Access Wideband Audiology Immitance database using R and dplyr (Voss PI)

Nicholas Horton (nhorton@amherst.edu)

June 23, 2016

Introduction

This document is intended to describe how to access data from a MySQL database using R. It utilizes a database of wideband acoustic immitance variables from humans with normal hearing (see https://projectreporter.nih.gov/project_info_description.cfm?aid=8769352&icde=30039221&ddparam=&ddvalue=&ddsub=&cr=10&csb=default&cs=ASC for more details).

A relevant paper on the topic of data management and databases in R can be found at http://chance.amstat.org/2015/04/setting-the-stage.

Accessing data from a database using SQL commands

First I will demonstrate how to access data using SQL (structured query language) commands and the `dbGetQuery()` function. First a connection to the database is set up.

```
library(mosaic)
library(RMySQL)
```

```
## Loading required package: DBI
```

```
con <- dbConnect(MySQL(), host = "scidb.smith.edu",
                 user = "waiuser", password = "smith_waiDB",
                 dbname = "wai")
```

Next a series of SQL queries can be sent to the database. These return R dataframes.

```
dbGetQuery(con, "SHOW TABLES")
```

```
##   Tables_in_wai
## 1  Measurements
## 2       PI_Info
## 3       Subject
```

```
dbGetQuery(con, "EXPLAIN PI_Info")
```

```
##          Field          Type Null Key Default Extra
## 1   Identifier   varchar(20)  YES        <NA>
## 2      PI_Year       int(11)  YES        <NA>
## 3           PI   varchar(40)  YES        <NA>
## 4  Affiliation  varchar(500)  YES        <NA>
## 5        Email   varchar(30)  YES        <NA>
## 6        Title  varchar(140)  YES        <NA>
## 7          Pub   varchar(30)  YES        <NA>
## 8         Date      char(20)  YES        <NA>
## 9          URL  varchar(140)  YES        <NA>
## 10    PI_Notes varchar(1500)  YES        <NA>
```

```
ds <- dbGetQuery(con, "SELECT * from Measurements LIMIT 10")
ds
```

```
##    Identifier Sub_Number Session Left_Ear MEP Instrument  Ear_Area    Freq
## 1  Voss_ASA14          1       1        0   0          1 0.0000442 210.938
## 2  Voss_ASA14          1       1        0   0          1 0.0000442 234.375
## 3  Voss_ASA14          1       1        0   0          1 0.0000442 257.812
## 4  Voss_ASA14          1       1        0   0          1 0.0000442 281.250
## 5  Voss_ASA14          1       1        0   0          1 0.0000442 304.688
## 6  Voss_ASA14          1       1        0   0          1 0.0000442 328.125
## 7  Voss_ASA14          1       1        0   0          1 0.0000442 351.562
## 8  Voss_ASA14          1       1        0   0          1 0.0000442 375.000
## 9  Voss_ASA14          1       1        0   0          1 0.0000442 398.438
## 10 Voss_ASA14          1       1        0   0          1 0.0000442 421.875
##    Absorbance     Zmag      Zang
## 1   0.0417482 82170700 -0.234778
## 2   0.0430154 78968100 -0.234903
## 3   0.0471408 68093700 -0.235638
## 4   0.0599458 60912400 -0.233472
## 5   0.0729253 56467300 -0.231156
## 6   0.0817873 51378900 -0.230572
## 7   0.0925911 48610600 -0.228987
## 8   0.1148400 45332000 -0.225251
## 9   0.1222960 42558100 -0.225022
## 10  0.1383710 39362700 -0.223418
```

Accessing a database using dplyr commands

Alternatively, a connection can be made to the server by creating a series of dplyr table objects.

```
library(mosaic)
db <- src_mysql(dbname = "wai", host = "scidb.smith.edu", user = "waiuser",
          password="smith_waiDB")
Measurements <- tbl(db, "Measurements")
PI_Info <- tbl(db, "PI_Info")
Subject <- tbl(db, "Subject")
```

