Data 607 - Week 12 Assignment

Adam Douglas

11/19/2018

Relational Database

Our data comes courtesy of the `nycflights13` database and is a large recordset with all the flights arriving and departing from NYC airports in 2013.

The data has been preloaded in a relational database (PostgreSQL) and our goal is to ferry the data from the relational database to a NoSQL database (MongoDB). First we set up our connection to PostgreSQL:

```r
# Set up our connection to PostgreSQL
host <- "localhost"
usr <- "postgres"
pass <- getOption("pass")
port <- 5432
database <- "nycflights"
```

Then we get the data and load it into a data frame:

```r
# Get all data from the flights table
conn <- dbConnect(RPostgreSQL::PostgreSQL(),
                  host = host, dbname = database,
                  user = usr, password = pass)

flights <- dbGetQuery(conn, "select * from flights")
```

Finally, we close our connection:

```r
dbDisconnect(conn)
```

NoSQL Database

To get our data into MongoDB, we use a similar methodology as above with PostgreSQL by creating a connection object. For this, we use the `mongolite` package:

```r
# Open the connection
mon <- mongo(collection="flights", db="Data607")
```

Now we can easily load the data using that connection object:

```r
mon$insert(flights)
```

```
## List of 5
##  $ nInserted  : num 336776
##  $ nMatched   : num 0
##  $ nRemoved   : num 0
##  $ nUpserted  : num 0
##  $ writeErrors: list()
```

Our load went nice and smoothly, with all 336776 records being loaded.

Some Analysis in MongoDB

Now that we loaded the data into MongoDB, we can query collection we created via the same connection object we used to load the documents. As an example, let's display one document:

```
rec <- data.frame(mon$iterate()$one())
rec
```

```
##   row_names year month day dep_time sched_dep_time dep_delay arr_time
## 1         1 2013     1   1      517            515         2      830
##   sched_arr_time arr_delay carrier flight tailnum origin dest air_time
## 1            819        11      UA   1545  N14228    EWR  IAH      227
##   distance hour minute           time_hour
## 1     1400    5     15 2013-01-01 05:00:00
```

We can query the collection in a manner similar to a relational database, though the syntax is quite different.

Here, we count the total flights by month:

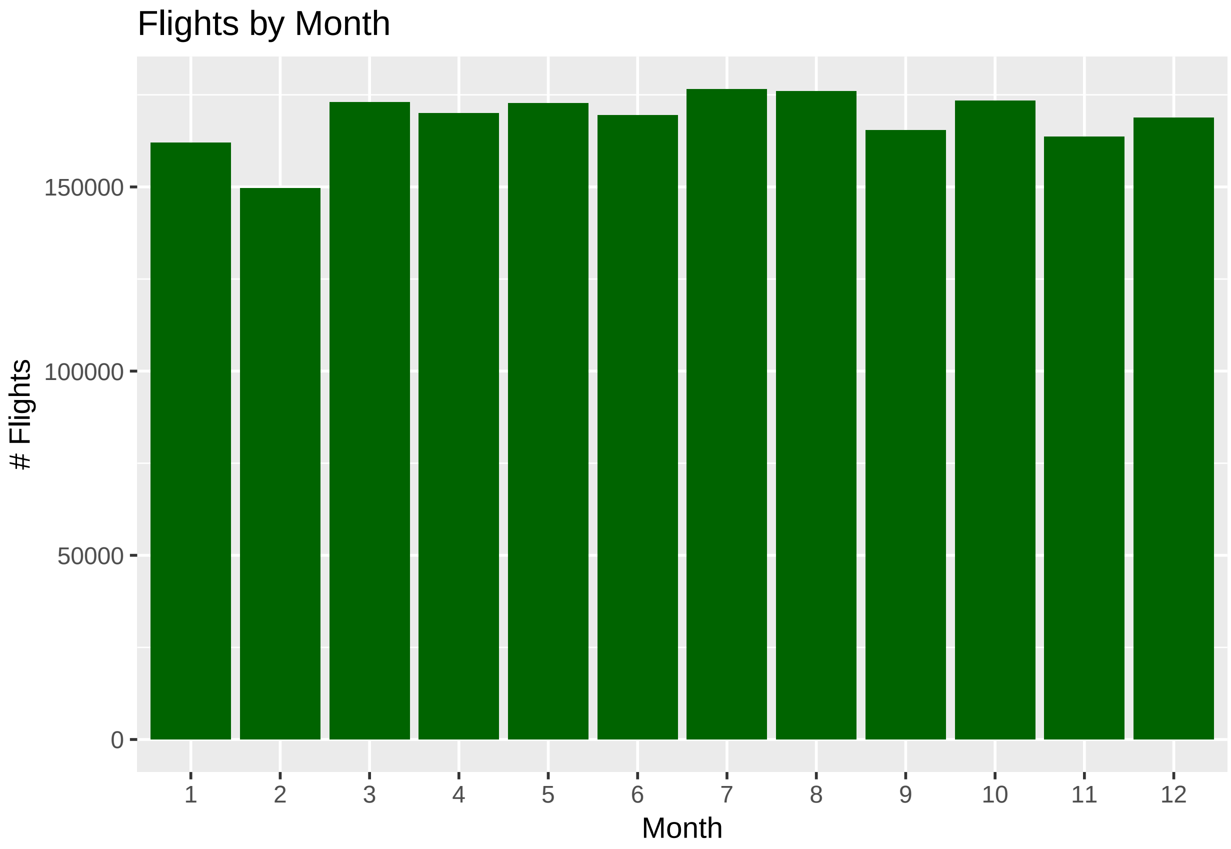
```
byMonth <-
  mon$aggregate('[
                  {
                    "$group":
                    {
                      "_id": "$month",
                      "num_flights":
                      {
                        "$sum": 1
                      }
                    }
                  },
                  {
                    "$sort":
                      {
                        "_id": 1
                      }
                  },
                  {
                    "$project":
                    {
                      "_id": 0,
                      "num_flights": 1,
                      "month": "$_id"
                    }
                  }
              ]')

head(byMonth)
```

```
##   num_flights month
## 1      162024     1
## 2      149706     2
## 3      173004     3
## 4      169980     4
## 5      172776     5
## 6      169458     6
```

And then we can use R to display the data if we choose:

```
byMonth %>% ggplot(aes(x=as.factor(month), y=num_flights)) +
  geom_col(fill="darkgreen") + ggtitle("Flights by Month") + ylab("# Flights") +
  xlab("Month")
```



Finally, we ensure we close our connections:

```
mon$disconnect()
```