Project Notebook

Keith Engwall

1/23/2018

Project Objectives

Create a predictive model for identifying patients with Diabetes

See Capstone Proposal for details.

Project Dataset

Practice Fusion De-Identified Data Set containing EHR data for approximately 10,000 de-identified patients, including data points for diagnoses, medication, transcript data, and lab observations. See the Data Dictionary for details.

General Notes

Working with SQLite

The dataset is contained within an SQLite database file (420.7MB). To load the data into R requires installation and loading of **RSQLite** and **DBI** R packages. One of the dependencies is the tibble package. During the install, I was asked whether to install the binary version (1.3.4) or the source version (1.4.1), which would need compilation. I wasn't comfortable enough to explore compiling R package code yet, so I went with the 1.3.4 version.

I found this brief example of how to connect to and query an SQLite database file to be very helpful in getting up and going quickly.

```
#Connect to SQLite file
con <- dbConnect(SQLite(), dbname="data.db")

#Define query and store it in my_query
my_query <- dbSendQuery(con, "SELECT name from person_table")
#Fetch data using query and store it in my_data
my_data <- dbFetch(my_query)

#Clear the results cache from my_query
dbClearResult(my_query)

#Perform additional queries

#Disconnect from the database file
dbDisconnect(con)
```

dplyr also has sqlite functions

Project Notes

R Packages

The following packages are needed to work on the project.

Note that dbplyr is required to make database connections using dplyr functions.

```
library(tidyr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(dbplyr)
```

```
##
## Attaching package: 'dbplyr'
```

```
## The following objects are masked from 'package:dplyr':
##
##     ident, sql
```

Connect to database

Note that dbplyr library is required in order for this command to work

```
# Connect to database
my_db <- src_sqlite("compData.db")
```

Load relevant tables

The tables of interest are:

- *training_diagnosis* contains diagnosis information (we need this in order to identify the patients diagnosed with diabetes)
- *training_allergy* contains allergy information for patient
- *training_medication* contains medication information for patient
- *training_patient* contains gender and year of birth
- *training_smoke* combines patient info with smoking statuses
- *lab tables* contain lab results
  - *training_labResult* contains a record of lab results for a particular transcript
  - *training_labPanel* links training_labResult and training_labObservation
  - *training_labObservation* contains details regarding lab tests

```
diagnosis_tbl <- tbl(my_db, sql("SELECT DiagnosisGuid, PatientGuid, ICD9Code, StartYear, StopYear, Acut
allergy_tbl <- tbl(my_db, sql("SELECT AllergyGuid, PatientGuid, AllergyType, StartYear as AllergyStartY
medication_tbl <- tbl(my_db, sql("SELECT MedicationGuid, PatientGuid, MedicationNdcCode, MedicationName
patient_tbl <- tbl(my_db, sql("SELECT PatientGuid, Gender, YearOfBirth FROM training_patient"))
```

```
transcript_tbl <- tbl(my_db, sql("SELECT TranscriptGuid, PatientGuid, VisitYear, Height, Weight, BMI, Sy
smoke_tbl <- tbl(my_db, sql("SELECT PatientGuid, SmokeEffectiveYear, SmokingStatus_Description, Smoking
# to get a single table that links the observation data back to the patient,
# join training_labResult, training_labPanel, and training labObservation
lab_tbl <- left_join(left_join(tbl(my_db,sql("SELECT LabResultGuid, PatientGuid FROM training_labResult
```

Identify target population

All Type 1 Diabetes diagnoses have an ICD9Code that starts with 205. We have a table with all of the diagnoses and one with only the diabetes diagnoses.

```
diabetes_tbl <- tbl(my_db, sql("SELECT DiagnosisGuid, PatientGuid, ICD9Code, StartYear, StopYear, Acute
```

```
## Observations: ??
```

```
## Warning in rsqlite_fetch(res@ptr, n = n): Column `StopYear`: mixed type,
## first seen values of type string, coercing other values of type integer
```

```
## Variables: 6
## $ DiagnosisGuid <chr> "AC80E20C-EEF2-4AA4-B9C8-00661EF88886", "43EA46E...
## $ PatientGuid   <chr> "D76FC581-580A-4988-B13F-D4E9BD6763EA", "F6E5C45...
## $ ICD9Code      <chr> "250.71", "250.61", "250.03", "250.61", "250.03"...
## $ StartYear     <int> 0, 2010, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2011, 0, ...
## $ StopYear      <chr> "NULL", "NULL", "NULL", "NULL", "NULL", "NULL", ...
## $ Acute         <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
```

