SOC4001 Procesamiento avanzado de bases de datos en R

Tarea 3

Ponderación: 12% de la nota final del curso

Entrega: Desde el momento de entrega, los estudiantes tienen plazo hasta el domingo 19 de Octubre a las 23:59pm para completar esta tarea.

Formato: Desarrollar esta tarea en un RScript, agregando comentarios cuando sea necesario.

El código a continuación carga la Base de Datos Histórica Proyectos Adjudicados ANID (ex-conicyt) y extrae una selección de variables que son almacenados en el objecto `data_anid`.

```
library("tidyverse")
library("readr")

path <- url("https://raw.githubusercontent.com/ANID-GITHUB/Historico-de-Proyectos-Adjudicados/da63cab4fa

data_anid <- read_delim(path, delim = ";")

data_anid <- data_anid %>% rename(codigo_proyecto = CODIGO_PROYECTO, anno = ANO_FALLO, sexo = SEXO, area
```

Descripción de los datos: La Agencia Nacional de Investigación y Desarrollo (ANID) cada año adjudica financiamiento para proyectos en Ciencia y Tecnología a través de sus diferentes concursos. La base de datos denominada "BDH_Proyectos" contiene la información disponible de proyectos adjudicados por la Agencia (antes del 2020, CONICYT) desde el año 1982 hasta el 2020, con fecha de corte al 31 de diciembre del 2020. Cada fila representa una iniciativa adjudicada. Los datos deben verse así:

```
## # A tibble: 6 x 5
##   codigo_proyecto  anno sexo   area                           monto
##   <chr>           <dbl> <chr>  <chr>                          <dbl>
## 1 1820005          1982 HOMBRE CIENCIAS NATURALES               300
## 2 1820006          1982 HOMBRE CIENCIAS MEDICAS Y DE LA SALUD   130
## 3 1820009          1982 HOMBRE CIENCIAS NATURALES               506
## 4 1820010          1982 HOMBRE HUMANIDADES                      335
## 5 1820015          1982 HOMBRE CIENCIAS NATURALES               260
## 6 1820043          1982 HOMBRE CIENCIAS AGRICOLAS               464
```

1) Usando los comandos `group_by()` y `summarise()` produce la siguiente tabla y asígnala al objeto `tabla_1`. El resultado debe verse así y corresponde al monto promedio asignado a cada investigadores de caga sexo en cada area:

```
tabla_1 <- data_anid  %>% group_by(area,anno,sexo) %>%
  summarise(across(c(monto), ~mean(.x,rm=T)))
tabla_1 %>% head()
```

```
## # A tibble: 6 x 4
```

```
## # Groups:   area, anno [4]
##   area                anno sexo   monto
##   <chr>              <dbl> <chr>  <dbl>
## 1 CIENCIAS AGRICOLAS  1982 HOMBRE  408.
## 2 CIENCIAS AGRICOLAS  1982 MUJER   549
## 3 CIENCIAS AGRICOLAS  1983 HOMBRE  382.
## 4 CIENCIAS AGRICOLAS  1984 HOMBRE  355.
## 5 CIENCIAS AGRICOLAS  1984 MUJER   373
## 6 CIENCIAS AGRICOLAS  1985 HOMBRE  414.
```

2) Carga la base de datos con el IPC anual y guárdala en un objecto llamado `datos_ipc`. Para los años con valores perdidos en la variable `datos_ipc$ipc`, usa la función `fill()` para asignales el valor correspondiente al año siguiente. Conserva sólo las variables `anno` e `ipc`. Los datos deben verse así:

```
datos_ipc <- read_csv("https://raw.githubusercontent.com/mebucca/dar_soc4001/master/homework/ipc.csv")
```

```
## New names:
## Rows: 39 Columns: 3
## -- Column specification
## -------------------------------------------------------- Delimiter: "," dbl
## (3): ...1, anno, ipc
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

```
datos_ipc <- datos_ipc  %>% fill(ipc,.direction = "up")
datos_ipc %>% head()
```

```
## # A tibble: 6 x 3
##    ...1  anno   ipc
##   <dbl> <dbl> <dbl>
## 1     1  1982  2.79
## 2     2  1983  8.98
## 3     3  1984  6.53
## 4     4  1985 10.1
## 5     5  1986  6.45
## 6     6  1987  6.54
```

3) Usando algunos de los comandos `_join()` junta los datos en `tabla_1` y `datos_ipc` preservando toda la información disponible en `tabla_1`. El resultado debe verse así:

```
tabla_1 <- tabla_1 %>% left_join(datos_ipc, by="anno")
tabla_1 %>% head()
```