Let's explore the `PI_Info` table.

```
PI_Info  %>% summarise(total = n())
```

```
## Source: mysql 5.5.47-0ubuntu0.14.04.1 [waiuser@scidb.smith.edu:/wai]
## From: <derived table> [?? x 1]
##
##    total
##    (dbl)
## 1      3
## ..   ...
```

```
PI_Info %>% collect() %>% data.frame()   # collect() is a bad idea when dealing with large tables!
```

```
##      Identifier PI_Year                                    PI
## 1     Voss_2014    2014                 Susan Voss;Abur;Horton
## 2 Rosowski_2012    2012                         Rosowski, J.J.
```

```
## 3    Voss_ASA14    2014 Susan E. Voss; Defne Abur; Hiwot Kassaye
##
## 1
## 2 Eaton-Peabody Laboratory, Massachusetts Eye and Ear Infirmary, Boston; Department of Otology and La
## 3
##                            Email
## 1                svoss@smith.edu
## 2 John_Rosowski@meei.harvard.edu
## 3                svoss@smith.edu
##                                                                                          Title
## 1                                                 Intrasubject Variability in Power Reflectance
## 2             Ear-Canal Reflectance, Umbo Velocity, and Tympanometry in Normal-Hearing Adults
## 3 Comparisons of reflectance measurements across measurements sessions, instruments, and ages
##                              Pub       Date
## 1              J Am Acad Audiol 10/04/2014
## 2                 Ear & Hearing 11/06/2015
## 3 Acoustical Society of America       2014
##                                                       URL
## 1 http://www.ncbi.nlm.nih.gov/pubmed/?term=abur+voss+2014
## 2             http://www.ncbi.nlm.nih.gov/pubmed/21857517
## 3                    http://dx.doi.org/10.1121/1.4877464
##
## 1
## 2 HearID (Mimosa Acoustics); \nNormal Criteria as follows: \n(1) There was no history of significant
## 3
```

Let's explore the Subjects table.

```
Subject  %>% summarise(total = n())
```

```
## Source: mysql 5.5.47-0ubuntu0.14.04.1 [waiuser@scidb.smith.edu:/wai]
## From: <derived table> [?? x 1]
##
##    total
##    (dbl)
## 1     52
## ..   ...
```

```
Subject %>% collect()  # be careful with collect() with large tables!
```

```
## Source: local data frame [52 x 11]
##
##       Identifier Sub_Number Session_Total   Age Female  Race Ethnicity
##            (chr)      (chr)         (int) (int)  (int) (int)     (int)
## 1      Voss_2014          1             4    20      1     0         0
## 2      Voss_2014          2             8    20      1     0         0
## 3      Voss_2014          3             7    21      1     0         0
## 4      Voss_2014          4             4    20      1     0         0
## 5      Voss_2014          5             4    19      1     0         0
## 6      Voss_2014          6             6    20      1     0         0
## 7      Voss_2014          8             3    20      1     0         0
## 8      Voss_2014          9             4    22      1     0         0
## 9  Rosowski_2012          3             1    30      1     5         2
## 10 Rosowski_2012          6             1    29      0     5         2
## ..           ...        ...           ...   ...    ...   ...       ...
```

```
## Variables not shown: Left_Ear_Status (int), Right_Ear_Status (int),
##   Sub_Notes (chr), ID (dbl)
```

Let's explore the Measurements table.

```
Measurements %>% summarise(total = n())
```

```
## Source: mysql 5.5.47-0ubuntu0.14.04.1 [waiuser@scidb.smith.edu:/wai]
## From: <derived table> [?? x 1]
##
##     total
##     (dbl)
## 1  107226
## ..    ...
```

