

Plots

Eduardo Yuki Yada

Imports

```
library(tidyverse)
library(yaml)
```

Loading data

```
load('../dataset/processed_data.RData')
load('../dataset/processed_dictionary.RData')

columns_list <- yaml.load_file("../auxiliar/columns_list.yaml")

outcome_column <- params$outcome_column
```

Plots

```
medianWithoutNA <- function(x) {
  median(x[which(!is.na(x))])
}

k = 6

df[columns_list$outcome_columns] <- lapply(df[columns_list$outcome_columns], as.numeric)

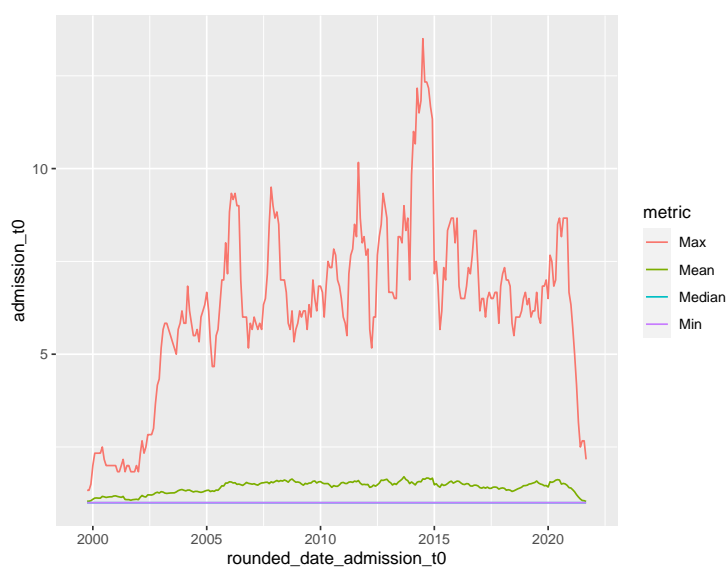
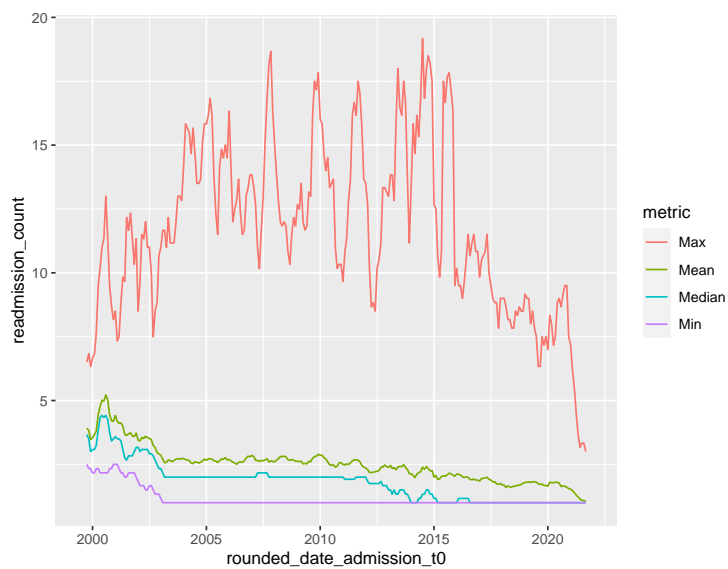
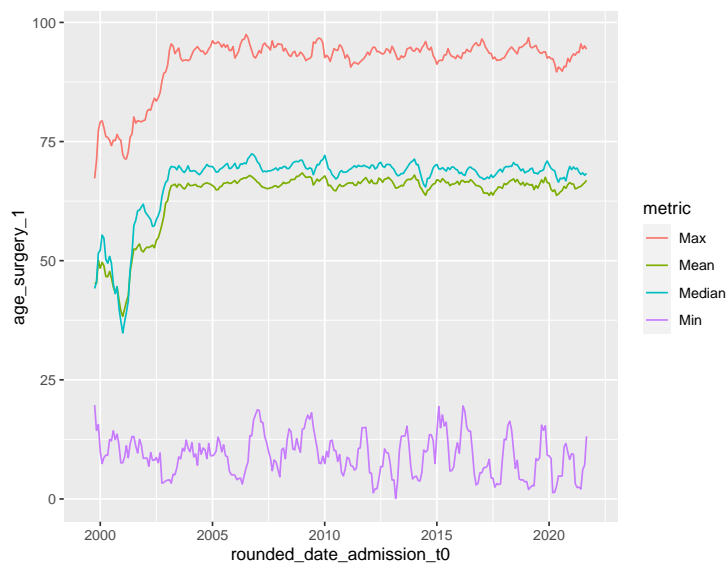
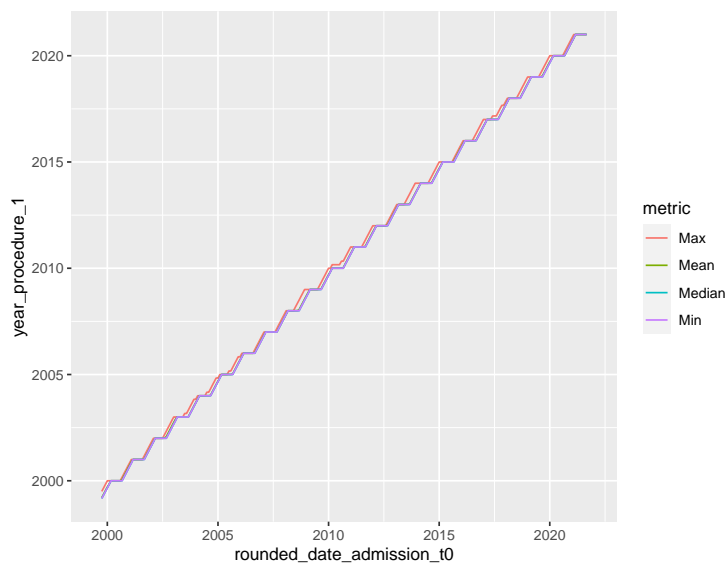
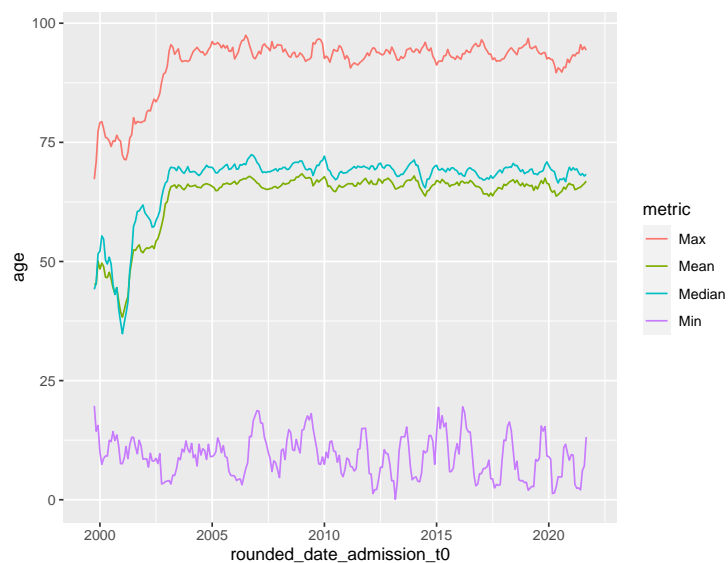
for (column in columns_list$numerical_columns){
  if (mean(is.na(df[[column]])) > 0.5) next

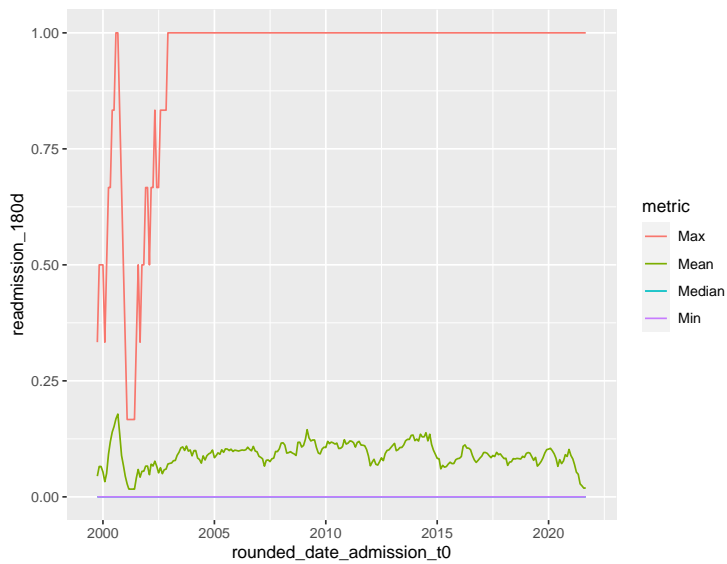
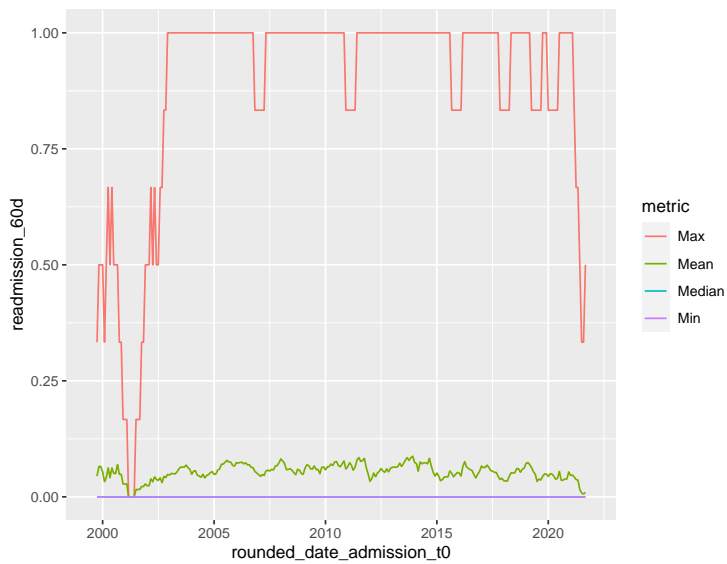
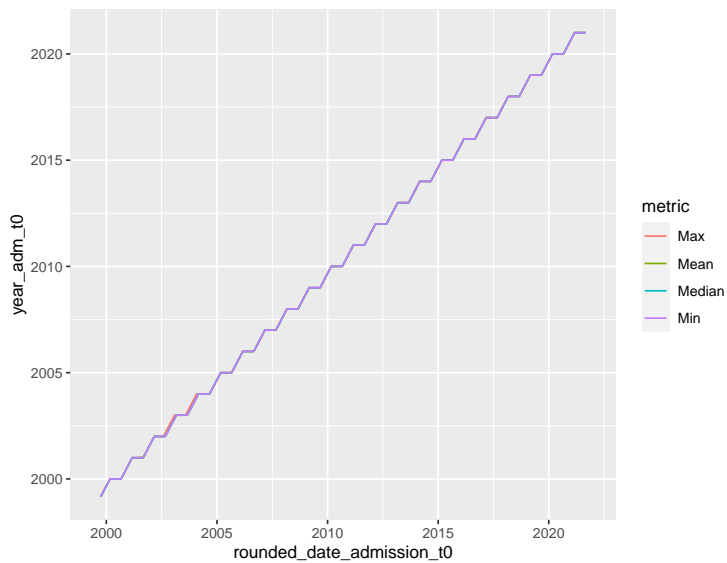
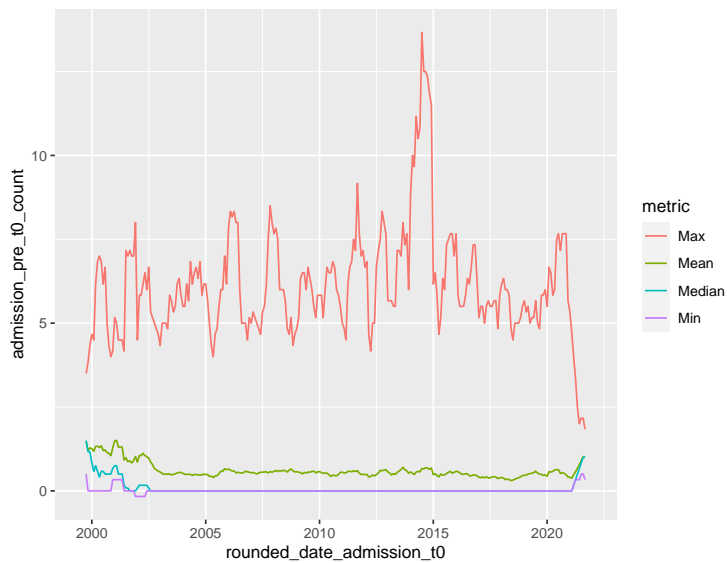
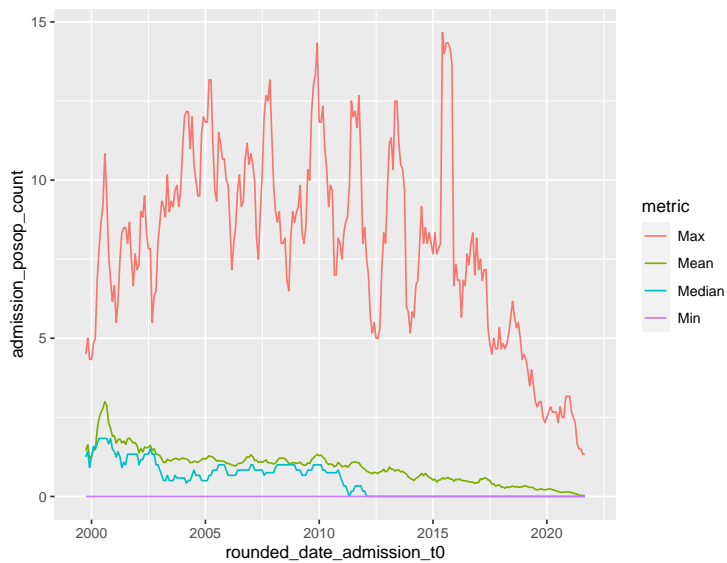
  df %>%
    mutate(rounded_date_admission_t0 = lubridate::floor_date(date_admission_t0,
                                                              'month')) %>%

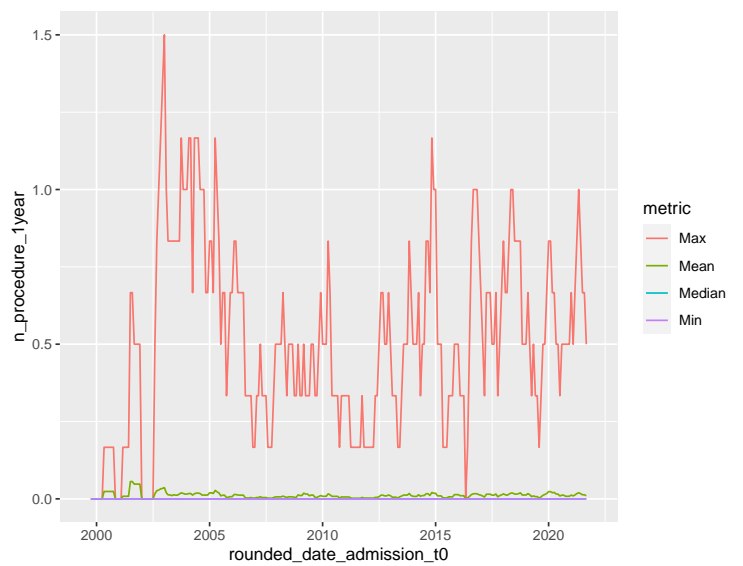
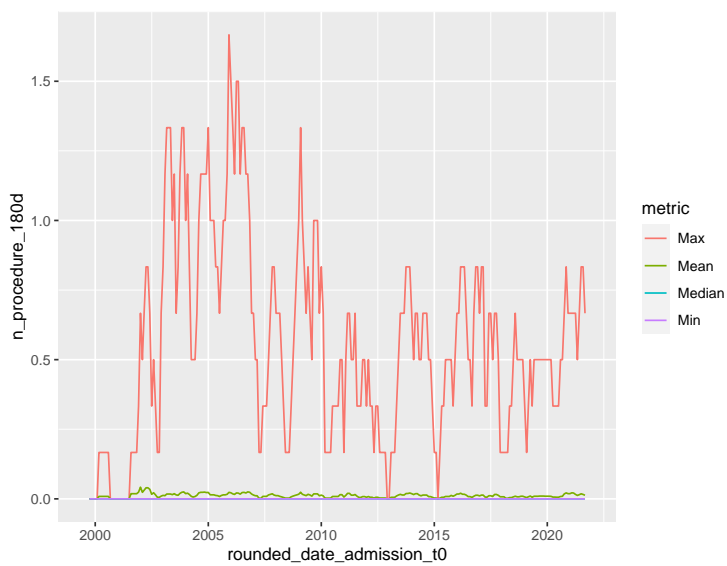
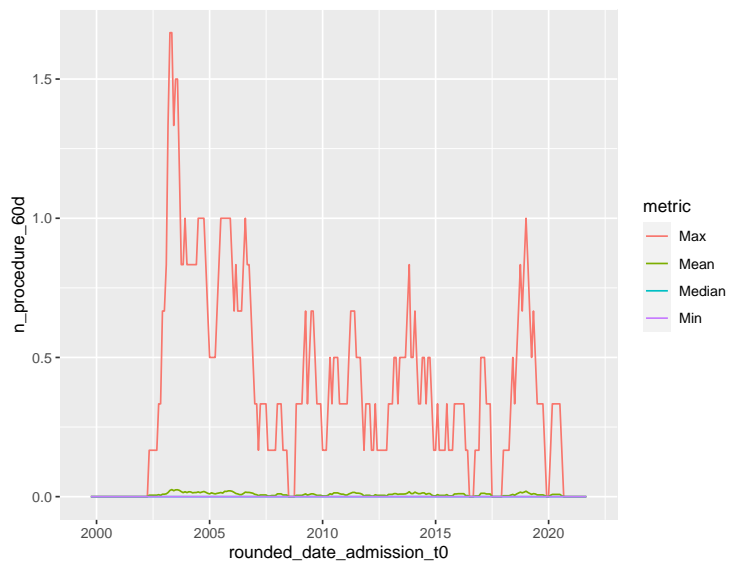
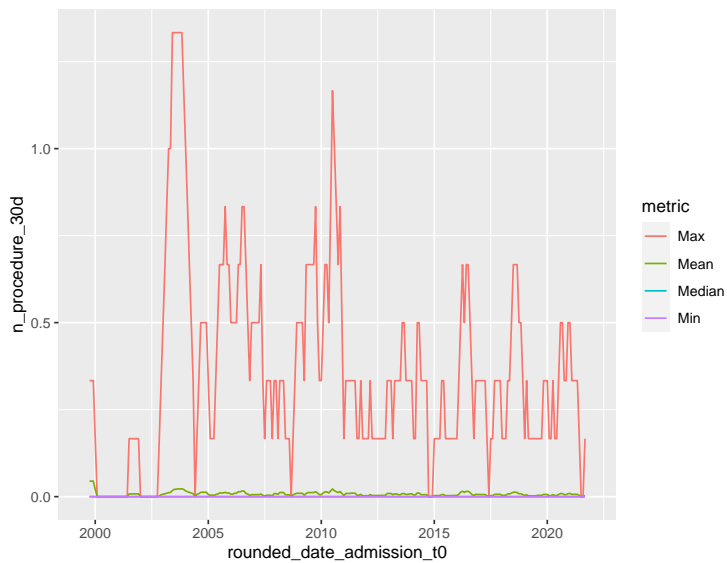
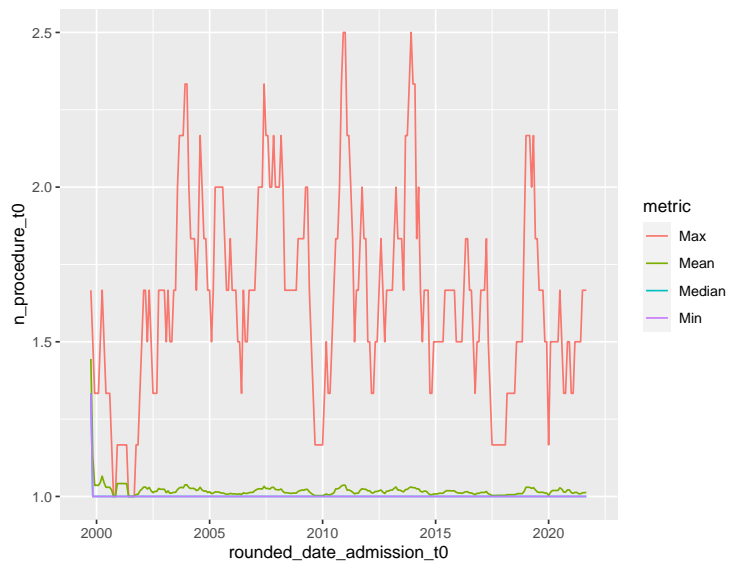
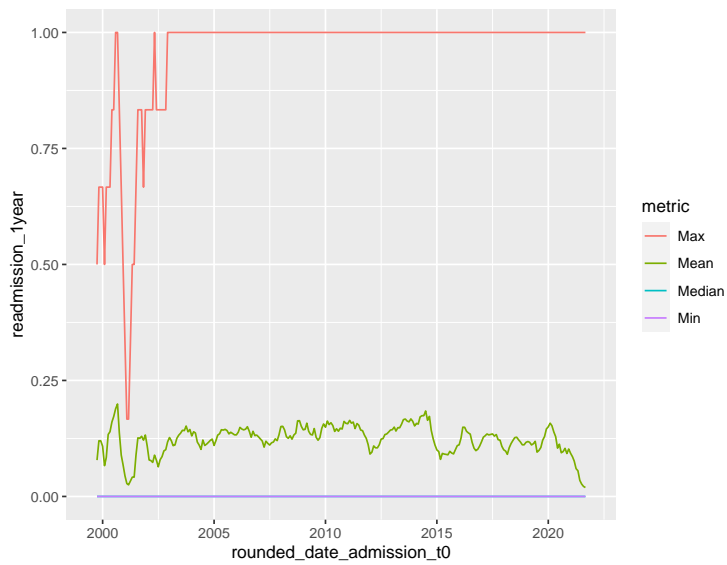
    group_by(rounded_date_admission_t0) %>%
    summarise('Mean' = mean(!sym(column), na.rm = T),
              'Min' = min(!sym(column), na.rm = T),
              'Median' = medianWithoutNA(!sym(column)),
              'Max' = max(!sym(column), na.rm = T)) %>%
    ungroup %>%
    arrange(rounded_date_admission_t0) %>%
    mutate(Mean = zoo::rollmean(Mean, k = k, fill = NA),
           Min = zoo::rollmean(Min, k = k, fill = NA),
           Median = zoo::rollmean(Median, k = k, fill = NA),
           Max = zoo::rollmean(Max, k = k, fill = NA)) %>%
    pivot_longer(!rounded_date_admission_t0, names_to = 'metric') %>%
    ggplot(aes(x = rounded_date_admission_t0, y = value, color = metric)) +
      geom_line() +
      geom_vline(xintercept = lubridate::ymd('2020-03-01'), linetype = "dashed") +
      labs(y = column)-> p

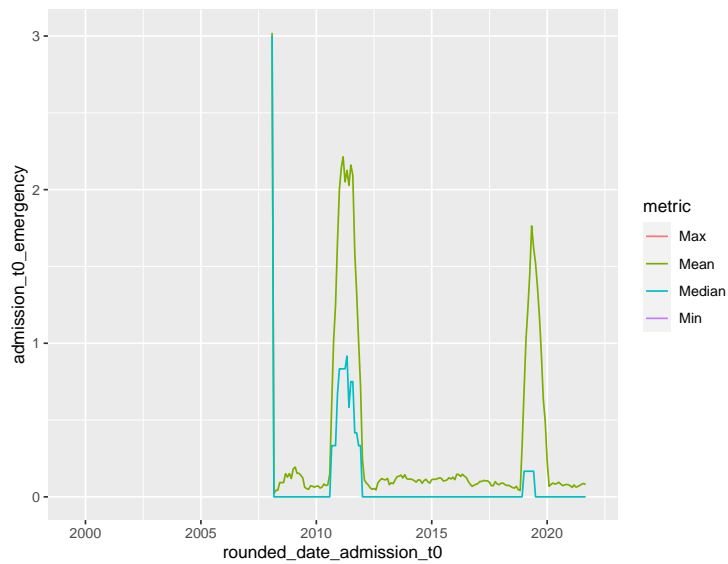
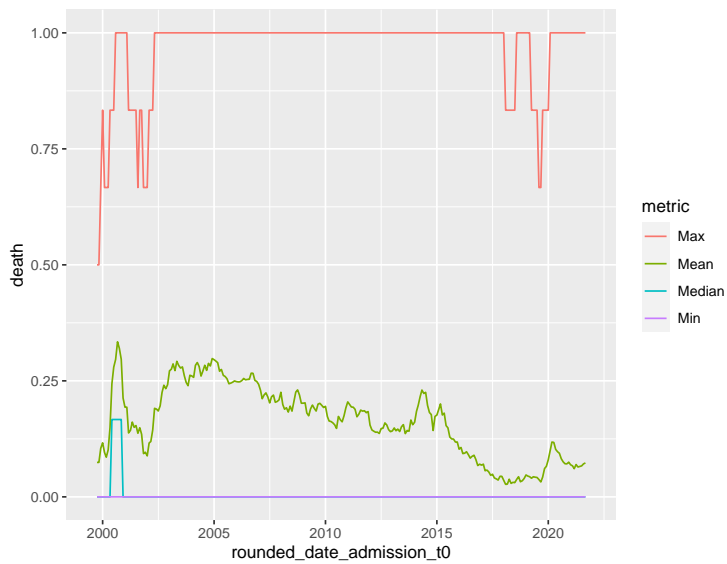
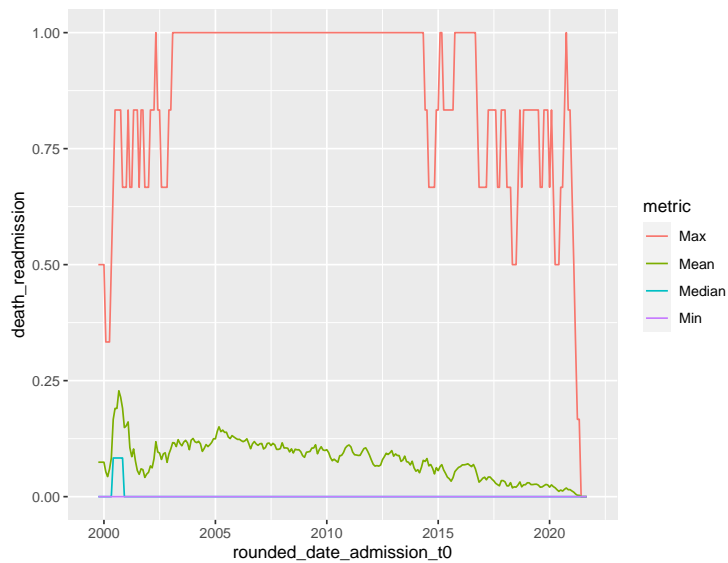
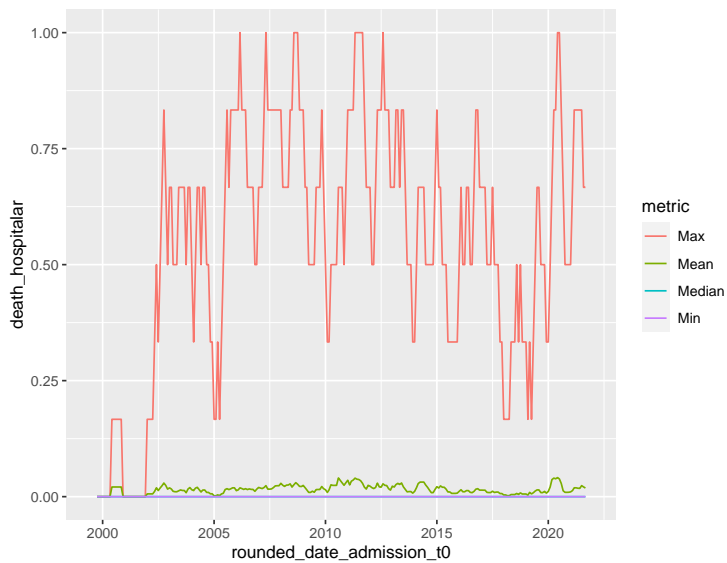
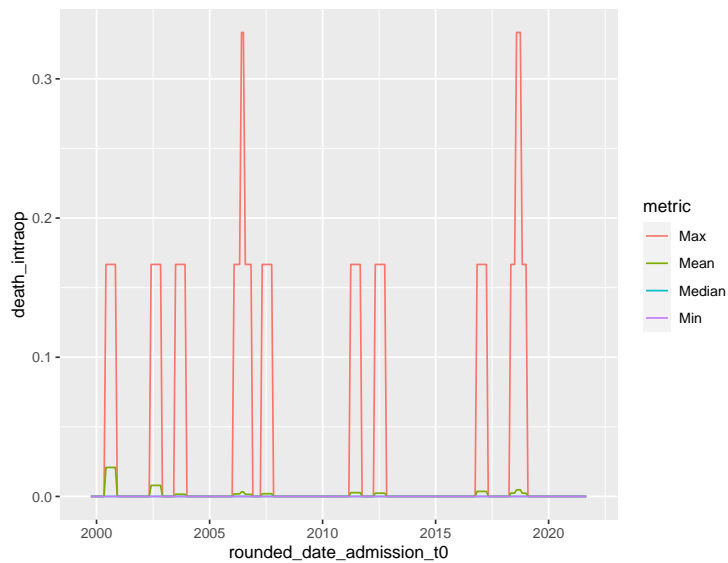
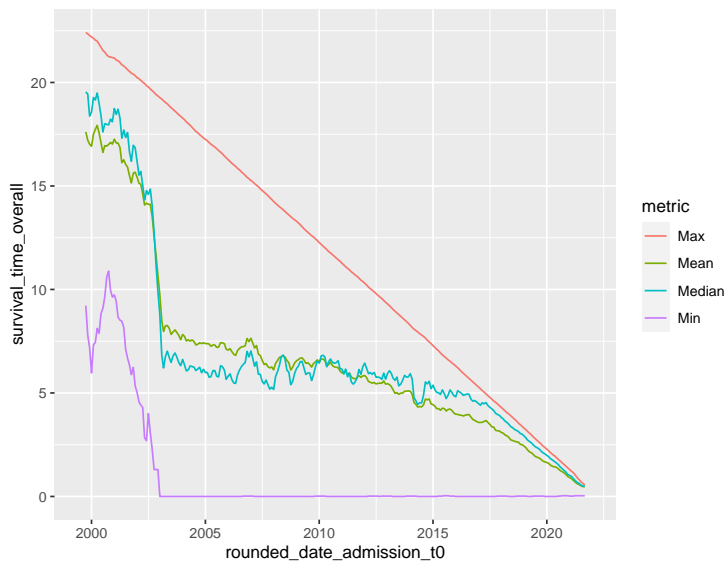
  print(p)
```

