

Supplement 2

Pain characteristics

Peter Kamerman

Last knitted: 27 June 2019

Contents

Import data	2
Check for missingness	2
Pain as the reason for coming for testing	4
Point estimates	4
95% confidence intervals for the point estimates	6
95% confidence interval of the difference in proportions	8
Body sites affected by pain	9
Point estimates	9
95% confidence intervals for the point estimates	13
95% confidence interval of the difference in proportions	17
Number of body sites affected by pain	20
Point estimates	20
95% confidence intervals for the point estimates	24
95% confidence interval for the difference in medians	25
Body site with the worst pain	27
Point estimates	27
95% confidence intervals for the point estimates	31
95% confidence interval of the difference in proportions	35
Pain intensity	37
Point estimates	37
95% confidence intervals for the point estimates	41
95% confidence interval of the difference in means	41
Did you receive treatment for your pain?	43
Point estimates	43
95% confidence intervals for the point estimates	46
95% confidence interval of the difference in proportions	47
Session information	49

Note: only includes participants with pain

This script generates summaries of key descriptors of the pain phenotype for the subgroup of the cohort with pain, with and without conditioning on HIV status.

We present the data in tabular and graphical format, and calculate the precision of the estimates using bootstrap 95% confidence intervals.

To describe any differences between the HIV+ and HIV- groups, we have calculated 95% confidence intervals of the difference in mean/proportion.

Import data

```
# Import test result data and select columns
test <- read_rds('data-cleaned/hiv_test.rds') %>%
  select(PID,
         test_result) %>%
  rename(HIV_positive = test_result) %>%
  mutate(HIV_positive = ifelse(HIV_positive == 'HIV positive',
                              yes = 'yes',
                              no = 'no'))

# Import pain data and filter for those with pain
pain <- read_rds('data-cleaned/wbpq.rds') %>%
  filter(pain_in_last_month == 'yes') %>%
  filter(pain_worst > 0)

# Join datasets
data <- pain %>%
  left_join(test) %>%
  select(-starts_with('pain_cause'))
```

Check for missingness

```
# Extract columns used in this series of analyses
tmp <- data %>%
  select(HIV_positive, pain_in_last_month, pain_reason_for_visit, pain_worst, pain_treatment,
         head, shoulders, arms, hands, chest, abdomen, low_back, genitals,
         legs, feet, joints, muscles)

## Whole cohort
tmp %>%
  profile_missing() %>%
  mutate(pct_missing = round(100 * pct_missing)) %>%
  arrange(pct_missing)

## # A tibble: 17 x 3
##   feature          num_missing pct_missing
##   <fct>              <int>         <dbl>
## 1 pain_in_last_month         0           0
## 2 pain_worst                 0           0
## 3 pain_treatment             0           0
## 4 head                      0           0
## 5 shoulders                 0           0
## 6 arms                     0           0
## 7 hands                    0           0
## 8 chest                    0           0
## 9 abdomen                  0           0
```

```

## 10 low_back          0          0
## 11 genitals          0          0
## 12 legs              1          0
## 13 feet              1          0
## 14 joints            0          0
## 15 HIV_positive      2          1
## 16 pain_reason_for_visit 3          1
## 17 muscles           7          2

## By HIV status
### HIV-
tmp %>%
  filter(HIV_positive == 'no') %>%
  profile_missing() %>%
  mutate(pct_missing = round(100 * pct_missing)) %>%
  arrange(pct_missing)

## # A tibble: 17 x 3
##   feature          num_missing pct_missing
##   <fct>             <int>         <dbl>
## 1 HIV_positive      0             0
## 2 pain_in_last_month 0             0
## 3 pain_worst        0             0
## 4 pain_treatment    0             0
## 5 head              0             0
## 6 shoulders         0             0
## 7 arms              0             0
## 8 hands             0             0
## 9 chest             0             0
## 10 abdomen          0             0
## 11 low_back         0             0
## 12 genitals         0             0
## 13 legs             1             0
## 14 feet             1             0
## 15 joints           0             0
## 16 pain_reason_for_visit 3             1
## 17 muscles          5             2

### HIV+
tmp %>%
  filter(HIV_positive == 'yes') %>%
  profile_missing() %>%
  mutate(pct_missing = round(100 * pct_missing)) %>%
  arrange(pct_missing)

## # A tibble: 17 x 3
##   feature          num_missing pct_missing
##   <fct>             <int>         <dbl>
## 1 HIV_positive      0             0
## 2 pain_in_last_month 0             0
## 3 pain_reason_for_visit 0             0
## 4 pain_worst        0             0
## 5 pain_treatment    0             0
## 6 head              0             0
## 7 shoulders         0             0
## 8 arms              0             0

```

```
## 9 hands 0 0
## 10 chest 0 0
## 11 abdomen 0 0
## 12 low_back 0 0
## 13 genitals 0 0
## 14 legs 0 0
## 15 feet 0 0
## 16 joints 0 0
## 17 muscles 2 6
```

```
# Remove rows with missing HIV test results (n = 4)
```

```
data %<>%
  filter(!is.na(HIV_positive))
```

Pain as the reason for coming for testing

Point estimates

```
# Total cohort
```

```
data %>%
  filter(!is.na(pain_reason_for_visit)) %>%
  group_by(pain_reason_for_visit) %>%
  summarise(count = n()) %>%
  ungroup() %>%
  mutate(total = sum(count)) %>%
  mutate(proportion = round(count/total, 2)) %>%
  kable(caption = 'Pain was the reason for getting tested - total cohort (point estimates)')
```

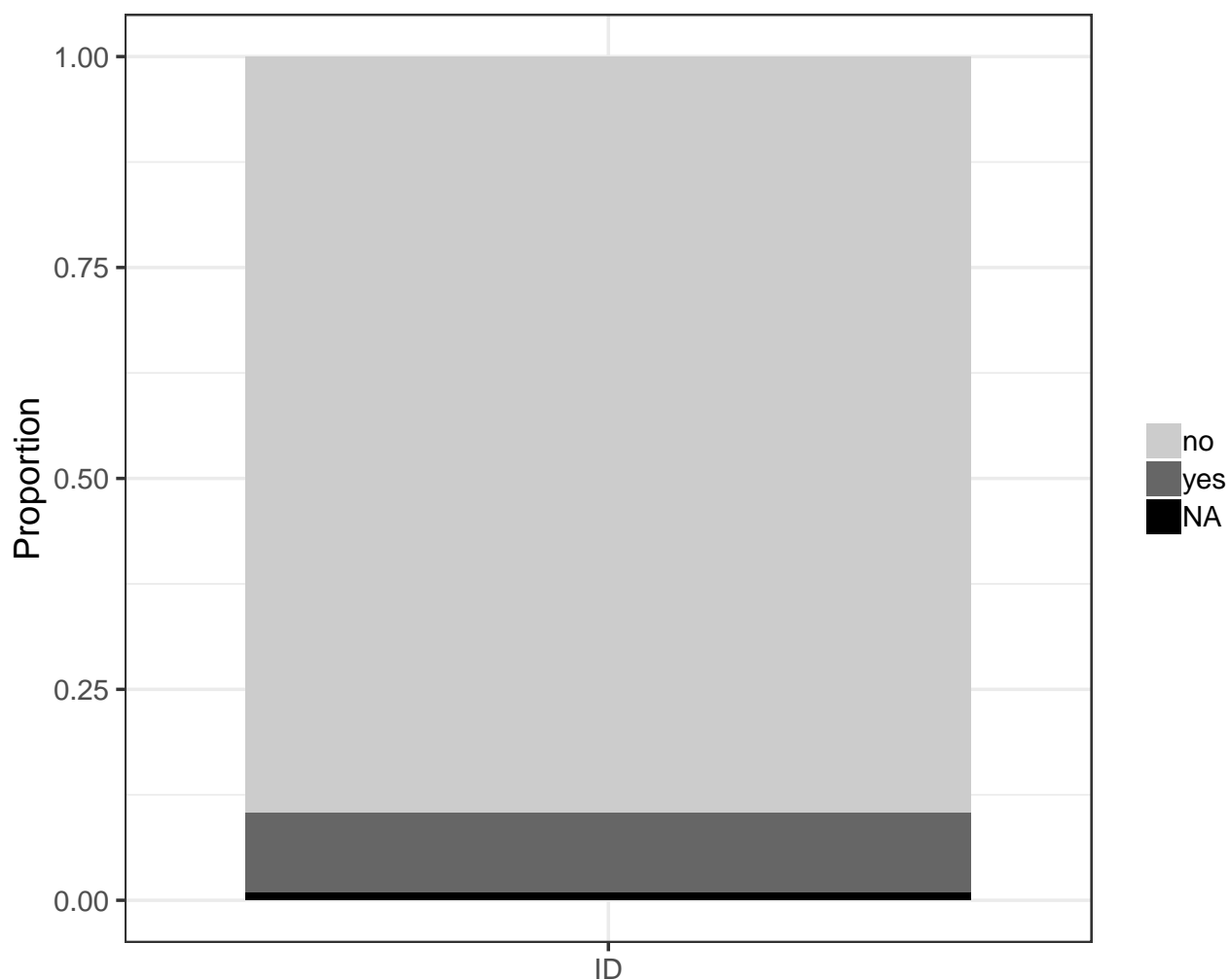
pain_reason_for_visit	count	total	proportion
no	283	313	0.9
yes	30	313	0.1

```
## Plot
```

```
ggplot(data = data) +
  aes(x = 'ID',
      fill = pain_reason_for_visit) +
  geom_bar(position = position_fill()) +
  scale_fill_manual(values = pal,
                    na.value = '#000000') +
  labs(title = 'Pain was the reason for getting tested',
       subtitle = 'Whole cohort',
       y = 'Proportion') +
  theme(legend.title = element_blank(),
        axis.title.x = element_blank())
```

Pain was the reason for getting tested

Whole cohort



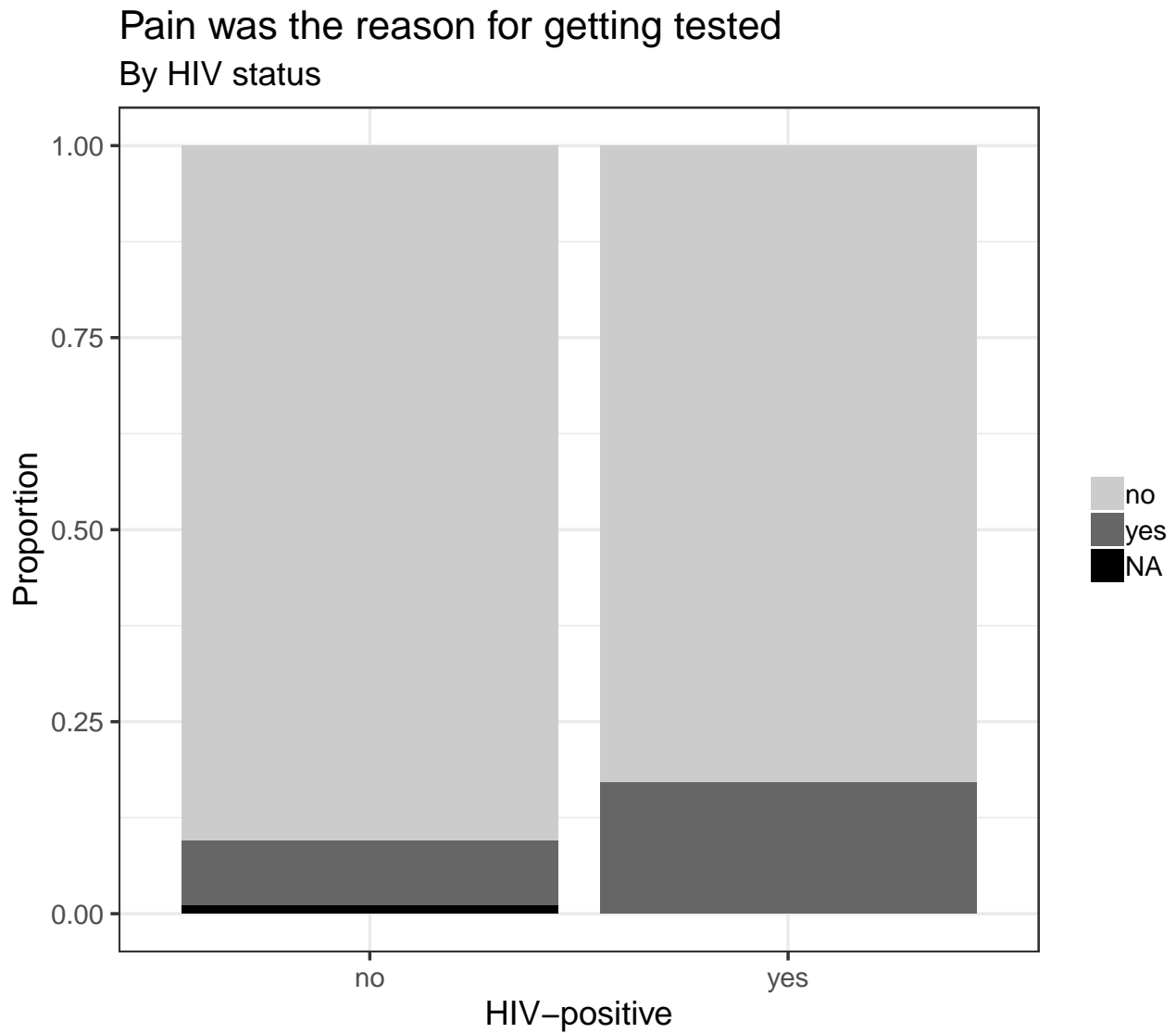
Count and proportion by HIV status

```
data %>%
  filter(!is.na(pain_reason_for_visit)) %>%
  filter(!is.na(HIV_positive)) %>%
  group_by(HIV_positive, pain_reason_for_visit) %>%
  summarise(count = n()) %>%
  group_by(HIV_positive) %>%
  mutate(total = sum(count)) %>%
  mutate(proportion = round(count/total, 2)) %>%
  kable(caption = 'Pain was the reason for getting tested - by HIV status (point estimates)')
```

HIV_positive	pain_reason_for_visit	count	total	proportion
no	no	254	278	0.91
no	yes	24	278	0.09
yes	no	29	35	0.83
yes	yes	6	35	0.17

Plot

```
ggplot(data = data) +
  aes(x = HIV_positive,
      fill = pain_reason_for_visit) +
  geom_bar(position = position_fill()) +
  scale_fill_manual(values = pal,
                    na.value = '#000000') +
  labs(title = 'Pain was the reason for getting tested',
       subtitle = 'By HIV status',
       y = 'Proportion',
       x = 'HIV-positive') +
  theme(legend.title = element_blank())
```



95% confidence intervals for the point estimates

```
# Boot functions
func_tmp <- function(d, i){
  data <- d[i, ]
  data <- data %>%
```

```

    filter(!is.na(pain_reason_for_visit))
  prop <- mean(data$pain_reason_for_visit == 'yes')
  prop
}

# Whole cohort
set.seed(2019)
boot_tmp <- boot(data = data,
  statistic = func_tmp,
  R = 999,
  stype = 'i')

bootci_tmp <- boot.ci(boot_tmp,
  type = 'perc')

tibble(pain_reason_for_visit = 'yes',
  proportion = round(boot_tmp$t0, 2),
  `lower 95% CI` = round(bootci_tmp$percent[[4]], 2),
  `upper 95% CI` = round(bootci_tmp$percent[[5]], 2)) %>%
  kable(caption = 'Pain was the reason for getting tested - total cohort (95% CI)')