Identify allergy data for target population

```
# join allergy_tbl to diabetes_tbl to get allergy data for diabetic patients
diabetes_allergy_tbl <- as_tibble(left_join(diabetes_tbl, allergy_tbl, by = c("PatientGuid"))) %>%
  select(PatientGuid, AllergyGuid, AllergyType, ReactionName, SeverityName, AllergyMedicationNdcCode = N

# add has_allergy to diabetes_allergy_tbl to indicate whether diabetic patient has allergies
diabetes_allergy_tbl <- diabetes_allergy_tbl %>%
  mutate(has_allergy = as.integer(!is.na(diabetes_allergy_tbl$AllergyGuid)))

glimpse(diabetes_allergy_tbl)
```

```
## Observations: 219
## Variables: 7
## $ PatientGuid              <chr> "D76FC581-580A-4988-B13F-D4E9BD6763EA...
## $ AllergyGuid              <chr> NA, NA, NA, NA, NA, "2CDB164D-4AB8-4B...
## $ AllergyType              <chr> NA, NA, NA, NA, NA, "Medication", "Me...
## $ ReactionName             <chr> NA, NA, NA, NA, NA, "Shortness of bre...
## $ SeverityName             <chr> NA, NA, NA, NA, NA, "Modest", "Mild",...
## $ AllergyMedicationNdcCode <chr> NA, NA, NA, NA, NA, "68462033905", "6...
## $ has_allergy              <int> 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0...
```

Identify medication data for target population

Exclude medication specifically given in response to diabetes diagnosis (?)

```
# join medication_tbl to diabetes_tbl to get medication data for diabetic patients
# filter out medication that is linked to the diabetes diagnosis
diabetes_medication_tbl <- as_tibble(left_join(diabetes_tbl, medication_tbl, by = c("PatientGuid"))) %>%
  filter(DiagnosisGuid.x != DiagnosisGuid.y) %>%
  select(PatientGuid, MedicationGuid, MedicationNdcCode)

# add has_meds to diabetes_medication_tbl to indicate whether diabetic patient has medication
diabetes_medication_tbl <- diabetes_medication_tbl %>%
  mutate(has_meds = as.integer(!is.na(diabetes_medication_tbl$MedicationGuid)))

glimpse(diabetes_medication_tbl)
```

```
## Observations: 1,099
## Variables: 4
## $ PatientGuid       <chr> "D76FC581-580A-4988-B13F-D4E9BD6763EA", "D76...
## $ MedicationGuid    <chr> "18D1EC29-6EB4-48C0-AF06-95E7F74D1270", "2D0...
## $ MedicationNdcCode <chr> "49483022110", "00247200674", "38245067973",...
## $ has_meds          <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
```

Identify patient information for target population

```
# join patient_tbl to diabetes_tbl to get gender & age data for diabetic patients
diabetes_patient_tbl <- as_tibble(left_join(diabetes_tbl, patient_tbl, by = c("PatientGuid"))) %>%
  select(PatientGuid, Gender, YearOfBirth) %>% glimpse()
```

```
## Observations: 188
## Variables: 3
## $ PatientGuid <chr> "D76FC581-580A-4988-B13F-D4E9BD6763EA", "F6E5C45F-...
## $ Gender      <chr> "M", "F", "F", "M", "F", "F", "F", "M", "F", "M", ...
## $ YearOfBirth <int> 1927, 1960, 1959, 1947, 1968, 1934, 1987, 1949, 19...
```

Identify smoking information for target population

Need to parse through results to identify smokers, former smokers, etc. How should these be grouped?

```
# join smoke_tbl to diabetes_tbl to get smoking data for diabetic patients
diabetes_smoke_tbl <- as_tibble(left_join(diabetes_tbl, smoke_tbl, by = c("PatientGuid"))) %>%
  select(PatientGuid, SmokeEffectiveYear, SmokingStatus_Description, SmokingStatus_NISTCode) %>%
  arrange(SmokingStatus_NISTCode) %>%
  glimpse()
```

```
## Observations: 191
## Variables: 4
## $ PatientGuid               <chr> "83706824-EF87-4916-ACEE-4727FF3A1C1...
## $ SmokeEffectiveYear        <int> 2011, 2012, 2012, 2011, 2011, 2011, ...
## $ SmokingStatus_Description <chr> "Not a current tobacco user", "Not a...
## $ SmokingStatus_NISTCode    <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
```

Identify lab information for target population

```
# join lab_tbl to diabetes_tbl to get lab results data for diabetic patients
diabetes_lab_tbl <- as_tibble(left_join(diabetes_tbl, lab_tbl, by = c("PatientGuid"))) %>%
```

```
  select(PatientGuid, LabObservationGuid, HL7Identifier, HL7Text, HL7CodingSystem, ObservationValue, Uni

# add has_labs to diabetes_lab_tbl to indicate whether diabetic patient has lab results
diabetes_lab_tbl <- diabetes_lab_tbl %>%
  mutate(has_labs = as.integer(!is.na(diabetes_lab_tbl$LabObservationGuid)))

glimpse(diabetes_lab_tbl)
```

```
## Observations: 312
## Variables: 12
## $ PatientGuid        <chr> "D76FC581-580A-4988-B13F-D4E9BD6763EA", "F6...
## $ LabObservationGuid <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ HL7Identifier      <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ HL7Text            <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ HL7CodingSystem    <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ ObservationValue   <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ Units              <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ ReferenceRange     <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ AbnormalFlags      <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ ResultStatus       <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ ObservationYear    <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ has_labs           <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
```

disconnect from database

This is getting an error. May have to use a different method to connect

```
dbDisconnect(my_db)
```