)) +#the range of colors
  xlab('YEAR') + ylab("UNEMPLOYMENT DATA") + ggtitle(STATEID)
}

get_state_county_unemployment_data_sql(db, "'MA'")#Plot the data in Massachussets
```

```
## Warning: position_dodge requires non-overlapping x intervals
```



```r
get_state_county_unemployment_data_sql(db, "'NV'")#Plot the data in Nevada
```

```
## Warning: position_dodge requires non-overlapping x intervals
```



Submission

You need to submit an .Rmd extension file as well as the generated pdf file. Be sure to state all the assumptions and give explanations as comments in the .Rmd file wherever needed to help us assess your submission. Please name the submission file LAST_FirstInitial_1.Rmd for example for John Smith's 1st assignment, the file should be named Smith_J_1.Rmd.