```

pain_reason_for_visit	proportion	lower 95% CI	upper 95% CI
yes	0.1	0.06	0.13

```

# By HIV status (HIV- reported first)
set.seed(2019)
boot_tmp <- data %>%
  group_by(HIV_positive) %>%
  nest() %>%
  mutate(boot = map(.x = data,
    ~ boot(data = .x,
      statistic = func_tmp,
      R = 999,
      stype = 'i')))) %>%
  mutate(boot_ci = map(.x = boot,
    ~ boot.ci(.x,
      type = 'perc'))))

tibble(`HIV positive` = c('no', 'yes'),
  pain_reason_for_visit = c('yes', 'yes'),
  proportion = c(round(boot_tmp$boot[[1]]$t0, 2),
    round(boot_tmp$boot[[2]]$t0, 2)),
  `lower 95% CI` = c(round(boot_tmp$boot_ci[[1]]$percent[[4]], 2),
    round(boot_tmp$boot_ci[[2]]$percent[[4]], 2)),
  `upper 95% CI` = c(round(boot_tmp$boot_ci[[1]]$percent[[5]], 2),
    round(boot_tmp$boot_ci[[2]]$percent[[5]], 2)) %>%
  kable(caption = 'Pain was the reason for getting tested - by HIV status (95% CI)')

```

HIV positive	pain_reason_for_visit	proportion	lower 95% CI	upper 95% CI
no	yes	0.09	0.05	0.12
yes	yes	0.17	0.06	0.29

95% confidence interval of the difference in proportions

The difference in the proportion of HIV-positive participants who got tested because of pain vs participants who were HIV-negative (HIV+ minus HIV-).

```
# Boot function
func_tmp <- function(d, i){
  data <- d[i, ]
  data <- data %>%
    filter(!is.na(pain_reason_for_visit)) %>%
    filter(!is.na(HIV_positive))
  data_hiv <- filter(data, HIV_positive == 'yes')
  data_nohiv <- filter(data, HIV_positive == 'no')
  prop_yes <- mean(data_hiv$pain_reason_for_visit == 'yes')
  prop_no <- mean(data_nohiv$pain_reason_for_visit == 'yes')
  prop_yes - prop_no
}

# Confidence interval of the difference in proportions (HIV+ minus HIV-)
set.seed(2019)
boot_tmp <- boot(data = data,
  statistic = func_tmp,
  R = 999,
  stype = 'i')

bootci_tmp <- boot.ci(boot_tmp,
  type = 'perc')

tibble_tmp <- tibble(`difference in proportion` = round(boot_tmp$t0, 2),
  `lower 95% CI` = round(bootci_tmp$percent[[4]], 2),
  `upper 95% CI` = round(bootci_tmp$percent[[5]], 2))

tibble_tmp %>%
  kable(caption = 'Pain as the reason for getting tested - 95% CI of the difference (HIV+ minus HIV-)')

```

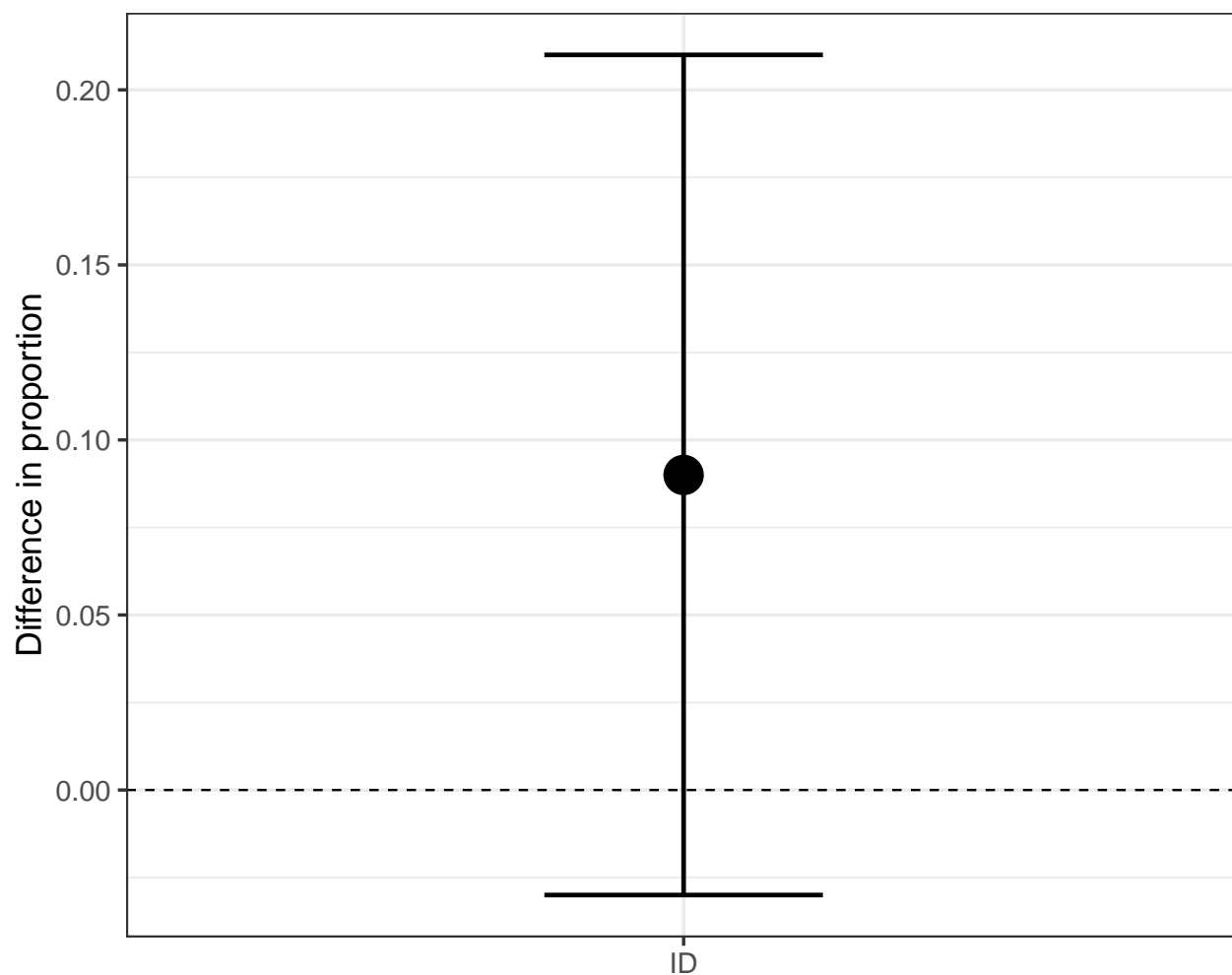
difference in proportion	lower 95% CI	upper 95% CI
0.09	-0.03	0.21

```
# Plot
ggplot(data = tibble_tmp) +
  aes(x = 'ID',
    y = `difference in proportion`,
    ymin = `lower 95% CI`,
    ymax = `upper 95% CI`) +
  geom_point(size = 8) +
  geom_errorbar(size = 1,
    width = 0.3) +
  geom_hline(yintercept = 0,
    linetype = 2) +
  labs(title = 'Pain was the reason for getting tested',
    subtitle = '95% CI of the difference in proportion (HIV+ minus HIV-)',
    y = 'Difference in proportion') +
  theme(axis.title.x = element_blank())

```


Pain was the reason for getting tested

95% CI of the difference in proportion (HIV+ minus HIV-)



Body sites affected by pain

Point estimates

```
# Total cohort
data %>%
  gather(key = site,
         value = pain_present,
         head, shoulders, arms, hands, chest, abdomen, low_back,
         genitals, legs, feet, joints, muscles) %>%
  group_by(site, pain_present) %>%
  summarise(count = n()) %>%
  group_by(site) %>%
  mutate(total = sum(count)) %>%
  group_by(site, pain_present) %>%
  mutate(proportion = round(count/total, 2)) %>%
```

```

select(-total) %>%
filter(pain_present == 'yes') %>%
arrange(desc(proportion)) %>%
kable(caption = 'Body sites affected by pain - total cohort')

```

site	pain_present	count	proportion
head	yes	211	0.67
low_back	yes	143	0.45
abdomen	yes	104	0.33
chest	yes	99	0.31
shoulders	yes	91	0.29
feet	yes	87	0.28
legs	yes	86	0.27
joints	yes	79	0.25
arms	yes	60	0.19
hands	yes	55	0.17
muscles	yes	51	0.16
genitals	yes	43	0.14

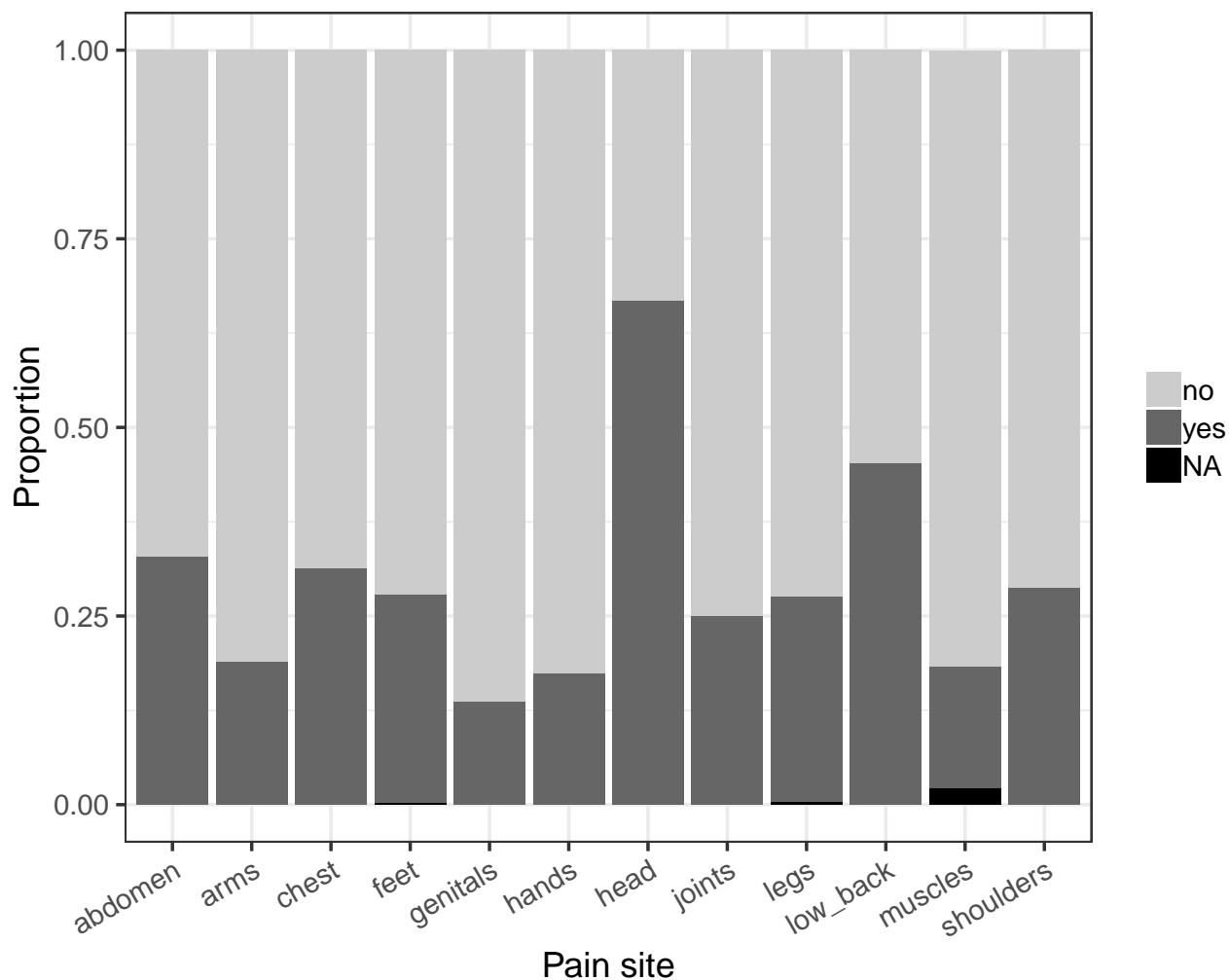
```

## Plot
data %>%
  gather(key = site,
         value = pain_present,
         head, shoulders, arms, hands, chest, abdomen, low_back,
         genitals, legs, feet, joints, muscles) %>%
  ggplot(data = .) +
  aes(x = site,
       fill = pain_present) +
  geom_bar(position = position_fill()) +
  scale_fill_manual(values = pal,
                    na.value = '#000000') +
  labs(title = 'Body sites affected by pain',
       subtitle = 'Whole cohort',
       y = 'Proportion',
       x = 'Pain site') +
  theme(legend.title = element_blank(),
        axis.text.x = element_text(angle = 30,
                                     hjust = 1))

```

Body sites affected by pain

Whole cohort



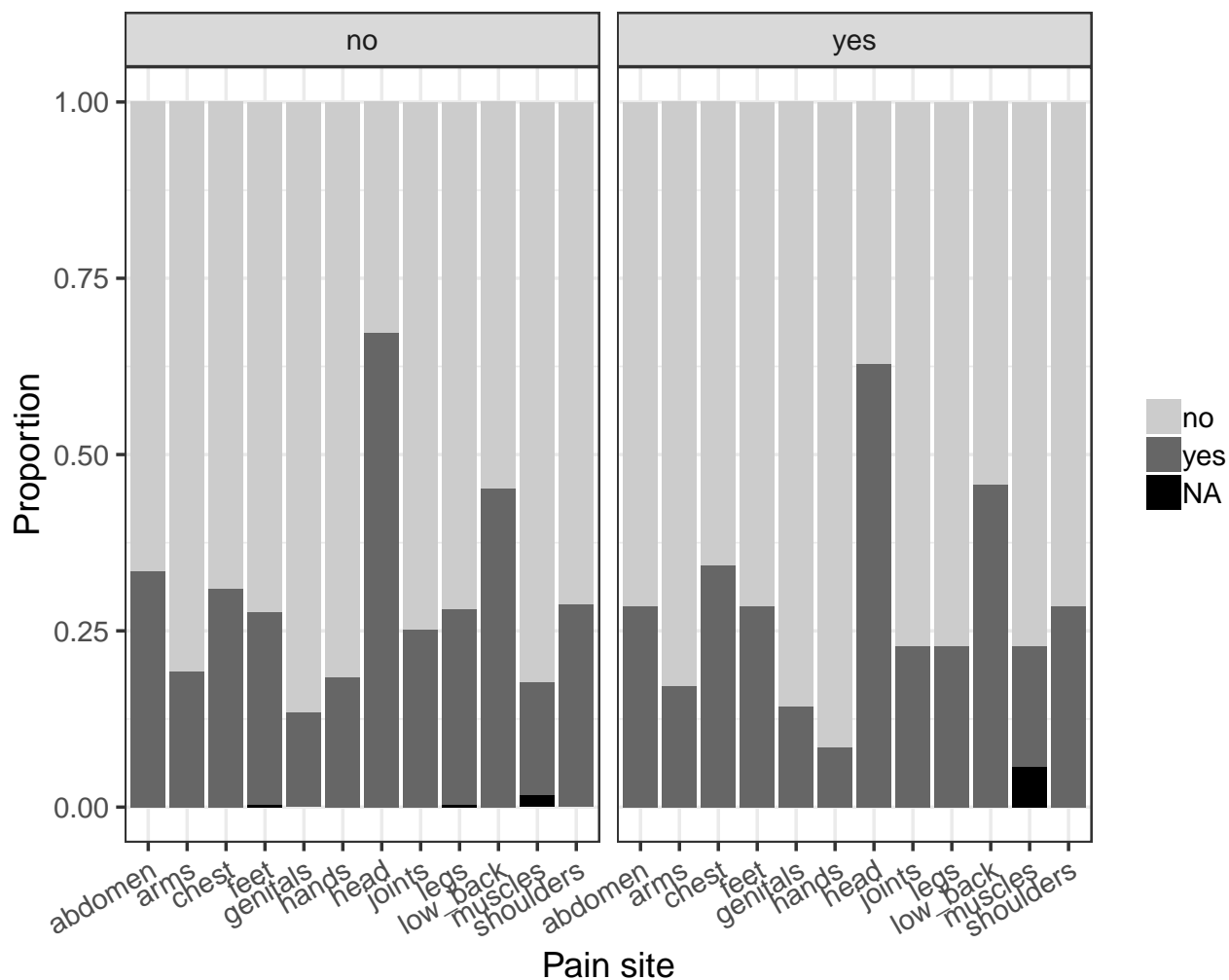
```
# By HIV status
data %>%
  gather(key = site,
         value = pain_present,
         head, shoulders, arms, hands, chest, abdomen, low_back,
         genitals, legs, feet, joints, muscles) %>%
  group_by(HIV_positive, site, pain_present) %>%
  summarise(count = n()) %>%
  group_by(HIV_positive, site) %>%
  mutate(total = sum(count)) %>%
  group_by(HIV_positive, site, pain_present) %>%
  mutate(proportion = round(count/total, 2)) %>%
  select(-total) %>%
  filter(pain_present == 'yes' & !is.na(HIV_positive)) %>%
  arrange(site, HIV_positive, desc(proportion)) %>%
  kable(caption = 'Body sites affected by pain - by HIV status')
```

