

# SAS\_proj

Pavan Kalyan

2024-02-04

```
library(tidyverse)

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.4      v readr      2.1.5
## v forcats    1.0.0      v stringr   1.5.1
## v ggplot2    3.4.4      v tibble    3.2.1
## v lubridate  1.9.3      v tidyr     1.3.1
## v purrr      1.0.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(lubridate)

allergy <- read.csv("/cloud/project/allergy.csv")
condition <- read.csv("/cloud/project/condition.csv")
encounter <- read.csv("/cloud/project/encounter.csv")
lab <- read.csv("/cloud/project/lab.csv")
location <- read.csv("/cloud/project/location.csv")
patient <- read.csv("/cloud/project/patient.csv")
practitioner <- read.csv("/cloud/project/practitioner.csv")
procedure <- read.csv("/cloud/project/procedure.csv")
vital_sign <- read.csv("/cloud/project/vital_sign.csv")
medication <- read.csv("/cloud/project/medication.csv")
```

Changing to YYYY-MM-DD format Filtering the Data Generating a frequency table

```
condition <- read.csv("/cloud/project/condition.csv")
condition$condition_date <- as.Date(condition$condition_date,format="%m/%d/%Y")

Imp_date <- condition %>%
  filter(str_detect(code, "^I48"),
         condition_date >= as.Date("2007-01-01") & condition_date <= as.Date("2019-01-01"))
freq_table <- Imp_date %>%
  count(code, description)%>%
  arrange(n)
print(freq_table)

##      code      description      n
## 1 I4892    Unspecified atrial flutter  57
## 2 I482     Chronic atrial fibrillation 114
## 3 I480     Paroxysmal atrial fibrillation 228
## 4 I481     Persistent Atrial Fibrillation 376
```

```
## 5 I4891 Unspecified atrial fibrillation 627
```

Convert medication\_name to uppercase for consistent comparison by using toupper Frequency table for medication Names and frequency table for cohort

```
library(dplyr)
library(stringi)
library(data.table)

##
## Attaching package: 'data.table'

## The following objects are masked from 'package:lubridate':
##
##   hour, isoweek, mday, minute, month, quarter, second, wday, week,
##   yday, year

## The following objects are masked from 'package:dplyr':
##
##   between, first, last

## The following object is masked from 'package:purrr':
##
##   transpose

medication <- medication %>%
  mutate(
    Cohort = case_when(
      ndc %in% c(3089421, 636297747) | grepl("APIXABAN", toupper(medication_name)) ~ "NOAC",
      ndc %in% c(5970108) | grepl("DABIGATRAN", toupper(medication_name)) ~ "NOAC",
      ndc %in% c(50458577) | grepl("RIVAROXABAN", toupper(medication_name)) ~ "NOAC",
      ndc %in% c(31722327) | grepl("WARFARIN", toupper(medication_name)) ~ "Warfarin",
      ndc %in% c(2802100) | grepl("ASPIRIN", toupper(medication_name)) ~ "Aspirin",
      TRUE ~ NA_character_
    ),
    CohortN = case_when(
      Cohort == "NOAC" ~ 1,
      Cohort == "Warfarin" ~ 2,
      Cohort == "Aspirin" ~ 3,
      TRUE ~ NA_integer_
    )
  )

medication$request_date <- as.Date(medication$request_date, format="%m/%d/%Y")
medication_filtered <- medication %>%
  filter(request_date >= '2017-01-01' & request_date <= '2021-01-01')

freq_cohorts <- medication_filtered %>%
  count(Cohort) %>%
  arrange(desc(n))

freq_med_names <- medication_filtered %>%
  count(medication_name) %>%
  arrange(desc(n))
```

```

medication_unique <- medication_filtered %>%
  distinct(patient_id, Cohort, medication_name, request_date, .keep_all = TRUE)

library(dplyr)
library(lubridate)

library(dplyr)
library(lubridate)

patient <- patient %>%
  mutate(birth_date = as.Date(birth_date, format = "%m/%d/%Y"), death_date = as.Date(death_date, format = "%m/%d/%Y"))

excluded_ids_based_on_condition <- condition %>%
  filter(substr(code, 1, 3) %in% c("M81", "I97")) %>%
  distinct(patient_id) %>%
  pull(patient_id)

excluded_ids_based_on_procedure <- procedure %>%
  filter(code %in% c("B215YZZ", "B2151ZZ", "B2151ZZ")) %>%
  distinct(patient_id) %>%
  pull(patient_id)

cohort_1p <- patient %>%
  inner_join(medication_filtered, by = "patient_id") %>%
  mutate(age = interval(birth_date, request_date) / dyears(1)) %>%
  filter(age > 18,
         !patient_id %in% excluded_ids_based_on_condition,
         !patient_id %in% excluded_ids_based_on_procedure) %>%
  select(patient_id, gender, death_date, death_flag, race, Cohort, CohortN, request_date, birth_date, age)

cohort_1r <- cohort_1p %>%
  mutate(gender = tools::toTitleCase(gender),
         race = tools::toTitleCase(race),
         Index_Date = request_date,
         cohort1n = ifelse(CohortN == 1, 1, 2))

cohort_freq_m <- cohort_1r %>%
  count(Cohort) %>%
  arrange(desc(n))

library(dplyr)
library(stringr)

chf_condition <- condition %>%
  filter(str_sub(code, 1, 3) %in% c("I50")) %>%
  distinct(patient_id) %>%
  mutate(CHF = 1)

hyp_condition <- condition %>%
  filter(str_sub(code, 1, 3) %in% c("I10", "I11", "I12", "I13", "I14", "I15")) %>%
  distinct(patient_id) %>%

