Homework13a

Steve Simon

This file was created on 2020-07-24 and last modified on 2021-07-17.

Note: this solution uses R and SQLite. An alternate solution using SAS and Oracle is also available.

Note: Some of the names used in this code are arbitrary and you can choose whatever names you want. To emphasize which names can be modified at your discretion, I am using names of famous statisticians.

The statistician being honored in this code is Hirotugu Akaike.

Q1. Count the number of records after an inner join of acupuncture_baseline_results and acupuncture_one_year_results. Count the number of records after a left join of acupuncture_baseline_results and acupuncture_one_year_results. Why are these numbers different?

```
library(sqldf)
```

```
## Loading required package: gsubfn
```

```
## Loading required package: proto
```

```
## Loading required package: RSQLite
```

```
akaike <- dbConnect(SQLite(),
  dbname="../data/melange.sqlite")
hirotugu3a <- dbGetQuery(conn=akaike,  "
    select count(*) as n
      from acupuncture_baseline_results as b
      join acupuncture_one_year_results as o
      on b.id=o.id
")
hirotugu3a
```

```
##     n
## 1 301
```

```
hirotugu3b <- dbGetQuery(conn=akaike,  "
    select count(*) as n
      from acupuncture_baseline_results as b
      left join acupuncture_one_year_results as o
      on b.id=o.id
")
hirotugu3b
```

```
##     n
## 1 401
```

```
dbDisconnect(conn=akaike)
```

Q2. Compute the average pk score at baseline, the average score at one year, and the average change score. Without running any formal statistical tests, tell us whether you think the pk scores are increasing, decreasing, or staying about the same.

```
library(sqldf)
akaike <- dbConnect(SQLite(),
  dbname="../data/melange.sqlite")
hirotugu4 <- dbGetQuery(conn=akaike,  "
    select
      avg(b.pk1) as pk1_avg,
      avg(o.pk5) as pk5_avg,
      avg(b.pk1)-avg(o.pk5) as change_score
      from acupuncture_baseline_results as b
      join acupuncture_one_year_results as o
      on b.id=o.id
")

hirotugu4
```

```
##    pk1_avg  pk5_avg change_score
## 1 25.56894 19.08245     6.486489
```

```
dbDisconnect(conn=akaike)
```

Q3. Display all the pk1 values for patients 64 and older.

```
library(sqldf)
akaike <- dbConnect(SQLite(),
  dbname="../data/melange.sqlite")
hirotugu5 <- dbGetQuery(conn=akaike,  "
    select
      d.id, d.age, b.pk1
      from acupuncture_demographics as d
      inner join acupuncture_baseline_results as b
      on d.id=b.id
      where d.age >= 64
")

hirotugu5
```

```
##     id age   pk1
## 1  131  64 14.50
## 2  172  64 74.25
## 3  217  65 22.50
## 4  252  64 29.75
## 5  380  65 41.25
## 6  578  64 13.00
## 7  590  65 20.50
## 8  634  64 70.00
## 9  734  64 18.00
## 10 884  65 17.50
```

```
dbDisconnect(conn=akaike)
```

Q4. There are 100 patients with baseline values but no values at one year. Use a left join to identify these patients. Print the ids of the first ten of these patients.

```
library(sqldf)
akaike <- dbConnect(SQLite(),
  dbname="../data/melange.sqlite")
hirotugu6 <- dbGetQuery(conn=akaike,  "
    select
      b.id as unmatched_ids
      from acupuncture_baseline_results as b
      left join acupuncture_one_year_results as o
      on b.id=o.id
    where o.id is null
    limit 10
")

hirotugu6
```

```
##    unmatched_ids
## 1            100
## 2            101
## 3            105
## 4            138
## 5            139
## 6            151
## 7            154
## 8            159
## 9            164
## 10           166
```

```
dbDisconnect(conn=akaike)
```

Q5. Compute the intersection of the ids from acupuncture_baseline_results and acupuncture_one_year_results. Display the first ten rows of data only.

```
library(sqldf)
walker <- dbConnect(SQLite(),
  dbname="../data/melange.sqlite")
helen1 <- dbGetQuery(conn=walker,  "
    select id
      from acupuncture_baseline_results
    intersect
    select id
      from acupuncture_one_year_results
    limit 10
")

helen1
```

```
##     id
```

```
## 1  104
## 2  108
## 3  112
## 4  113
## 5  114
## 6  126
## 7  130
## 8  131
## 9  135
## 10 137
```

```
dbDisconnect(conn=walker)
```

Q6. Compute the union of the ids from acupuncture_baseline_results and acupuncture_one_year_results. Display the first ten rows of data only.

```
library(sqldf)
walker <- dbConnect(SQLite(),
  dbname="../data/melange.sqlite")
helen2 <- dbGetQuery(conn=walker,  "
    select id
      from acupuncture_baseline_results
    union
    select id
      from acupuncture_one_year_results
    order by id
    limit 10
")

helen2
```

```
##     id
## 1  100
## 2  101
## 3  104
## 4  105
## 5  108
## 6  112
## 7  113
## 8  114
## 9  126
## 10 130
```

```
dbDisconnect(conn=walker)
```

Q7. In a previous question, you were asked to list the first ten ids that were in acupuncture_baseline_results but not in acupuncture_one_year_results. Use the set operator "minus" to achieve the same goal. Note: for SQLite, use "except" instead of "minus".

```
library(sqldf)
walker <- dbConnect(SQLite(),
  dbname="../data/melange.sqlite")
```

```
helen3 <- dbGetQuery(conn=walker,  "
    select id
      from acupuncture_baseline_results
    except
    select id
      from acupuncture_one_year_results
    limit 10
")

helen3
```

```
##     id
## 1  100
## 2  101
## 3  105
## 4  138
## 5  139
## 6  151
## 7  154
## 8  159
## 9  164
## 10 166
```

```
dbDisconnect(conn=walker)
```