HIV_positive	site	pain_present	count	proportion
no	abdomen	yes	94	0.33
yes	abdomen	yes	10	0.29
no	arms	yes	54	0.19
yes	arms	yes	6	0.17
no	chest	yes	87	0.31
yes	chest	yes	12	0.34
no	feet	yes	77	0.27
yes	feet	yes	10	0.29
no	genitals	yes	38	0.14
yes	genitals	yes	5	0.14
no	hands	yes	52	0.19
yes	hands	yes	3	0.09
no	head	yes	189	0.67
yes	head	yes	22	0.63
no	joints	yes	71	0.25
yes	joints	yes	8	0.23
no	legs	yes	78	0.28
yes	legs	yes	8	0.23
no	low_back	yes	127	0.45
yes	low_back	yes	16	0.46
no	muscles	yes	45	0.16
yes	muscles	yes	6	0.17
no	shoulders	yes	81	0.29
yes	shoulders	yes	10	0.29

```
## Plot
data %>%
  gather(key = site,
         value = pain_present,
         head, shoulders, arms, hands, chest, abdomen, low_back,
         genitals, legs, feet, joints, muscles) %>%
  ggplot(data = .) +
  aes(x = site,
      fill = pain_present) +
  geom_bar(position = position_fill()) +
  scale_fill_manual(values = pal,
                    na.value = '#000000') +
  labs(title = 'Body sites affected by pain',
       subtitle = 'By HIV status',
       y = 'Proportion',
       x = 'Pain site') +
  facet_wrap(~ HIV_positive,
             ncol = 2) +
  theme(legend.title = element_blank(),
        axis.text.x = element_text(angle = 30,
                                     hjust = 1))
```

Body sites affected by pain

By HIV status



95% confidence intervals for the point estimates

```
# Boot functions
func_tmp <- function(d, i){
  data <- d[i, ]
  data <- data %>%
    filter(!is.na(pain_present))
  prop <- mean(data$pain_present == 'yes')
  prop
}

# Whole cohort
set.seed(2019)
boot_tmp <- data %>%
  gather(key = site,
         value = pain_present,
         head, shoulders, arms, hands, chest, abdomen, low_back,
```

```

      genitals, legs, feet, joints, muscles) %>%
group_by(site) %>%
nest() %>%
mutate(boot = map(.x = data,
  ~ boot(data = .x,
    statistic = func_tmp,
    R = 999,
    stype = 'i',
    parallel = 'multicore',
    ncpus = 7))) %>%
mutate(boot_ci = map(.x = boot,
  ~ boot.ci(.x,
    type = 'perc'))))

tibble(`body site` = c('head', 'shoulders', 'arms', 'hands', 'chest', 'abdomen',
  'low_back', 'genitals', 'legs', 'feet', 'joints', 'muscles'),
`pain present` = rep('yes', 12),
proportion = c(round(boot_tmp$boot[[1]]$t0, 2),
  round(boot_tmp$boot[[2]]$t0, 2),
  round(boot_tmp$boot[[3]]$t0, 2),
  round(boot_tmp$boot[[4]]$t0, 2),
  round(boot_tmp$boot[[5]]$t0, 2),
  round(boot_tmp$boot[[6]]$t0, 2),
  round(boot_tmp$boot[[7]]$t0, 2),
  round(boot_tmp$boot[[8]]$t0, 2),
  round(boot_tmp$boot[[9]]$t0, 2),
  round(boot_tmp$boot[[10]]$t0, 2),
  round(boot_tmp$boot[[11]]$t0, 2),
  round(boot_tmp$boot[[12]]$t0, 2)),
`lower 95% CI` = c(round(boot_tmp$boot_ci[[1]]$percent[[4]], 2),
  round(boot_tmp$boot_ci[[2]]$percent[[4]], 2),
  round(boot_tmp$boot_ci[[3]]$percent[[4]], 2),
  round(boot_tmp$boot_ci[[4]]$percent[[4]], 2),
  round(boot_tmp$boot_ci[[5]]$percent[[4]], 2),
  round(boot_tmp$boot_ci[[6]]$percent[[4]], 2),
  round(boot_tmp$boot_ci[[7]]$percent[[4]], 2),
  round(boot_tmp$boot_ci[[8]]$percent[[4]], 2),
  round(boot_tmp$boot_ci[[9]]$percent[[4]], 2),
  round(boot_tmp$boot_ci[[10]]$percent[[4]], 2),
  round(boot_tmp$boot_ci[[11]]$percent[[4]], 2),
  round(boot_tmp$boot_ci[[12]]$percent[[4]], 2)),
`upper 95% CI` = c(round(boot_tmp$boot_ci[[1]]$percent[[5]], 2),
  round(boot_tmp$boot_ci[[2]]$percent[[5]], 2),
  round(boot_tmp$boot_ci[[3]]$percent[[5]], 2),
  round(boot_tmp$boot_ci[[4]]$percent[[5]], 2),
  round(boot_tmp$boot_ci[[5]]$percent[[5]], 2),
  round(boot_tmp$boot_ci[[6]]$percent[[5]], 2),
  round(boot_tmp$boot_ci[[7]]$percent[[5]], 2),
  round(boot_tmp$boot_ci[[8]]$percent[[5]], 2),
  round(boot_tmp$boot_ci[[9]]$percent[[5]], 2),
  round(boot_tmp$boot_ci[[10]]$percent[[5]], 2),
  round(boot_tmp$boot_ci[[11]]$percent[[5]], 2),
  round(boot_tmp$boot_ci[[12]]$percent[[5]], 2))) %>%
kable(caption = 'Body sites affected by pain - whole cohort (95% CI)')

```

body site	pain present	proportion	lower 95% CI	upper 95% CI
head	yes	0.67	0.62	0.72
shoulders	yes	0.29	0.24	0.34
arms	yes	0.19	0.15	0.23
hands	yes	0.17	0.14	0.22
chest	yes	0.31	0.26	0.37
abdomen	yes	0.33	0.28	0.37
low_back	yes	0.45	0.40	0.51
genitals	yes	0.14	0.10	0.18
legs	yes	0.27	0.22	0.33
feet	yes	0.28	0.23	0.33
joints	yes	0.25	0.20	0.30
muscles	yes	0.17	0.13	0.21

```

# By HIV status (HIV- reported first)
set.seed(2019)
boot_tmp <- data %>%
  gather(key = site,
         value = pain_present,
         head, shoulders, arms, hands, chest, abdomen, low_back,
         genitals, legs, feet, joints, muscles) %>%
  group_by(HIV_positive, site) %>%
  nest() %>%
  mutate(boot = map(.x = data,
                    ~ boot(data = .x,
                          statistic = func_tmp,
                          R = 999,
                          stype = 'i',
                          parallel = 'multicore',
                          ncpus = 7))) %>%
  mutate(boot_ci = map(.x = boot,
                      ~ boot.ci(.x,
                                type = 'perc'))))

tibble(`HIV positive` = rep(c('no', 'yes'), 12),
       `body site` = rep(c('head', 'shoulders', 'arms', 'hands', 'chest', 'abdomen',
                          'low_back', 'genitals', 'legs', 'feet', 'joints', 'muscles'),
                          each = 2),
       `pain present` = rep('yes', 24),
       proportion = c(round(boot_tmp$boot[[1]]$t0, 2),
                      round(boot_tmp$boot[[2]]$t0, 2),
                      round(boot_tmp$boot[[3]]$t0, 2),
                      round(boot_tmp$boot[[4]]$t0, 2),
                      round(boot_tmp$boot[[5]]$t0, 2),
                      round(boot_tmp$boot[[6]]$t0, 2),
                      round(boot_tmp$boot[[7]]$t0, 2),
                      round(boot_tmp$boot[[8]]$t0, 2),
                      round(boot_tmp$boot[[9]]$t0, 2),
                      round(boot_tmp$boot[[10]]$t0, 2),
                      round(boot_tmp$boot[[11]]$t0, 2),
                      round(boot_tmp$boot[[12]]$t0, 2),
                      round(boot_tmp$boot[[13]]$t0, 2),
                      round(boot_tmp$boot[[14]]$t0, 2),

```

```

round(boot_tmp$boot[[15]]$t0, 2),
round(boot_tmp$boot[[16]]$t0, 2),
round(boot_tmp$boot[[17]]$t0, 2),
round(boot_tmp$boot[[18]]$t0, 2),
round(boot_tmp$boot[[19]]$t0, 2),
round(boot_tmp$boot[[20]]$t0, 2),
round(boot_tmp$boot[[21]]$t0, 2),
round(boot_tmp$boot[[22]]$t0, 2),
round(boot_tmp$boot[[23]]$t0, 2),
round(boot_tmp$boot[[24]]$t0, 2)),
`lower 95% CI` = c(round(boot_tmp$boot_ci[[1]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[2]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[3]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[4]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[5]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[6]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[7]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[8]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[9]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[10]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[11]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[12]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[13]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[14]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[15]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[16]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[17]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[18]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[19]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[20]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[21]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[22]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[23]]$percent[[4]], 2),
round(boot_tmp$boot_ci[[24]]$percent[[4]], 2)),
`upper 95% CI` = c(round(boot_tmp$boot_ci[[1]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[2]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[3]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[4]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[5]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[6]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[7]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[8]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[9]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[10]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[11]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[12]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[13]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[14]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[15]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[16]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[17]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[18]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[19]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[20]]$percent[[5]], 2),

```



```

round(boot_tmp$boot_ci[[21]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[22]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[23]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[24]]$percent[[5]], 2))) %>%
kable(caption = 'Body sites affected by pain - by HIV status (95% CI)')

```

HIV positive	body site	pain present	proportion	lower 95% CI	upper 95% CI
no	head	yes	0.67	0.62	0.72
yes	head	yes	0.63	0.46	0.77
no	shoulders	yes	0.29	0.23	0.34
yes	shoulders	yes	0.29	0.14	0.43
no	arms	yes	0.19	0.15	0.23
yes	arms	yes	0.17	0.06	0.29
no	hands	yes	0.19	0.14	0.23
yes	hands	yes	0.09	0.00	0.17
no	chest	yes	0.31	0.26	0.36
yes	chest	yes	0.34	0.17	0.51
no	abdomen	yes	0.33	0.27	0.40
yes	abdomen	yes	0.29	0.14	0.43
no	low_back	yes	0.45	0.40	0.51
yes	low_back	yes	0.46	0.31	0.60
no	genitals	yes	0.14	0.10	0.17
yes	genitals	yes	0.14	0.03	0.29
no	legs	yes	0.28	0.22	0.33
yes	legs	yes	0.23	0.11	0.37
no	feet	yes	0.28	0.22	0.33
yes	feet	yes	0.29	0.14	0.46
no	joints	yes	0.25	0.20	0.30
yes	joints	yes	0.23	0.09	0.37
no	muscles	yes	0.16	0.12	0.21
yes	muscles	yes	0.18	0.06	0.31

95% confidence interval of the difference in proportions

The difference in the proportion of HIV-positive participants with pain at a body site vs participants who were HIV-negative (HIV+ minus HIV-).

```

# Boot function
func_tmp <- function(d, i){
  data <- d[i, ]
  data <- data %>%
    filter(!is.na(pain_present)) %>%
    filter(!is.na(HIV_positive))
  data_hiv <- filter(data, HIV_positive == 'yes')
  data_nohiv <- filter(data, HIV_positive == 'no')
  prop_yes <- mean(data_hiv$pain_present == 'yes')
  prop_no <- mean(data_nohiv$pain_present == 'yes')
  prop_yes - prop_no
}

# Confidence interval of the difference in proportions (HIV+ minus HIV-)
set.seed(2019)

```

```

boot_tmp <- data %>%
  gather(key = site,
         value = pain_present,
         head, shoulders, arms, hands, chest, abdomen, low_back,
         genitals, legs, feet, joints, muscles) %>%
  group_by(site) %>%
  nest() %>%
  mutate(boot = map(.x = data,
                    ~ boot(data = .x,
                          statistic = func_tmp,
                          R = 999,
                          stype = 'i',
                          parallel = 'multicore',
                          ncpus = 7))) %>%
  mutate(boot_ci = map(.x = boot,
                      ~ boot.ci(.x,
                                type = 'perc'))))

tibble_tmp <- tibble(`body site` = factor(c('head', 'shoulders', 'arms', 'hands', 'chest', 'abdomen',
                                           'low_back', 'genitals', 'legs', 'feet', 'joints', 'muscles'),
                                           levels = c('head', 'shoulders', 'arms', 'hands', 'chest', 'abdomen',
                                           'low_back', 'genitals', 'legs', 'feet', 'joints', 'muscles'),
                                           ordered = TRUE),
  `pain present` = rep('yes', 12),
  `difference in proportion` = c(round(boot_tmp$boot[[1]]$t0, 2),
                                round(boot_tmp$boot[[2]]$t0, 2),
                                round(boot_tmp$boot[[3]]$t0, 2),
                                round(boot_tmp$boot[[4]]$t0, 2),
                                round(boot_tmp$boot[[5]]$t0, 2),
                                round(boot_tmp$boot[[6]]$t0, 2),
                                round(boot_tmp$boot[[7]]$t0, 2),
                                round(boot_tmp$boot[[8]]$t0, 2),
                                round(boot_tmp$boot[[9]]$t0, 2),
                                round(boot_tmp$boot[[10]]$t0, 2),
                                round(boot_tmp$boot[[11]]$t0, 2),
                                round(boot_tmp$boot[[12]]$t0, 2)),
  `lower 95% CI` = c(round(boot_tmp$boot_ci[[1]]$percent[[4]], 2),
                    round(boot_tmp$boot_ci[[2]]$percent[[4]], 2),
                    round(boot_tmp$boot_ci[[3]]$percent[[4]], 2),
                    round(boot_tmp$boot_ci[[4]]$percent[[4]], 2),
                    round(boot_tmp$boot_ci[[5]]$percent[[4]], 2),
                    round(boot_tmp$boot_ci[[6]]$percent[[4]], 2),
                    round(boot_tmp$boot_ci[[7]]$percent[[4]], 2),
                    round(boot_tmp$boot_ci[[8]]$percent[[4]], 2),
                    round(boot_tmp$boot_ci[[9]]$percent[[4]], 2),
                    round(boot_tmp$boot_ci[[10]]$percent[[4]], 2),
                    round(boot_tmp$boot_ci[[11]]$percent[[4]], 2),
                    round(boot_tmp$boot_ci[[12]]$percent[[4]], 2)),
  `upper 95% CI` = c(round(boot_tmp$boot_ci[[1]]$percent[[5]], 2),
                    round(boot_tmp$boot_ci[[2]]$percent[[5]], 2),
                    round(boot_tmp$boot_ci[[3]]$percent[[5]], 2),
                    round(boot_tmp$boot_ci[[4]]$percent[[5]], 2),
                    round(boot_tmp$boot_ci[[5]]$percent[[5]], 2),
                    round(boot_tmp$boot_ci[[6]]$percent[[5]], 2),
                    round(boot_tmp$boot_ci[[7]]$percent[[5]], 2),
                    round(boot_tmp$boot_ci[[8]]$percent[[5]], 2),
                    round(boot_tmp$boot_ci[[9]]$percent[[5]], 2),
                    round(boot_tmp$boot_ci[[10]]$percent[[5]], 2),
                    round(boot_tmp$boot_ci[[11]]$percent[[5]], 2),
                    round(boot_tmp$boot_ci[[12]]$percent[[5]], 2))