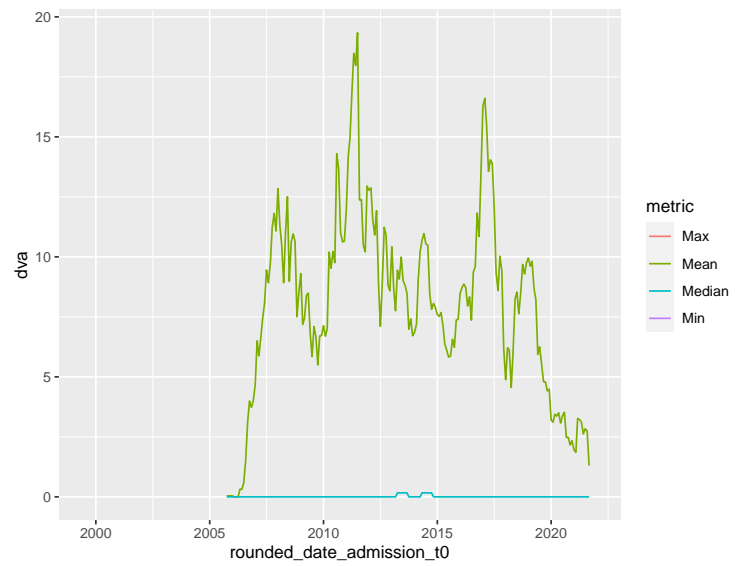
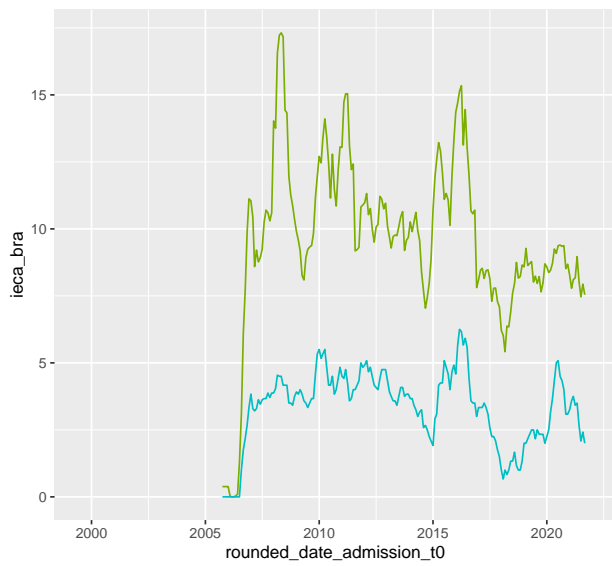
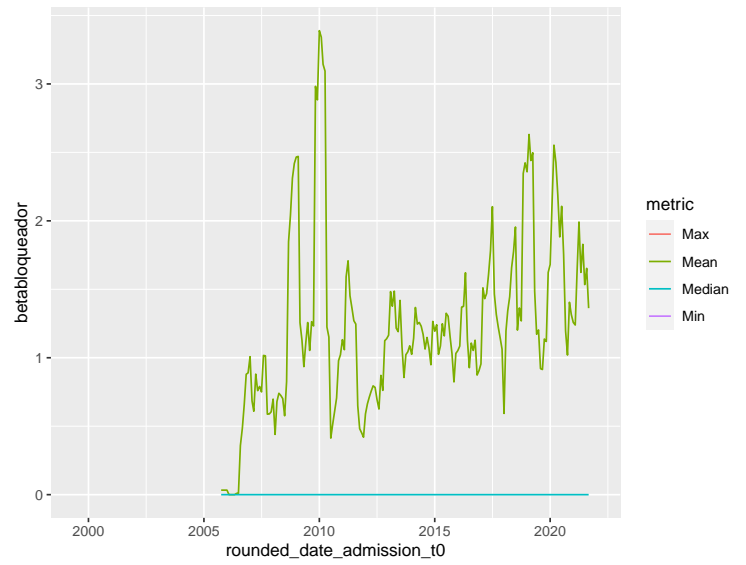
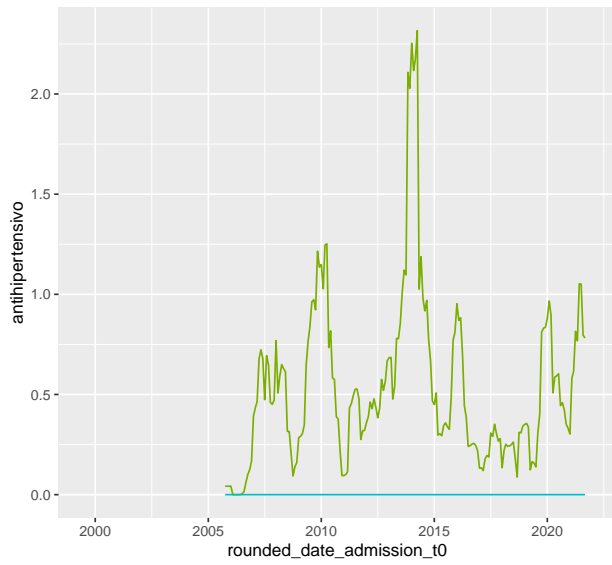
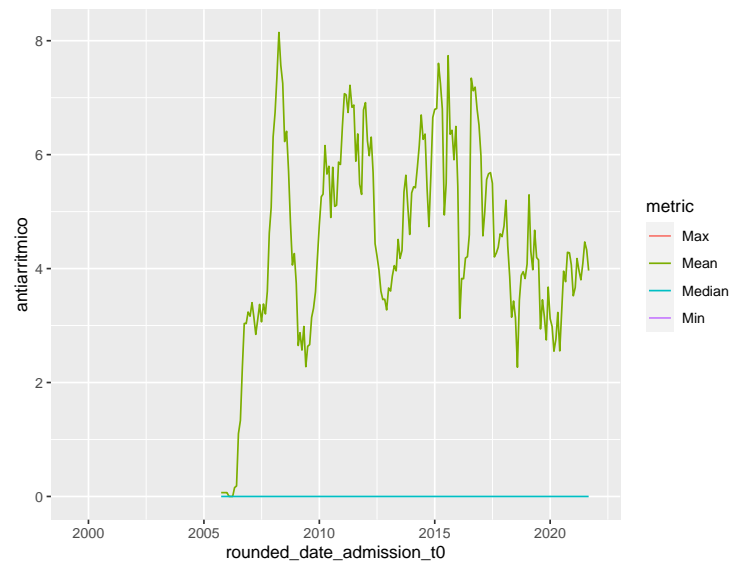
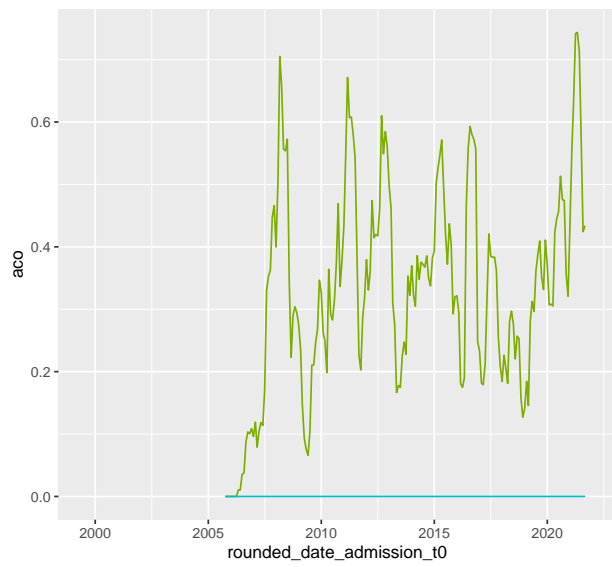
}

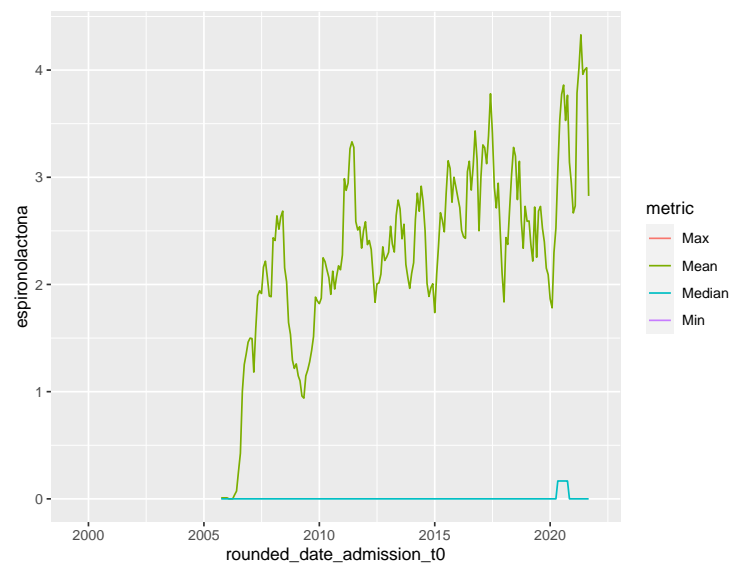
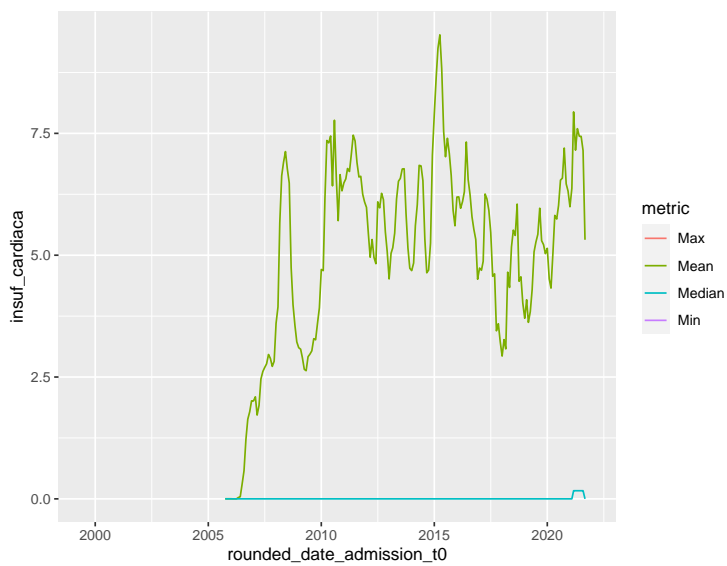
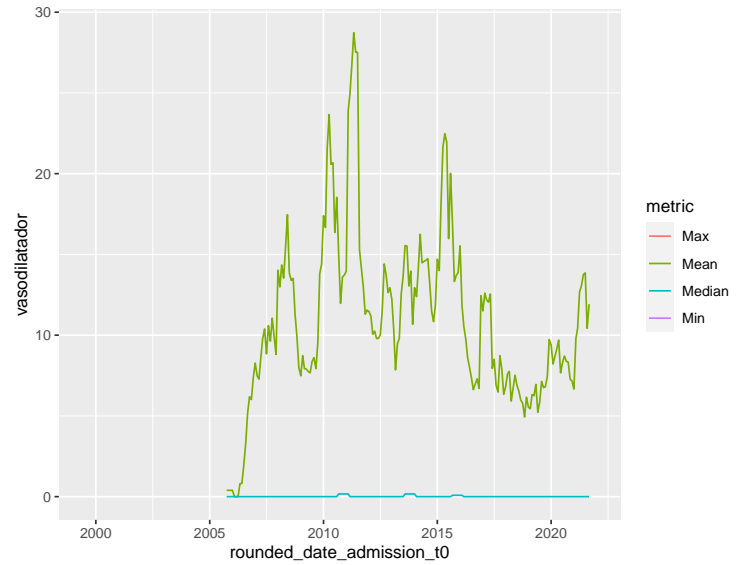
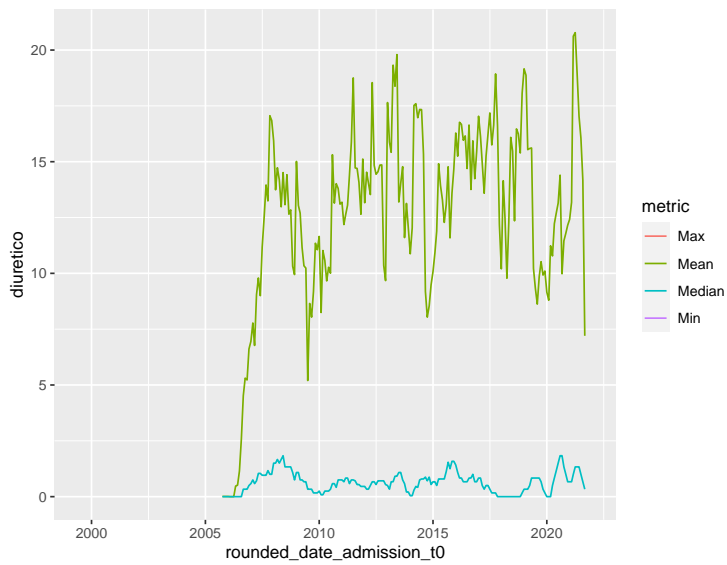
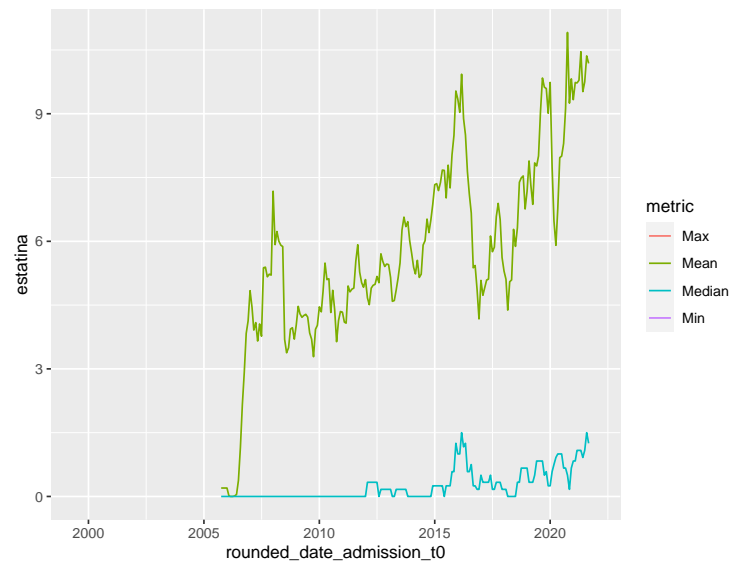
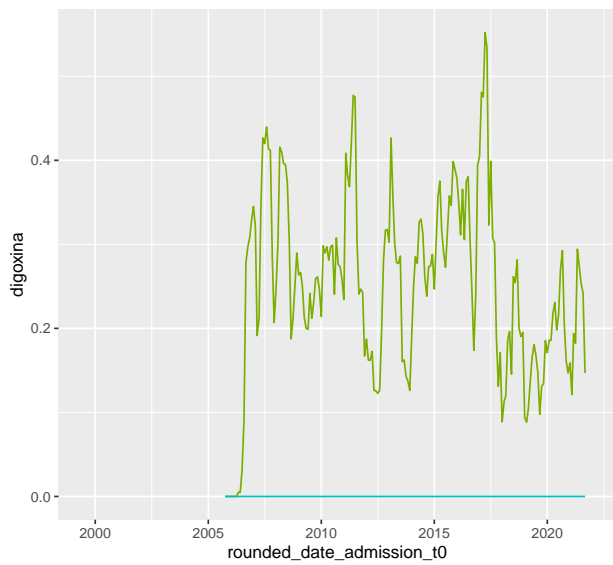


