ali_sim_params

Sarah Lotspeich

2024-02-26

```
## Read in summary data (one row per patient)
summ_data = read.csv("~/Documents/ali-data/summary_data.csv") |>
  ### Create indicator of >= 1 healthcare encounter (ED visit or hospital admission)
  dplyr::mutate(ANY_ENCOUNTERS = as.numeric(NUM_ENCOUNTERS > 0))
```

Distribution of age at encounter

```
## Model age at first encounter with a Poisson distribution
glm(formula = AGE_AT_ENCOUNTER ~ 1,
    family = poisson,
    data = summ_data) |>
  coefficients() |>
  exp()
```

```
## (Intercept)
##      45.662
```

Calculate error-prone ALI from EHR

```
## Calculate components of the error-prone version of ALI from EHR
summ_data = summ_data |>
  dplyr::mutate(CREAT_C_POINT = as.numeric(CREAT_C < 110 & SEX == "Male") +
                  as.numeric(CREAT_C < 100 & SEX == "Female"),
                ALB_POINT = as.numeric(ALB >= 3.5),
                BMI_POINT = as.numeric(BMI > 30),
                BP_SYSTOLIC_POINT = as.numeric(BP_SYSTOLIC > 140),
                BP_DIASTOLIC_POINT = as.numeric(BP_DIASTOLIC > 90),
                A1C_POINT = as.numeric(A1C >= 6.5),
                CHOL_POINT = as.numeric(CHOL >= 200),
                TRIG_POINT = as.numeric(TRIG >= 150),
                CRP_POINT = as.numeric(CRP >= 10),
                HCST_POINT = as.numeric(HCST > 50),
                U_ALBUMIN_CREAT_RATIO = ALB_U / CREAT_U,
                U_ALBUMIN_CREAT_RATIO_POINT = as.numeric(U_ALBUMIN_CREAT_RATIO < 30)
  )
## Define components of the Seemen et al. ALI
seemen_components = c("CREAT_C_POINT", "ALB_POINT", "BMI_POINT", "BP_SYSTOLIC_POINT",
```

```
                      "BP_DIASTOLIC_POINT", "A1C_POINT", "CHOL_POINT", "TRIG_POINT",
                      "CRP_POINT", "HCST_POINT")
### Number of non-missing components
ALI_DENOM = rowSums(!is.na(summ_data[, seemen_components]))
### Number of components equal to 1
ALI_NUM = rowSums(summ_data[, seemen_components], na.rm = TRUE)
summ_data$ALI_Seeman = ALI_NUM / ALI_DENOM
```

Naive model of ALI and healthcare utilization

```
## Naive model parameters
naive_mod = glm(formula = ANY_ENCOUNTERS ~ ALI_Seeman + AGE_AT_ENCOUNTER,
                family = "binomial",
                data = summ_data)
summary(naive_mod)
```

```
##
## Call:
## glm(formula = ANY_ENCOUNTERS ~ ALI_Seeman + AGE_AT_ENCOUNTER,
##     family = "binomial", data = summ_data)
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -1.567621   0.251493  -6.233 4.57e-10 ***
## ALI_Seeman        0.945232   0.341832   2.765  0.00569 **
## AGE_AT_ENCOUNTER  0.010266   0.005222   1.966  0.04928 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1250.7  on 999  degrees of freedom
## Residual deviance: 1234.7  on 997  degrees of freedom
## AIC: 1240.7
##
## Number of Fisher Scoring iterations: 4
```

Naive proportions "yes" per ALI component

```
## probability of stressor = YES
colMeans(summ_data[, seemen_components], na.rm = TRUE)
```

```
##      CREAT_C_POINT          ALB_POINT          BMI_POINT  BP_SYSTOLIC_POINT
##          0.2500000          0.9870130          0.4549098          0.1450000
## BP_DIASTOLIC_POINT          A1C_POINT         CHOL_POINT         TRIG_POINT
##          0.0580000          0.2490119          0.3138501          0.3316391
##          CRP_POINT         HCST_POINT
##          0.3111111          0.0000000
```

Percent missing data per ALI component

```
## probability of stressor = NA
colMeans(is.na(summ_data[, seemen_components]), na.rm = TRUE)
```

```
##      CREAT_C_POINT          ALB_POINT          BMI_POINT  BP_SYSTOLIC_POINT
##              0.996              0.153              0.002              0.000
## BP_DIASTOLIC_POINT          A1C_POINT         CHOL_POINT         TRIG_POINT
##              0.000              0.494              0.213              0.213
##          CRP_POINT        HCST_POINT
##              0.955              0.983
```