```

```

round(boot_tmp$boot_ci[[7]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[8]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[9]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[10]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[11]]$percent[[5]], 2),
round(boot_tmp$boot_ci[[12]]$percent[[5]], 2)))

```

```
tibble_tmp %>%
```

```

kable(caption = 'Body sites affected by pain - 95% CI of the difference (HIV+ minus HIV-)')

```

body site	pain present	difference in proportion	lower 95% CI	upper 95% CI
head	yes	-0.04	-0.21	0.12
shoulders	yes	0.00	-0.15	0.16
arms	yes	-0.02	-0.14	0.12
hands	yes	-0.10	-0.19	0.01
chest	yes	0.03	-0.13	0.20
abdomen	yes	-0.05	-0.21	0.11
low_back	yes	0.01	-0.17	0.19
genitals	yes	0.01	-0.11	0.15
legs	yes	-0.05	-0.20	0.11
feet	yes	0.01	-0.15	0.17
joints	yes	-0.02	-0.17	0.13
muscles	yes	0.02	-0.11	0.16

```
# Plot
```

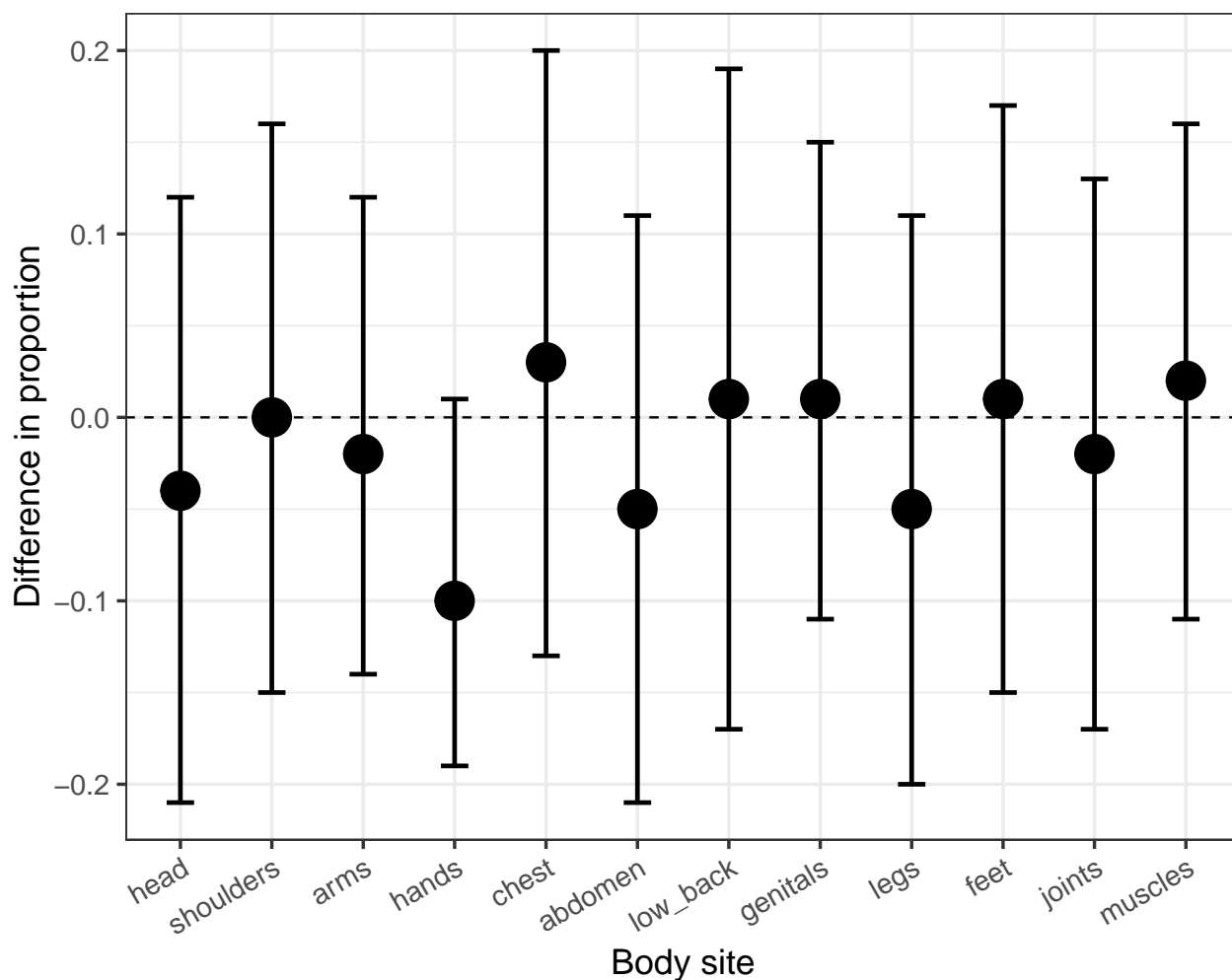
```

ggplot(data = tibble_tmp) +
  aes(x = `body site`,
      y = `difference in proportion`,
      ymin = `lower 95% CI`,
      ymax = `upper 95% CI`) +
  geom_point(size = 8) +
  geom_errorbar(size = 1,
               width = 0.3) +
  geom_hline(yintercept = 0,
             linetype = 2) +
  labs(title = 'Body sites affected by pain',
       subtitle = '95% CI of the difference in proportion (HIV+ minus HIV-)',
       y = 'Difference in proportion',
       x = 'Body site') +
  theme(axis.text.x = element_text(angle = 30,
                                    hjust = 1))

```

Body sites affected by pain

95% CI of the difference in proportion (HIV+ minus HIV-)



Number of body sites affected by pain

Point estimates

```
# Total cohort
data %>%
  gather(key = site,
         value = pain_present,
         head, shoulders, arms, hands, chest, abdomen, low_back,
         genitals, legs, feet, joints, muscles) %>%
  filter(pain_present == 'yes') %>%
  group_by(PID, pain_present) %>%
  summarise(count = n()) %>%
  ungroup() %>%
  summarise(mean = round(mean(count), 2),
```

```

SD = round(sd(count), 2),
min = min(count),
max = max(count)) %>%
kable(caption = 'Number of body sites with pain - total cohort')

```

mean	SD	min	max
3.51	2.18	1	12

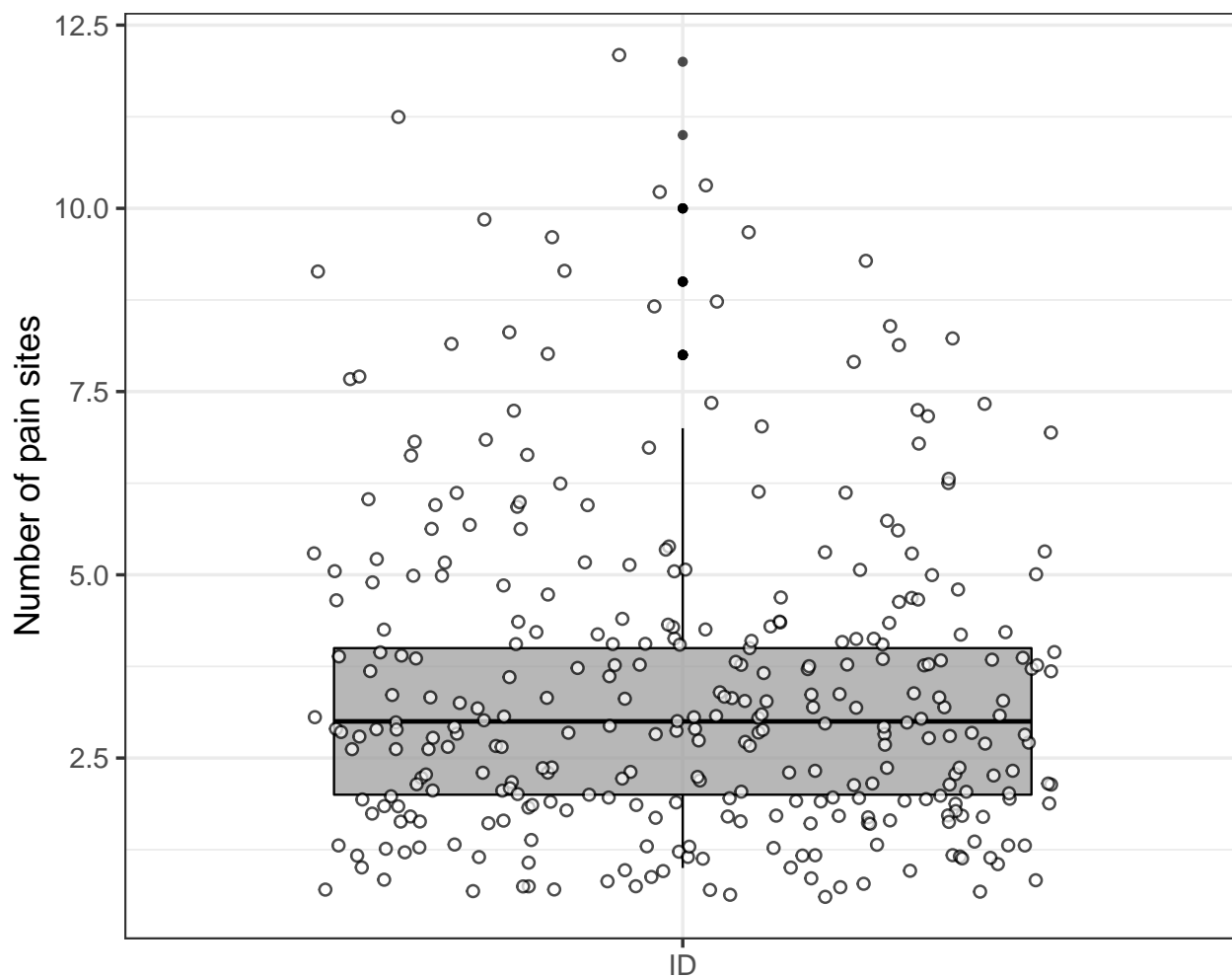
```

## Plot
data %>%
  gather(key = site,
         value = pain_present,
         head, shoulders, arms, hands, chest, abdomen, low_back,
         genitals, legs, feet, joints, muscles) %>%
  filter(pain_present == 'yes') %>%
  group_by(PID, pain_present) %>%
  summarise(count = n()) %>%
  ungroup() %>%
  ggplot(data = .) +
  aes(x = 'ID',
      y = count) +
  geom_boxplot(alpha = 0.7,
              colour = '#000000',
              fill = '#999999') +
  geom_point(position = position_jitter(),
            fill = '#FFFFFF',
            alpha = 0.7,
            stroke = 0.8,
            size = 2,
            shape = 21) +
  scale_fill_manual(values = pal,
                   na.value = '#000000') +
  labs(title = 'Number of body sites with pain',
       subtitle = 'Whole cohort',
       y = 'Number of pain sites') +
  theme(legend.title = element_blank(),
        axis.title.x = element_blank())

```

Number of body sites with pain

Whole cohort



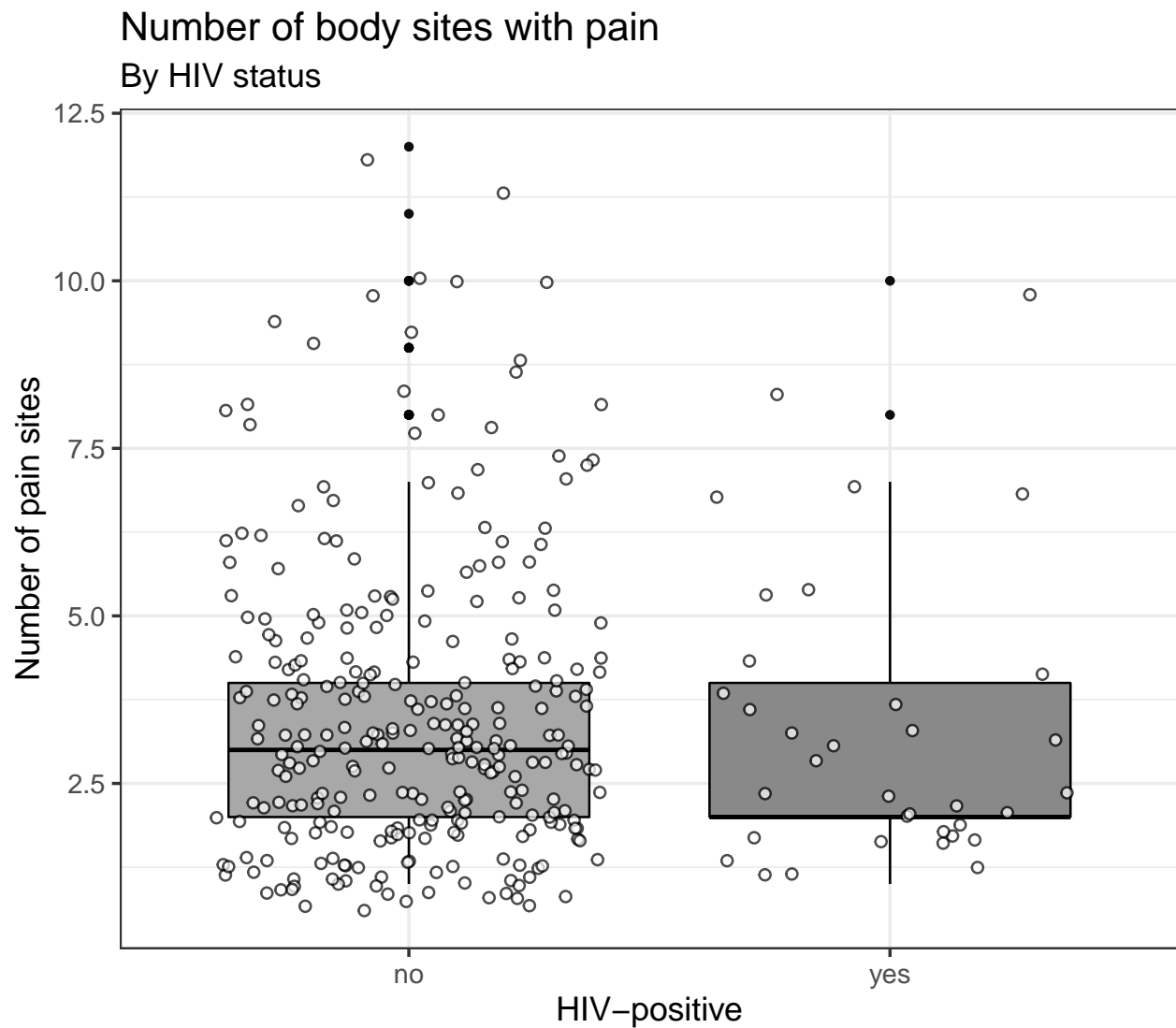
By HIV status

```
data %>%
  gather(key = site,
         value = pain_present,
         head, shoulders, arms, hands, chest, abdomen, low_back,
         genitals, legs, feet, joints, muscles) %>%
  filter(pain_present == 'yes') %>%
  group_by(PID, HIV_positive, pain_present) %>%
  summarise(count = n()) %>%
  group_by(HIV_positive) %>%
  summarise(mean = round(mean(count), 2),
            SD = round(sd(count), 2),
            min = min(count),
            max = max(count)) %>%
  kable(caption = 'Number of body sites with pain - by HIV status')
```

