Supplement 2

Baseline characteristics of RCTs for neuropathic pain

Peter Kamerman

Last knitted: 16 July 2020

Contents

---

1 Pain intensity inclusion thresholds

- Original data source: Finnerup et al 2015, Appendix 4[1].
  - All numeric values were used (after removing duplicate entries).
  - All visual analogue values were converted to a 0 to 10 scale.

```
# Inclusion criteria thresholds extracted from the supplementary
# materials of Finnerup et al., 2015
cut_off <- c(5, 4, 4, 2, 4, 3, 4, 6, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
             4, 6, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
             4, 4, 4, 4, 4, 4, 4, 3, 3, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 5, 3, 4,
             4, 4, 4, 4, 4, 4, 4, 4, 5, 5, 5, 5, 4, 4, 4, 3, 5, 4, 4, 4, 5, 5,
             4, 4, 5, 3, 4, 4, 4, 5, 4, 3, 3, 3, 3, 3, 3, 3, 4, 7, 5, 4, 3, 2,
             3, 4, 4, 2.5, 2.5, 4, 4, 4, 4, 5, 4, 4, 8, 4, 4, 4, 5, 3, 4, 4, 4)
```

[1] Finnerup NB, Attal N, Haroutounian S, McNicol E, Baron R, Dworkin RH, Gilron I, Haanpää M, Hansson P, Jensen TS, Kamerman PR, Lund K, Moore A, Raja SN, Rice ASC, Rowbotham M, Sena E, Siddall P, Smith BH, Wallace M. Pharmacotherapy for neuropathic pain in adults: a systematic review and meta-analysis. Lancet Neurol 2015;14:162–173. doi:10.1016/S1474-4422(14)70251-0

```r
# Generate summary statistics
## 6-number summary
summary(cut_off)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.000   4.000   4.000   4.015   4.000   8.000
```

```r
## Tabular summary of inclusion thresholds
data.frame(cutoff = cut_off) %>%
    group_by(cutoff) %>%
    summarise(count = n(),
              percent = round(100 * (count/length(cut_off)), 1)) %>%
    knitr::kable(., caption = 'Summary of pain inclusion thresholds')
```

Table 1: Summary of pain inclusion thresholds

cutoff	count	percent
2.0	2	1.5
2.5	2	1.5
3.0	16	12.2
4.0	93	71.0
5.0	14	10.7
6.0	2	1.5
7.0	1	0.8
8.0	1	0.8

---

2 Explore baseline pain intensity data

2.1 Import data

- Original data source: Finnerup et al 2015, Appendix 4[1],

```r
data <- read_csv('data/baseline-data.csv')
```

2.2 Explore data

2.2.1 Median

```r
data %>%
    filter(mean_or_median == 'median') %>%
    summarise(min = round(min(value, na.rm = TRUE), 1),
              Q25 = round(quantile(value, probs = 0.25, na.rm = TRUE), 1),
              mean = round(mean(value, na.rm = TRUE), 1),
              median = round(median(value, na.rm = TRUE), 1),
              Q75 = round(quantile(value, probs = 0.75, na.rm = TRUE), 1),
              max = round(max(value, na.rm = TRUE), 1)) %>%
    knitr::kable(., caption = 'Summary: median values')
```

Table 2: Summary: median values

min	Q25	mean	median	Q75	max
4.7	4.9	5.8	5	5.9	8.4

2.2.2 Mean

```
data %>%
    filter(mean_or_median == 'mean') %>%
    summarise(min = round(min(value, na.rm = TRUE), 1),
              Q25 = round(quantile(value, probs = 0.25, na.rm = TRUE), 1),
              mean = round(mean(value, na.rm = TRUE), 1),
              median = round(median(value, na.rm = TRUE), 1),
              Q75 = round(quantile(value, probs = 0.75, na.rm = TRUE), 1),
              max = round(max(value, na.rm = TRUE), 1)) %>%
    knitr::kable(., caption = 'Summary: mean values')
```

Table 3: Summary: mean values

min	Q25	mean	median	Q75	max
3.7	5	6.2	6.4	7.2	8.7

2.2.3 SD

```
data %>%
    filter(!is.na(sd)) %>%
    summarise(min = round(min(sd, na.rm = TRUE), 1),
              Q25 = round(quantile(sd, probs = 0.25, na.rm = TRUE), 1),
              mean = round(mean(sd, na.rm = TRUE), 1),
              median = round(median(sd, na.rm = TRUE), 1),
              Q75 = round(quantile(sd, probs = 0.75, na.rm = TRUE), 1),
              max = round(max(sd, na.rm = TRUE), 1)) %>%
    knitr::kable(., caption = 'Range of SD values')
```

Table 4: Range of SD values

min	Q25	mean	median	Q75	max
0.8	1.3	1.6	1.7	1.8	2.9

---

3 Weighted medians

- Three studies, four median values.
- Hahn et al., 2004 (active), Hahn et al., 2004 (placebo), Yuen et al., 2002, Low et al., 1995

```r
# Process data
median <- data %>%
    filter(included_in_analysis == 'yes') %>%
    filter(mean_or_median == 'median') %>%
    # Calculate total sample size
    mutate(total_sample = sum(sample_size)) %>%
    # Calculate weights
    mutate(weights = sample_size / total_sample)

## Weighted median
weighted.median(x = median$value,
                w = median$weights)
```

```
## [1] 5.07
```

