SIR-panc-cancer-BDIPMN

Tommaso Pollini

2023-03-24

Introduction

The aim of the study is to identify a subset of patients affected by a BD-IPMN of the pancreas where surveillance discontinuation could be considered. To compare the incidence of pancreatic cancer in our cohort with that of the general population we are going to calculate the Standardized Incidence Ratio (SIR) of pancreatic cancer.

In summary, first we will calculate the number of person-year that every patients has contributed to every age group. Afterwards, we will get the sum of the element wise multiplication between person-year and the crude rate of pancreatic cancer, which equals the number of expected cases. Finally we will calculate SIR and its 95% confidence interval.

```
# The first thing that we need is to read the dataset, and
# assigning it to the variable df. For the purpose of this
# markdown a randomly generated dataset will be used

df <- tibble(DOB = ymd(paste0(round(seq(1930, 1980, length = 2000)),
    "/1/1")), DSTART = DOB + runif(35, 1, 7500), DLAST = DSTART +
    runif(35, 100, 7500), Institution = rep(c("SGH", "B", "C",
    "D", "E"), 400), ID = c(1:2000), Sex = sample(c("MALE", "FEMALE"),
    size = 2000, replace = TRUE), OBS = sample(c(0, 1), size = 2000,
    replace = TRUE, prob = c(0.997, 0.003)))

# With the prob argument of the OBS variable, we have set
# the rate of observation at 0.3% as an example, changing
# this number alter the final SIR significantly
```

Table 1: Crude rate (per 100,000 people) of Pancreatic Cancer

	List A countries		Singapore	
Age (years)	Male	Female	Male	Female
0-4	0.02	0.02	0.0	0.0
5-9	0.02	0.00	0.0	0.0
10-14	0.04	0.02	0.0	0.0
15-19	0.02	0.09	0.2	0.2
20-24	0.12	0.17	0.0	0.0
25-29	0.19	0.27	0.3	0.1
30-34	0.43	0.38	0.3	0.3
35-39	1.00	0.83	1.6	0.6
40-44	2.70	1.80	3.0	2.0
45-49	5.70	3.80	3.1	4.4
50-54	11.50	7.50	7.9	4.7
55-59	20.10	13.20	17.7	10.4
60-64	33.30	22.50	21.0	16.4
65-69	48.10	35.70	45.1	26.6
70-74	62.00	52.40	50.8	38.2
75-79	82.40	67.40	61.0	63.6
80-85	98.20	79.40	72.5	52.5
85+	106.90	93.20	71.8	83.7

[a] List A countries: Germany, Italy, Republic of Korea, The Netherlands, Spain, UK, England and Wales, and the USA

The crude rate of pancreatic cancer are inputted into different tibbles, that will be used later to calculate the number of expected cases

```
# Crude rate (per 100.000) for male per age group
RaM <- tibble(group_1 = 0.02, group_2 = 0.02, group_3 = 0.04,
    group_4 = 0.02, group_5 = 0.12, group_6 = 0.19, group_7 = 0.43,
    group_8 = 1, group_9 = 2.7, group_10 = 5.7, group_11 = 11.5,
    group_12 = 20.1, group_13 = 33.3, group_14 = 48.1, group_15 = 62,
    group_16 = 82.4, group_17 = 98.2, group_18 = 106.9)

# Crude rate (per 100.000) for female per age group
RaF <- tibble(group_1 = 0.02, group_2 = 0, group_3 = 0.02, group_4 = 0.09,
    group_5 = 0.17, group_6 = 0.27, group_7 = 0.38, group_8 = 0.83,
    group_9 = 1.8, group_10 = 3.8, group_11 = 7.5, group_12 = 13.2,
    group_13 = 22.5, group_14 = 35.7, group_15 = 52.4, group_16 = 67.4,
    group_17 = 79.4, group_18 = 93.2)

# Crude rate (per 100.000) for male in SINGAPORE per age
# group
sRaM <- tibble(group_1 = 0, group_2 = 0, group_3 = 0, group_4 = 0.2,
    group_5 = 0, group_6 = 0.3, group_7 = 0.3, group_8 = 1.6,
    group_9 = 3, group_10 = 3.1, group_11 = 7.9, group_12 = 17.7,
    group_13 = 21, group_14 = 45.1, group_15 = 50.8, group_16 = 61,
    group_17 = 72.5, group_18 = 71.8)
```

```r
# Crude rate (per 100.000) for female in SINGAPORE per age
# group
sRaF <- tibble(group_1 = 0, group_2 = 0, group_3 = 0, group_4 = 0.2,
    group_5 = 0, group_6 = 0.1, group_7 = 0.3, group_8 = 0.6,
    group_9 = 2, group_10 = 4.4, group_11 = 4.7, group_12 = 10.4,
    group_13 = 16.4, group_14 = 26.6, group_15 = 38.2, group_16 = 63.6,
    group_17 = 52.5, group_18 = 83.7)
```