HIV_positive	mean	SD	min	max
no	3.53	2.18	1	12

HIV_positive	mean	SD	min	max
yes	3.31	2.18	1	10

```
## Plot
data %>%
  gather(key = site,
          value = pain_present,
          head, shoulders, arms, hands, chest, abdomen, low_back,
          genitals, legs, feet, joints, muscles) %>%
  filter(pain_present == 'yes') %>%
  group_by(PID, HIV_positive, pain_present) %>%
  summarise(count = n()) %>%
  ungroup() %>%
  ggplot(data = .) +
  aes(x = HIV_positive,
       y = count,
       fill = HIV_positive) +
  geom_boxplot() + geom_boxplot(alpha = 0.7,
                                colour = '#000000',
                                fill = '#999999') +
  geom_point(position = position_jitter(),
             fill = '#FFFFFF',
             alpha = 0.7,
             stroke = 0.8,
             size = 2,
             shape = 21) +
  scale_fill_manual(values = pal,
                    na.value = '#000000') +
  labs(title = 'Number of body sites with pain',
       subtitle = 'By HIV status',
       y = 'Number of pain sites',
       x = 'HIV-positive') +
  theme(legend.position = 'none')
```



95% confidence intervals for the point estimates

```
# Use the Rcompanion groupwiseMean for numeric data

# Whole cohort
## Data
tmp <- data %>%
  gather(key = site,
         value = pain_present,
         head, shoulders, arms, hands, chest, abdomen, low_back,
         genitals, legs, feet, joints, muscles) %>%
  filter(pain_present == 'yes') %>%
  group_by(PID, pain_present) %>%
  summarise(count = n()) %>%
  ungroup()

## Calculation
set.seed(2019)
```



```

groupwiseMean(count ~ 1,
              data = tmp,
              bca = TRUE,
              R = 999) %>%
select(Mean, Bca.lower, Bca.upper) %>%
rename(mean = Mean,
       `lower 95% CI` = Bca.lower,
       `upper 95% CI` = Bca.upper) %>%
kable(caption = 'Number of body sites with pain - total cohort (95% CI')

```

mean	lower 95% CI	upper 95% CI
3.51	3.28	3.76

By HIV status (HIV- reported first)

Data

```

tmp <- data %>%
gather(key = site,
      value = pain_present,
      head, shoulders, arms, hands, chest, abdomen, low_back,
      genitals, legs, feet, joints, muscles) %>%
filter(pain_present == 'yes') %>%
group_by(PID, HIV_positive, pain_present) %>%
summarise(count = n()) %>%
ungroup()

```

Calculation

set.seed(2019)

```

groupwiseMean(count ~ HIV_positive,
              data = tmp,
              bca = TRUE,
              R = 999) %>%
select(HIV_positive, Mean, Bca.lower, Bca.upper) %>%
rename(mean = Mean,
       `lower 95% CI` = Bca.lower,
       `upper 95% CI` = Bca.upper) %>%
kable(caption = 'Number of body sites with pain - by HIV status (95% CI')

```

HIV_positive	mean	lower 95% CI	upper 95% CI
no	3.53	3.30	3.82
yes	3.31	2.71	4.11

95% confidence interval for the difference in medians

Boot function

```

func_tmp <- function(d, i){
  data <- d[i, ]
  tmp <- data %>%
gather(key = site,
      value = pain_present,
      head, shoulders, arms, hands, chest, abdomen, low_back,
      genitals, legs, feet, joints, muscles) %>%

```

```

  filter(pain_present == 'yes')
hiv_p <- tmp %>%
  filter(HIV_positive == 'yes') %>%
  group_by(PID, pain_present) %>%
  summarise(count = n()) %>%
  ungroup() %>%
  summarise(median = median(count))
hiv_n <- tmp %>%
  filter(HIV_positive == 'no') %>%
  group_by(PID, pain_present) %>%
  summarise(count = n()) %>%
  ungroup() %>%
  summarise(median = median(count))
hiv_p$median - hiv_n$median
}

# Difference in medians
set.seed(2019)
boot_tmp <- boot(data = data,
  R = 999,
  statistic = func_tmp,
  stype = 'i')

bootci_tmp <- boot.ci(boot_tmp,
  type = 'perc') # BCa gives extreme order statistics warning

tibble_tmp <- tibble(`difference in medians` = bootci_tmp$t0,
  `lower 95% CI` = bootci_tmp$percent[[4]],
  `upper 95% CI` = bootci_tmp$percent[[5]])

tibble_tmp %>%
  kable(caption = 'Number of body sites with pain - 95% CI of the difference (HIV+ minus HIV-)')

```

difference in medians	lower 95% CI	upper 95% CI
-1	-1.5	1.5

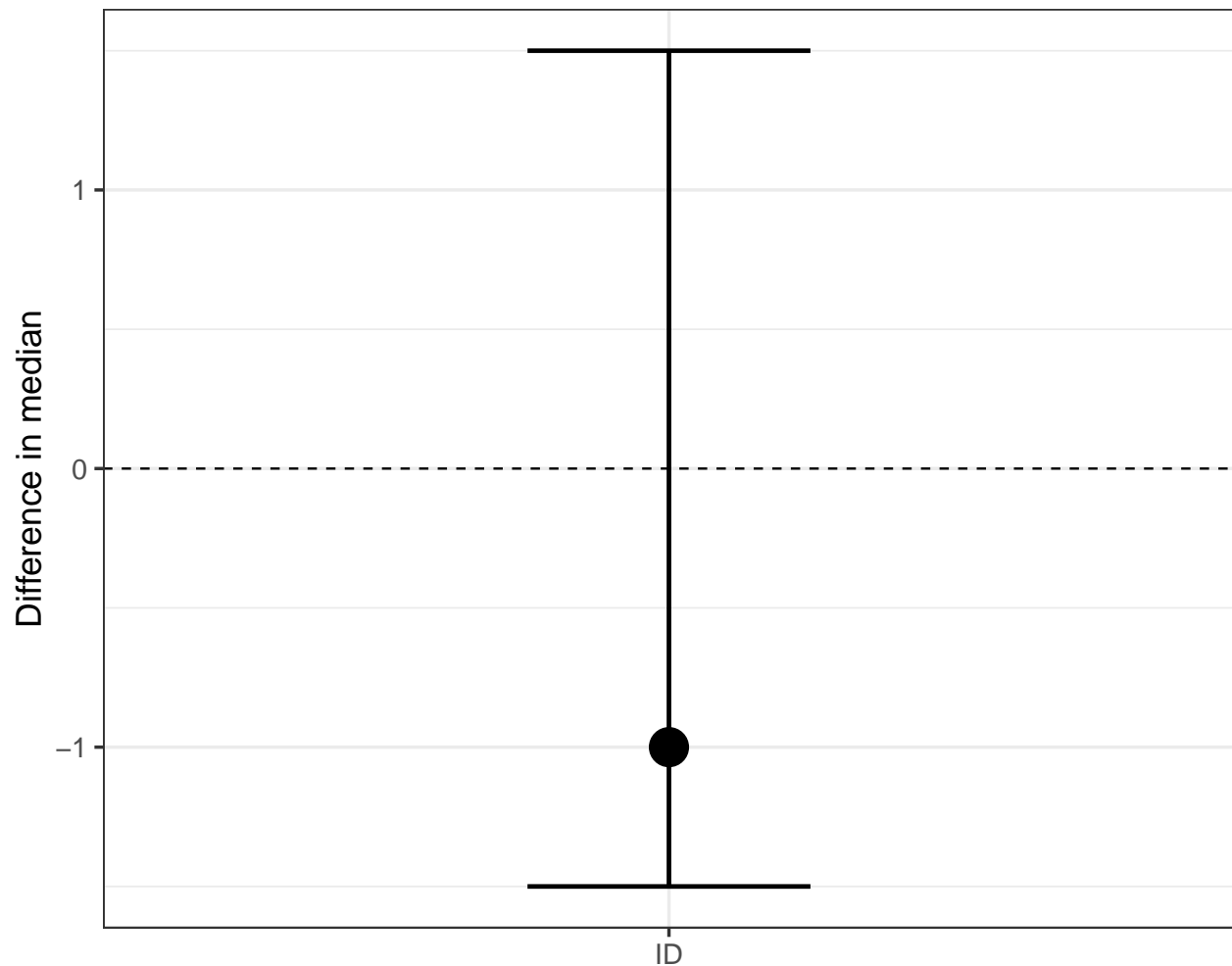
```

# Plot
ggplot(data = tibble_tmp) +
  aes(x = 'ID',
    y = `difference in medians`,
    ymin = `lower 95% CI`,
    ymax = `upper 95% CI`) +
  geom_point(size = 8) +
  geom_errorbar(size = 1,
    width = 0.3) +
  geom_hline(yintercept = 0,
    linetype = 2) +
  labs(title = 'Number of painful sites',
    subtitle = '95% CI of the difference in median (HIV+ minus HIV-)',
    y = 'Difference in median') +
  theme(axis.title.x = element_blank())

```

Number of painful sites

95% CI of the difference in median (HIV+ minus HIV-)



Body site with the worst pain

Point estimates

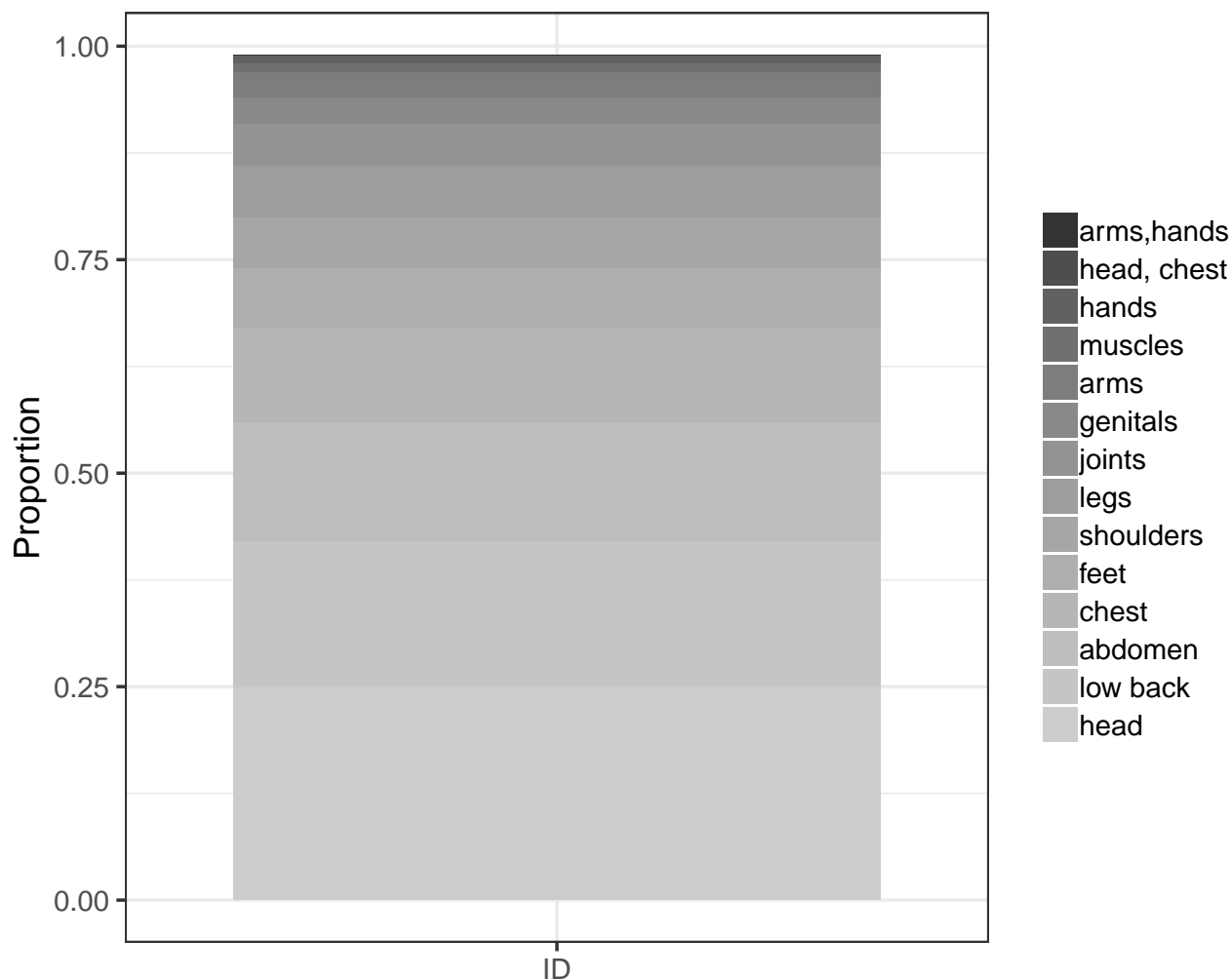
```
# Total cohort
data %>%
  select(PID, HIV_positive, site_of_worst_pain_1) %>%
  rename(`pain site` = site_of_worst_pain_1) %>%
  filter(!is.na(`pain site`)) %>%
  group_by(`pain site`) %>%
  summarise(count = n()) %>%
  arrange(desc(count)) %>%
  ungroup() %>%
  mutate(`total count` = sum(count),
         proportion = round(count / `total count`, 2)) %>%
  kable(caption = 'Site of the worst pain - whole cohort')
```

pain site	count	total count	proportion
head	78	307	0.25
low back	53	307	0.17
abdomen	44	307	0.14
chest	33	307	0.11
feet	22	307	0.07
legs	19	307	0.06
shoulders	19	307	0.06
joints	14	307	0.05
arms	8	307	0.03
genitals	8	307	0.03
muscles	4	307	0.01
hands	3	307	0.01
arms, hands	1	307	0.00
head, chest	1	307	0.00

```
## Plot
data %>%
  select(PID, HIV_positive, site_of_worst_pain_1) %>%
  rename(`pain site` = site_of_worst_pain_1) %>%
  filter(!is.na(`pain site`)) %>%
  group_by(`pain site`) %>%
  summarise(count = n()) %>%
  ungroup() %>%
  mutate(`total count` = sum(count),
         proportion = round(count / `total count`, 2)) %>%
  mutate(`pain site` = factor(`pain site`),
         `pain site` = fct_reorder(`pain site`, proportion)) %>%
ggplot(data = .) +
  aes(x = 'ID',
      y = proportion,
      fill = `pain site`) +
  geom_col() +
  scale_fill_grey() +
  labs(title = 'Site of the worst pain',
       subtitle = 'Whole cohort',
       y = 'Proportion') +
  theme(legend.title = element_blank(),
        axis.title.x = element_blank())
```

