Report

Signal detection of spontaneous medical device reports over time accounting for multiple comparisons

Ty Stanford et al.

Table of contents

1 Set up

1.1 Packages

```r
suppressPackageStartupMessages({
  library("readr")
  library("dplyr")
  library("tidyr")
  library("forcats")
  library("lubridate") # way to handle dates better than default R way
  library("ggplot2")
  library("ggrepel")
  library("knitr")
  library("gsDesign")
  library("arrow")
})
```

1.2 Load data

```r
sra_cum_bcpnn <-  read_parquet("out/sra_cum_bcpnn.parquet")


bcpnn_signif <-
  sra_cum_bcpnn %>%
  group_by(grps, dat_type, thresh) %>%
  arrange(dte) %>%
  dplyr::filter(reach_sig) %>%
  dplyr::filter(row_number() == 1) %>%
  ungroup()
```

2 Methods

2.1 Data aquisition

The data is thanks to curtis-murray at his MedicalDevicesNLP repo

- Natural language processing of the TGA spontaneous reports of medical device database (DAEN)
- Each record has an estimate of P(`topic == "pain"` | `Level`, `Doc`) using hierarchical stochastic block modelling (hSBM)
- P(`topic == "pain"` | `Level`, `Doc`) estimates for each record are roughly interpreted as the proportion of the NLP analysed free text that is considered as using/describing words related to pain

And example record and processing values:

- [to include here]

3 Analysis choices:

- Data structures - cumulative vs snapshot
- Threshold choose
- How many “looks”
- how to choose alpha spending
- sample size limitations for maxsprt