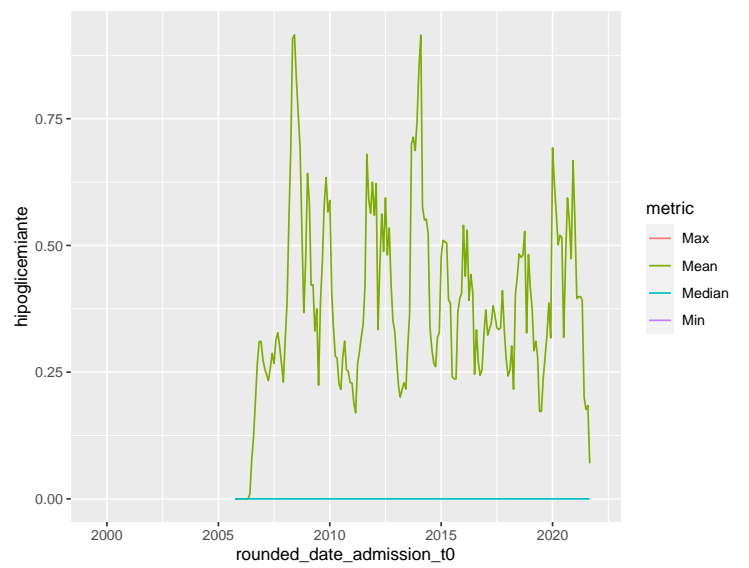
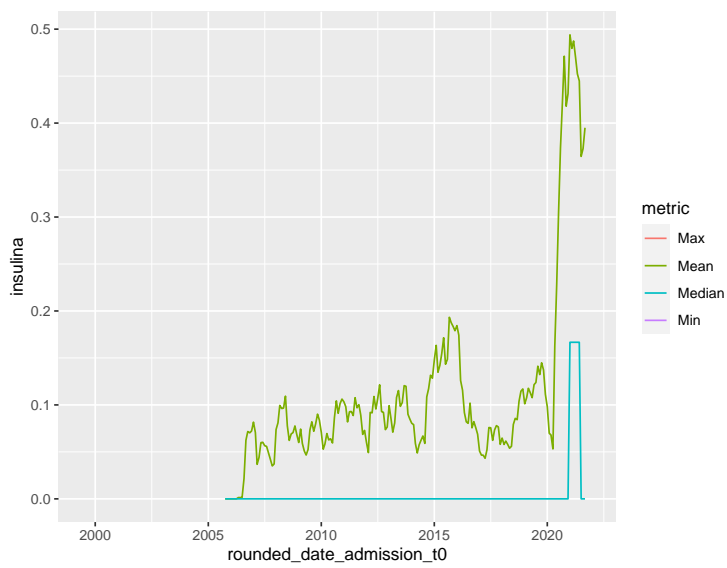
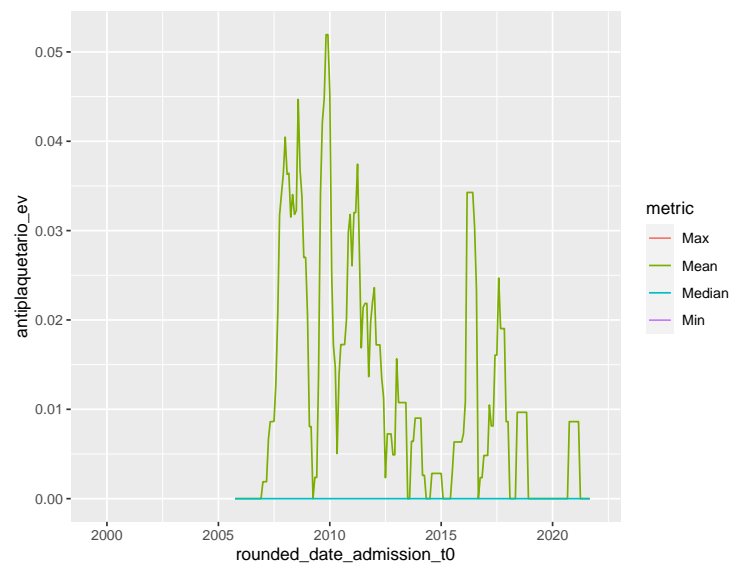
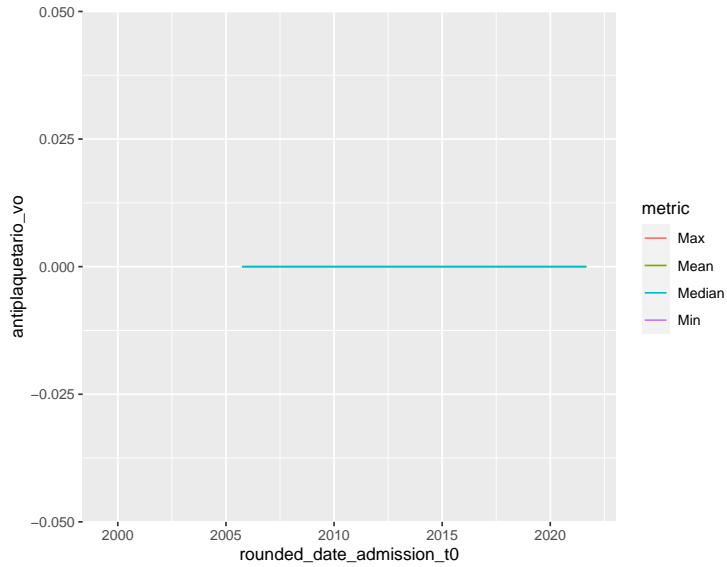
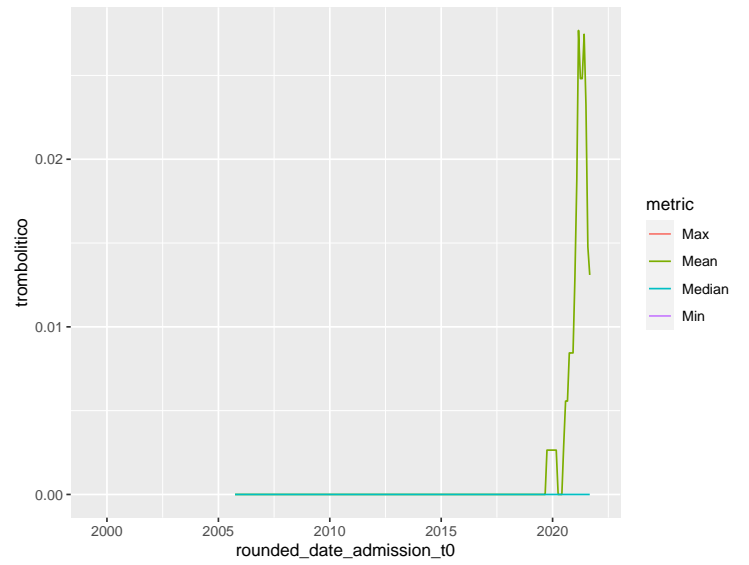
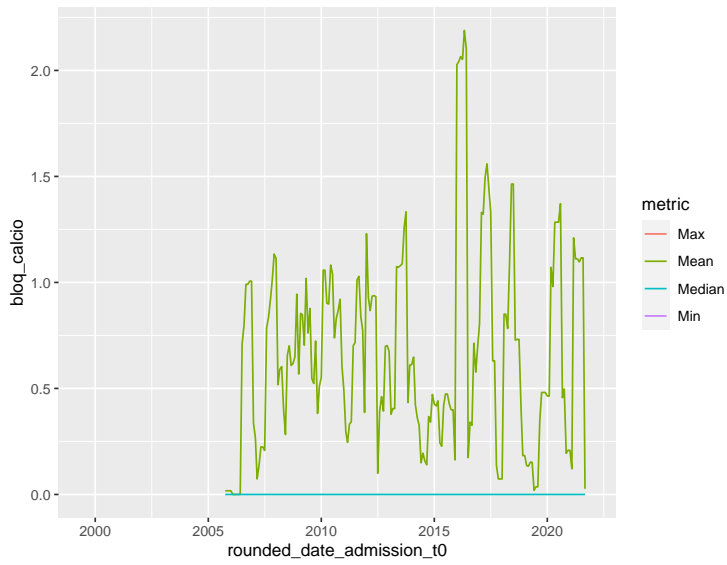


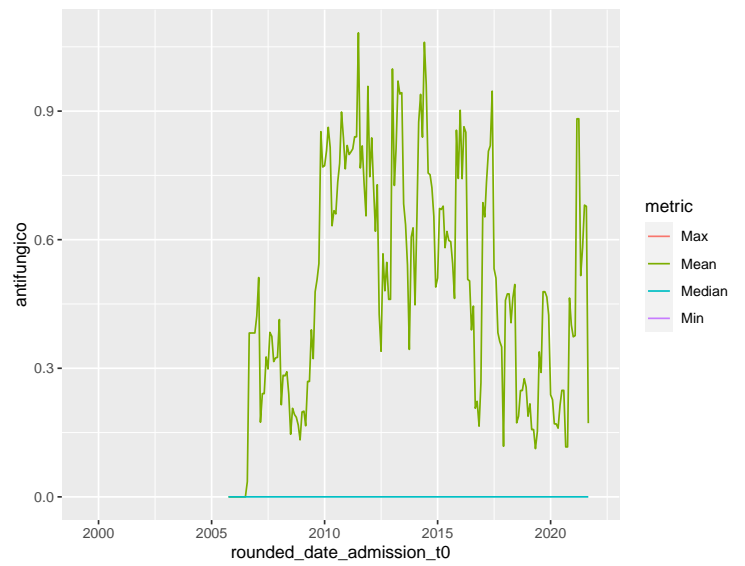
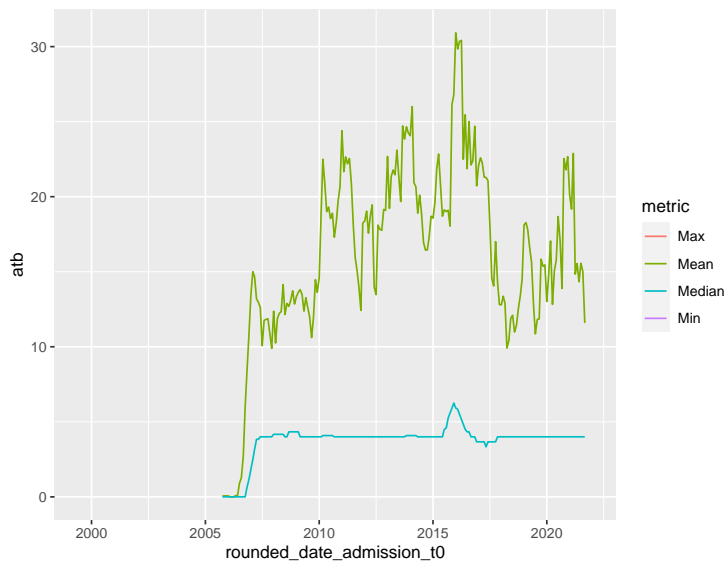
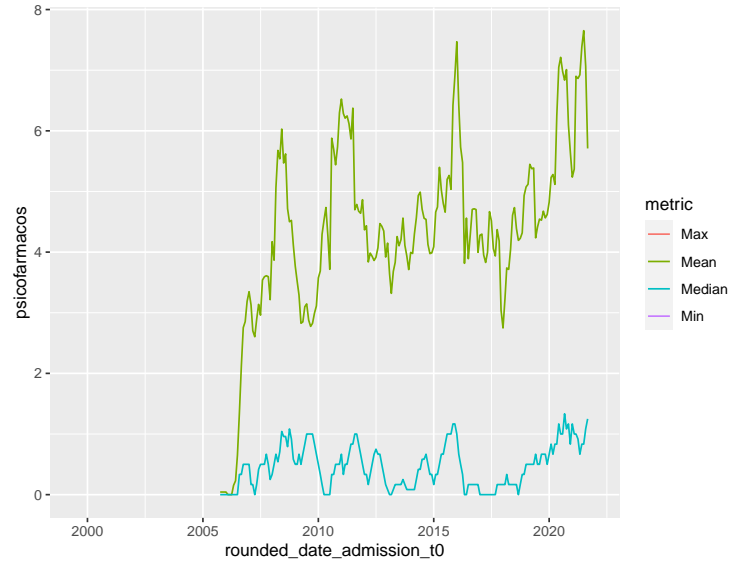
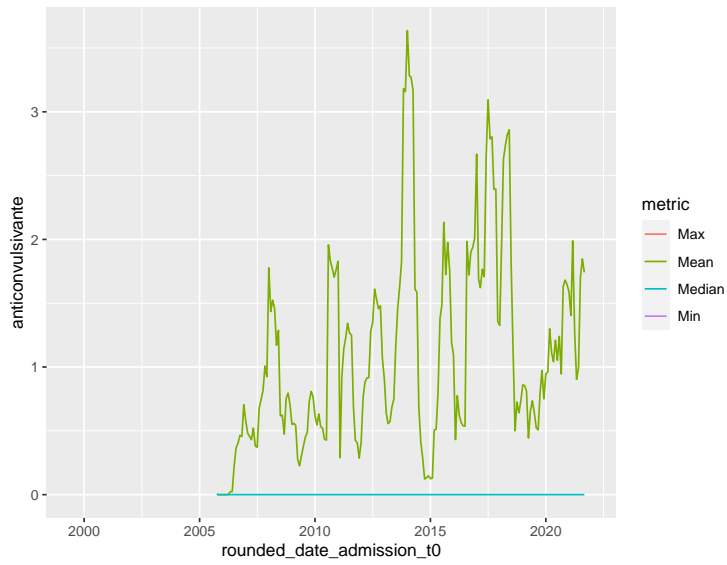
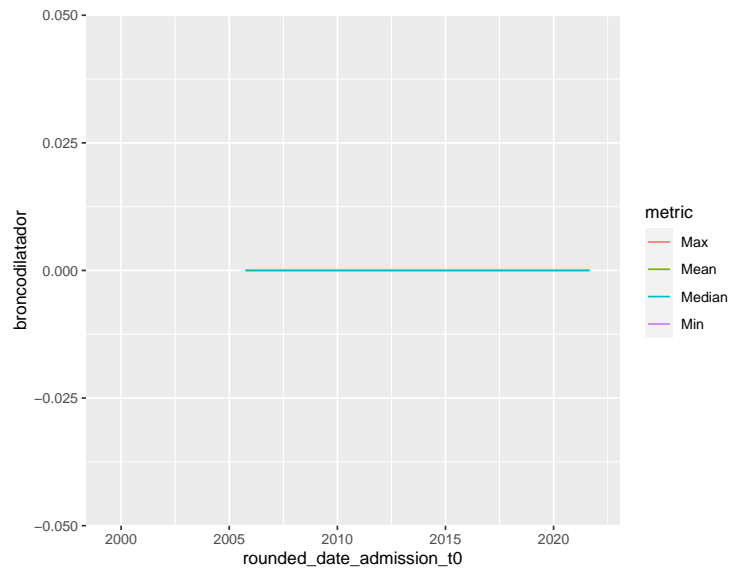
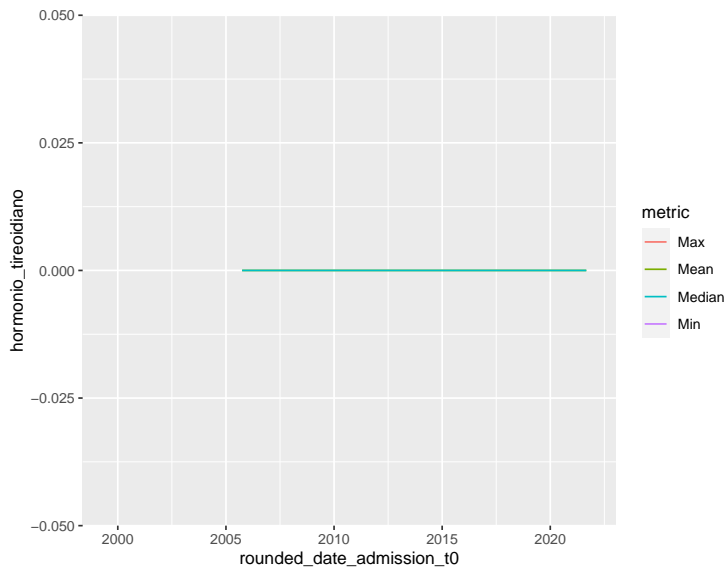


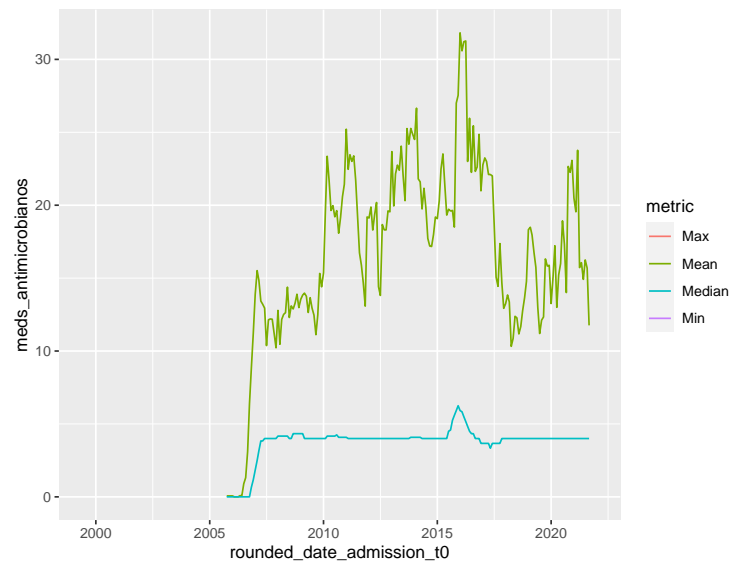
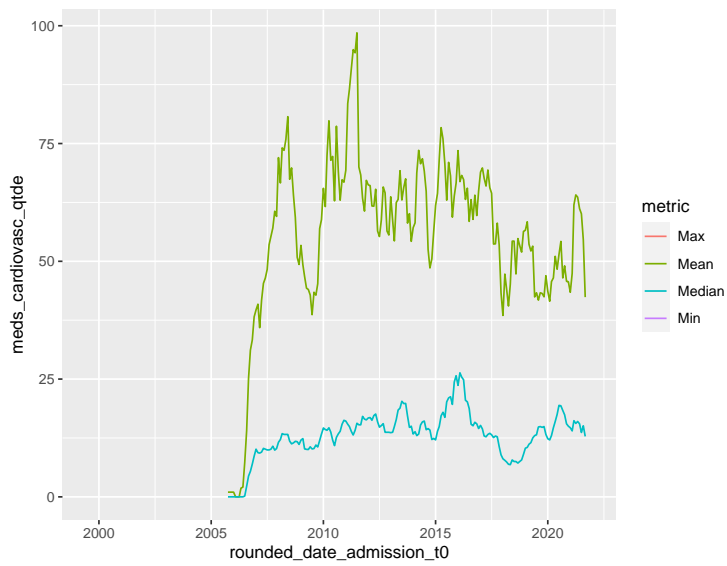
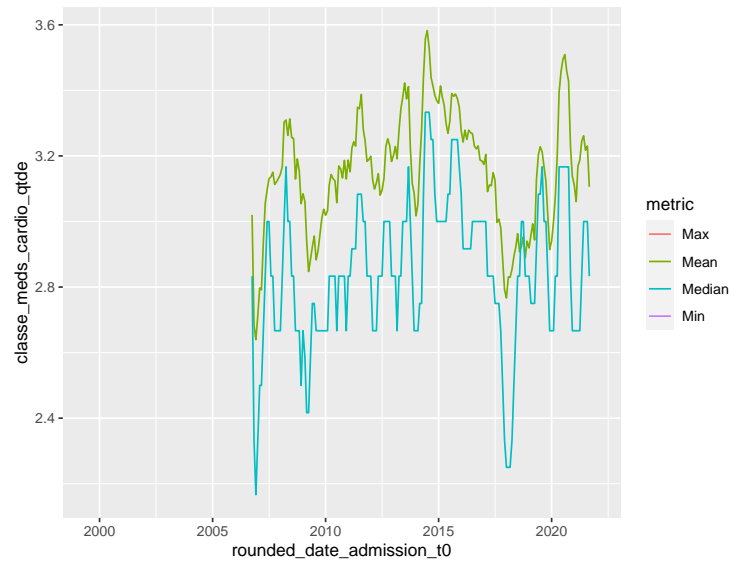
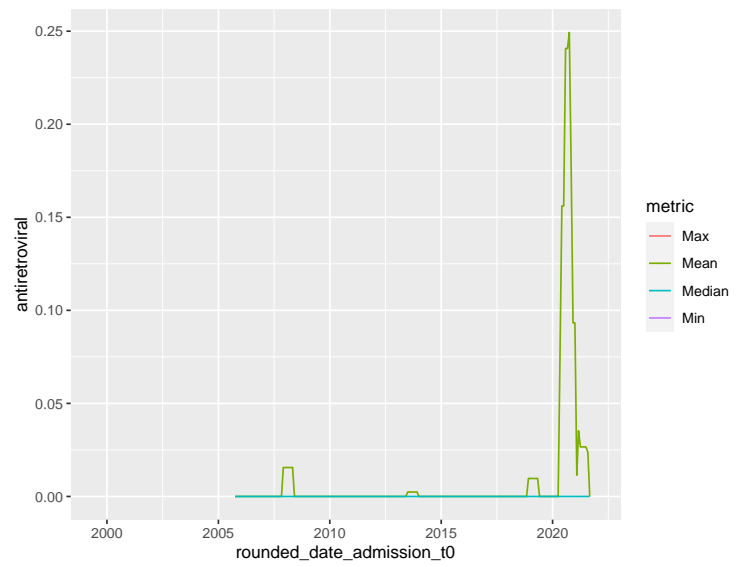
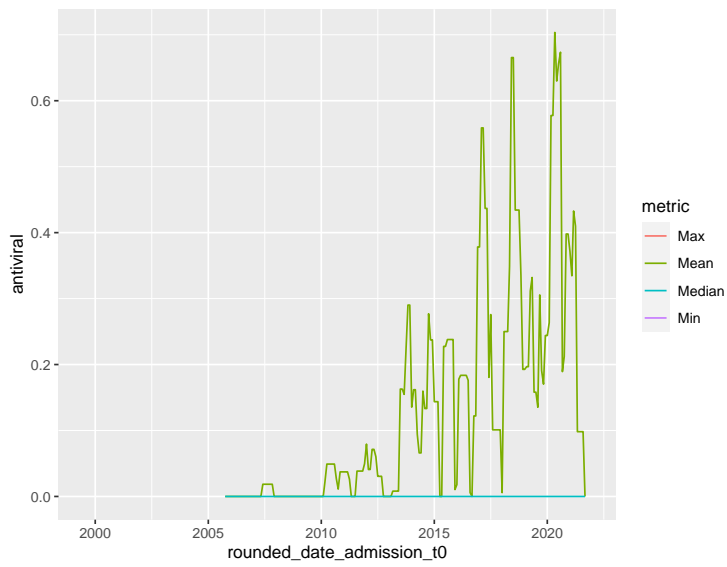