Overall Person-Year

To calculate SIR, we need to measure the number of expected cases, given the number of person-years stratified by age group and the incidence of pancreatic cancer in the respective age group. Crude rates of pancreatic cancer in the general population were obtained from the IARC dataset for Germany, Italy, Republic of Korea, The Netherlands, Spain, UK, England and Wales, and the USA. Data for Singapore was extracted from the singaporean national registry of disease. Therefore, the number of expected cases was measured separately for patients from SGH (Singapore General Hospital) from patients from all other institutions in the study.

Using the dplyr package in R, we create a new column "s" with a seq.Date object from the date of the first observation (DSTART) to the last date of surveillance or the date when a pancreatic malignancy was identified (DLAST), with days increment.

```r
# The following is to calculate person-years of patients
# from all institution excepts Singapore
person_year <- df %>%
    filter(Institution != "SGH") %>%
    rowwise() %>%
    mutate(s = list(seq.Date(DSTART, DLAST, by = "days"))) %>%
    unnest(s) %>%
    # create a new column with the corresponding age group
    # for every date in seq.Date
mutate(age = s - DOB, group = as.numeric(floor((age/365.25)/5) +
    1), group = ifelse(group > 18, 18, group)) %>%
    group_by(ID, DOB, DSTART, DLAST, group, Sex, OBS) %>%
    summarise(n = round(n()/365.25, 2)) %>%
    ungroup %>%
    arrange(group) %>%
    pivot_wider(names_from = "group", names_prefix = "group_",
        values_from = "n", values_fill = 0)
```

```
## `summarise()` has grouped output by 'ID', 'DOB', 'DSTART', 'DLAST', 'group',
## 'Sex'. You can override using the `.groups` argument.
```

```r
# The following is to calculate person-years of patients
# from Singapore
singapore_person_year <- df %>%
    filter(Institution == "SGH") %>%
    rowwise() %>%
    mutate(s = list(seq.Date(DSTART, DLAST, by = "days"))) %>%
    unnest(s) %>%
    # create a new column with the corresponding age group
    # for every date in seq.Date
```

```
mutate(age = s - DOB, group = as.numeric(floor((age/365.25)/5) +
    1), group = ifelse(group > 18, 18, group)) %>%
    group_by(ID, DOB, DSTART, DLAST, group, Sex, OBS) %>%
    summarise(n = round(n()/365.25, 2)) %>%
    ungroup %>%
    arrange(group) %>%
    pivot_wider(names_from = "group", names_prefix = "group_",
        values_from = "n", values_fill = 0)
```

```
## `summarise()` has grouped output by 'ID', 'DOB', 'DSTART', 'DLAST', 'group',
## 'Sex'. You can override using the `.groups` argument.
```

```
totalpy <- person_year %>%
    select(starts_with("group_")) %>%
    summarise(across(where(is.numeric), sum)) %>%
    pivot_longer(., cols = starts_with("group_"))
totpy <- sum(totalpy$value)

sin_totalpy <- singapore_person_year %>%
    select(starts_with("group_")) %>%
    summarise(across(where(is.numeric), sum)) %>%
    pivot_longer(., cols = starts_with("group_"))
sin_totpy <- sum(totalpy$value)

py <- totpy + sin_totpy
```

Table 2: Person year

ID	DOB	DSTART	DLAST	Sex	OBS	group_1	group_2	group_3	group_4
3	1930-01-01	1931-06-26	1943-06-27	FEMALE	0	3.52	5.00	3.49	0.00
9	1930-01-01	1932-06-23	1939-09-05	MALE	0	2.52	4.68	0.00	0.00
15	1930-01-01	1931-11-17	1948-06-22	FEMALE	0	3.12	5.00	5.00	3.47
18	1930-01-01	1930-10-18	1943-02-04	FEMALE	0	4.21	5.00	3.09	0.00
19	1930-01-01	1931-07-02	1950-10-14	FEMALE	0	3.50	5.00	5.00	5.00
20	1930-01-01	1931-06-06	1948-07-06	MALE	0	3.58	5.00	5.00	3.51