Site of the worst pain

Whole cohort

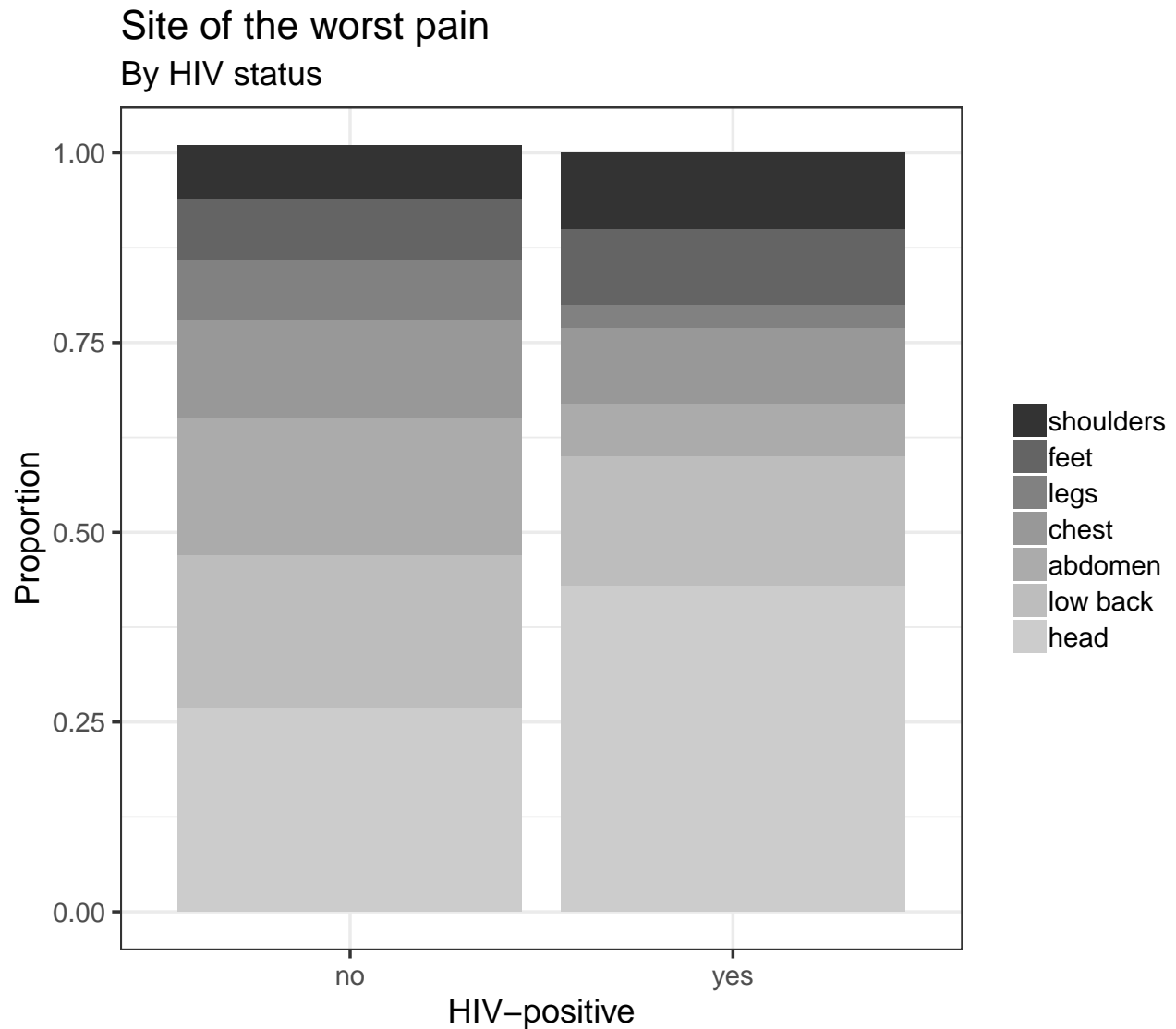


```
# By HIV status for sites >= 5% (see table above)
sites <- c('head', 'low back', 'abdomen', 'chest', 'feet', 'legs', 'shoulders')
data %>%
  select(PID, HIV_positive, site_of_worst_pain_1) %>%
  rename(`pain site` = site_of_worst_pain_1) %>%
  filter(!is.na(`pain site`)) %>%
  filter(`pain site` %in% sites) %>%
  group_by(HIV_positive, `pain site`) %>%
  summarise(count = n()) %>%
  group_by(HIV_positive) %>%
  mutate(`total count` = sum(count),
         proportion = round(count / `total count`, 2)) %>%
  arrange(`pain site`, HIV_positive) %>%
  kable(caption = 'Site of the worst pain - by HIV status for sites >= 5%')
```

HIV_positive	pain site	count	total count	proportion
no	abdomen	42	238	0.18
yes	abdomen	2	30	0.07

HIV_positive	pain site	count	total count	proportion
no	chest	30	238	0.13
yes	chest	3	30	0.10
no	feet	19	238	0.08
yes	feet	3	30	0.10
no	head	65	238	0.27
yes	head	13	30	0.43
no	legs	18	238	0.08
yes	legs	1	30	0.03
no	low back	48	238	0.20
yes	low back	5	30	0.17
no	shoulders	16	238	0.07
yes	shoulders	3	30	0.10

```
## Plot
data %>%
  select(PID, HIV_positive, site_of_worst_pain_1) %>%
  rename(`pain site` = site_of_worst_pain_1) %>%
  filter(!is.na(`pain site`)) %>%
  filter(`pain site` %in% sites) %>%
  group_by(HIV_positive, `pain site`) %>%
  summarise(count = n()) %>%
  mutate(`total count` = sum(count),
         proportion = round(count / `total count`, 2)) %>%
  mutate(`pain site` = factor(`pain site`),
         `pain site` = fct_reorder(`pain site`, proportion)) %>%
  ungroup() %>%
  ggplot(data = .) +
  aes(x = HIV_positive,
      y = proportion,
      fill = `pain site`) +
  geom_col() +
  scale_fill_grey() +
  labs(title = 'Site of the worst pain',
       subtitle = 'By HIV status',
       y = 'Proportion',
       x = 'HIV-positive') +
  theme(legend.title = element_blank())
```



95% confidence intervals for the point estimates

```
# Boot functions
func_tmp <- function(d, i, site){
  data <- d[i, ]
  mean(data$site_of_worst_pain_1 == site,
        na.rm = TRUE)
}

# Whole cohort
## Bootstrap CI for all sites
set.seed(2019)
map_tmp <- map(.x = unique(data$site_of_worst_pain_1[!is.na(data$site_of_worst_pain_1)]),
               ~ boot(data = data,
                      site = .x,
                      R = 999,
                      statistic = func_tmp,
```

```

      stype = 'i'))

map_tmp <- map(.x = map_tmp,
              ~ boot.ci(.x,
                        type = 'perc'))

## Tabulate
tibble_tmp <- tibble(site = unique(data$site_of_worst_pain_1[!is.na(data$site_of_worst_pain_1)]),
                    proportion = c(round(map_tmp[[1]]$t0, 2),
                                   round(map_tmp[[2]]$t0, 2),
                                   round(map_tmp[[3]]$t0, 2),
                                   round(map_tmp[[4]]$t0, 2),
                                   round(map_tmp[[5]]$t0, 2),
                                   round(map_tmp[[6]]$t0, 2),
                                   round(map_tmp[[7]]$t0, 2),
                                   round(map_tmp[[8]]$t0, 2),
                                   round(map_tmp[[9]]$t0, 2),
                                   round(map_tmp[[10]]$t0, 2),
                                   round(map_tmp[[11]]$t0, 2),
                                   round(map_tmp[[12]]$t0, 2),
                                   round(map_tmp[[13]]$t0, 2),
                                   round(map_tmp[[14]]$t0, 2)),
                    `lower 95% CI` = c(round(map_tmp[[1]]$percent[4], 2),
                                       round(map_tmp[[2]]$percent[4], 2),
                                       round(map_tmp[[3]]$percent[4], 2),
                                       round(map_tmp[[4]]$percent[4], 2),
                                       round(map_tmp[[5]]$percent[4], 2),
                                       round(map_tmp[[6]]$percent[4], 2),
                                       round(map_tmp[[7]]$percent[4], 2),
                                       round(map_tmp[[8]]$percent[4], 2),
                                       round(map_tmp[[9]]$percent[4], 2),
                                       round(map_tmp[[10]]$percent[4], 2),
                                       round(map_tmp[[11]]$percent[4], 2),
                                       round(map_tmp[[12]]$percent[4], 2),
                                       round(map_tmp[[13]]$percent[4], 2),
                                       round(map_tmp[[14]]$percent[4], 2)),
                    `upper 95% CI` = c(round(map_tmp[[1]]$percent[5], 2),
                                       round(map_tmp[[2]]$percent[5], 2),
                                       round(map_tmp[[3]]$percent[5], 2),
                                       round(map_tmp[[4]]$percent[5], 2),
                                       round(map_tmp[[5]]$percent[5], 2),
                                       round(map_tmp[[6]]$percent[5], 2),
                                       round(map_tmp[[7]]$percent[5], 2),
                                       round(map_tmp[[8]]$percent[5], 2),
                                       round(map_tmp[[9]]$percent[5], 2),
                                       round(map_tmp[[10]]$percent[5], 2),
                                       round(map_tmp[[11]]$percent[5], 2),
                                       round(map_tmp[[12]]$percent[5], 2),
                                       round(map_tmp[[13]]$percent[5], 2),
                                       round(map_tmp[[14]]$percent[5], 2))) %>%

  arrange(desc(proportion))

tibble_tmp %>%
  kable(caption = 'Site of the worst pain - whole cohort (95% CI)')

```


site	proportion	lower 95% CI	upper 95% CI
head	0.25	0.21	0.30
low back	0.17	0.13	0.21
abdomen	0.14	0.11	0.18
chest	0.11	0.07	0.15
feet	0.07	0.05	0.10
shoulders	0.06	0.04	0.09
legs	0.06	0.04	0.09
joints	0.05	0.03	0.07
genitals	0.03	0.01	0.05
arms	0.03	0.01	0.05
hands	0.01	0.00	0.02
muscles	0.01	0.00	0.03
head, chest	0.00	0.00	0.01
arms, hands	0.00	0.00	0.01

```

# By HIV status (>= 5%)
sites <- c('head', 'low back', 'abdomen', 'chest', 'feet', 'legs', 'shoulders')

## Extract HIV+ data and bootstrap CI
hiv_p <- data %>%
  filter(HIV_positive == 'yes')

set.seed(2019)
map_hiv_p <- map(.x = sites,
  ~ boot(data = hiv_p,
    site = .x,
    R = 999,
    statistic = func_tmp,
    stype = 'i'))

map_hiv_p <- map(.x = map_hiv_p,
  ~ boot.ci(.x,
    type = 'perc'))

## Extract HIV- data and bootstrap CI
hiv_n <- data %>%
  filter(HIV_positive == 'no')

set.seed(2019)
map_hiv_n <- map(.x = sites,
  ~ boot(data = hiv_n,
    site = .x,
    R = 999,
    statistic = func_tmp,
    stype = 'i'))

map_hiv_n <- map(.x = map_hiv_n,
  ~ boot.ci(.x,
    type = 'perc'))

## Tabulate
tibble_tmp <- tibble(`HIV positive` = rep(c('no', 'yes'),

```

```

      times = 7),
site = rep(sites, each = 2),
proportion = c(round(map_hiv_n[[1]]$t0, 2),
               round(map_hiv_p[[1]]$t0, 2),
               round(map_hiv_n[[2]]$t0, 2),
               round(map_hiv_p[[2]]$t0, 2),
               round(map_hiv_n[[3]]$t0, 2),
               round(map_hiv_p[[3]]$t0, 2),
               round(map_hiv_n[[4]]$t0, 2),
               round(map_hiv_p[[4]]$t0, 2),
               round(map_hiv_n[[5]]$t0, 2),
               round(map_hiv_p[[5]]$t0, 2),
               round(map_hiv_n[[6]]$t0, 2),
               round(map_hiv_p[[6]]$t0, 2),
               round(map_hiv_n[[7]]$t0, 2),
               round(map_hiv_p[[7]]$t0, 2)),
`lower 95% CI` = c(round(map_hiv_n[[1]]$percent[4], 2),
                  round(map_hiv_p[[1]]$percent[4], 2),
                  round(map_hiv_n[[2]]$percent[4], 2),
                  round(map_hiv_p[[2]]$percent[4], 2),
                  round(map_hiv_n[[3]]$percent[4], 2),
                  round(map_hiv_p[[3]]$percent[4], 2),
                  round(map_hiv_n[[4]]$percent[4], 2),
                  round(map_hiv_p[[4]]$percent[4], 2),
                  round(map_hiv_n[[5]]$percent[4], 2),
                  round(map_hiv_p[[5]]$percent[4], 2),
                  round(map_hiv_n[[6]]$percent[4], 2),
                  round(map_hiv_p[[6]]$percent[4], 2),
                  round(map_hiv_n[[7]]$percent[4], 2),
                  round(map_hiv_p[[7]]$percent[4], 2)),
`upper 95% CI` = c(round(map_hiv_n[[1]]$percent[5], 2),
                  round(map_hiv_p[[1]]$percent[5], 2),
                  round(map_hiv_n[[2]]$percent[5], 2),
                  round(map_hiv_p[[2]]$percent[5], 2),
                  round(map_hiv_n[[3]]$percent[5], 2),
                  round(map_hiv_p[[3]]$percent[5], 2),
                  round(map_hiv_n[[4]]$percent[5], 2),
                  round(map_hiv_p[[4]]$percent[5], 2),
                  round(map_hiv_n[[5]]$percent[5], 2),
                  round(map_hiv_p[[5]]$percent[5], 2),
                  round(map_hiv_n[[6]]$percent[5], 2),
                  round(map_hiv_p[[6]]$percent[5], 2),
                  round(map_hiv_n[[7]]$percent[5], 2),
                  round(map_hiv_p[[7]]$percent[5], 2)))

tibble_tmp %>%
  kable(caption = 'Site of the worst pain - by HIV status (95% CI)')

```

HIV positive	site	proportion	lower 95% CI	upper 95% CI
no	head	0.24	0.19	0.29
yes	head	0.38	0.23	0.56
no	low back	0.18	0.13	0.22
yes	low back	0.15	0.03	0.29
no	abdomen	0.15	0.11	0.20

HIV positive	site	proportion	lower 95% CI	upper 95% CI
yes	abdomen	0.06	0.00	0.15
no	chest	0.11	0.08	0.15
yes	chest	0.09	0.00	0.20
no	feet	0.07	0.04	0.10
yes	feet	0.09	0.00	0.20
no	legs	0.07	0.04	0.10
yes	legs	0.03	0.00	0.09
no	shoulders	0.06	0.03	0.09
yes	shoulders	0.09	0.00	0.19