```
## # A tibble: 6 x 6
## # Groups:   area, anno [4]
##   area                anno sexo   monto  ...1   ipc
##   <chr>              <dbl> <chr>  <dbl> <dbl> <dbl>
## 1 CIENCIAS AGRICOLAS  1982 HOMBRE  408.     1  2.79
## 2 CIENCIAS AGRICOLAS  1982 MUJER   549      1  2.79
## 3 CIENCIAS AGRICOLAS  1983 HOMBRE  382.     2  8.98
```

```
## 4 CIENCIAS AGRICOLAS  1984 HOMBRE  355.     3  6.53
## 5 CIENCIAS AGRICOLAS  1984 MUJER   373      3  6.53
## 6 CIENCIAS AGRICOLAS  1985 HOMBRE  414.     4 10.1
```

4) Crea la nueva variable `monto_precios2021` multiplicando las variables `monto` e `ipc`. Posteriormente remueve las variables `monto` e `ipc`. El resultado debe verse así:

```
tabla_1 <- tabla_1 %>% mutate(monto_precios2021 = monto*ipc) %>% select(-c(monto,ipc))
tabla_1
```

```
## # A tibble: 608 x 5
## # Groups:   area, anno [277]
##    area                anno sexo    ...1 monto_precios2021
##    <chr>              <dbl> <chr>  <dbl>             <dbl>
##  1 CIENCIAS AGRICOLAS  1982 HOMBRE     1             1141.
##  2 CIENCIAS AGRICOLAS  1982 MUJER      1             1534.
##  3 CIENCIAS AGRICOLAS  1983 HOMBRE     2             3428.
##  4 CIENCIAS AGRICOLAS  1984 HOMBRE     3             2317.
##  5 CIENCIAS AGRICOLAS  1984 MUJER      3             2437.
##  6 CIENCIAS AGRICOLAS  1985 HOMBRE     4             4198.
##  7 CIENCIAS AGRICOLAS  1986 HOMBRE     5             7220.
##  8 CIENCIAS AGRICOLAS  1986 MUJER      5             3161.
##  9 CIENCIAS AGRICOLAS  1987 HOMBRE     6            15665.
## 10 CIENCIAS AGRICOLAS  1987 MUJER      6             9688.
## # ... with 598 more rows
## # i Use `print(n = ...)` to see more rows
```

5) Usando el comando `pivot_wider()` transforma los datos de la siguiente manera.

```
tabla_1 %>%
  pivot_wider(names_from=sexo, values_from=monto_precios2021)
```

```
## # A tibble: 277 x 6
## # Groups:   area, anno [277]
##    area                anno  ...1   HOMBRE   MUJER `SIN INFORMACION`
##    <chr>              <dbl> <dbl>    <dbl>   <dbl>              <dbl>
##  1 CIENCIAS AGRICOLAS  1982     1    1141.   1534.                 NA
##  2 CIENCIAS AGRICOLAS  1983     2    3428.     NA                  NA
##  3 CIENCIAS AGRICOLAS  1984     3    2317.   2437.                 NA
##  4 CIENCIAS AGRICOLAS  1985     4    4198.     NA                  NA
##  5 CIENCIAS AGRICOLAS  1986     5    7220.   3161.                 NA
##  6 CIENCIAS AGRICOLAS  1987     6   15665.   9688.                 NA
##  7 CIENCIAS AGRICOLAS  1988     7   41485.     NA                  NA
##  8 CIENCIAS AGRICOLAS  1989     8   44183.     NA                  NA
##  9 CIENCIAS AGRICOLAS  1990     9   77329.  84307.                 NA
## 10 CIENCIAS AGRICOLAS  1991    10 1118740. 462418.                 NA
## # ... with 267 more rows
## # i Use `print(n = ...)` to see more rows
```

6) Usa la función `replace_na()` para reemplazar los valores perdidos en las variables `HOMBRE` y `MUJER` por ceros. El resultado debe verse así:

```
tabla_1 %>%
  pivot_wider(names_from=sexo, values_from=monto_precios2021) %>%
  replace_na(list(HOMBRE = 0, MUJER = 0))
```

```
## # A tibble: 277 x 6
## # Groups:   area, anno [277]
##    area                anno  ...1   HOMBRE   MUJER `SIN INFORMACION`
##    <chr>              <dbl> <dbl>    <dbl>   <dbl>             <dbl>
##  1 CIENCIAS AGRICOLAS  1982     1    1141.   1534.                NA
##  2 CIENCIAS AGRICOLAS  1983     2    3428.      0                 NA
##  3 CIENCIAS AGRICOLAS  1984     3    2317.   2437.                NA
##  4 CIENCIAS AGRICOLAS  1985     4    4198.      0                 NA
##  5 CIENCIAS AGRICOLAS  1986     5    7220.   3161.                NA
##  6 CIENCIAS AGRICOLAS  1987     6   15665.   9688.                NA
##  7 CIENCIAS AGRICOLAS  1988     7   41485.      0                 NA
##  8 CIENCIAS AGRICOLAS  1989     8   44183.      0                 NA
##  9 CIENCIAS AGRICOLAS  1990     9   77329.  84307.                NA
## 10 CIENCIAS AGRICOLAS  1991    10 1118740. 462418.                NA
## # ... with 267 more rows
## # i Use `print(n = ...)` to see more rows
```

7) Crea una nueva variable llamada `dif_hombremujer` que mida la diferencia entre el monto asignado a hombres y mujeres `= HOMBRE - MUJER`. Posteriormente conserva sólo las variables `anno`, `area` y `dif_hombremujer`. El resultado debe verse así:

```
tabla_1 %>% select(anno,sexo,area,monto_precios2021) %>%
  pivot_wider(names_from=sexo, values_from=monto_precios2021) %>%
  replace_na(list(HOMBRE = 0, MUJER = 0)) %>%
  mutate(dif_hombremujer = HOMBRE - MUJER) %>%select(anno,area,dif_hombremujer)
```

```
## # A tibble: 277 x 3
## # Groups:   area, anno [277]
##     anno area                dif_hombremujer
##    <dbl> <chr>                         <dbl>
##  1  1982 CIENCIAS AGRICOLAS            -394.
##  2  1983 CIENCIAS AGRICOLAS            3428.
##  3  1984 CIENCIAS AGRICOLAS            -120.
##  4  1985 CIENCIAS AGRICOLAS            4198.
##  5  1986 CIENCIAS AGRICOLAS            4060.
##  6  1987 CIENCIAS AGRICOLAS            5978.
##  7  1988 CIENCIAS AGRICOLAS           41485.
##  8  1989 CIENCIAS AGRICOLAS           44183.
##  9  1990 CIENCIAS AGRICOLAS           -6978.
## 10  1991 CIENCIAS AGRICOLAS          656322.
## # ... with 267 more rows
## # i Use `print(n = ...)` to see more rows
```

8) Usando el comando `pivot_wider()` modifica la tabla producida en (7) y produce la siguiente tabla:

```
tabla_1 %>% select(anno,sexo,area,monto_precios2021) %>%
  pivot_wider(names_from=sexo, values_from=monto_precios2021) %>%
```

```
  replace_na(list(HOMBRE = 0, MUJER = 0)) %>%
  mutate(dif_hombremujer = HOMBRE - MUJER) %>%  select(anno,area,dif_hombremujer) %>%
  pivot_wider(names_from=area,values_from=dif_hombremujer)
```

```
## # A tibble: 39 x 10
## # Groups:   anno [39]
##     anno CIENC~1 CIENC~2 CIENC~3 CIENC~4 HUMAN~5 INGEN~6 MULTI~7 NO AP~8 SIN I~9
##    <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
##  1  1982   -394.   -147.  7.66e1  5.70e2  1.02e3   -115.      NA      NA      NA
##  2  1983   3428.   3425. -3.44e2 -6.00e2  1.78e3   3360.      NA      NA      NA
##  3  1984   -120.    305.  6.36e1 -9.96e1  2.20e3   -125.      NA      NA      NA
##  4  1985   4198.    189.  3.89e2 -2.45e2 -8.91e1    165.      NA      NA      NA
##  5  1986   4060.   1025.  4.24e3  7.19e2  8.38e2   3544.      NA      NA      NA
##  6  1987   5978.  16343. -5.09e2  1.89e3  1.66e3 -14112.      NA      NA      NA
##  7  1988  41485.  11794. -6.31e3  1.10e3 -6.32e3  29806.      NA      NA      NA
##  8  1989  44183.  14803. -1.60e3 -2.29e3  5.30e3  14003.      NA      NA      NA
##  9  1990  -6978.  30467.  8.11e3 -1.00e4 -2.93e4   2569.      NA      NA      NA
## 10  1991 656322.  18343.  1.17e5  4.36e3  2.91e3  84903.      NA      NA      NA
## # ... with 29 more rows, and abbreviated variable names
## #   1: 'CIENCIAS AGRICOLAS', 2: 'CIENCIAS MEDICAS Y DE LA SALUD',
## #   3: 'CIENCIAS NATURALES', 4: 'CIENCIAS SOCIALES', 5: HUMANIDADES,
## #   6: 'INGENIERIA Y TECNOLOGIA', 7: MULTIDISCIPLINARIO, 8: 'NO APLICA',
## #   9: 'SIN INFORMACION'
## # i Use 'print(n = ...)' to see more rows
```

9) Elige el valor correspondiente a una celda cualquiera y describe la información que comunica.

Cada celda indica la diferencia entre el promedio de recursos asignado a hombres y mujeres en un determinado año y área de la ciencia.