4 Plots

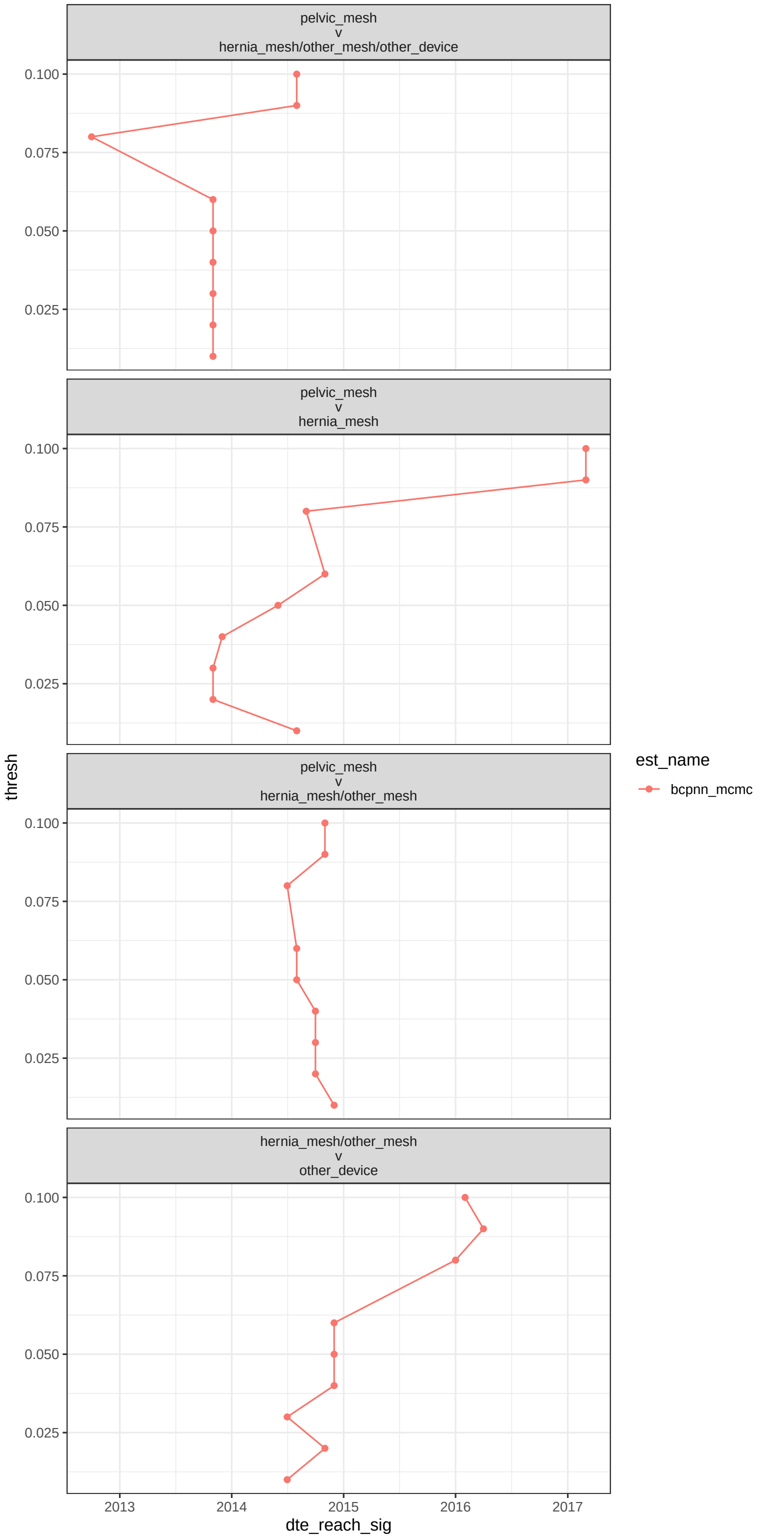
```r
bcpnn_signif_plt <-
  bcpnn_signif %>%
  # keep only multiples of 0.01 (too many colours otherwise)
  dplyr::filter(abs(100 * thresh - floor(100 * thresh)) < 1e-6) %>%
  mutate(
    grps = gsub(" v ", "\nv\n", grps),
    grps = fct_inorder(grps)
  )

thresholds <- sort(unique(bcpnn_signif_plt[["thresh"]]))
length(thresholds)
```

```
[1] 9
```

```r
thresh_scale <- rev(hcl.colors(length(thresholds) + 1, "Inferno"))[-1]
# thresh_scale <- rev(hcl.colors(length(thresholds), "SunsetDark"))

bcpnn_signif_plt %>%
  arrange(grps, thresh) %>%
  ggplot(., aes(x = dte_reach_sig, y = thresh, col = est_name)) +
  # ggplot(., aes(x = dte_reach_sig, y = est_name, col = factor(thresh))) +
  geom_point() +
  geom_path(aes(group = est_name)) +
  # scale_colour_viridis_c(option = "B", direction = -1) +
  # scale_colour_manual(values = thresh_scale) +
  facet_wrap(~ grps, ncol = 1) +
  theme_bw()
```


pelvic_mesh
v
hernia_mesh/other_mesh/other_device
pelvic_mesh
v
hernia_mesh
pelvic_mesh
v
hernia_mesh/other_mesh
hernia_mesh/other_mesh
v
other_device
0.100
0.075
0.050
0.025
thresh
2013
2014
2015
2016
2017
dte_reach_sig
est_name
bcpnn_mcmc

```
sra_cum_bcpnn_plt <-
  sra_cum_bcpnn %>%
  # keep only multiples of 0.01 (too many colours otherwise)
  dplyr::filter(abs(100 * thresh - floor(100 * thresh)) < 1e-6) %>%
  mutate(
    grps = gsub(" v ", "\nv\n", grps),
    grps = fct_inorder(grps)
  )

thresholds <- sort(unique(sra_cum_bcpnn_plt[["thresh"]]))
length(thresholds)
```