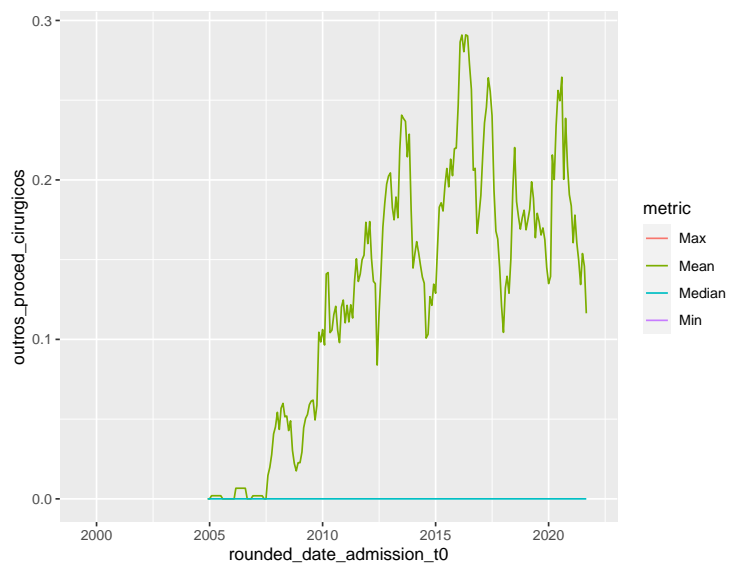
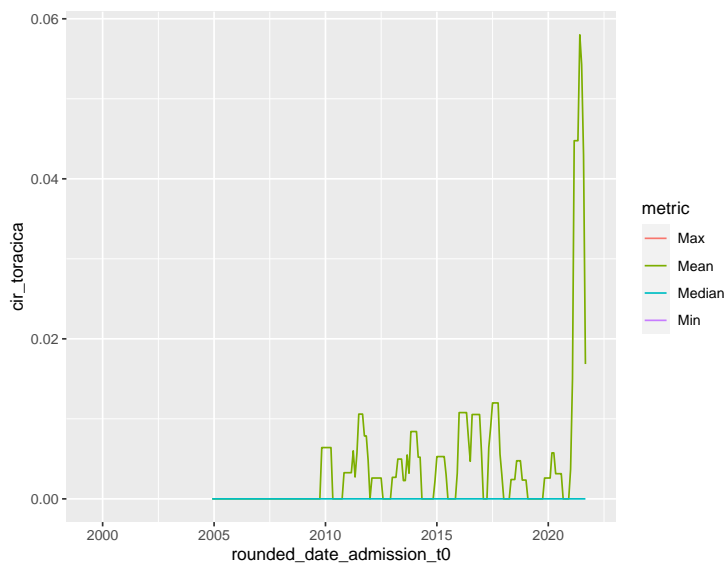
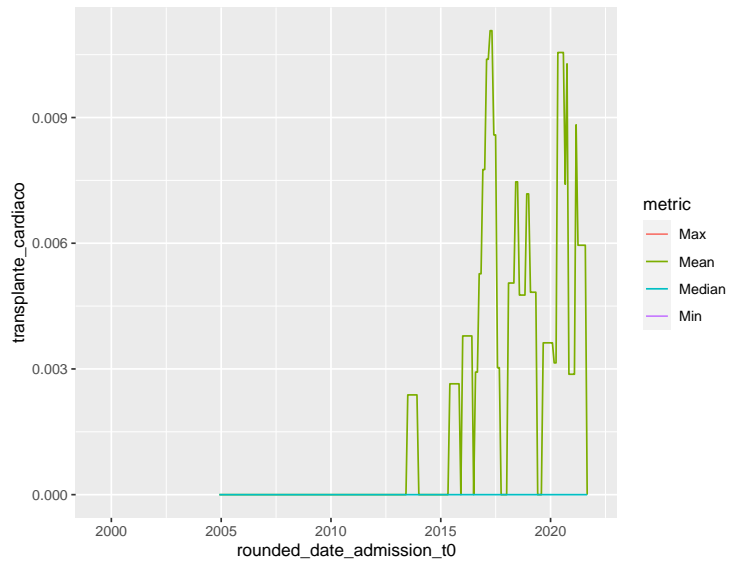
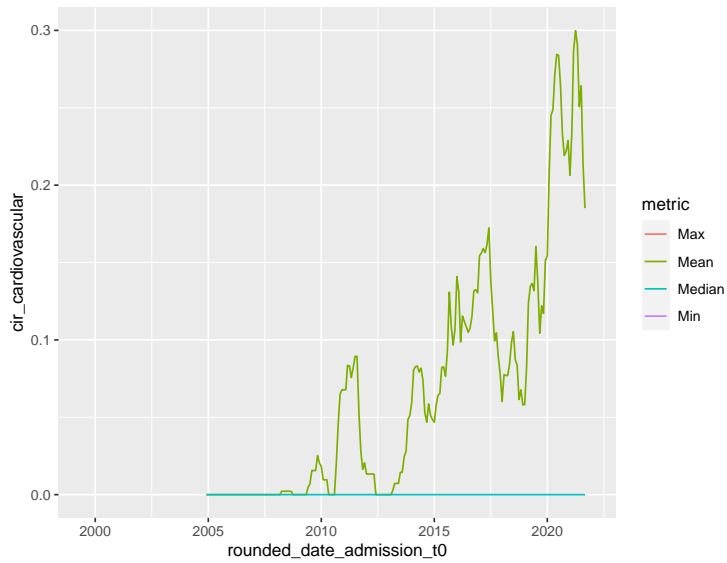
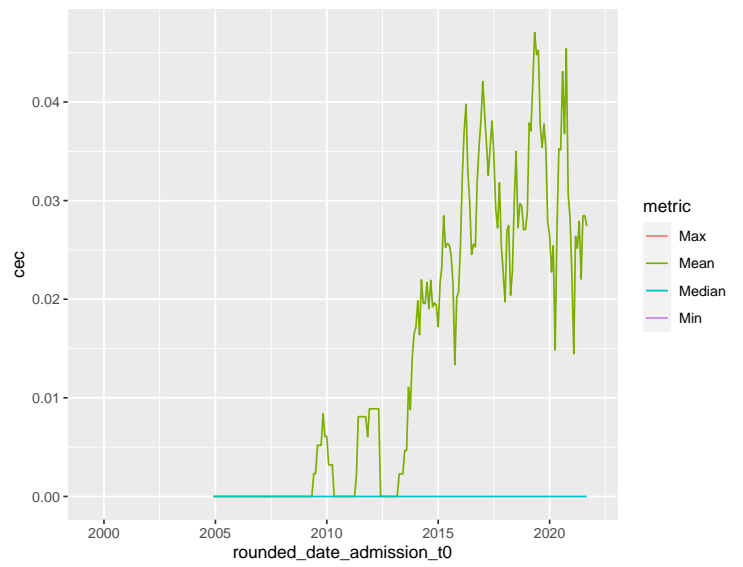
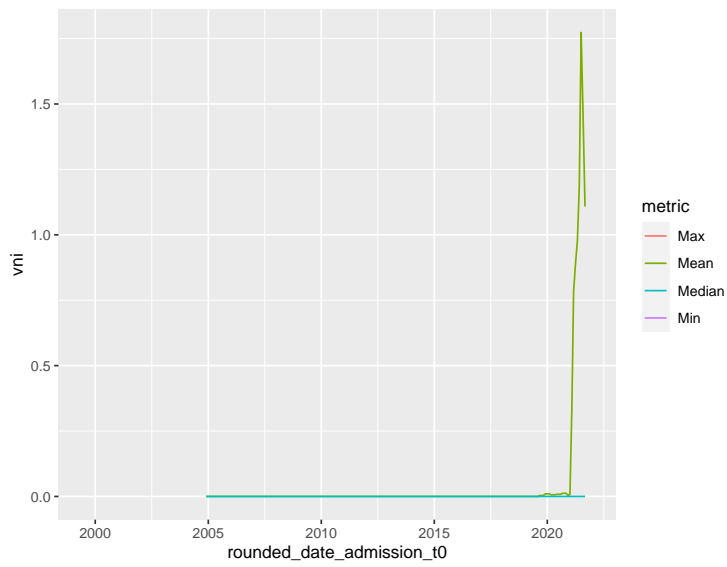


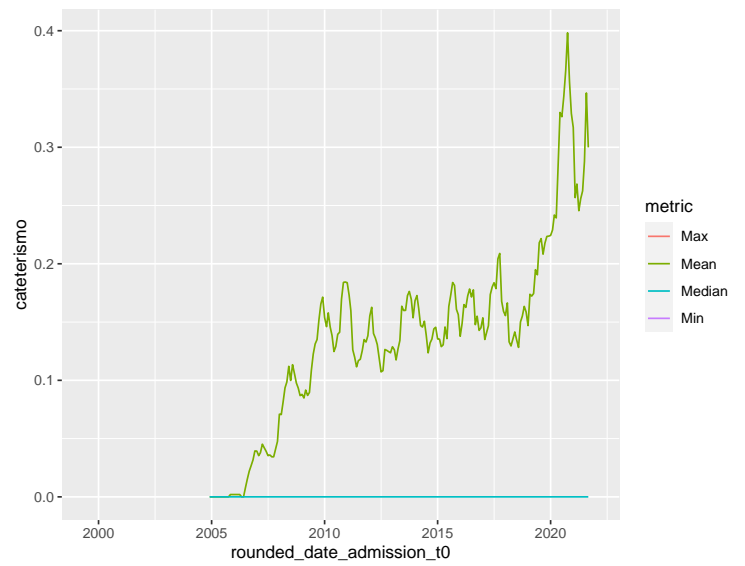
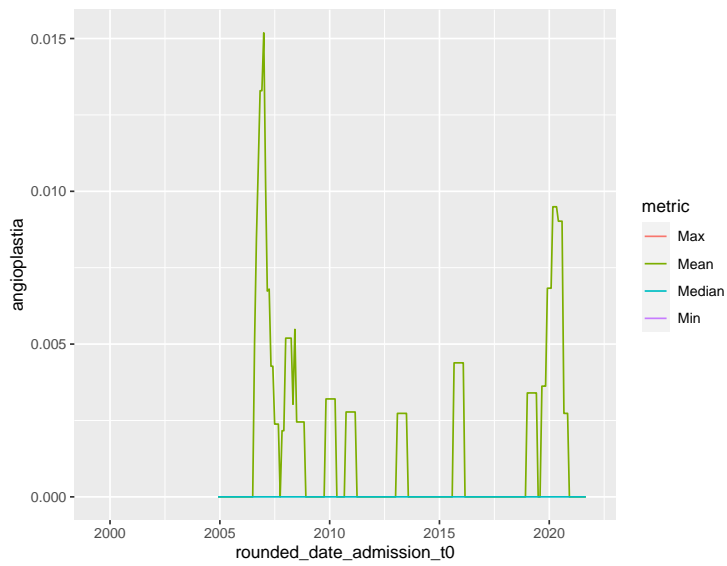
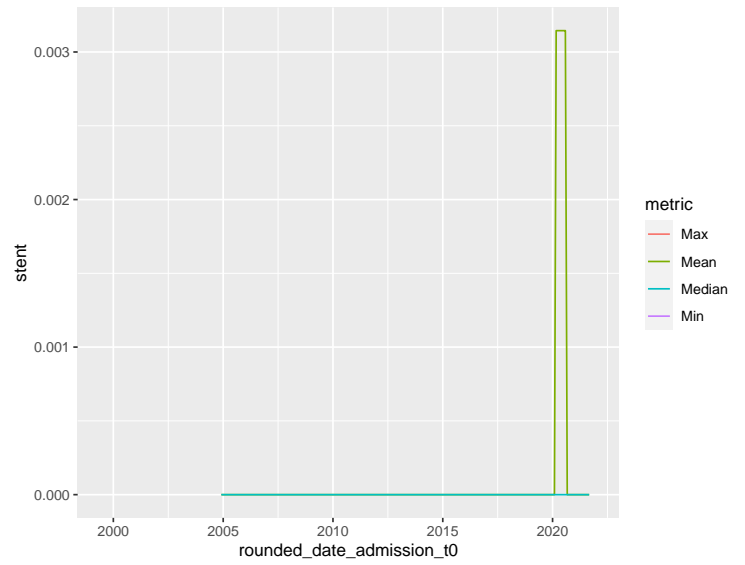
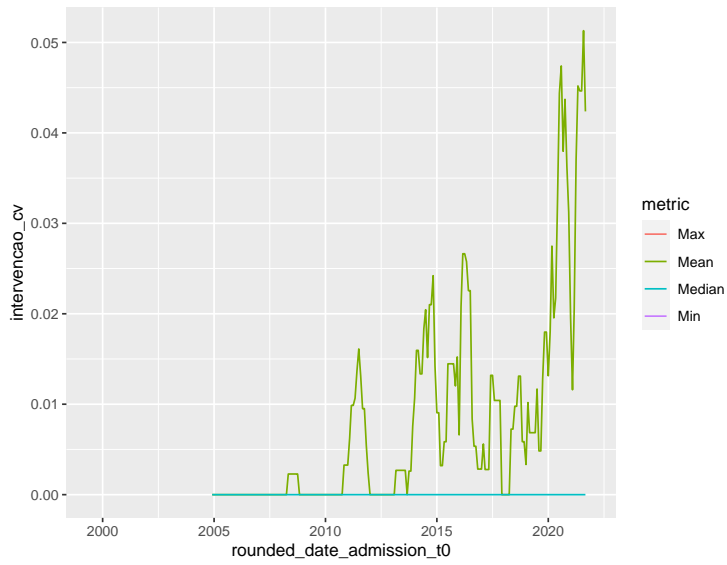
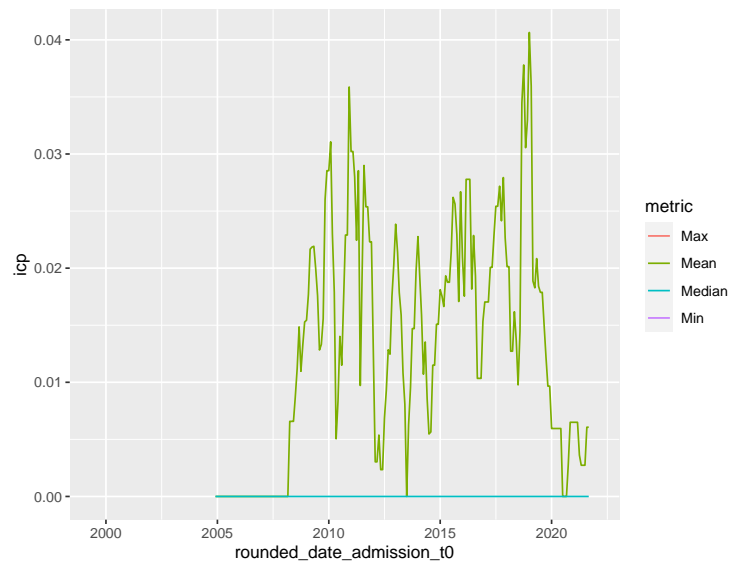
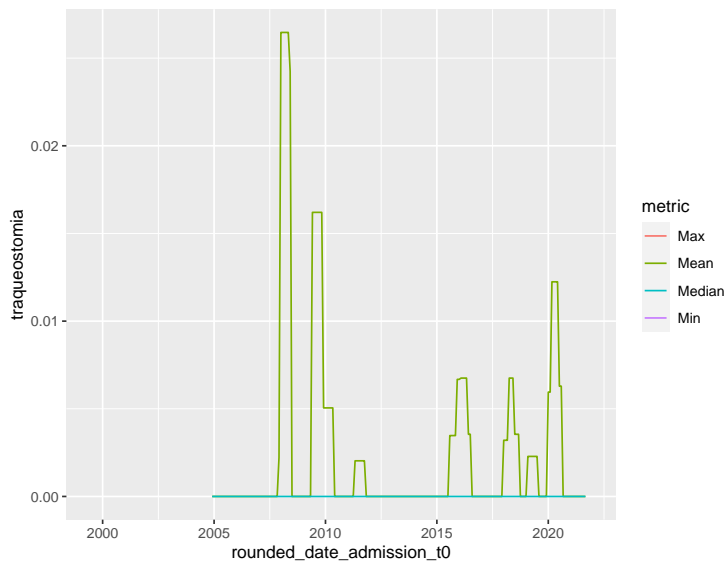