```



```

    filter(grepl("Omeprazole|Pantoprazole|Lansoprazole|Rabeprazole|Esomeprazole|Dexlansoprazole", medication_name, ignore.case = TRUE)) %>%
    distinct(patient_id) %>%
    mutate(PPI = 1)

H2Anta_medication <- medication %>%
  filter(grepl("Cimetidine|Ranitidine|Famotidine|Nizatidine|Roxatidine|Lafutidine", medication_name, ignore.case = TRUE)) %>%
  distinct(patient_id) %>%
  mutate(H2Anta= 1)

AntiArr_medication<- medication %>%
  filter(grepl("Quinidine|Procainamide|Mexiletine|Propafenone|Flecainide|Amiodarone|Bretylium|Dronedarone", medication_name, ignore.case = TRUE)) %>%
  distinct(patient_id) %>%
  mutate(AntiArr= 1)

digi_medication <- medication %>%
  filter(grepl("Digoxin", medication_name, ignore.case = TRUE)) %>%
  distinct(patient_id) %>%
  mutate(Digi= 1)

Statin_medication <- medication %>%
  filter(grepl("Atorvastatin|Fluvastatin|Lovastatin|Pitavastatin|Pravastatin|Roxuvastatin|Simvastatin", medication_name, ignore.case = TRUE)) %>%
  distinct(patient_id) %>%
  mutate(Statin= 1)

Cohort_2 <- cohort_1r %>%
  left_join(chf_condition, by = "patient_id") %>%
  left_join(hyp_condition, by = "patient_id") %>%
  left_join(Diab_condition, by = "patient_id") %>%
  left_join(Strok_condition, by = "patient_id") %>%
  left_join(Vsc_condition, by = "patient_id") %>%
  left_join(AbRenal_condition , by = "patient_id") %>%
  left_join(AbLiver_condition, by = "patient_id") %>%
  left_join(bleed_condition, by = "patient_id") %>%
  left_join(Alcoh_condition, by = "patient_id") %>%
  left_join(nsaid_medication, by= "patient_id") %>%
  left_join(antiplat_medication, by = "patient_id") %>%
  left_join(ppi_medication,by="patient_id") %>%
  left_join(H2Anta_medication,by= "patient_id") %>%
  left_join(AntiArr_medication,by= "patient_id") %>%
  left_join(digi_medication,by= "patient_id") %>%
  left_join(Statin_medication,by="patient_id")

Cohort_2 <- Cohort_2 %>%
  arrange(patient_id)

Cohort_3 <- Cohort_2 %>%
  mutate(
    Gender = tools::toTitleCase(gender),
    female = as.integer(toupper(gender) == "FEMALE"),
    Age1 = case_when(
      age >= 65 & age < 75 ~ 1,
      age >= 75 ~ 2,