95% confidence interval of the difference in proportions

```
# Boot function
func_tmp <- function(d, i, site){
  data <- d[i, ]
  hiv_p <- filter(data, HIV_positive == 'yes')
  hiv_p <- mean(hiv_p$site_of_worst_pain_1 == site,
               na.rm = TRUE)
  hiv_n <- filter(data, HIV_positive == 'no')
  hiv_n <- mean(hiv_n$site_of_worst_pain_1 == site,
               na.rm = TRUE)
  hiv_p - hiv_n
}

# Bootstrap CI of the difference in proportions for all sites with estimate >= 5%
set.seed(2019)
map_tmp <- map(.x = sites,
               ~ boot(data = data,
                      site = .x,
                      R = 999,
                      statistic = func_tmp,
                      stype = 'i'))

map_tmp <- map(.x = map_tmp,
               ~ boot.ci(.x,
                        type = 'perc'))

# Tabulate
tibble_tmp <- tibble(site = sites,
                    `difference in proportion` = c(round(map_tmp[[1]]$t0, 2),
                                                    round(map_tmp[[2]]$t0, 2),
                                                    round(map_tmp[[3]]$t0, 2),
                                                    round(map_tmp[[4]]$t0, 2),
                                                    round(map_tmp[[5]]$t0, 2),
                                                    round(map_tmp[[6]]$t0, 2),
                                                    round(map_tmp[[7]]$t0, 2)),
                    `lower 95% CI` = c(round(map_tmp[[1]]$percent[4], 2),
                                         round(map_tmp[[2]]$percent[4], 2),
                                         round(map_tmp[[3]]$percent[4], 2),
                                         round(map_tmp[[4]]$percent[4], 2),
                                         round(map_tmp[[5]]$percent[4], 2),
                                         round(map_tmp[[6]]$percent[4], 2),
                                         round(map_tmp[[7]]$percent[4], 2))
```

```

        round(map_tmp[[6]]$percent[4], 2),
        round(map_tmp[[7]]$percent[4], 2)),
`upper 95% CI` = c(round(map_tmp[[1]]$percent[5], 2),
        round(map_tmp[[2]]$percent[5], 2),
        round(map_tmp[[3]]$percent[5], 2),
        round(map_tmp[[4]]$percent[5], 2),
        round(map_tmp[[5]]$percent[5], 2),
        round(map_tmp[[6]]$percent[5], 2),
        round(map_tmp[[7]]$percent[5], 2)))

tibble_tmp %>%
  kable(caption = 'Site of the worst pain - 95% CI of the difference (HIV+ minus HIV-)')

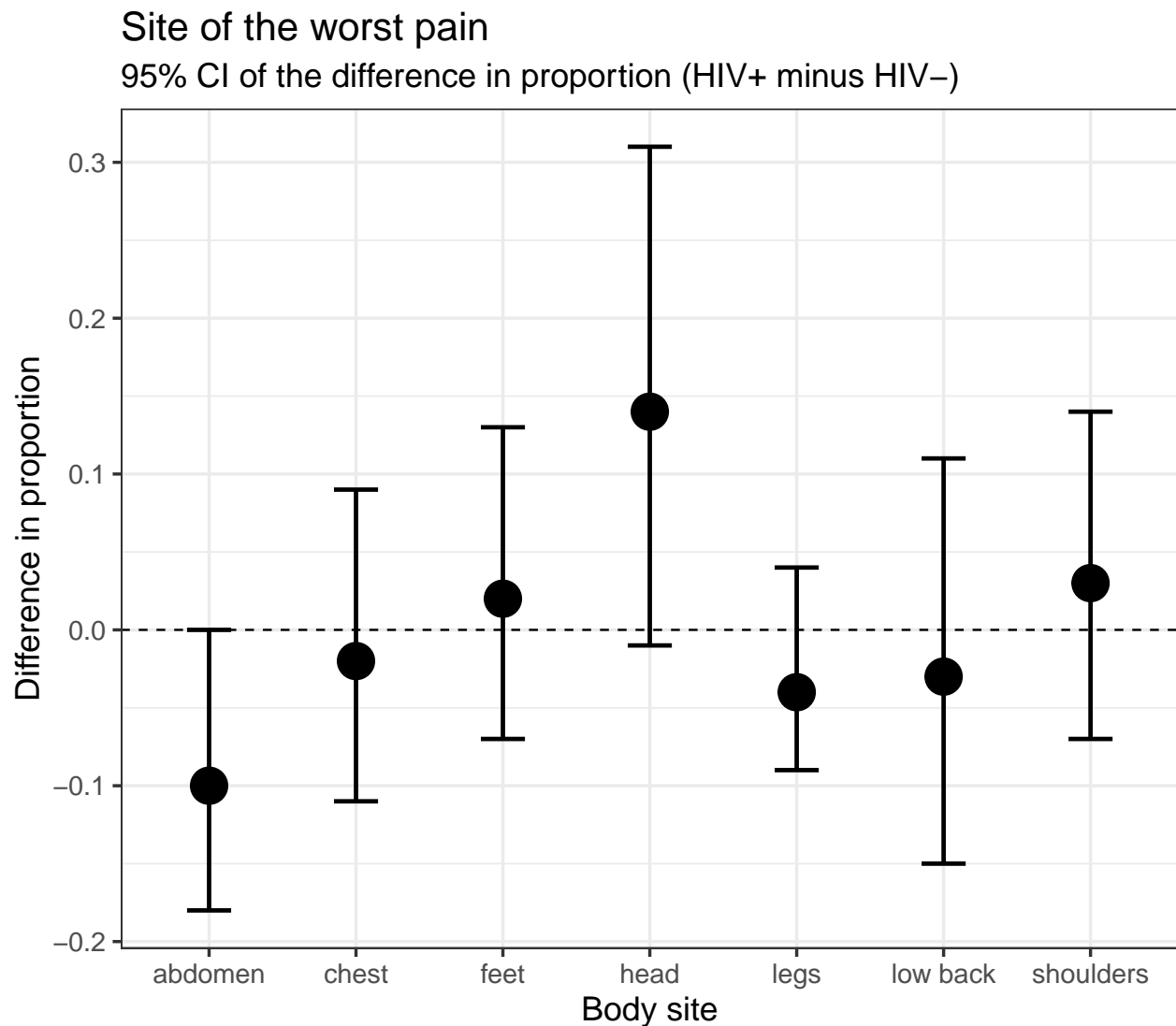
```

site	difference in proportion	lower 95% CI	upper 95% CI
head	0.14	-0.01	0.31
low back	-0.03	-0.15	0.11
abdomen	-0.10	-0.18	0.00
chest	-0.02	-0.11	0.09
feet	0.02	-0.07	0.13
legs	-0.04	-0.09	0.04
shoulders	0.03	-0.07	0.14

```

# Plot
ggplot(data = tibble_tmp) +
  aes(x = site,
      y = `difference in proportion`,
      ymin = `lower 95% CI`,
      ymax = `upper 95% CI`) +
  geom_point(size = 8) +
  geom_errorbar(size = 1,
               width = 0.3) +
  geom_hline(yintercept = 0,
             linetype = 2) +
  labs(title = 'Site of the worst pain',
       subtitle = '95% CI of the difference in proportion (HIV+ minus HIV-)',
       y = 'Difference in proportion',
       x = 'Body site')

```



Pain intensity

Worst pain in the last week.

Point estimates

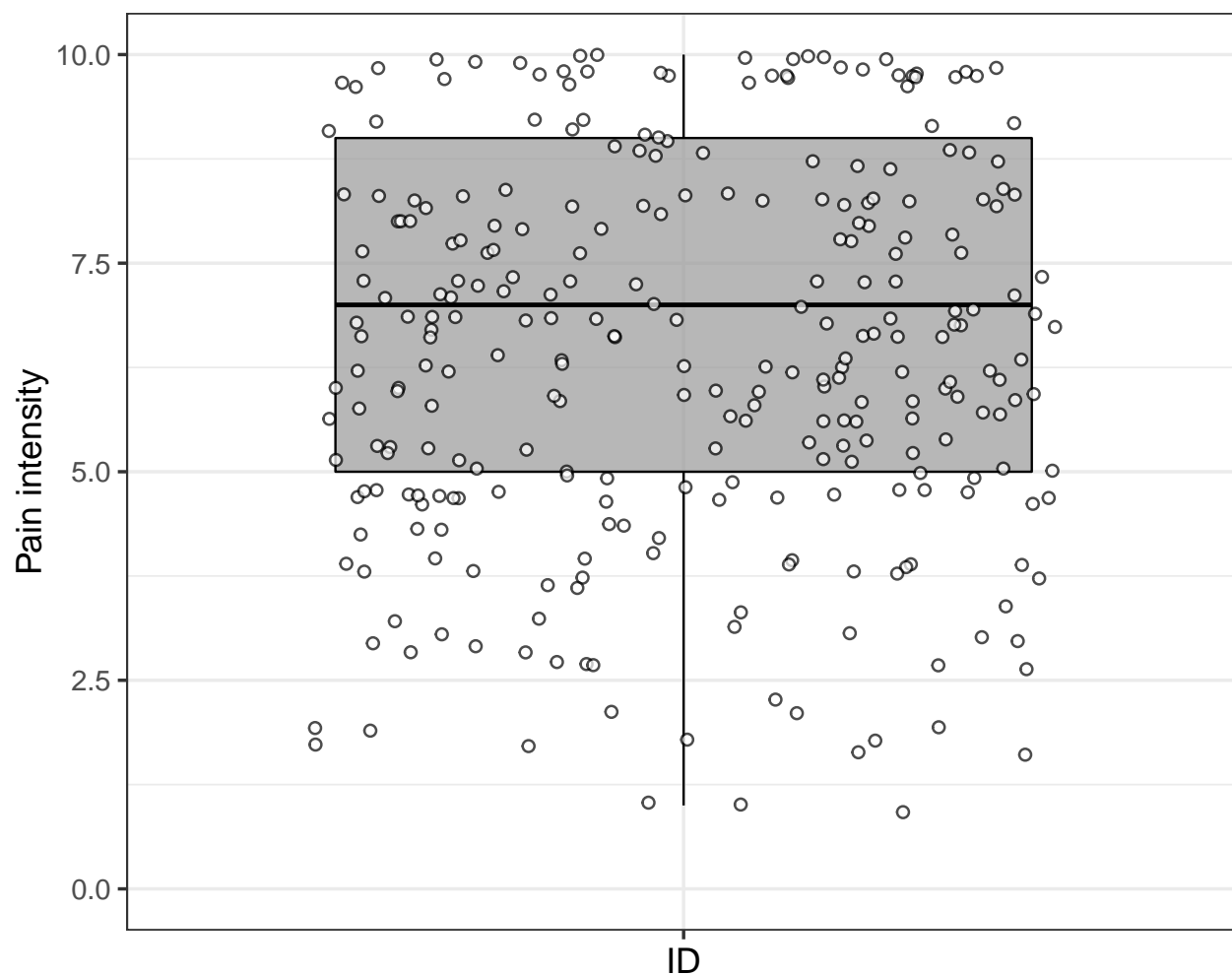
```
# Fix
# Total cohort
data %>%
  select(pain_worst) %>%
  summarise(mean = round(mean(pain_worst), 2),
            SD = round(sd(pain_worst), 2),
            min = min(pain_worst),
            max = max(pain_worst)) %>%
  kable(caption = 'Pain intensity - whole cohort')
```

mean	SD	min	max
6.78	2.41	1	10

```
## Plot
data %>%
  select(pain_worst) %>%
  ggplot(data = .) +
  aes(x = 'ID',
      y = pain_worst) +
  geom_boxplot(alpha = 0.7,
               colour = '#000000',
               fill = '#999999') +
  geom_point(position = position_jitter(),
             fill = '#FFFFFF',
             alpha = 0.7,
             stroke = 0.8,
             size = 2,
             shape = 21) +
  scale_y_continuous(limits = c(0, 10)) +
  labs(title = 'Worst pain in the past week',
       subtitle = 'Full cohort',
       y = 'Pain intensity') +
  theme(axis.text.x = element_blank())
```

Worst pain in the past week

Full cohort



```
# By HIV status
data %>%
  select(HIV_positive, pain_worst) %>%
  group_by(HIV_positive) %>%
  summarise(mean = round(mean(pain_worst), 2),
            SD = round(sd(pain_worst), 2),
            min = min(pain_worst),
            max = max(pain_worst)) %>%
  kable(caption = 'Pain intensity - by HIV status')
```

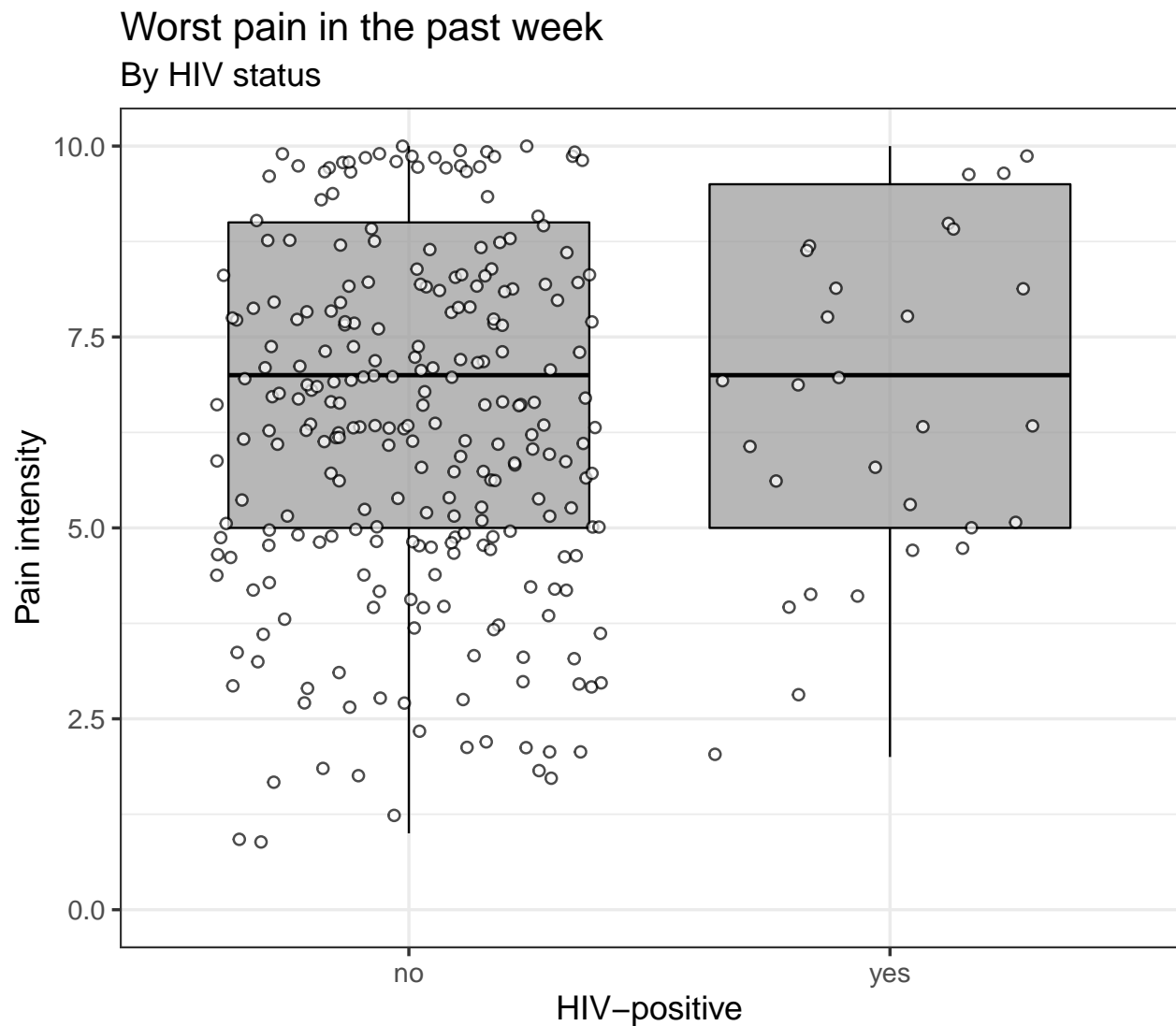
HIV_positive	mean	SD	min	max
no	6.73	2.42	1	10
yes	7.17	2.38	2	10

```
## Plot
data %>%
  select(HIV_positive, pain_worst) %>%
  ggplot(data = .) +
```

```

aes(x = HIV_positive,
    y = pain_worst) +
geom_boxplot(alpha = 0.7,
             colour = '#000000',
             fill = '#999999') +
geom_point(position = position_jitter(),
           fill = '#FFFFFF',
           alpha = 0.7,
           stroke = 0.8,
           size = 2,
           shape = 21) +
scale_y_continuous(limits = c(0, 10)) +
labs(title = 'Worst pain in the past week',
     subtitle = 'By HIV status',
     y = 'Pain intensity',
     x = 'HIV-positive')