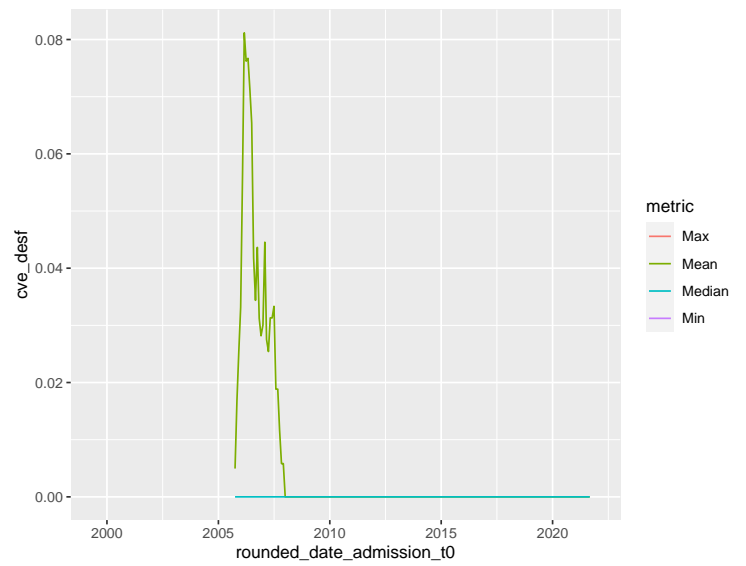
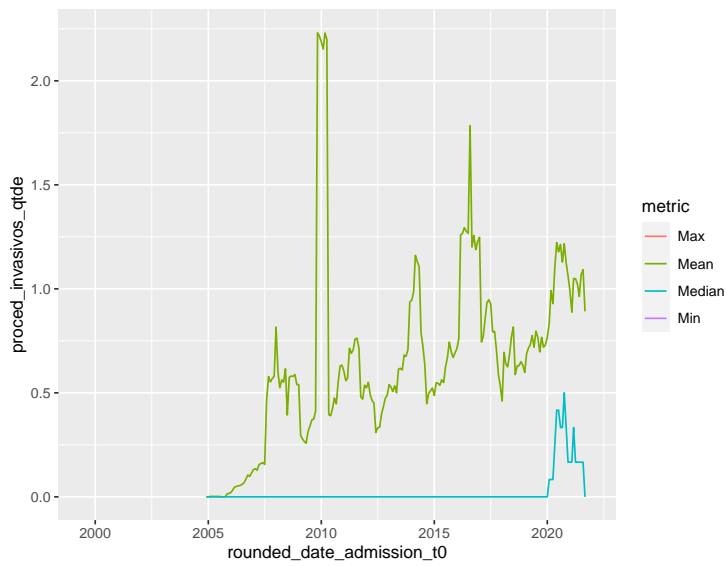
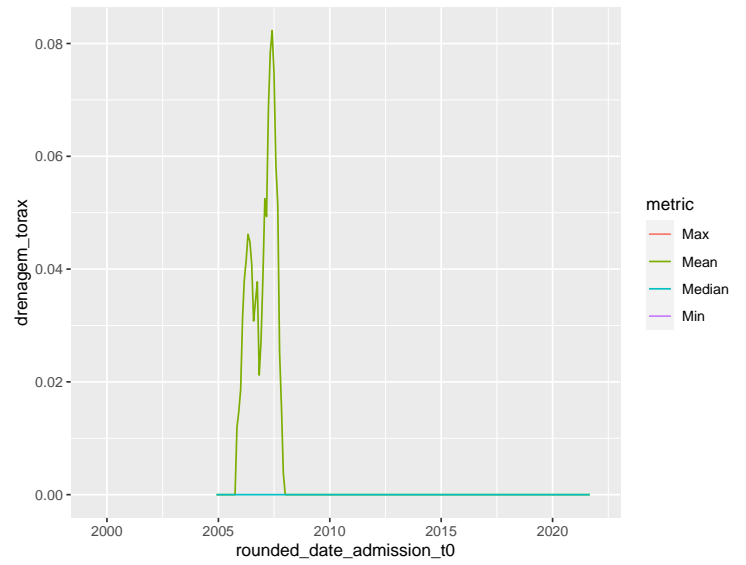
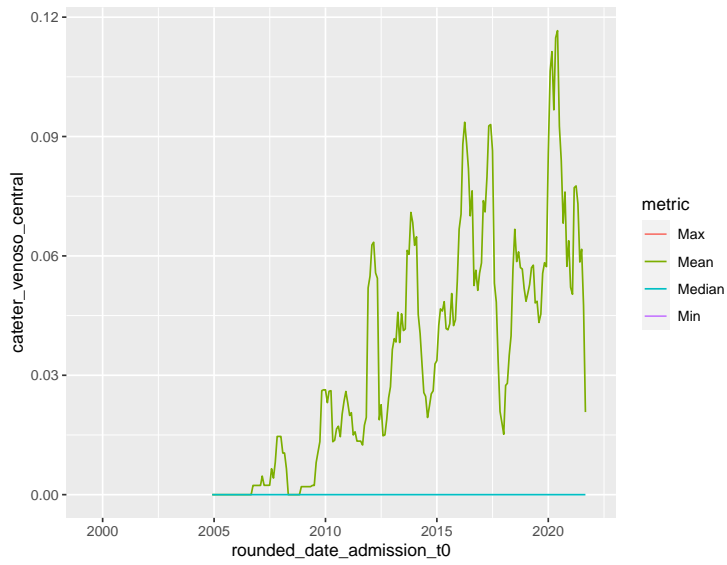
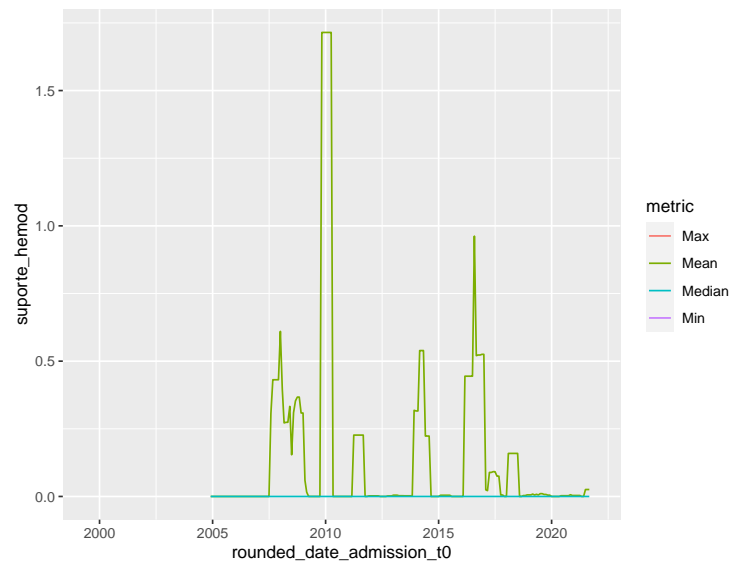
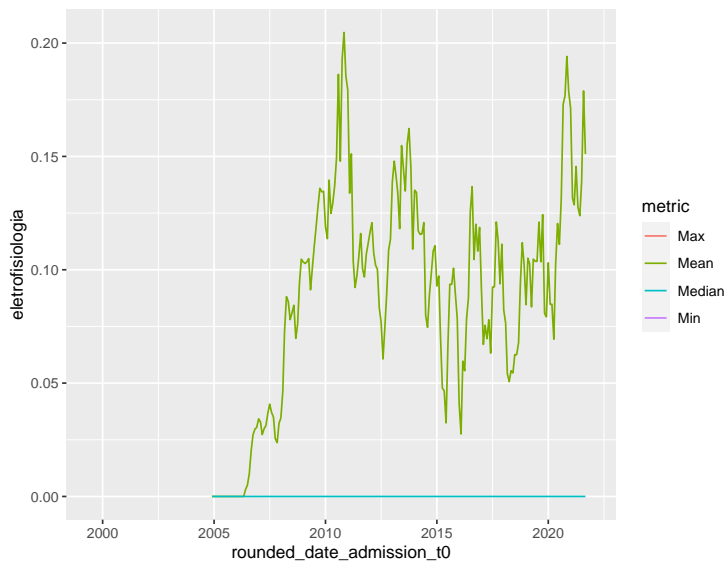


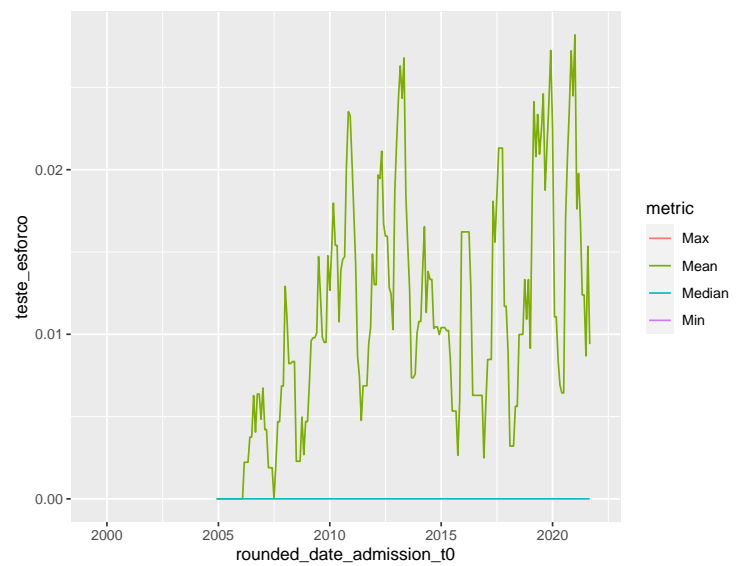
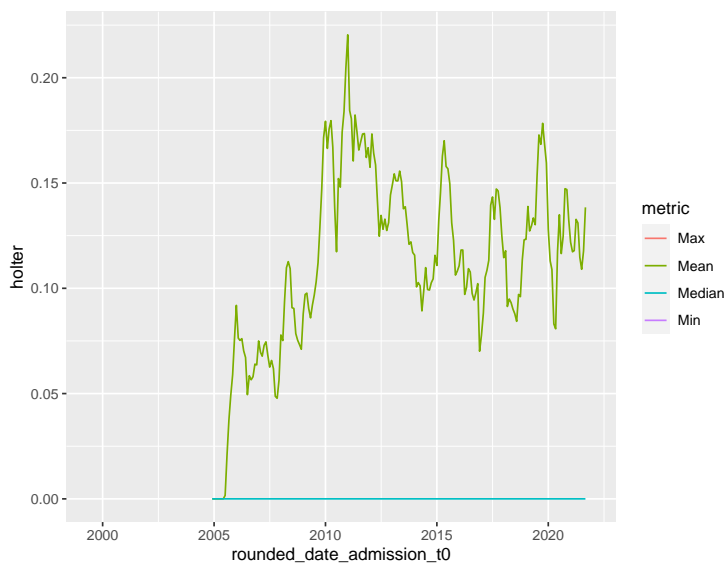
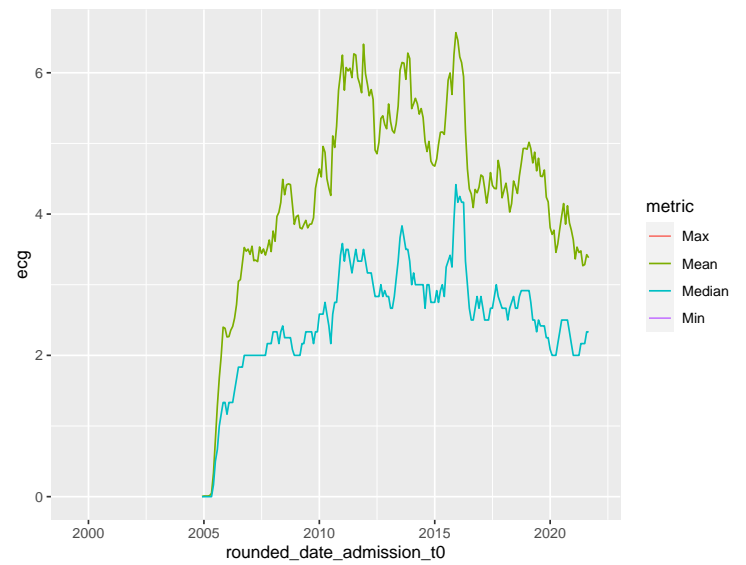
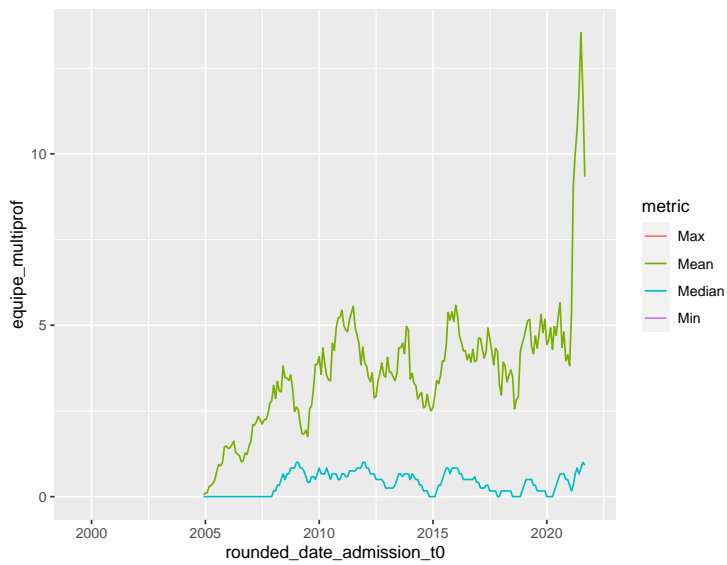
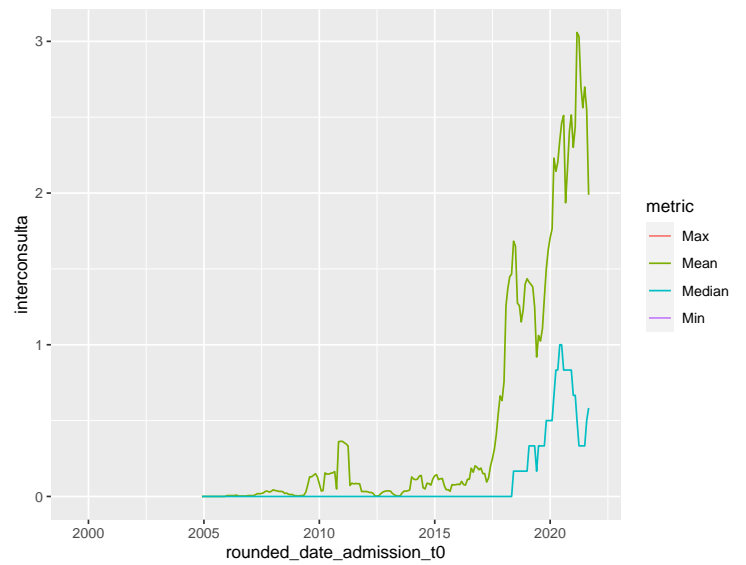
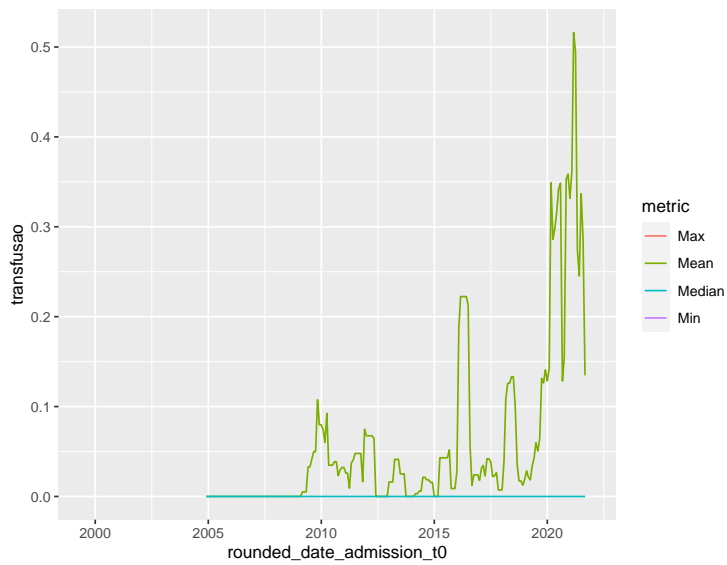


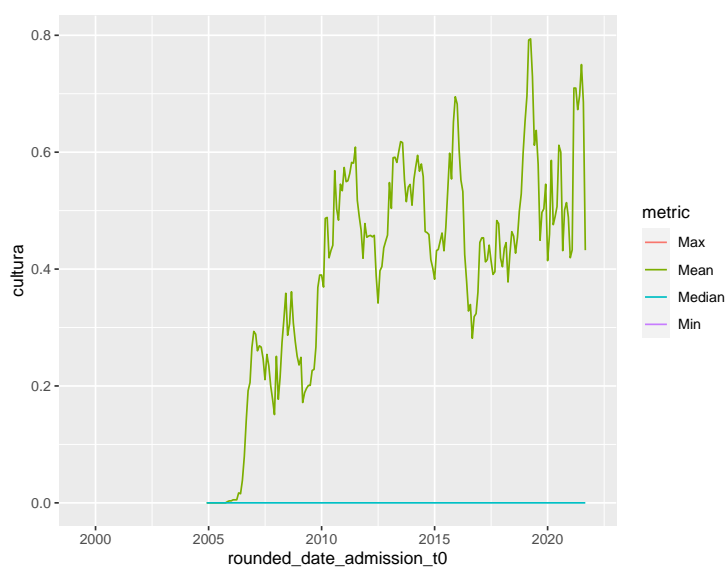
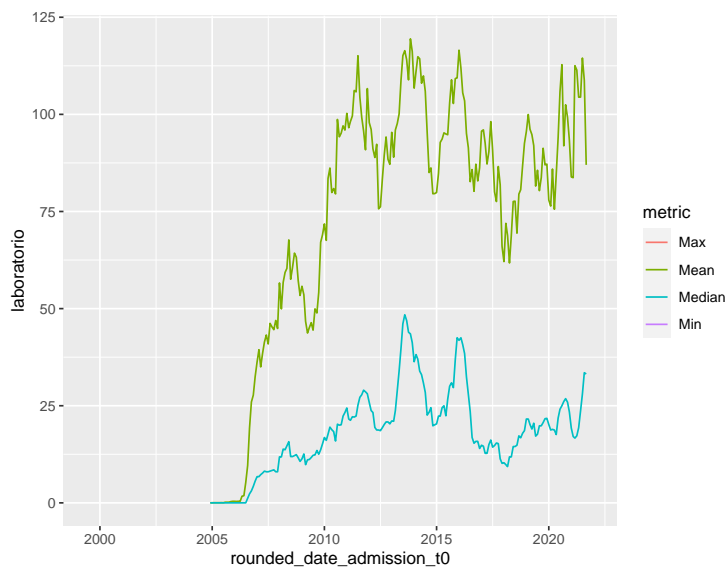
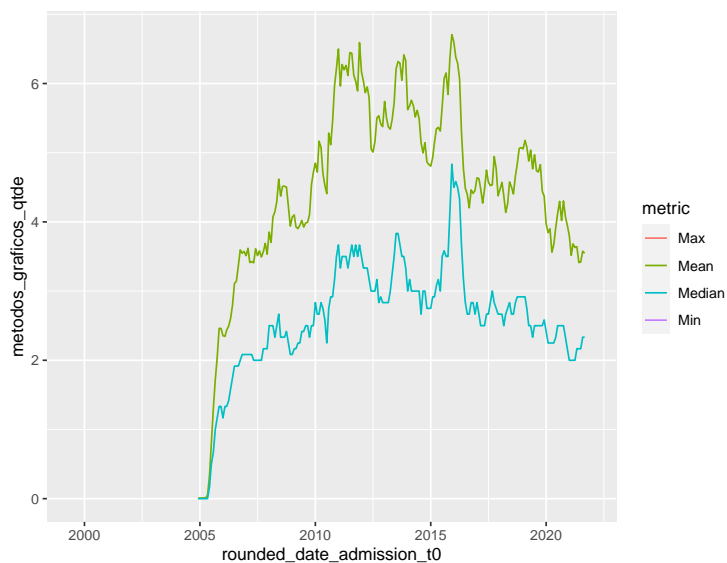
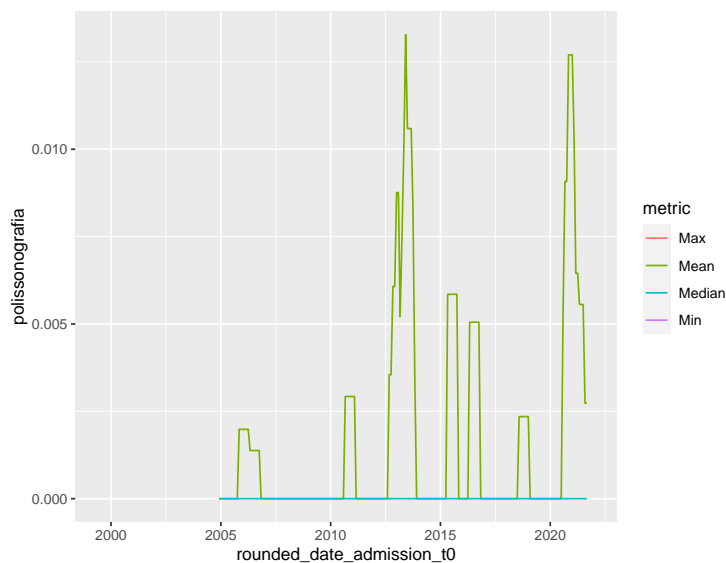
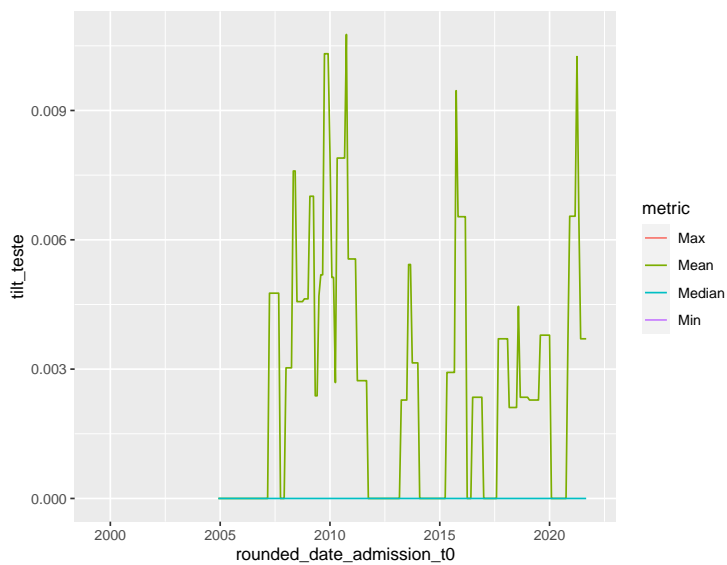
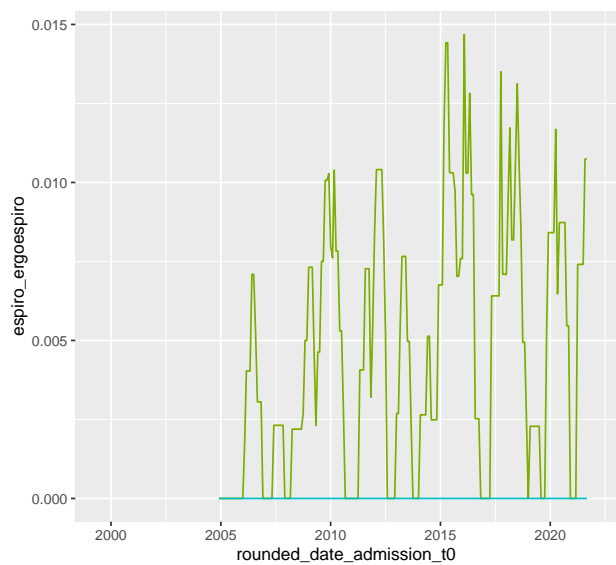