```

```

    TRUE ~ 0
  ),
  Age2 = as.integer(age >= 75),
  AgeCat = case_when(
    age >= 65 & age < 75 ~ "65=< to 75",
    age >= 75 ~ "75<",
    TRUE ~ "<65"
  ),
  new_strok = if_else(STROK == 1, 2, STROK),
  CHA2DS2 = Age1 + female +
    ifelse(is.na(CHF), 0, CHF) +
    ifelse(is.na(HYP), 0, HYP) +
    ifelse(is.na(DIAB), 0, DIAB) +
    new_strok +
    ifelse(is.na(VSC), 0, VSC),
  drugtherapy = as.integer(AntiPlat == 1 | NSAID == 1),
  HASBLED = ifelse(is.na(HYP), 0, HYP) +
    ifelse(is.na(AbRenal), 0, AbRenal) +
    ifelse(is.na(AbLiver), 0, AbLiver) +
    ifelse(is.na(bleed), 0, bleed) +
    ifelse(is.na(STROK), 0, STROK) +
    ifelse(is.na(Alcoh), 0, Alcoh) +
    drugtherapy + Age2,
  Year = if_else(!is.na(Index_Date), as.integer(format(as.Date(Index_Date, format = "%Y-%m-%d"), "%Y")),
) %>%
select(
  patient_id, Index_Date, Gender, birth_date, death_date, age, death_flag,
  STROK, bleed, race, Cohort, AgeCat, CohortN, CHA2DS2, cohortln, HASBLED, Year
)

```

#### #EFFECTIVENESS IN REDUCING STROKE AND BLEEDING

```

freq1 <- Cohort_3 %>%
  count(Cohort, STROK) %>%
  arrange(Cohort, STROK)

freq2 <- Cohort_3 %>%
  count(Cohort, bleed) %>%
  arrange(Cohort, bleed)

```

#### #Treatment Pattern And Drug Profile

```

Med_1 <- medication %>%
  filter(patient_id %in% cohort_1p$patient_id)

Med_2 <- Med_1 %>%
  filter(request_date >= as.Date('2017-01-01') & request_date <= as.Date('2021-01-01')) %>%
  mutate(
    Category = case_when(
      grepl('Bromfenac|Celecoxib|Diclofenac|Etodolac|Fenoprofen|Flurbiprofen|Ibuprofen|Indomethacin|Ketorolac', medication_name) ~ 'NSAID',
      grepl('Aspirin|Clopidogrel|Prasugrel|Ticlopidine|Cilostazol|Abciximab|Tirofiban|Dipyridamole|Ticagrelor', medication_name) ~ 'Antiplatelet',
      grepl('Omeprazole|Pantoprazole|Lansoprazole|Rabeprazole|Esomeprazole|Dexlansoprazole', medication_name) ~ 'PPI',
      grepl('Cimetidine|Ranitidine|Famotidine|Nizatidine|Roxatidine|Lafutidine', medication_name, ignore.case = TRUE) ~ 'H2 blocker',
      grepl('Quinidine|Procainamide|Mexiletine|Propafenone|Flecainide|Amiodarone|Bretylium|Dronedarone', medication_name) ~ 'Antiarrhythmic'
    )
  )

```

```

    grepl('Digoxin', medication_name, ignore.case = TRUE) ~ "Digoxin",
    grepl('Atorvastatin|Fluvastatin|Lovastatin|Pitavastatin|Pravastatin|Rosuvastatin|Simvastatin', medication_name,
    TRUE ~ NA_character_
  )
)

medication_summary <- Med_2 %>%
  group_by(patient_id) %>%
  summarise(
    Num_Presc = n(),
    Num_Cat = n_distinct(Category)
  )

trt_pattern <- Cohort_3 %>%
  select(patient_id, Cohort, CohortN, cohortln) %>%
  left_join(medication_summary, by = "patient_id") %>%
  replace_na(list(Num_Presc = 0, Num_Cat = 0))

#HRU Table
encounter_summary <- encounter %>%
  mutate(encounter_start_date = mdy_hm(encounter_start_date)) %>%
  filter(encounter_start_date >= as.Date('2017-01-01') & encounter_start_date <= as.Date('2022-11-30'))
  group_by(patient_id) %>%
  summarise(
    encounter_count = n(),
    first_encounter = min(encounter_start_date),
    last_encounter = max(encounter_start_date)
  )

encounter_1 <- encounter %>%
  select(patient_id, encounter_type)

HRU <- Cohort_3 %>%
  inner_join(encounter_1, by = "patient_id") %>%
  select(patient_id, encounter_type, Cohort, CohortN, cohortln)

## Warning in inner_join(., encounter_1, by = "patient_id"): Detected an unexpected many-to-many relationship
## i Row 9 of `x` matches multiple rows in `y`.
## i Row 70 of `y` matches multiple rows in `x`.
## i If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.

#Overall Survival Time to Event
condition <- condition %>% mutate(last_date = ymd(condition_date))
lab <- lab %>% mutate(last_date = mdy_hm(result_date))
medication <- medication %>% mutate(last_date = ymd(request_date))
procedure <- procedure %>% mutate(last_date = mdy_hm(procedure_date))
encounter <- encounter %>% mutate(last_date = mdy(encounter_start_date))
vital_sign <- vital_sign %>% mutate(last_date = mdy(vital_date))
patient <- patient %>% mutate(last_date = ymd(birth_date))