---

4 Pooled mean

- Formula sourced from: https://www.ncbi.nlm.nih.gov/books/NBK56512/

```r
# Make bootstrap function
pooled_mean <- function(d, i){
    # Select data
    data <- d[i, ]

    # Process data
    mean <- data %>%
      # Get the required data
      filter(included_in_analysis == 'yes') %>%
      filter(mean_or_median == 'mean')

    # Calculate preliminary values
    mean <- mean %>%
      mutate(mean_by_n = value * sample_size)

    # Calculate pooled mean
    sum(mean$mean_by_n) / sum(mean$sample_size)
}

# Perform bootstrap
set.seed(2019)
mean_pooled <- boot(data = data,
                    statistic = pooled_mean,
                    R = 999,
                    stype = 'i',
                    parallel = 'multicore',
                    ncpus = 4)

# Pooled mean
mean_pooled$t0
```

```
## [1] 6.243982
```

```r
# Confidence interval of the pooled mean (percentile method)
boot.ci(mean_pooled, type = 'perc')
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 999 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = mean_pooled, type = "perc")
##
## Intervals :
## Level     Percentile
## 95%   ( 5.735,  6.652 )
## Calculations and Intervals on Original Scale
```

---

5 Pooled SD

- Formula sourced from: https://www.ncbi.nlm.nih.gov/books/NBK56512/

```r
# Make bootstrap function
pooled_sd <- function(d, i){
    # Select data
    data <- d[i, ]

    # Process the data
    sd <- data %>%
      filter(included_in_analysis == 'yes') %>%
      filter(mean_or_median == 'mean') %>%
      filter(!is.na(sd)) %>%
      # Square the SD
      mutate(SD_squared = sd^2) %>%
      # Calculate sample size -1
      mutate(n_minus_1 = sample_size - 1) %>%
      # Get the number of groups
      mutate(k = length(unique(.$authors))) %>%
      # Calculate (n-1)SD^2
      mutate(numerator = SD_squared * n_minus_1)

    # Pooled SD
    sqrt(sum(sd$numerator) / (sum(sd$sample_size) - sd$k[[1]]))
}

# Perform bootstrap
set.seed(2019)
sd_pooled <- boot(data = data,
                  statistic = pooled_sd,
                  R = 999,
                  stype = 'i',
                  parallel = 'multicore',
                  ncpus = 4)

# Pooled mean
```

```r
sd_pooled$t0
```

```
## [1] 1.665597
```

```r
# Confidence interval of the pooled mean (percentile method)
boot.ci(sd_pooled, type = 'perc')
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 999 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = sd_pooled, type = "perc")
##
## Intervals :
## Level     Percentile
## 95%   ( 1.362,  1.982 )
## Calculations and Intervals on Original Scale
```

---

6 Session information

```r
sessionInfo()
```

```
## R version 4.0.2 (2020-06-22)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Catalina 10.15.5
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRblas.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] boot_1.3-25         spatstat_1.64-1     rpart_4.1-15
##  [4] nlme_3.1-148        spatstat.data_1.4-3 forcats_0.5.0
##  [7] stringr_1.4.0       dplyr_1.0.0         purrr_0.3.4
## [10] readr_1.3.1         tidyr_1.1.0         tibble_3.0.1
## [13] ggplot2_3.3.2       tidyverse_1.3.0
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.4.6          lubridate_1.7.9       lattice_0.20-41
##  [4] deldir_0.1-25         assertthat_0.2.1      digest_0.6.25
##  [7] R6_2.4.1              cellranger_1.1.0      backports_1.1.8
## [10] reprex_0.3.0          evaluate_0.14         highr_0.8
## [13] httr_1.4.1            tensor_1.5            pillar_1.4.4
## [16] rlang_0.4.6           readxl_1.3.1          rstudioapi_0.11
## [19] blob_1.2.1            Matrix_1.2-18         goftest_1.2-2
```

```
## [22] rmarkdown_2.3          splines_4.0.2         polyclip_1.10-0
## [25] munsell_0.5.0          broom_0.5.6           compiler_4.0.2
## [28] modelr_0.1.8           xfun_0.15             pkgconfig_2.0.3
## [31] mgcv_1.8-31            htmltools_0.5.0       tidyselect_1.1.0
## [34] fansi_0.4.1            crayon_1.3.4          dbplyr_1.4.4
## [37] withr_2.2.0            grid_4.0.2            jsonlite_1.6.1
## [40] gtable_0.3.0           lifecycle_0.2.0       DBI_1.1.0
## [43] magrittr_1.5           scales_1.1.1          cli_2.0.2
## [46] stringi_1.4.6          fs_1.4.1              xml2_1.3.2
## [49] ellipsis_0.3.1         generics_0.0.2        vctrs_0.3.1
## [52] spatstat.utils_1.17-0 tools_4.0.2           glue_1.4.1
## [55] hms_0.5.3              parallel_4.0.2        abind_1.4-5
## [58] yaml_2.2.1             colorspace_1.4-1      rvest_0.3.5
## [61] knitr_1.29             haven_2.3.1
```