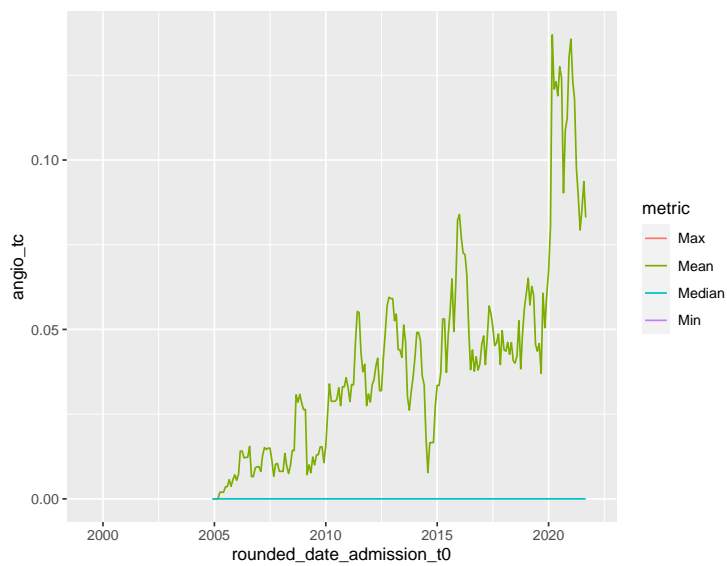
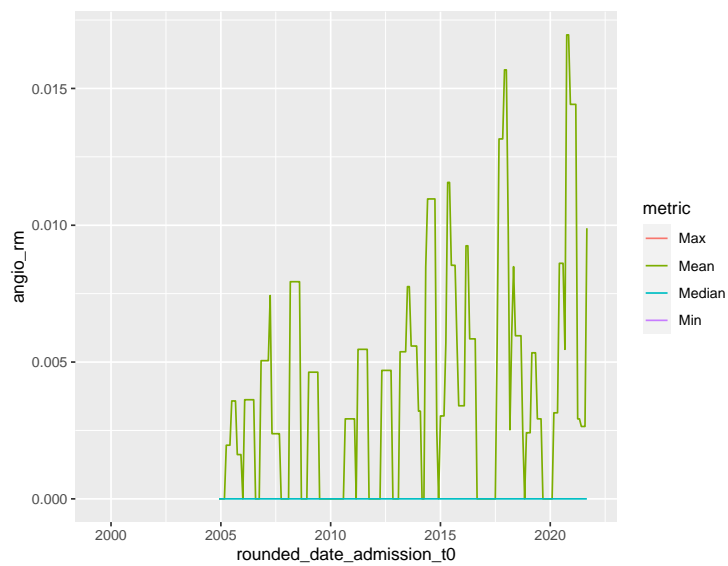
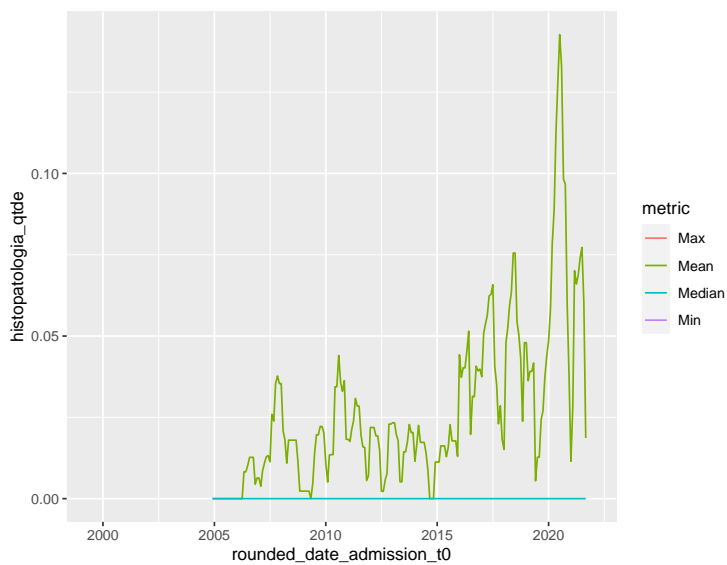
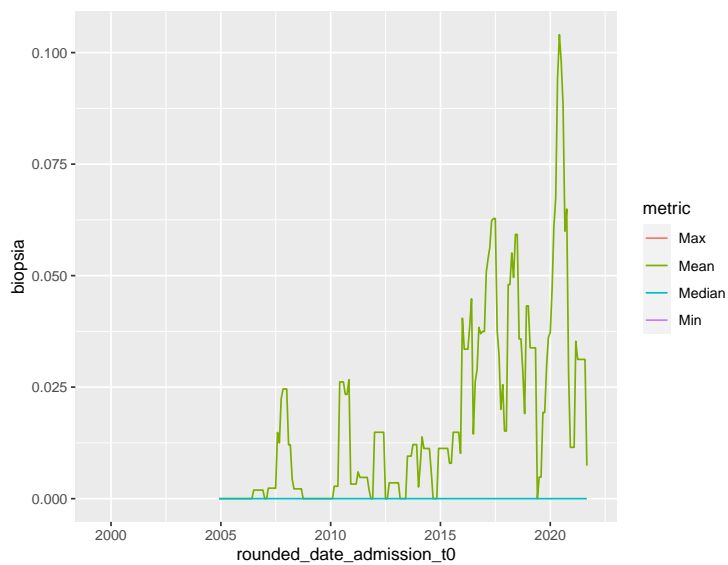
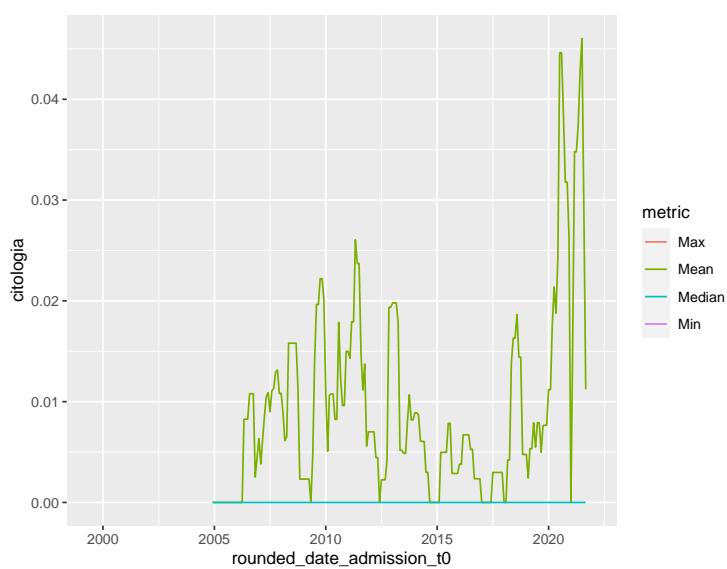
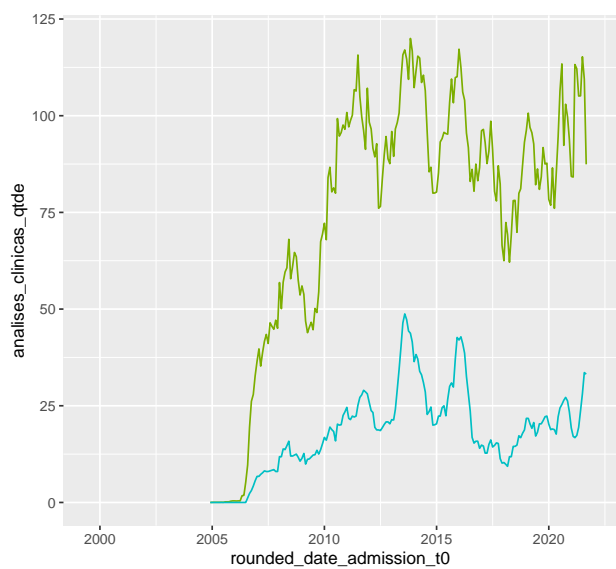


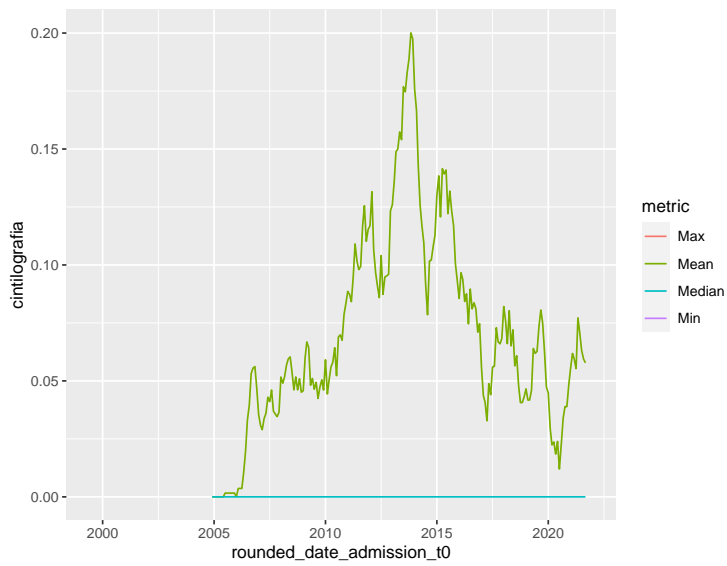
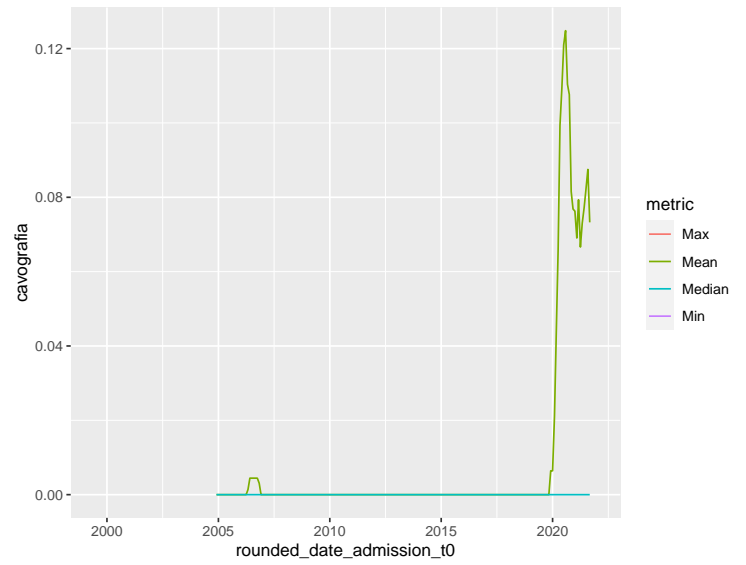
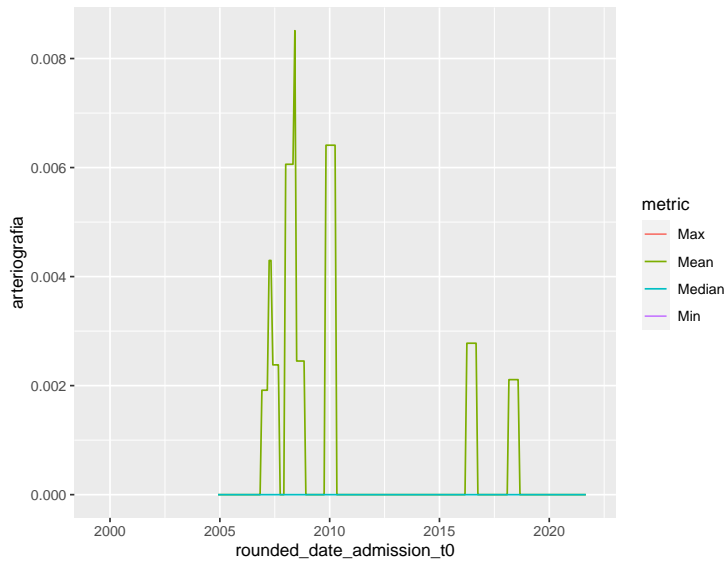
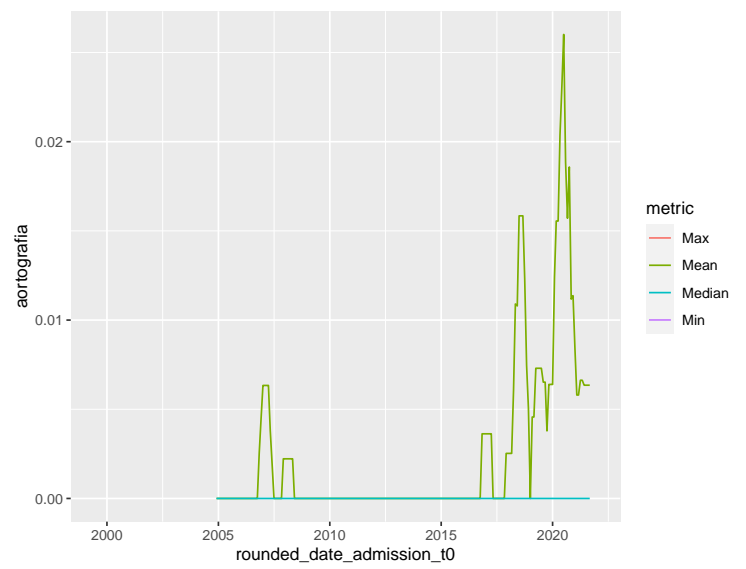
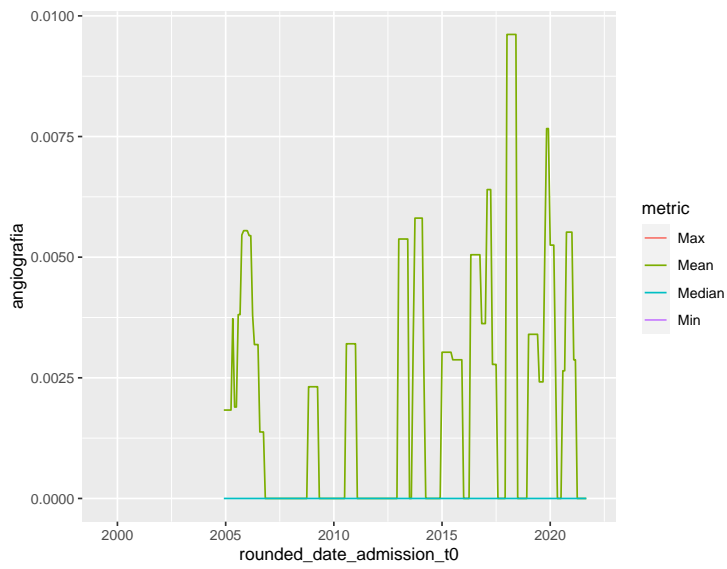


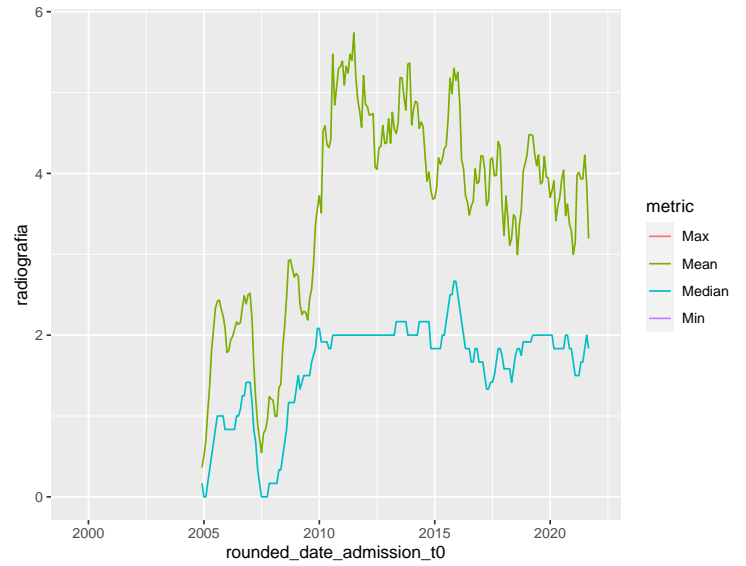
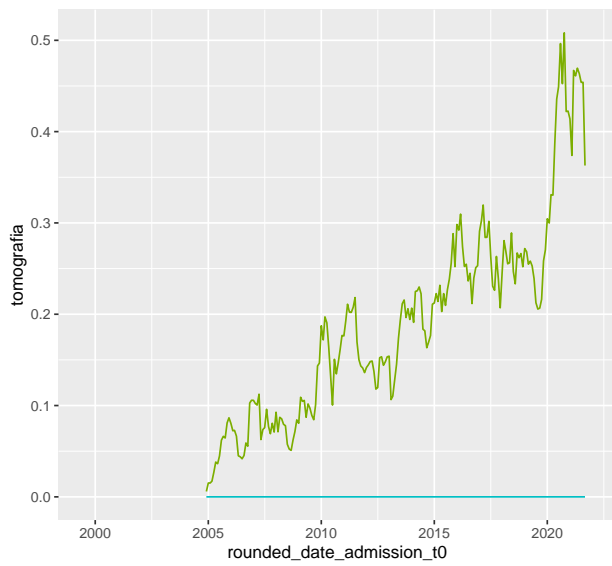
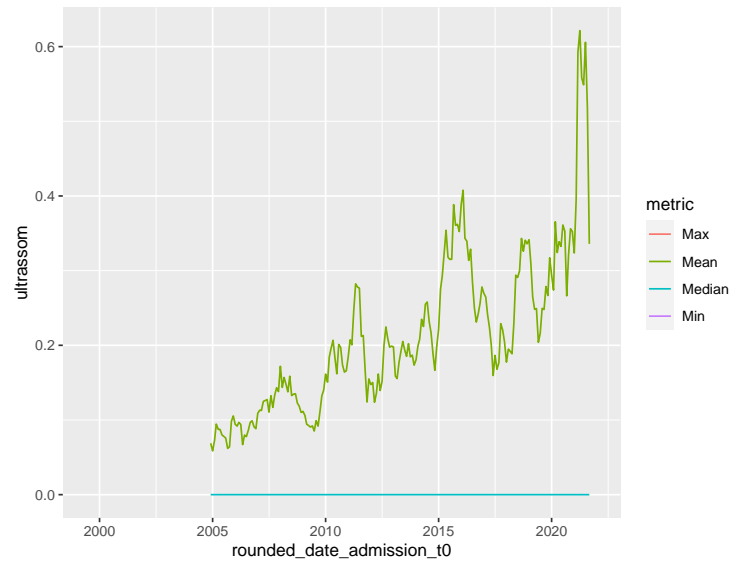
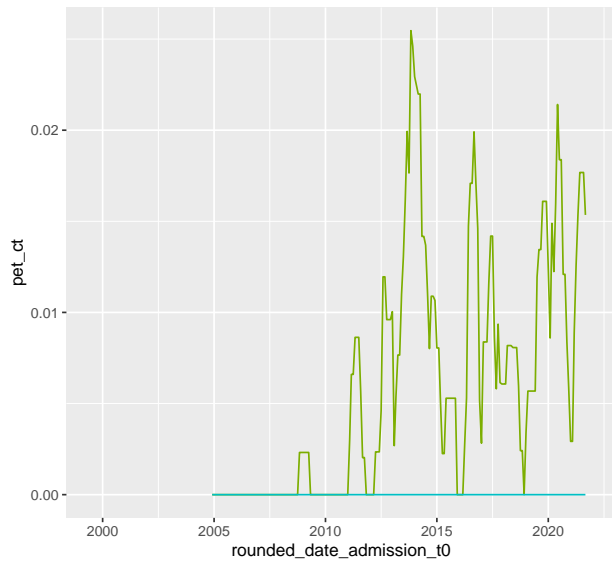
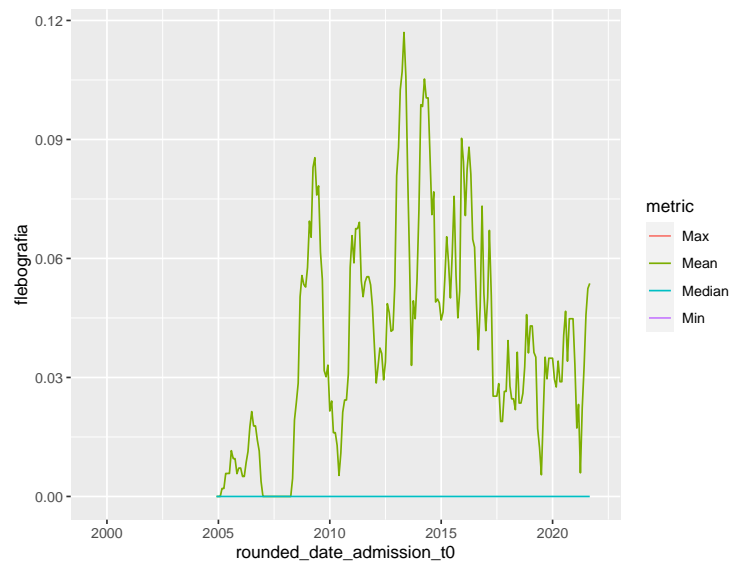
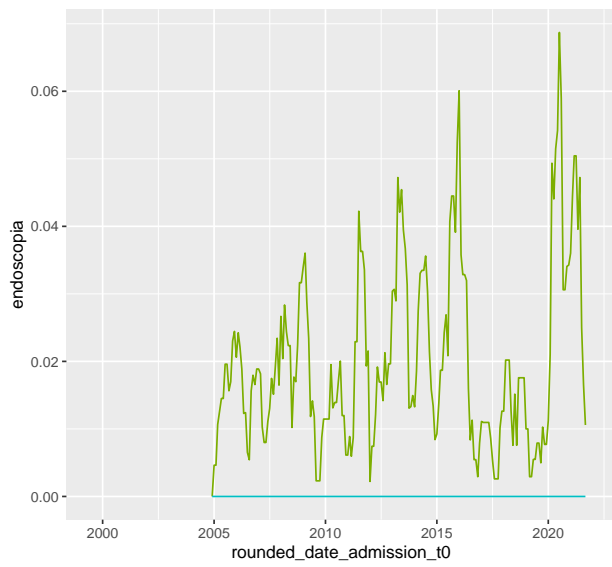