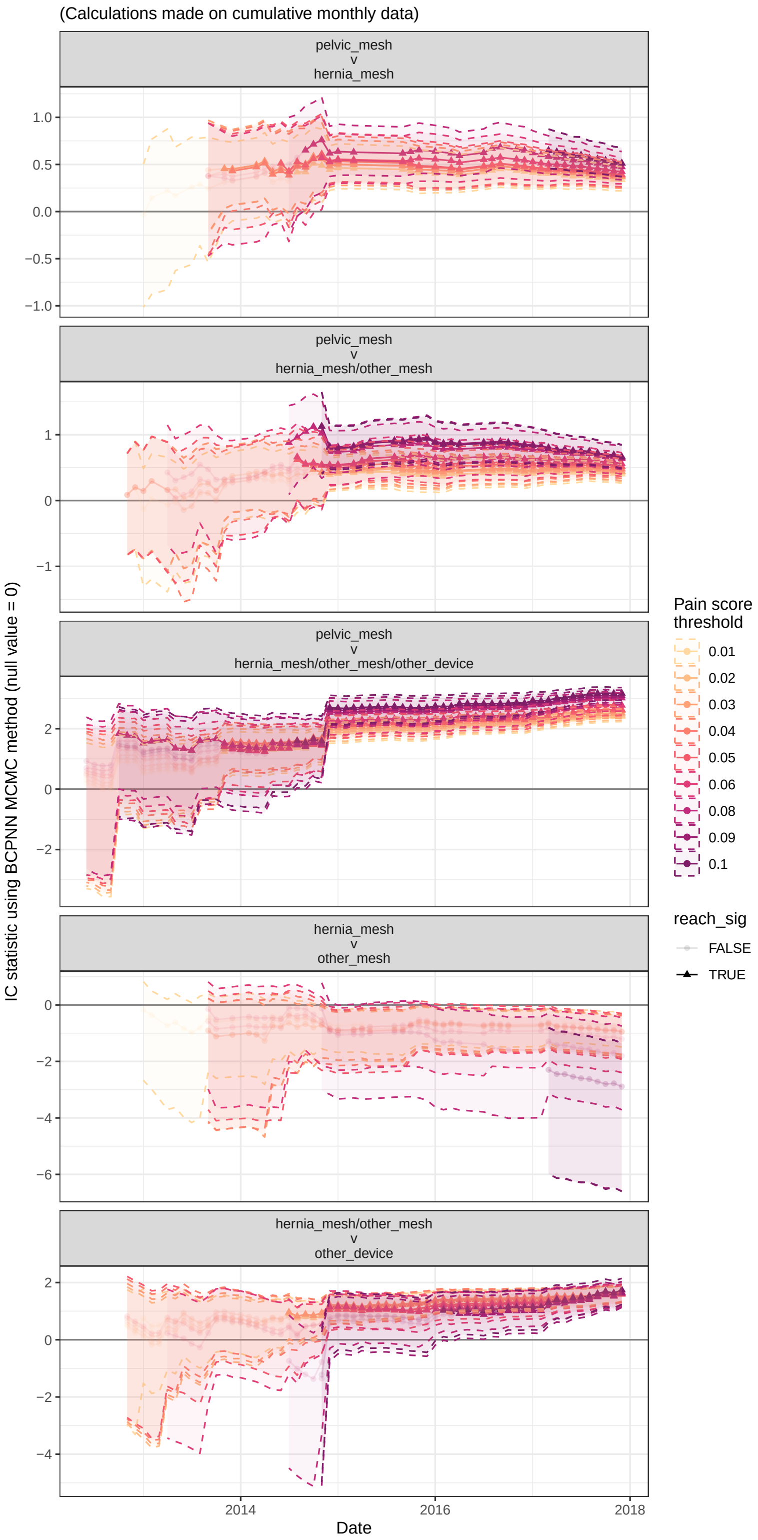
```
[1] 9
```

```
thresh_scale <- rev(hcl.colors(length(thresholds), "SunsetDark"))


sra_cum_bcpnn_plt %>%
  ggplot(
    .,
    aes(
      dte,
      est,
      ymax = ci_hi,
      ymin = ci_lo,
      col = factor(thresh),
      fill = factor(thresh),
      group = factor(thresh),
      alpha = reach_sig,
      shape = reach_sig
    )
  ) %+%
  geom_hline(aes(yintercept = 0), col = "grey50") %+% # null value
  geom_line() %+%
  geom_point() %+%
  geom_ribbon(alpha = 0.05, lty = 2)  %+%
  facet_wrap(~ grps, scales = "free_y", ncol = 1) %+%
  labs(
    subtitle = "(Calculations made on cumulative monthly data)",
    y = "IC statistic using BCPNN MCMC method (null value = 0)",
```

```r
    x = "Date",
    col = "Pain score\nthreshold",
    fill = "Pain score\nthreshold"
  ) %+%
  scale_colour_manual(values = thresh_scale, aesthetics = c("colour", "fill")) %+%
  theme_bw()
```

```
Warning: Using alpha for a discrete variable is not advised.
```