```

```

latest_dates <- bind_rows(
  select(condition, patient_id, last_date),
  select(lab, patient_id, last_date),
  select(medication, patient_id, last_date),
  select(procedure, patient_id, last_date),
  select(encounter, patient_id, last_date),
  select(vital_sign, patient_id, last_date),
  select(patient, patient_id, last_date)
) %>%
  filter(patient_id %in% Cohort_3$patient_id) %>%
  group_by(patient_id) %>%
  summarize>Last_Followup = max(last_date, na.rm = TRUE))

OS <- merge(latest_dates, Cohort_3, by = "patient_id", all.x = TRUE)
OS <- OS %>%
  mutate(
    start_date = as.Date(Index_Date),
    ADT = if_else(!is.na(death_date), death_date, Last_Followup),
    Event = if_else(!is.na(as.Date(death_date)), 1, 0),
    CNSR = if_else(is.na(as.Date(death_date)), 1, 0),
    EVNTDESC = case_when(
      !is.na(death_date) ~ "Death",
      is.na(death_date) & !is.na(Last_Followup) ~ "No Event: Censored at Last Activity Date",
      is.na(death_date) & is.na(Last_Followup) ~ "No Event: Censored at Index Date",
      TRUE ~ NA_character_
    ),
    AVAL = (as.numeric(ADT-start_date) + 1) / (365.25 / 12)
  ) %>%
  select(patient_id, AVAL, CNSR, EVNTDESC, ADT, start_date, Last_Followup, Cohort, CohortN)

#vital sign change in CV parameters

vsl <- vital_sign %>%
  left_join(Cohort_3, by = "patient_id") %>%
  mutate(

    index_date = as.Date(Index_Date, format="%m/%d/%Y"),
    death_date = as.Date(death_date),
    vital_date = as.Date(vital_date, format="%m/%d/%Y")
  ) %>%
  filter(
    patient_id %in% Cohort_3$patient_id,
    vital_date > (index_date - days(30)),
    loinc %in% c('8462-4', '8480-6', '8867-4')
  ) %>%
  select(names(vital_sign), "index_date", "Cohort", "CohortN", "cohortln", "death_date") %>%
  arrange(patient_id, loinc, vital_date)

Loinc_codes <- c("8462-4", "8480-6", "8867-4")

vs2_base <- vsl %>%
  filter(vital_date > index_date - days(30) & vital_date < index_date + days(30),
    loinc %in% Loinc_codes,

```



```

      !is.na(value)) %>%
    arrange(patient_id, loinc, vital_date)

vs2_basel <- vs2_base %>%
  group_by(patient_id, loinc) %>%
  filter(row_number() == max(row_number())) %>%
  ungroup() %>%
  rename(Base = value)

vs_post <- vs1 %>%
  filter(vital_date > index_date & (!is.na(death_date) & vital_date < death_date | is.na(death_date)),
        loinc %in% Loinc_codes,
        !is.na(value)) %>%
  arrange(patient_id, loinc, vital_date)

vs_post_last <- vs_post %>%
  group_by(patient_id, loinc) %>%
  slice_max(order_by = vital_date) %>%
  ungroup()

```

## Merging baseline and post-baseline datasets.

```

vs_post_last <- vs_post_last %>%
  rename(Post_Base = value)

vital_sign_analysis <- merge(vs2_basel, vs_post_last, by = c("patient_id", "loinc"), all = TRUE)

vital_sign_analysis <- vital_sign_analysis %>%
  mutate(CHG=Post_Base-Base)

```