Note:
A random sample of 6 cases is reported

Person-year by Sex

```
fem_pry <- person_year %>%
    filter(Sex == "FEMALE") %>%
    summarise(across(OBS:last_col(), ~sum(., is.na = 0, 0)))
```

```
men_pry <- person_year %>%
    filter(Sex == "MALE") %>%
    summarise(across(OBS:last_col(), ~sum(., is.na = 0, 0)))
```

```r
# Same calculations is made in patients from Singapore

singapore_fem_pry <- singapore_person_year %>%
    filter(Sex == "FEMALE") %>%
    summarise(across(OBS:last_col(), ~sum(., is.na = 0, 0)))

singapore_men_pry <- singapore_person_year %>%
    filter(Sex == "MALE") %>%
    summarise(across(OBS:last_col(), ~sum(., is.na = 0, 0)))
```

Given n age categories, the number of expected cases is equal to $E = (P_1I_1 + P_2I_2 + P_3I_3...P_nI_n)$ with P being the number of person-years and I the incidence of pancreatic cancer in the n_{th} age group

```r
FemaleEx <- bind_rows(RaF, fem_pry) %>%
    summarise(across(group_1:group_18 & !OBS, ~prod(.))) %>%
    replace(is.na(.), 0) %>%
    mutate(total_female_expected = (rowSums(., na.rm = TRUE)/1e+05))
```

```r
MaleEx <- bind_rows(RaM, men_pry) %>%
    summarise(across(group_1:group_18 & !OBS, ~prod(.))) %>%
    replace(is.na(.), 0) %>%
    mutate(total_male_expected = (rowSums(., na.rm = TRUE)/1e+05))
```

```r
singapore_FemaleEx <- bind_rows(sRaF, singapore_fem_pry) %>%
    summarise(across(group_1:group_18 & !OBS, ~prod(.))) %>%
    replace(is.na(.), 0) %>%
    mutate(total_female_expected = (rowSums(., na.rm = TRUE)/1e+05))

singapore_MaleEx <- bind_rows(sRaM, singapore_men_pry) %>%
    summarise(across(group_1:group_18 & !OBS, ~prod(.))) %>%
    replace(is.na(.), 0) %>%
    mutate(total_male_expected = (rowSums(., na.rm = TRUE)/1e+05))
```

Standardized Incidence Ratio

After we obtained the number of expected cases, we can calculate the SIR, defined as:

$$SIR = \frac{\sum_{k=1}^{M} D_k}{\sum_{k=1}^{M} t_k \lambda_k^*}$$

where the total number of events observed in the cohort is $D = \sum_{k=1}^{M} D_k$ and the total number of expected events is $E^* = \sum_{k=1}^{M} E_k^* = \sum_{k=1}^{M} t_k \lambda_k^*$

To approximate the 95% confidence interval (95%CI) we can use the Wilson and Hilferty approximation of the chi-square percentiles:

$$\chi_{v,a} = v\left(1 - \frac{2}{9v} + Z_\alpha\sqrt{\frac{2}{9v}}\right)^3$$

Therefore, the lower limit of the 95%CI is equal to $SIR_L = \frac{D}{E^*}\left(1 - \frac{1}{9D} + \frac{Z_{\alpha/2}}{3\sqrt{D}}\right)^3$ with Z equal to -1.96 while the upper limit is equal to $SIR_U = \frac{D+1}{E^*}\left(1 - \frac{1}{9(D+1)} + \frac{z_{1-\alpha/2}}{3\sqrt{D+1}}\right)^3$ with Z equal to 1.96

```
mEx <- MaleEx$total_male_expected
mObs <- men_pry$OBS

fEx <- FemaleEx$total_female_expected
fObs <- fem_pry$OBS

sin_mEx <- singapore_MaleEx$total_male_expected
sin_mObs <- singapore_men_pry$OBS

sin_fEx <- singapore_FemaleEx$total_female_expected
sin_fObs <- singapore_fem_pry$OBS

totEx <- mEx + fEx + sin_mEx + sin_fEx
totObs <- mObs + fObs + sin_mObs + sin_fObs

SIR <- totObs/totEx
CIl <- SIR * (((1 - (1/(9 * totObs))) - (1.96/(3 * sqrt(totObs))))^3)
CIu <- SIR * ((totObs + 1)/totObs) * ((1 - (1/(9 * (totObs +
    1))) + (1.96/(3 * sqrt(totObs + 1))))^3)

mSIR <- (mObs + sin_mObs)/(mEx + sin_mEx)
mCIl <- mSIR * (((1 - (1/(9 * (mObs + sin_mObs)))) - (1.96/(3 *
    sqrt(mObs + sin_mObs))))^3)
mCIu <- mSIR * (((mObs + sin_mObs) + 1)/(mObs + sin_mObs)) *
    ((1 - (1/(9 * ((mObs + sin_mObs) + 1))) + (1.96/(3 * sqrt((mObs +
        sin_mObs) + 1))))^3)

fSIR <- (fObs + sin_fObs)/(fEx + sin_fEx)
fCIl <- fSIR * (((1 - (1/(9 * (fObs + sin_fObs)))) - (1.96/(3 *
    sqrt(fObs + sin_fObs))))^3)
fCIu <- fSIR * (((fObs + sin_fObs) + 1)/(fObs + sin_fObs)) *
    ((1 - (1/(9 * ((fObs + sin_fObs) + 1))) + (1.96/(3 * sqrt((fObs +
        sin_fObs) + 1))))^3)
```

```
options(width = 80)
print(paste("Total number of person-year is", py))
```

```
## [1] "Total number of person-year is 38373.3"
```

```
print(paste("The number of expected cases is", round(totEx, 2),
    "while observed cases are", round(totObs, 2)))
```

```
## [1] "The number of expected cases is 0.02 while observed cases are 7"
```

```
print(paste("The number of expected cases in male patients is",
    round((mEx + sin_mEx), 2), "while observed cases are", round((mObs +
        sin_mObs), 2)))
```

```
## [1] "The number of expected cases in male patients is 0.01 while observed cases are 5"
```

```
print(paste("The number of expected cases in female patients is",
    round((fEx + sin_fEx), 2), "while observed cases are", round((fObs +
        sin_fObs), 2)))
```

```
## [1] "The number of expected cases in female patients is 0.01 while observed cases are 2"
```

```
options(width = 80)
print(paste("The SIR for patients with a trivial BD-IPMN is:",
    round(SIR, 2), "with a 95%CI of", round(CIl, 2), "-", round(CIu,
        2)))
```

```
## [1] "The SIR for patients with a trivial BD-IPMN is: 328.07 with a 95%CI of 131.43 - 675.99"
```

```
print(paste("The SIR for male patients is:", round(mSIR, 2),
    "with a 95%CI of", round(mCIl, 2), "-", round(mCIu, 2)))
```

```
## [1] "The SIR for male patients is: 511.06 with a 95%CI of 164.7 - 1192.64"
```

```
print(paste("The SIR for female patients is:", round(fSIR, 2),
    "with a 95%CI of", round(fCIl, 2), "-", round(fCIu, 2)))
```

```
## [1] "The SIR for female patients is: 173.11 with a 95%CI of 19.44 - 625.01"
```

Appendix

The session info are below

```
## R version 4.2.2 (2022-10-31 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 22621)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] formatR_1.14     forcats_0.5.2    stringr_1.5.0    purrr_1.0.1
##  [5] tibble_3.1.8     ggplot2_3.4.0    tidyverse_1.3.2  kableExtra_1.3.4
##  [9] lubridate_1.9.0  timechange_0.2.0 dplyr_1.0.10     tidyr_1.2.1
## [13] knitr_1.41       readr_2.1.3
```

```
##
## loaded via a namespace (and not attached):
##  [1] tidyselect_1.2.0    xfun_0.36           haven_2.5.1
##  [4] gargle_1.2.1        colorspace_2.0-3    vctrs_0.5.1
##  [7] generics_0.1.3      htmltools_0.5.4     viridisLite_0.4.1
## [10] yaml_2.3.6          utf8_1.2.2          rlang_1.0.6
## [13] pillar_1.8.1        withr_2.5.0         glue_1.6.2
## [16] DBI_1.1.3           dbplyr_2.3.0        readxl_1.4.1
## [19] modelr_0.1.10       lifecycle_1.0.3     cellranger_1.1.0
## [22] munsell_0.5.0       gtable_0.3.1        rvest_1.0.3
## [25] evaluate_0.20       tzdb_0.3.0          fastmap_1.1.0
## [28] fansi_1.0.3         broom_1.0.2         googlesheets4_1.0.1
## [31] scales_1.2.1        backports_1.4.1     jsonlite_1.8.4
## [34] webshot_0.5.4       systemfonts_1.0.4   fs_1.5.2
## [37] hms_1.1.2           digest_0.6.31       stringi_1.7.12
## [40] grid_4.2.2          cli_3.6.0           tools_4.2.2
## [43] magrittr_2.0.3      crayon_1.5.2        pkgconfig_2.0.3
## [46] ellipsis_0.3.2      xml2_1.3.3          reprex_2.0.2
## [49] googledrive_2.0.0   assertthat_0.2.1    rmarkdown_2.20
## [52] svglite_2.1.1       httr_1.4.4          rstudioapi_0.14
## [55] R6_2.5.1            compiler_4.2.2
```