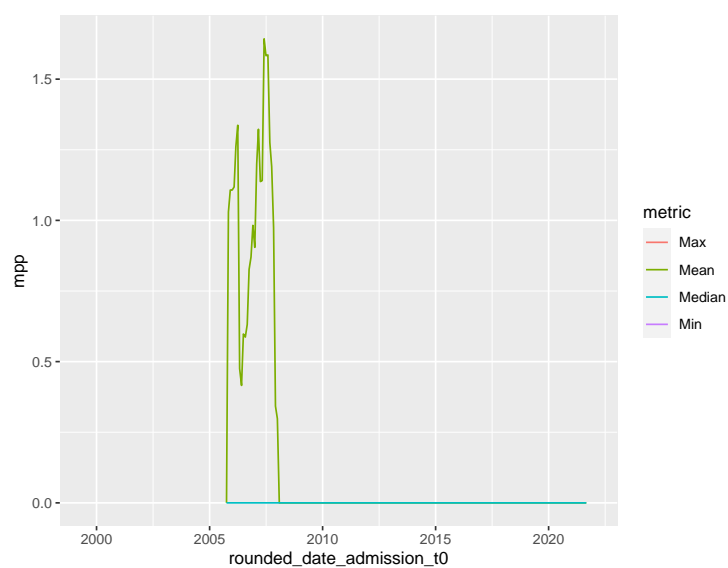
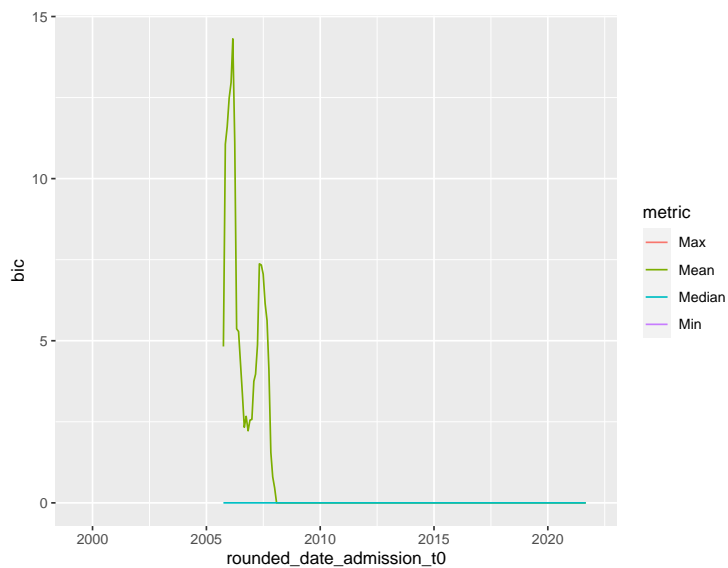
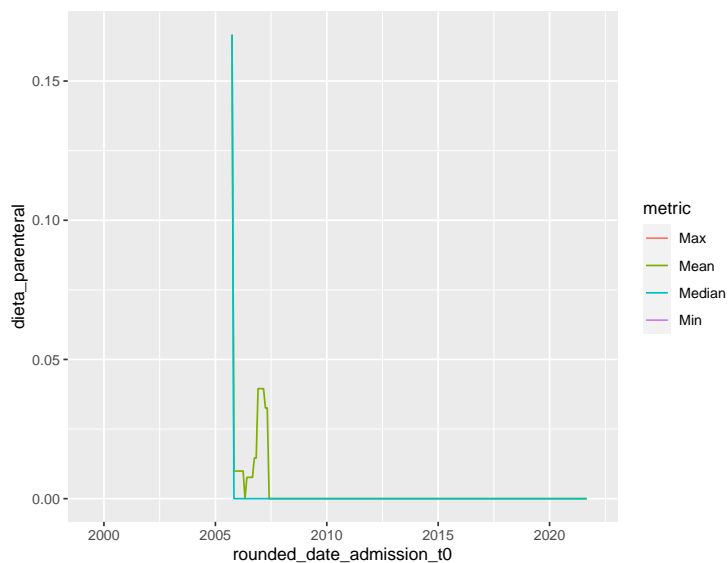
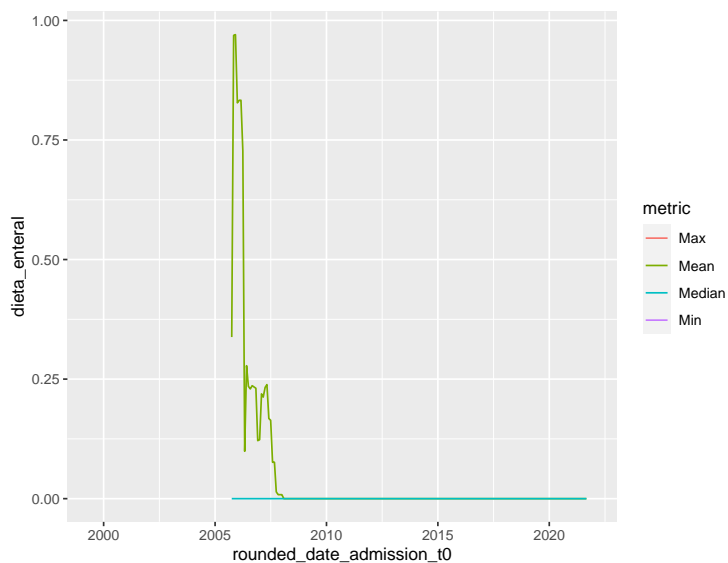
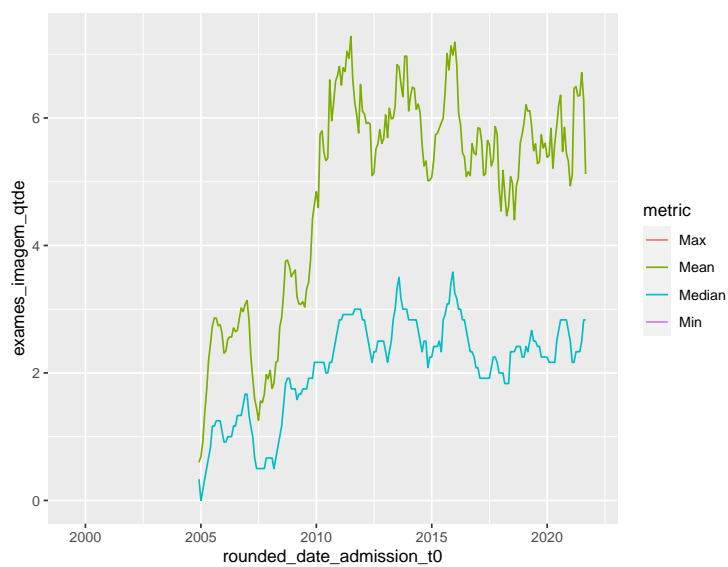
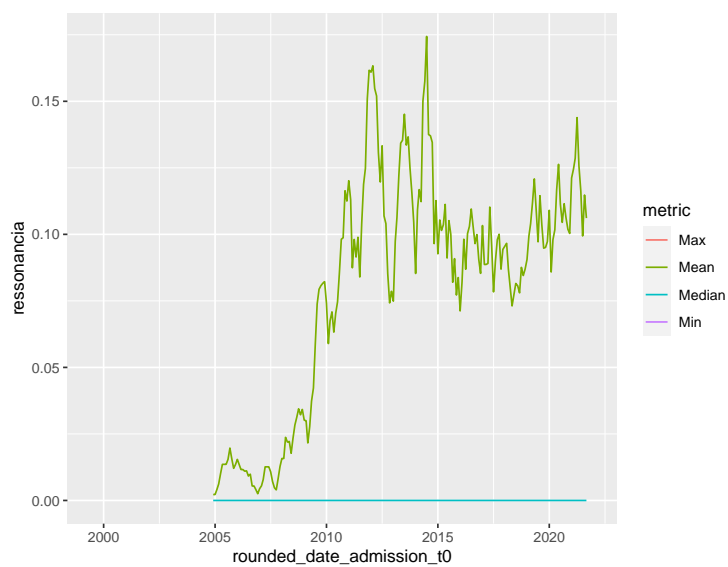












```
column = 'death_intraop_2'
for (column in columns_list$categorical_columns){
  if (mean(is.na(df[[column]])) > 0.5) next
  if (length(unique(df[[column]])) > 10) print(column)

  df %>%
    mutate(rounded_date_admission_t0 = lubridate::floor_date(date_admission_t0,
```

```

'month')) %>%
group_by(rounded_date_admission_t0, !!sym(column)) %>%
summarise(N = n()) %>%
ungroup %>%
arrange(rounded_date_admission_t0) %>%
mutate(N = zoo::rollmean(N, k = k, fill = NA)) %>%
group_by(rounded_date_admission_t0) %>%
mutate(percentage = N / sum(N)) %>%
ungroup %>%
tidyr::complete(rounded_date_admission_t0, !!sym(column),
                 fill = list(N = 0, percentage = 0)) %>%
ggplot(aes(x = rounded_date_admission_t0, y = percentage,
           color = !!sym(column), fill = !!sym(column))) +
  geom_area(alpha=0.6) +
  scale_y_continuous(labels = scales::percent) +
  labs(y = column)-> p

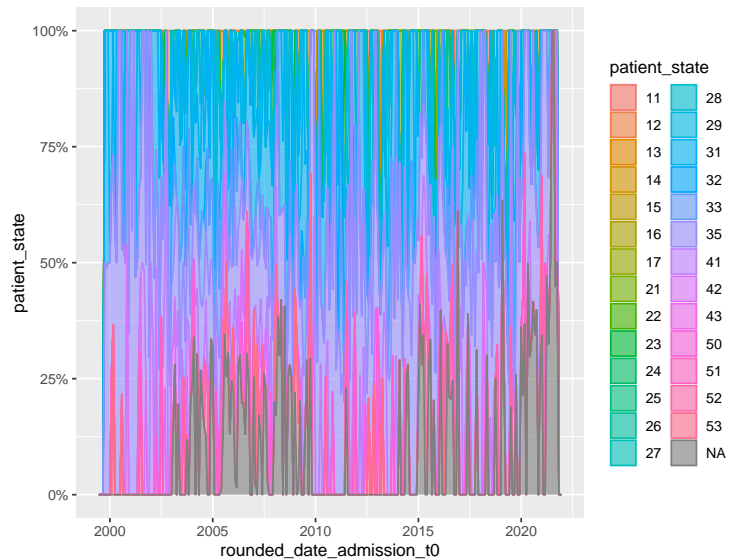
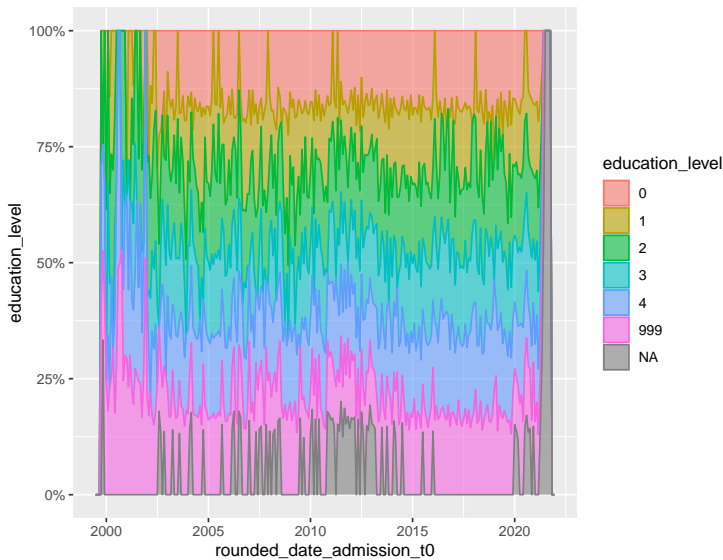
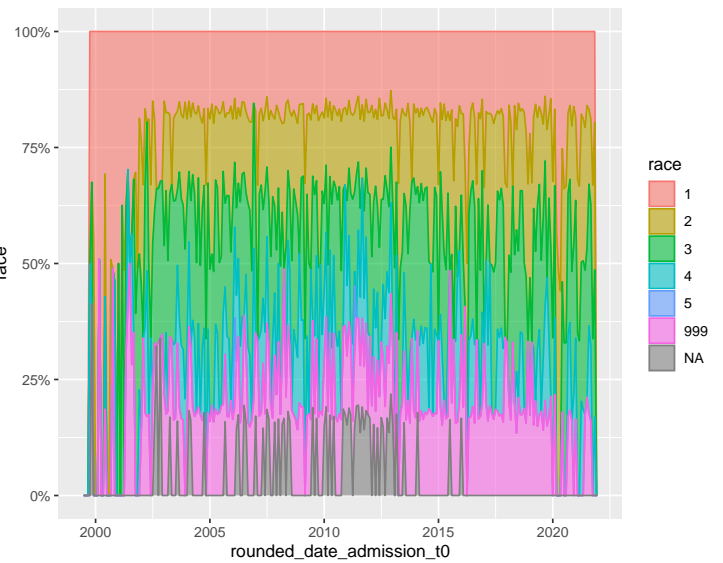
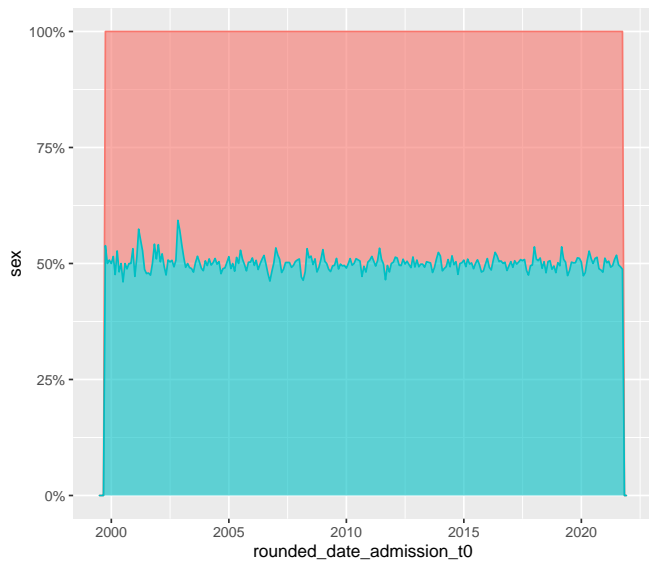
print(p)
}

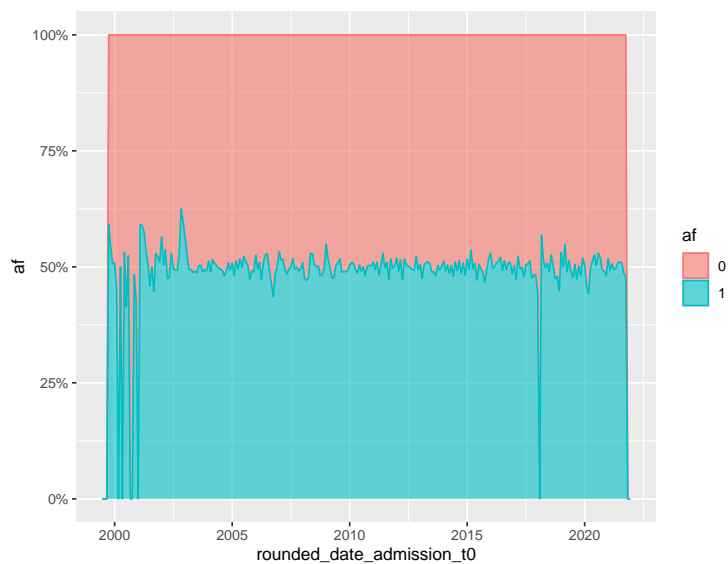
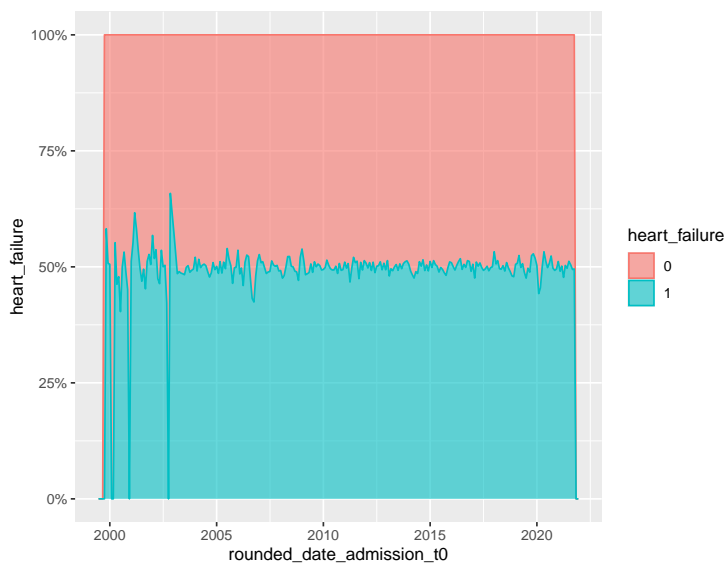
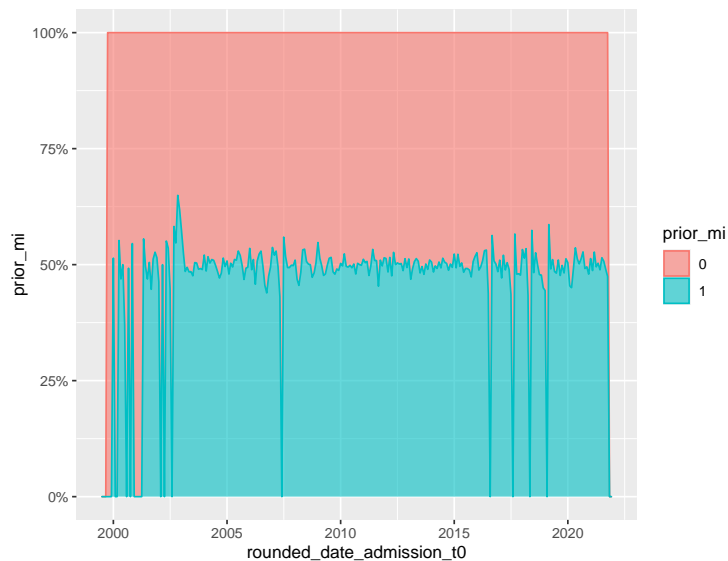
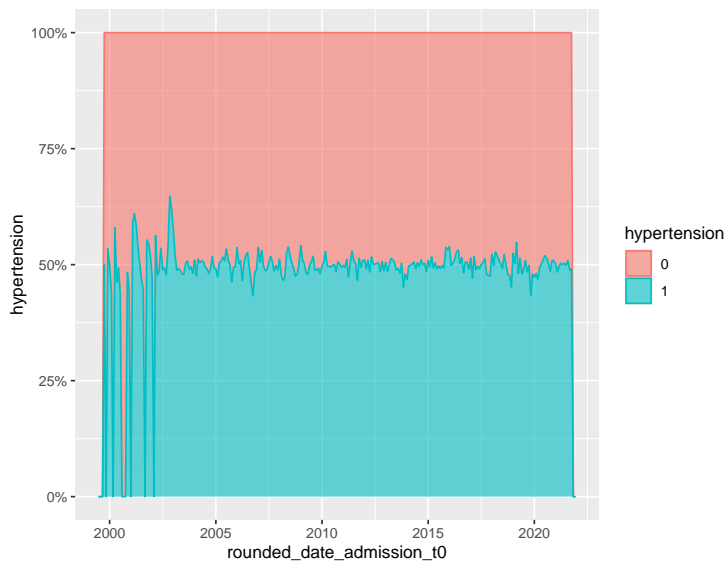
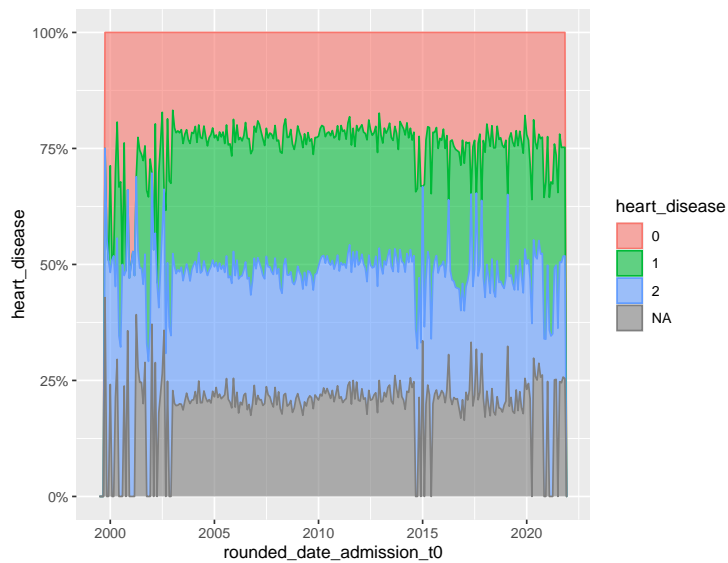
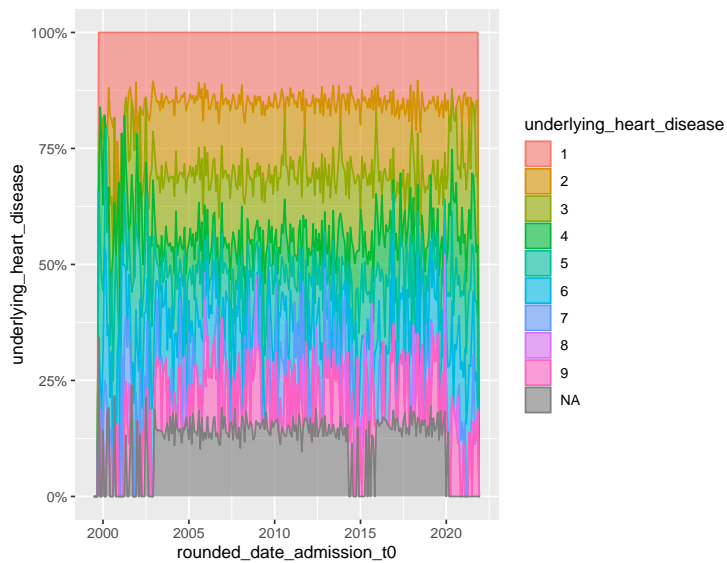
```

