R Practice

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.1.2
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##      filter, lag
```

```
## The following objects are masked from 'package:base':
##
##      intersect, setdiff, setequal, union
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.1.2
```

```
library(broom)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##      date, intersect, setdiff, union
```

```
clean_beneficiary <-
  function(beneficiary_df) {

    beneficiary_clean <-
      beneficiary_df %>%
      mutate(birth_dt = as.Date(x = as.character(BENE_BIRTH_DT), format = '%Y%m%d'),
             death_dt = as.Date(x = as.character(BENE_DEATH_DT),  format = '%Y%m%d'),
             death180 = death_dt - 180
             ) %>%
      select(member_id = DESYNPUF_ID,
             birth_dt,
             death_dt,
             sex = BENE_SEX_IDENT_CD,
```

```r
            race = BENE_RACE_CD,
            esrd = BENE_ESRD_IND,
            state_code = SP_STATE_CODE,
            death180)

    return(beneficiary_clean)
  }

ben_08 <- read.csv(file = "data/08_Beneficiary.csv") %>% clean_beneficiary()
ben_09 <- read.csv(file = "data/09_Beneficiary.csv") %>% clean_beneficiary()
ben_10 <- read.csv(file = "data/10_Beneficiary.csv") %>% clean_beneficiary()
```

```r
inpatient_claim <-
  read.csv("data/Inpatient_claims.csv") %>%
  mutate(
    in_claim_from_dt = as.Date(x = as.character(CLM_FROM_DT), format = '%Y%m%d'),
    in_claim_thru_dt = as.Date(x = as.character(CLM_THRU_DT), format = '%Y%m%d'),
    in_claim_admn_dt = as.Date(x = as.character(CLM_ADMSN_DT), format = '%Y%m%d'),
    in_claim_thru_yr = year(in_claim_thru_dt)
  ) %>%
  select(
    member_id = DESYNPUF_ID,
    in_claim_id = CLM_ID,
    in_claim_from_dt,
    in_claim_thru_dt,
    in_claim_thru_yr,
    in_claim_payment = CLM_PMT_AMT,
    in_claim_admn_dt
  )
```

```r
outpatient_claim <-
  read.csv("data/Outpatient_claims.csv") %>%
  mutate(
    out_claim_from_dt = as.Date(x = as.character(CLM_FROM_DT), format = '%Y%m%d'),
    out_claim_thru_dt = as.Date(x = as.character(CLM_THRU_DT), format = '%Y%m%d'),
    out_claim_thru_yr = year(out_claim_thru_dt)
  ) %>%
  select(
    member_id = DESYNPUF_ID,
    out_claim_id = CLM_ID,
    out_claim_payment = CLM_PMT_AMT,
    out_claim_from_dt,
    out_claim_thru_dt,
    out_claim_thru_yr
  )
```

```r
rx_claim <-
  read.csv("data/Rx_claims.csv") %>%
  mutate(service_dt = as.Date(x = as.character(SRVC_DT), format = '%Y%m%d'),
         service_yr = year(service_dt)) %>%
  select(
```

```r
    member_id = DESYNPUF_ID,
    rx_id = PDE_ID,
    service_dt,
    service_yr,
    patient_amt = PTNT_PAY_AMT,
    total_rx_amt = TOT_RX_CST_AMT)


# Which state spends the most, and which state spends the least per ESRD patient (Combining
# inpatient, outpatient and RX drugs). Answer separately for 2008, 2009 AND 2010.

claim_joiner <-
  function(beneficiary_df,
           year) {
    #' @Description: Takes a year and finds that spends the most / least per ESRD patient
    #' @param beneficiary_df: A data-frame to be used for joining
    #' @param year: A string, indicating year (2008, 2009, 2010)

    rx_year <-
      rx_claim %>%
      filter(service_yr == year) %>%
      group_by(member_id) %>%
      summarize(rx_amt = sum(total_rx_amt))

    inpatient_year <-
      inpatient_claim %>%
      filter(in_claim_thru_yr == year) %>%
      group_by(member_id) %>%
      summarize(inpatient_amt = sum(in_claim_payment))

    outpatient_year <-
      outpatient_claim %>%
      filter(out_claim_thru_yr == year) %>%
      group_by(member_id) %>%
      summarize(outpatient_amt = sum(out_claim_payment))

    beneficiary_amt <-
      beneficiary_df %>%
      left_join(y = rx_year,
                by = "member_id") %>%
      left_join(y = inpatient_year,
                by = "member_id") %>%
      left_join(y = outpatient_year,
                by = "member_id") %>%
      rowwise() %>%
      mutate(total_patient_amt = sum(rx_amt, inpatient_amt, outpatient_amt,
                                     na.rm = TRUE))
    return(beneficiary_amt)
  }

state_spender <-
  function(beneficiary_df) {

    beneficiary_state <-
```

```r
    beneficiary_df %>%
      filter(esrd == "Y") %>%
      group_by(state_code) %>%
      summarize(avg_spend = mean(total_patient_amt)) %>%
      arrange(desc(avg_spend))

  high_spender <-
    beneficiary_state %>% slice_head()

  low_spender <-
    beneficiary_state %>% slice_tail()

  state_spenders <- high_spender %>% bind_rows(low_spender)

  return(state_spenders)
}

claim_joiner(beneficiary_df = ben_08, year = "2008") %>% state_spender()
```

```
## # A tibble: 2 x 2
##   state_code avg_spend
##        <int>     <dbl>
## 1          9    28018.
## 2         41     5554.
```