(Calculations made on cumulative monthly data)
pelvic_mesh
v
hernia_mesh
1.0
0.5
0.0
−0.5
−1.0
pelvic_mesh
v
hernia_mesh/other_mesh
1
0
−1
pelvic_mesh
v
hernia_mesh/other_mesh/other_device
2
0
−2
hernia_mesh
v
other_mesh
0
−2
−4
−6
hernia_mesh/other_mesh
v
other_device
2
0
−2
−4
2014
2016
2018
Date
IC statistic using BCPNN MCMC method (null value = 0)
Pain score
threshold
0.01
0.02
0.03
0.04
0.05
0.06
0.08
0.09
0.1
reach_sig
FALSE
TRUE

5 Session information

```
format(Sys.time(), '%d %b %Y')
```

```
[1] "19 Jun 2023"
```

```
Sys.info() %>% as.data.frame(.)
```

```
                             .
sysname                Windows
release                 10 x64
version            build 19044
nodename       DESKTOP-R5P5N23
machine                 x86-64
login                       ty
user                        ty
effective_user              ty
```

```
sessionInfo()
```

```
R version 4.2.2 (2022-10-31 ucrt)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 10 x64 (build 19044)

Matrix products: default

locale:
[1] LC_COLLATE=English_Australia.utf8  LC_CTYPE=English_Australia.utf8
[3] LC_MONETARY=English_Australia.utf8 LC_NUMERIC=C
[5] LC_TIME=English_Australia.utf8

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
 [1] arrow_11.0.0.2  gsDesign_3.4.0  knitr_1.42      ggrepel_0.9.3
 [5] ggplot2_3.4.1   lubridate_1.9.2 forcats_1.0.0   tidyr_1.3.0
 [9] dplyr_1.1.0     readr_2.1.4

loaded via a namespace (and not attached):
 [1] Rcpp_1.0.10      pillar_1.8.1     compiler_4.2.2   tools_4.2.2
 [5] bit_4.0.5        digest_0.6.31    jsonlite_1.8.4   evaluate_0.20
 [9] lifecycle_1.0.3  tibble_3.1.8     gtable_0.3.1     timechange_0.2.0
[13] pkgconfig_2.0.3  rlang_1.0.6      cli_3.6.0        rstudioapi_0.14
[17] yaml_2.3.7       xfun_0.37        fastmap_1.1.0    withr_2.5.0
[21] generics_0.1.3   vctrs_0.5.2      hms_1.1.2        bit64_4.0.5
[25] grid_4.2.2       tidyselect_1.2.0 glue_1.6.2       R6_2.5.1
[29] fansi_1.0.4      rmarkdown_2.20   farver_2.1.1     tzdb_0.3.0
[33] purrr_1.0.1      magrittr_2.0.3   scales_1.2.1     ellipsis_0.3.2
[37] htmltools_0.5.4  assertthat_0.2.1 xtable_1.8-4     colorspace_2.1-0
[41] labeling_0.4.2   utf8_1.2.3       munsell_0.5.0
```