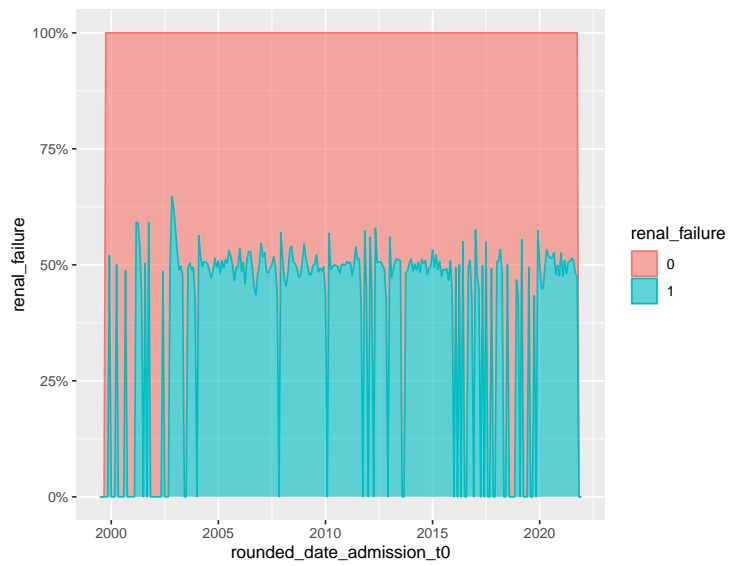
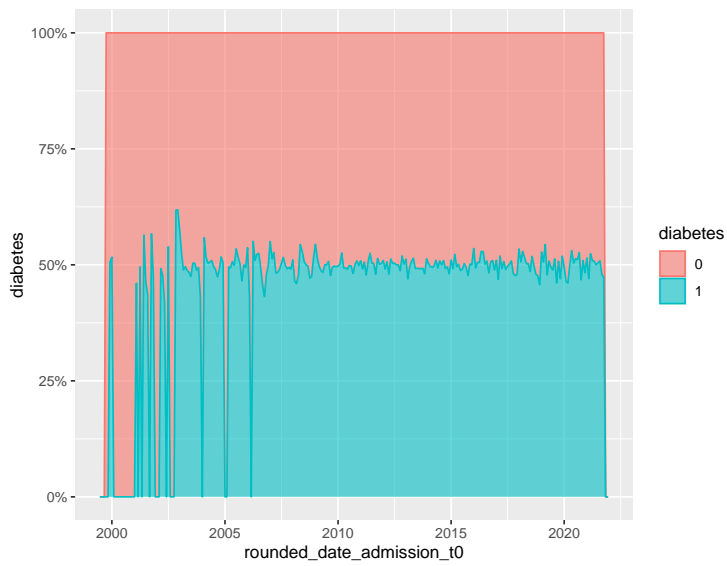
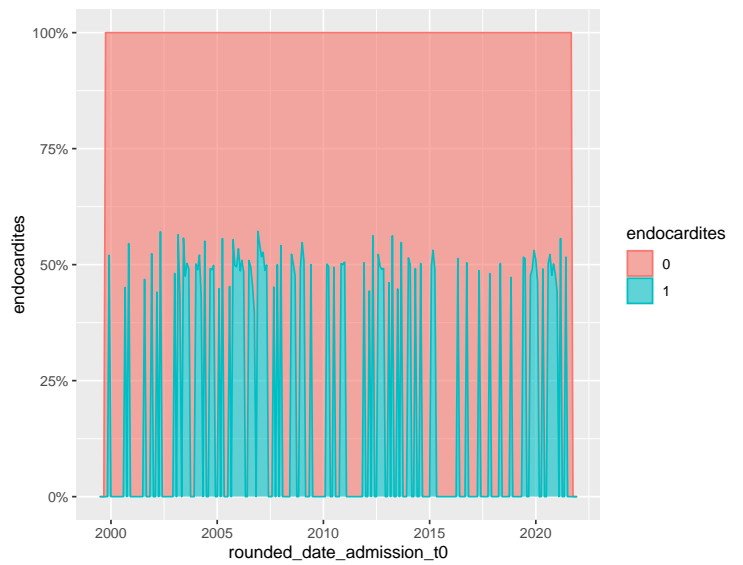
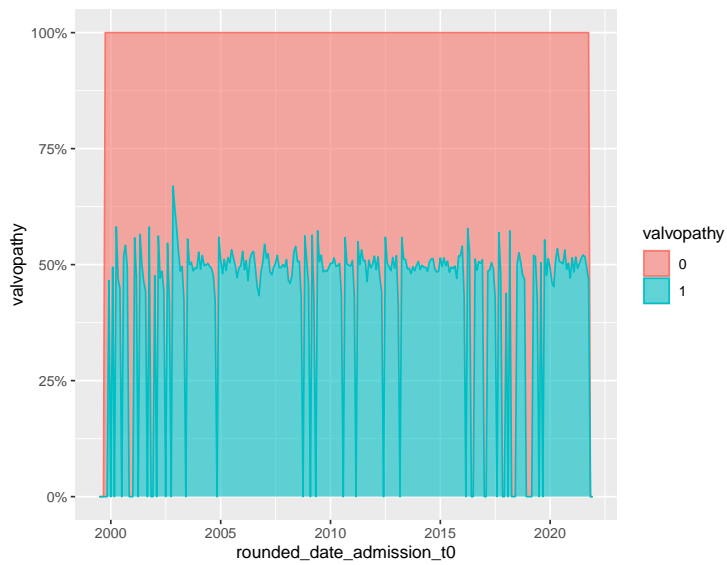
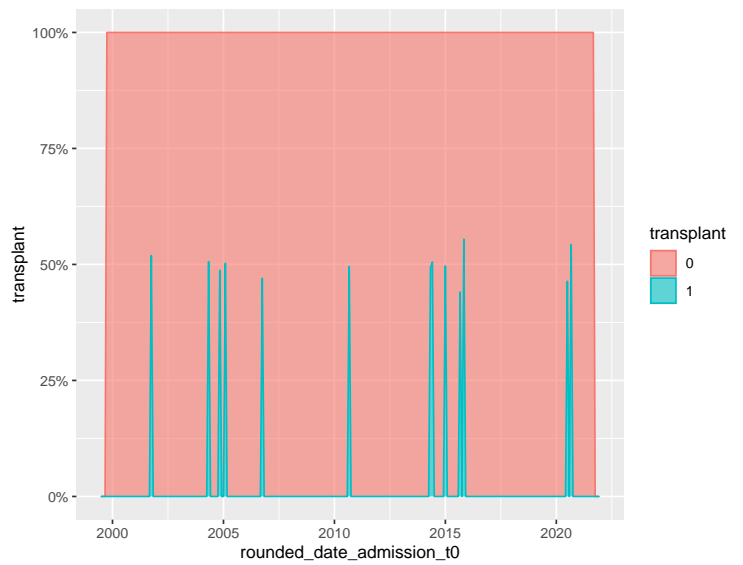
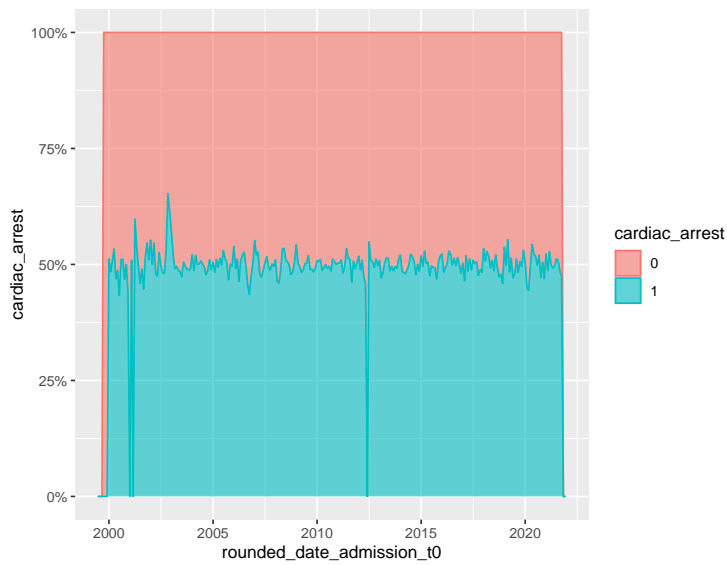
```
## [1] "patient_state"
```

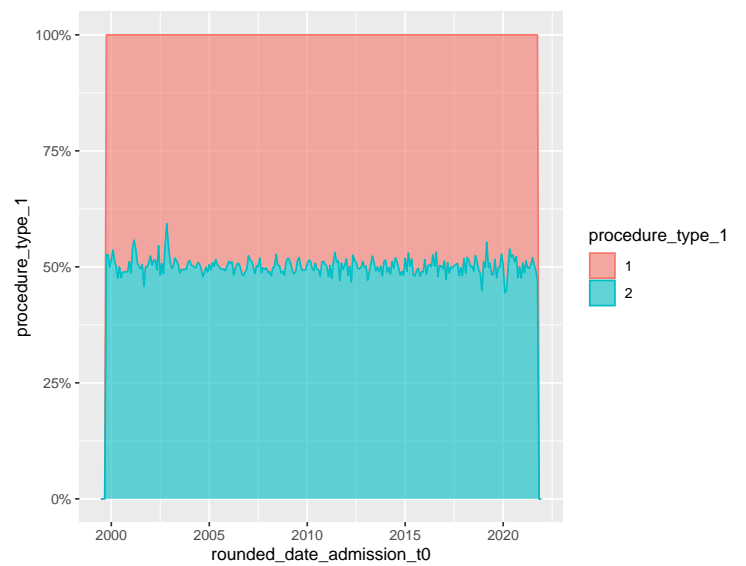
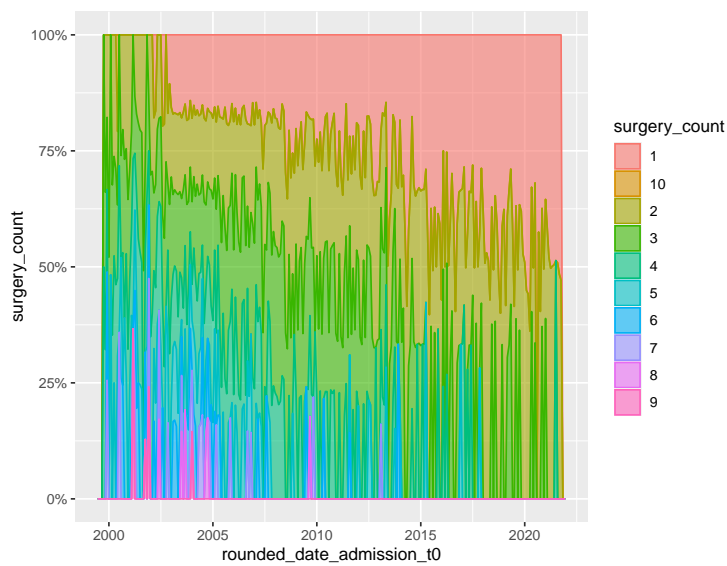
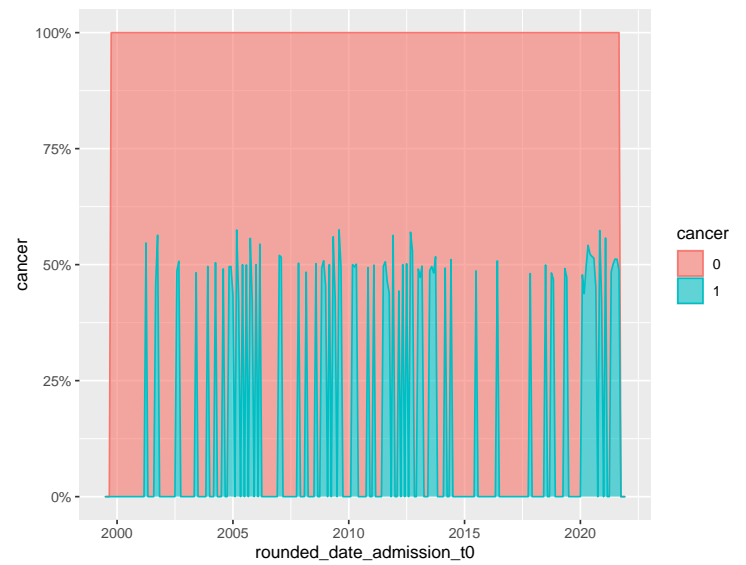
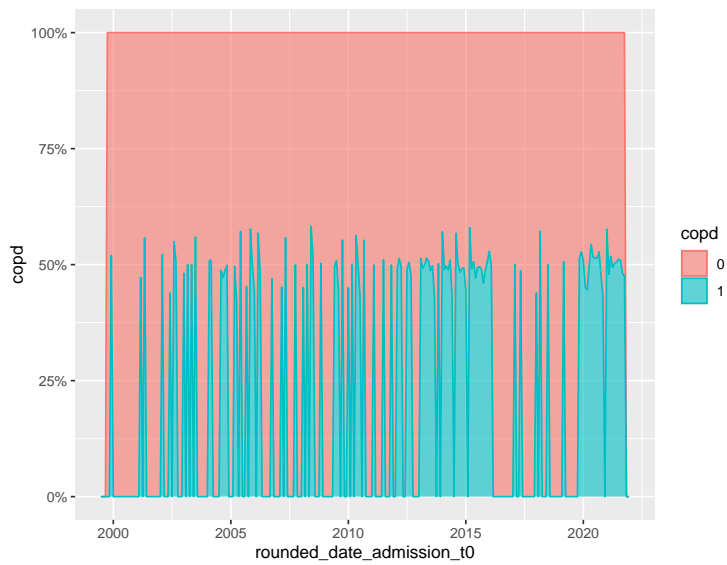
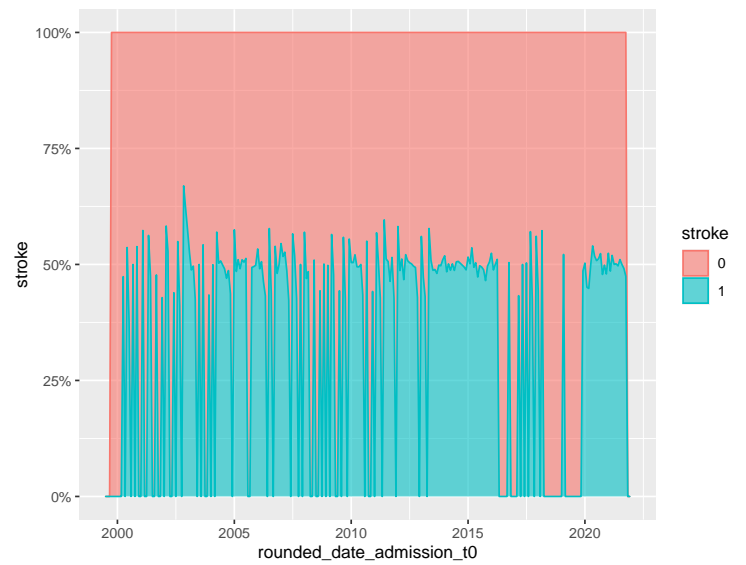
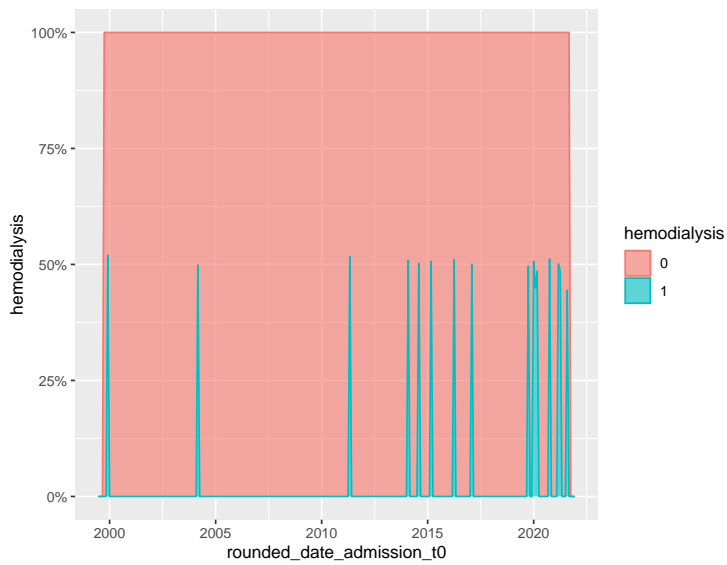
```
## [1] "icu_t0"
```

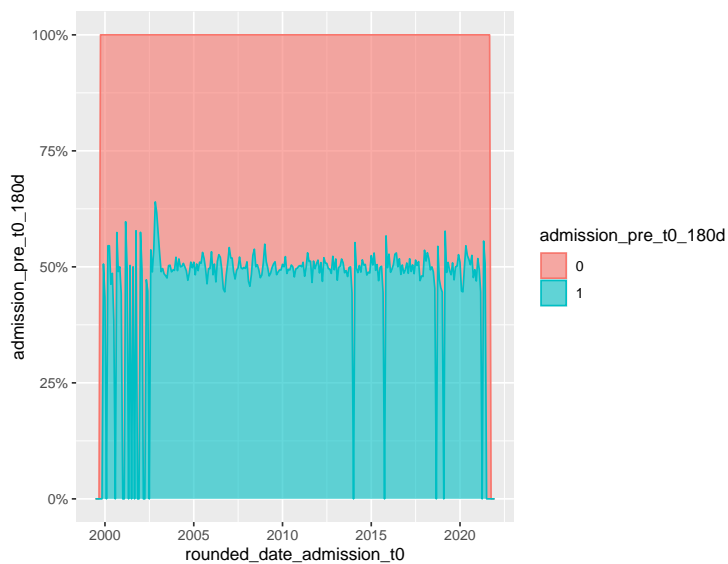
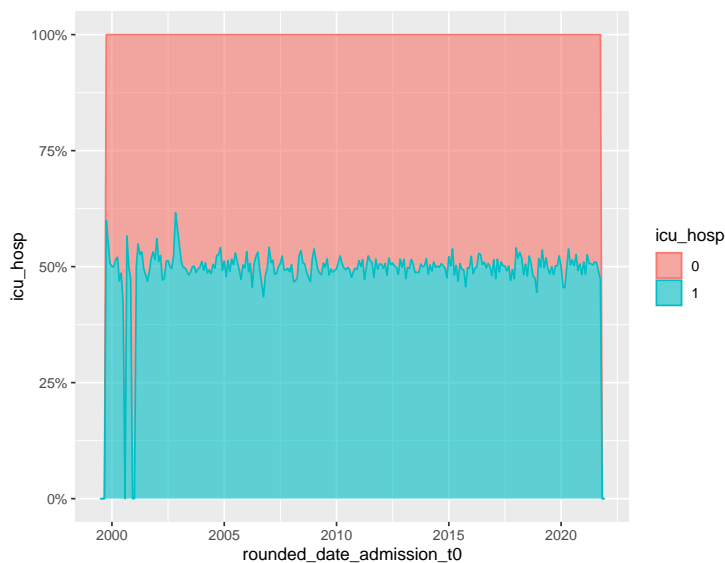
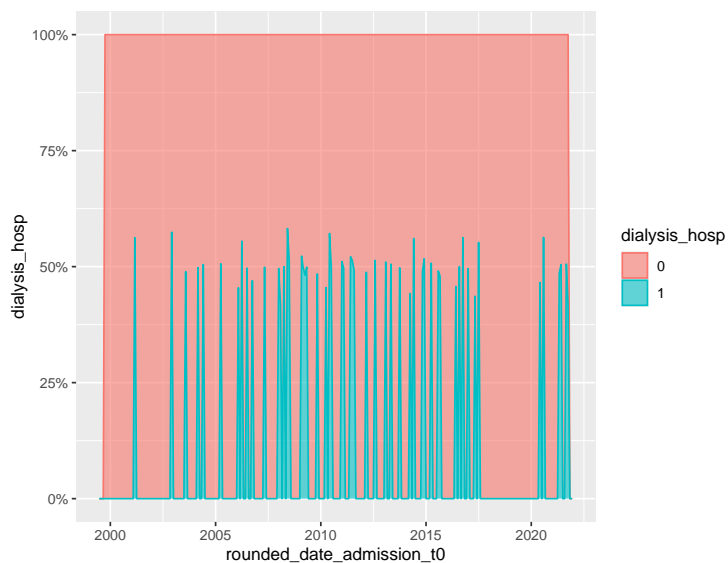
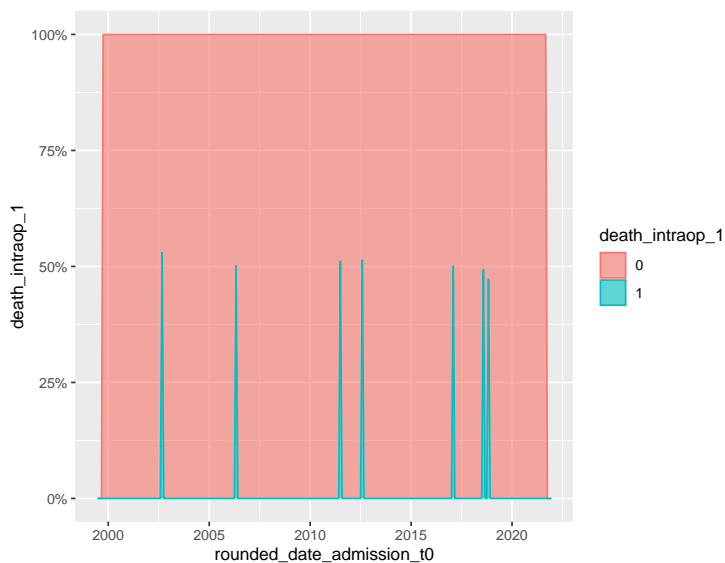
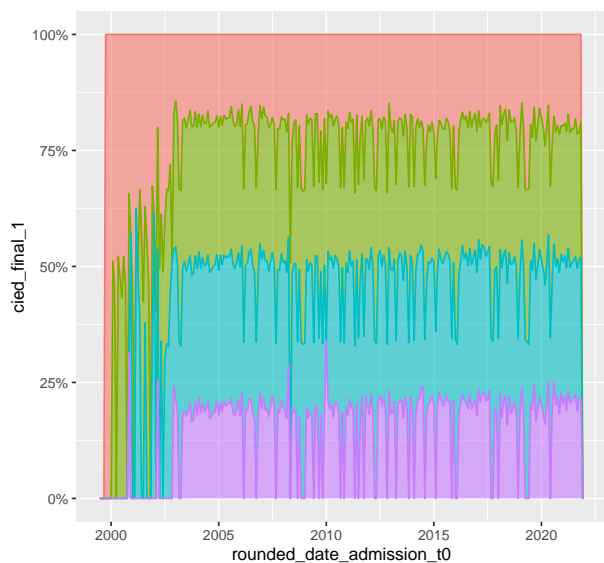
```
## [1] "dialysis_t0"
```











.61	2.82	20.16	20.99	21.77	22.99	24	25.47	26.4	27.66
.62	2.83	20.19	21	21.8	23	24.08	25.48	26.48	27.86
.63	2.85	20.22	21.01	21.82	23.05	24.09	25.54	26.58	27.9
.64	2.86	20.24	21.02	21.88	23.11	24.11	25.58	26.59	27.96
.65	2.87	20.3	21.05	21.89	23.12	24.15	25.6	26.67	28
.66	2.88	20.31	21.11	21.92	23.2	24.44	25.68	26.8	28.07
.67	2.89	20.35	21.18	213	23.25	24.45	25.72	26.89	28.1
.68	2.9	20.36	21.2	22	23.31	24.55	25.74	26.9	28.16
.69	2.91	20.7	21.24	22.05	23.35	24.69	25.79	26.96	28.16
.7	2.92	20.71	21.25	22.13	23.36	24.72	25.83	26.99	28.4
.72	2.93	20.74	21.31	22.2	23.53	24.75	25.88	27	28.46
.73	2.94	20.75	21.37	22.33	23.59	24.76	25.91	27.02	28.6
.74	2.95	20.77	21.43	22.37	23.61	24.78	25.94	27.04	28.73
.75	2.96	20.78	21.47	22.4	23.76	24.83	26	27.08	28.83
.76	2.97	20.87	21.5	22.65	23.82	24.94	26.04	27.13	28.86
.77	2.98	20.89	21.58	22.67	23.84	24.97	26.11	27.31	28.96
.78	2.99	20.93	21.6	22.75	23.85	25.05	26.2	27.59	29
.79	20	20.95	21.69	22.87	23.96	25.16	26.25	27.61	29.06
.8	20.05	20.96	21.71	22.91	23.98	25.17	26.35	27.62	29.13
.81	20.14	20.97	21.75	22.95	23.99	25.43	26.37	27.63	29.52