```r
claim_joiner(beneficiary_df = ben_09, year = "2009") %>% state_spender()
```

```
## # A tibble: 2 x 2
##   state_code avg_spend
##        <int>     <dbl>
## 1         28    16892.
## 2         53     4806.
```

```r
claim_joiner(beneficiary_df = ben_10, year = "2010") %>% state_spender()
```

```
## # A tibble: 2 x 2
##   state_code avg_spend
##        <int>     <dbl>
## 1         28    11021.
## 2          2     2143.
```

```r
# What is the average amount spent per ESRD patient on claims that were initiated in the final
# 180 days of life? (Combining inpatient, outpatient and RX drugs)

dead_esrd_08 <-
  ben_08 %>%
  filter(!is.na(death_dt),
         esrd == "Y")

dead_esrd_09 <-
  ben_09 %>%
```

```r
      filter(!is.na(death_dt),
             esrd == "Y")

dead_esrd_10 <-
  ben_10 %>%
  filter(!is.na(death_dt),
         esrd == "Y")

dead_esrd <-
  dead_esrd_08 %>%
  bind_rows(dead_esrd_09) %>%
  bind_rows(dead_esrd_10)

# death_finder <- r
spent_before_death <-
  function() {

    out_spent <-
      dead_esrd %>%
      right_join(y = outpatient_claim,
                 by = "member_id") %>%
      filter(out_claim_thru_dt >= death180 &
               out_claim_thru_dt <= death_dt) %>%
      group_by(member_id) %>%
      summarize(total_amt = sum(out_claim_payment))

    in_spent <-
      dead_esrd %>%
      right_join(y = inpatient_claim,
                 by = "member_id") %>%
      filter(in_claim_thru_dt >= death180 &
               in_claim_thru_dt <= death_dt) %>%
      group_by(member_id) %>%
      summarize(total_amt = sum(in_claim_payment))

    rx_spent <-
      dead_esrd %>%
      right_join(y = rx_claim,
                 by = "member_id") %>%
      filter(death180 <= service_dt) %>%
      group_by(member_id) %>%
      summarize(total_amt = sum(total_rx_amt))

    dead_spent <-
      out_spent %>%
      bind_rows(in_spent) %>%
      bind_rows(rx_spent) %>%
      group_by(member_id) %>%
      summarize(patient_amt = sum(total_amt))

    no_spenders <-
      dead_esrd %>%
      anti_join(y = dead_spent,
```

```
                 by = "member_id") %>%
      mutate(patient_amt = 0) %>%
      select(member_id,
             patient_amt)

    all_patient_amt <-
      dead_spent %>%
      bind_rows(no_spenders)

    return(all_patient_amt)
    }

# Ensure there are 436 - 387 = 49 patients that = 0 in the last dataset so the average is not offset
patients_spent_180 <- spent_before_death()

mean(patients_spent_180$patient_amt)
```

```
## [1] 5472.477
```

```
# Find the mean and use bootstrapping to manually generate a 95% confidence interval for
# members' age.

# 1. Confirm ben_08 has all unique members

ben_10 %>%
  anti_join(y = ben_08,
            by = "member_id")
```

```
## [1] member_id  birth_dt   death_dt   sex        race       esrd       state_code
## [8] death180
## <0 rows> (or 0-length row.names)
```

```
ben_09 %>%
  anti_join(y = ben_08,
            by = "member_id")
```

```
## [1] member_id  birth_dt   death_dt   sex        race       esrd       state_code
## [8] death180
## <0 rows> (or 0-length row.names)
```

```
# 2. Calculate Age

all_benefs <-
  ben_08 %>%
  mutate(age_08 =
           if_else(
             condition = !is.na(death_dt),
             true = as.numeric(death_dt - birth_dt) / 365.25 ,
             false = as.numeric(as.Date('2008-12-31', format = '%Y-%m-%d') - birth_dt) / 365.25
           ))
```

```r
# 3. Perform bootstrap
# set n = original size

mean_ages <- vector(mode = "numeric", length = 1000)

set.seed(seed = 1)
for (i in 1:length(mean_ages)) {
  boot_benef <- all_benefs %>% sample_n(116352, replace = TRUE)
  mean_ages[i] <- boot_benef %>% pull(age_08) %>% mean()

  }

quantile(mean_ages, probs = c(0.025, 0.5, .975))
```

```
##     2.5%      50%    97.5%
## 72.10885 72.17760 72.24648
```