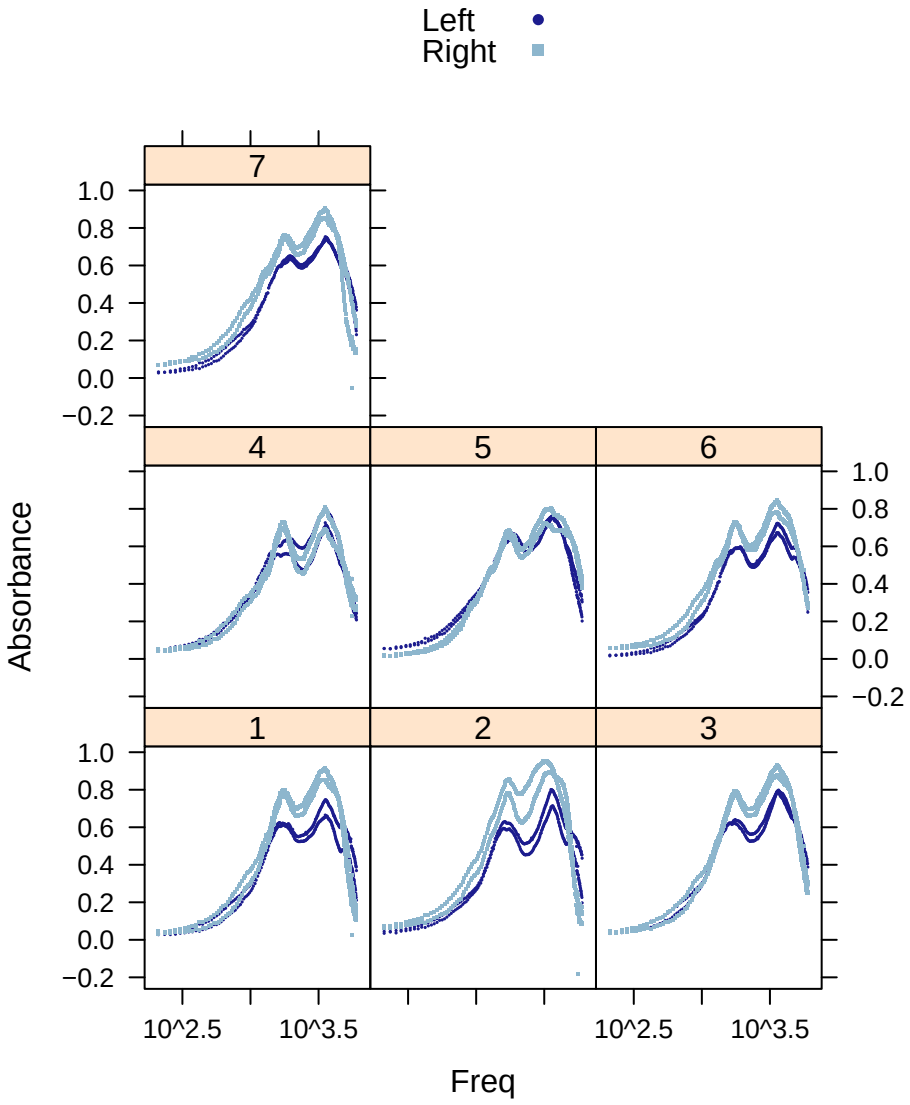
Let's download the data from a given subject

```
onesubj <-
  Measurements %>%
  filter(Identifier=="Voss_2014", Sub_Number==1) %>%
  collect %>%
  mutate(SessionNum = as.factor(Session))
head(onesubj)
```

```
## Source: local data frame [6 x 12]
##
##   Identifier Sub_Number Session Left_Ear   MEP Instrument Ear_Area  Freq
##        (chr)      (chr)   (int)    (int) (chr)      (int)    (chr) (dbl)
## 1  Voss_2014          1       1        0     0          1      NaN   211
## 2  Voss_2014          1       1        0     0          1      NaN   234
## 3  Voss_2014          1       1        0     0          1      NaN   258
## 4  Voss_2014          1       1        0     0          1      NaN   281
## 5  Voss_2014          1       1        0     0          1      NaN   305
## 6  Voss_2014          1       1        0     0          1      NaN   328
## Variables not shown: Absorbance (dbl), Zmag (dbl), Zang (dbl), SessionNum
##   (fctr)
```

Finally we can plot the results

```
onesubj <- mutate(onesubj,
  Ear=ifelse(Left_Ear==1, "Left", "Right"))
xyplot(Absorbance ~ Freq | SessionNum, group=Ear, auto.key=TRUE,
       scales=list(x=list(log=TRUE)),
       cex=0.2, data=onesubj)
```


Left
Right
7
4
5
6
1
2
3
Absorbance
Freq
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
10^2.5
10^3.5