```



95% confidence intervals for the point estimates

```
# Use the Rcompanion groupwiseMean for numeric data
```

```
# Data
```

```
tmp <- data %>%
```

```
  select(HIV_positive, pain_worst)
```

```
# Whole cohort
```

```
set.seed(2019)
```

```
groupwiseMean(pain_worst ~ 1,
```

```
  data = tmp,
```

```
  bca = TRUE,
```

```
  R = 999) %>%
```

```
  select(Mean, Bca.lower, Bca.upper) %>%
```

```
  rename(mean = Mean,
```

```
    `lower 95% CI` = Bca.lower,
```

```
    `upper 95% CI` = Bca.upper) %>%
```

```
  kable(caption = 'Pain at its worst in the past week - total cohort (95% CI)')
```

mean	lower 95% CI	upper 95% CI
6.78	6.49	7.05

```
# By HIV status (HIV- reported first)
```

```
set.seed(2019)
```

```
groupwiseMean(pain_worst ~ HIV_positive,
```

```
  data = tmp,
```

```
  bca = TRUE,
```

```
  R = 999) %>%
```

```
  select(HIV_positive, Mean, Bca.lower, Bca.upper) %>%
```

```
  rename(mean = Mean,
```

```
    `lower 95% CI` = Bca.lower,
```

```
    `upper 95% CI` = Bca.upper) %>%
```

```
  kable(caption = 'Pain at its worst in the past week - by HIV status (95% CI)')
```

HIV_positive	mean	lower 95% CI	upper 95% CI
no	6.73	6.46	7.03
yes	7.17	6.29	7.89

95% confidence interval of the difference in means

The difference in the mean intensity of the worst pain in the past week of HIV-positive participants vs participants who were HIV-negative (HIV+ minus HIV-).

```
# Boot function
```

```
func_tmp <- function(d, i){
```

```
  data <- d[i, ]
```

```
  data <- data %>%
```

```
    filter(!is.na(HIV_positive))
```

```
  data_hiv <- filter(data, HIV_positive == 'yes')
```

```
  data_nohiv <- filter(data, HIV_positive == 'no')
```

```

    mean_yes <- mean(data_hiv$pain_worst)
    mean_no <- mean(data_nohiv$pain_worst)
    mean_yes - mean_no
  }

# Confidence interval of the difference in proportions (HIV+ minus HIV-)
set.seed(2019)
boot_tmp <- boot(data = data,
                 statistic = func_tmp,
                 R = 999,
                 stype = 'i')

bootci_tmp <- boot.ci(boot_tmp,
                     type = 'perc') # BCa gave extreme order statistics

tibble_tmp <- tibble(`difference in mean` = round(boot_tmp$t0, 2),
                    `lower 95% CI` = round(bootci_tmp$percent[[4]], 2),
                    `upper 95% CI` = round(bootci_tmp$percent[[5]], 2))

tibble_tmp %>%
  kable(caption = 'Pain at its worst in the past week - 95% CI of the difference (HIV+ minus HIV-)')

```

difference in mean	lower 95% CI	upper 95% CI
0.44	-0.45	1.27

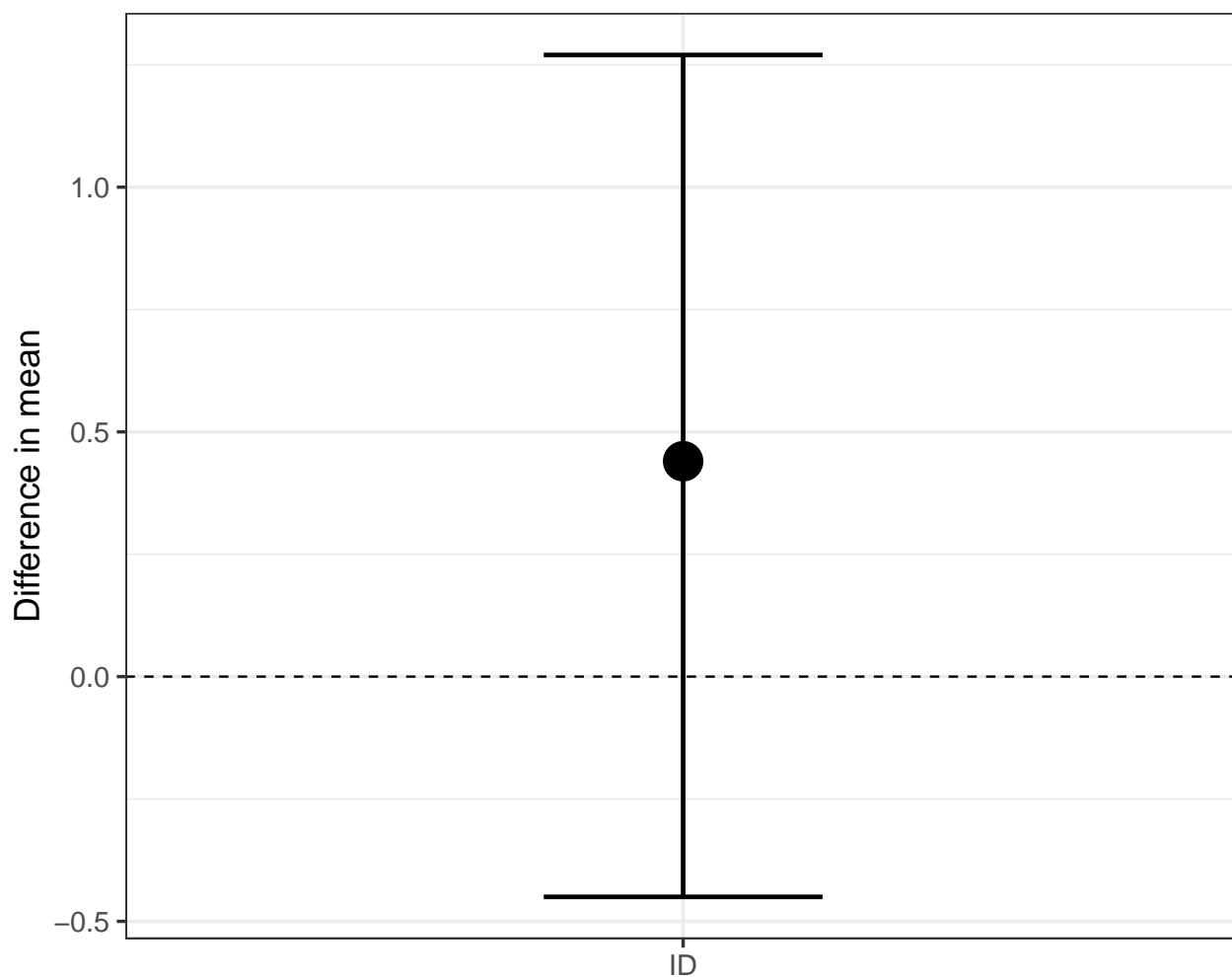
```

# Plot
ggplot(data = tibble_tmp) +
  aes(x = 'ID',
      y = `difference in mean`,
      ymin = `lower 95% CI`,
      ymax = `upper 95% CI`) +
  geom_point(size = 8) +
  geom_errorbar(size = 1,
               width = 0.3) +
  geom_hline(yintercept = 0,
             linetype = 2) +
  labs(title = 'Pain at its worst in the past week',
       subtitle = '95% CI of the difference in proportion (HIV+ minus HIV-)',
       y = 'Difference in mean') +
  theme(axis.title.x = element_blank())

```

Pain at its worst in the past week

95% CI of the difference in proportion (HIV+ minus HIV-)



Did you receive treatment for your pain?

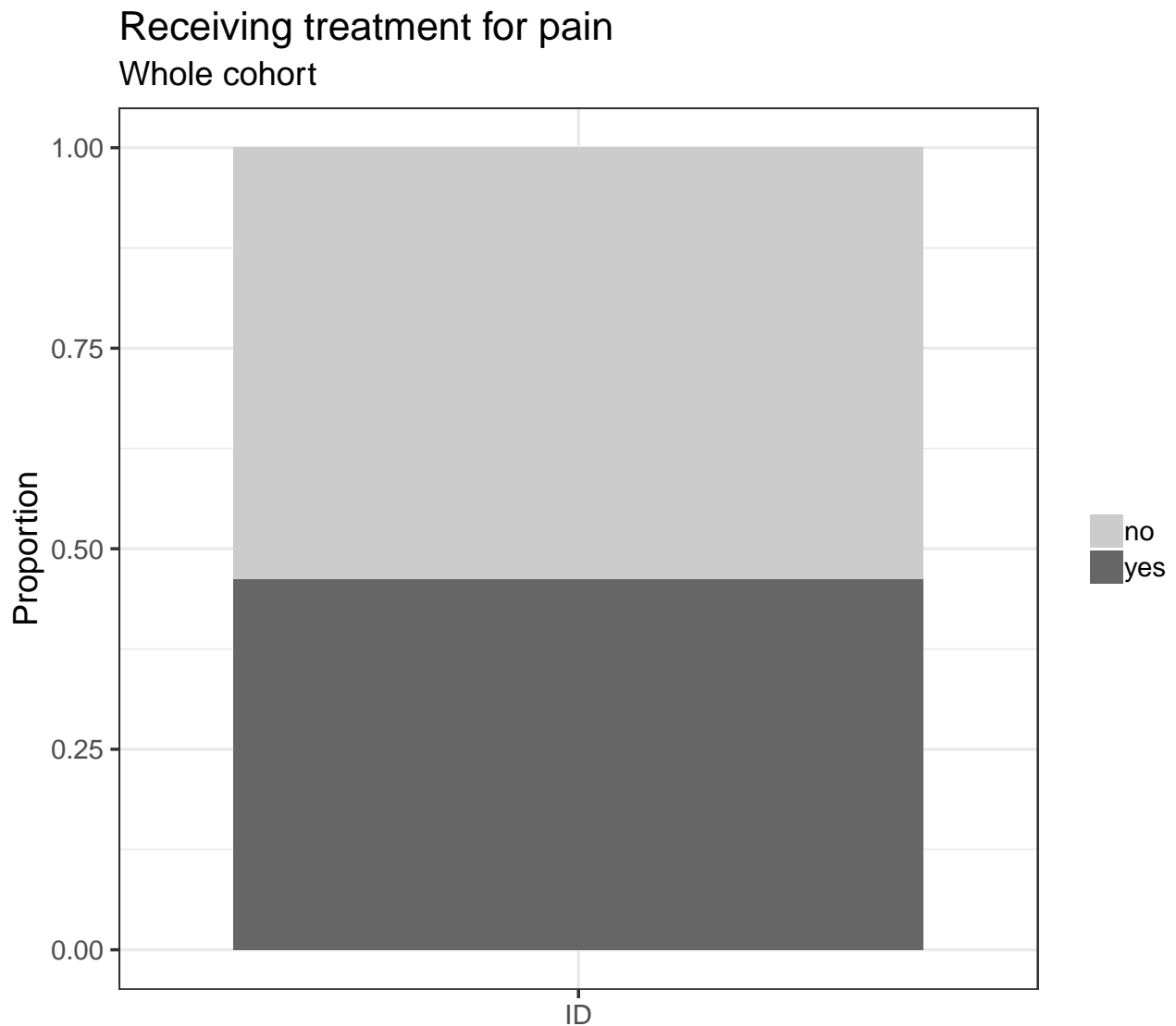
Receiving treatment includes self-medication.

Point estimates

```
# Whole cohort
data %>%
  select(pain_treatment) %>%
  group_by(pain_treatment) %>%
  summarise(count = n()) %>%
  ungroup() %>%
  mutate(total = sum(count)) %>%
  mutate(proportion = round(count / total, 2)) %>%
  kable(caption = 'Receiving pain treatment - total cohort (point estimates)')
```

pain_treatment	count	total	proportion
no	170	316	0.54
yes	146	316	0.46

```
## Plot
ggplot(data = data) +
  aes(x = 'ID',
      fill = pain_treatment) +
  geom_bar(position = position_fill()) +
  scale_fill_manual(values = pal,
                   na.value = '#000000') +
  labs(title = 'Receiving treatment for pain',
       subtitle = 'Whole cohort',
       y = 'Proportion') +
  theme(legend.title = element_blank(),
       axis.title.x = element_blank())
```



Count and proportion by HIV status

```

data %>%
  filter(!is.na(pain_treatment)) %>%
  filter(!is.na(HIV_positive)) %>%
  group_by(HIV_positive, pain_treatment) %>%
  summarise(count = n()) %>%
  group_by(HIV_positive) %>%
  mutate(total = sum(count)) %>%
  mutate(proportion = round(count/total, 2)) %>%
  kable(caption = 'Receiving pain treatment - by HIV status (point estimates)')

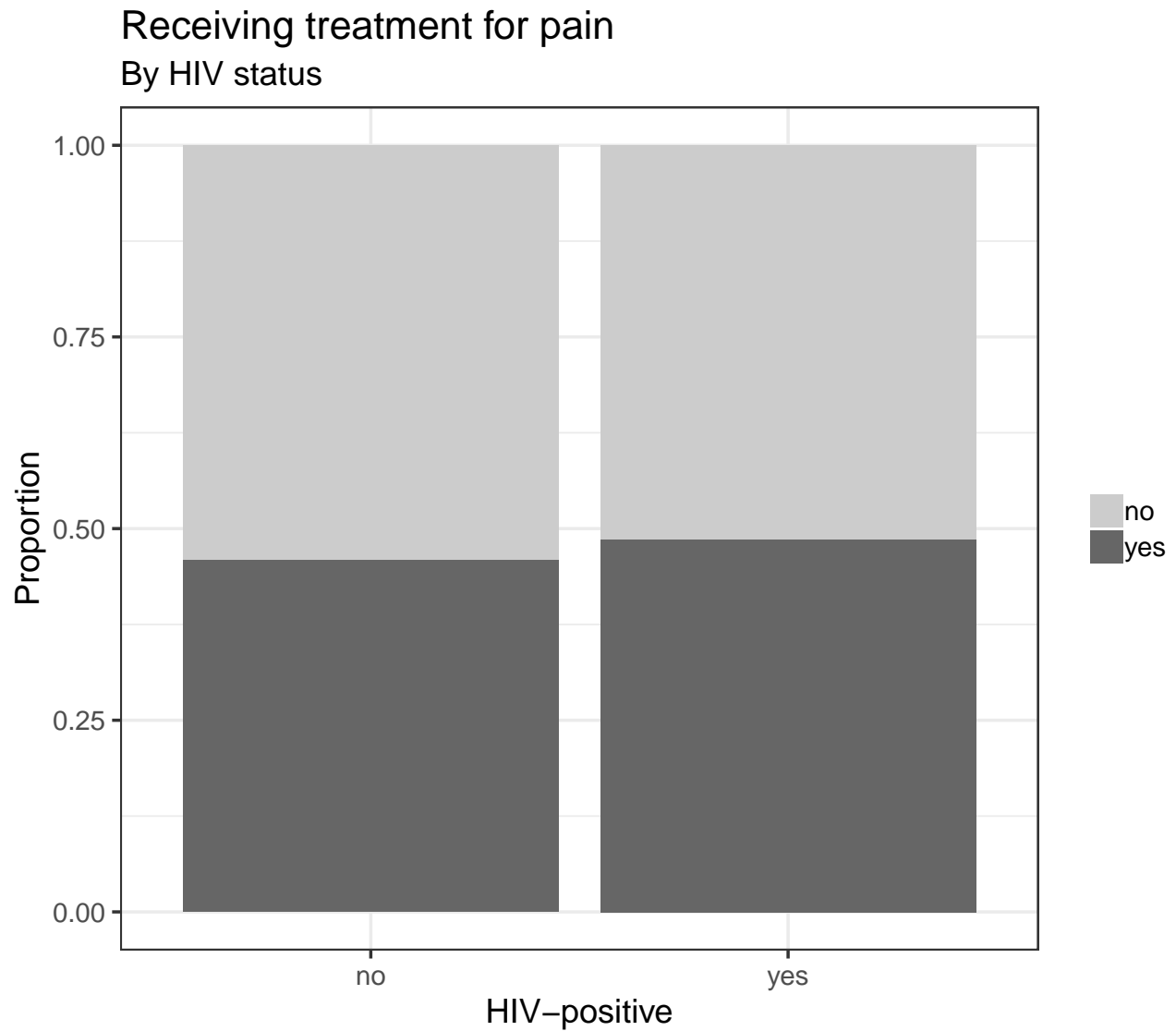
```

HIV_positive	pain_treatment	count	total	proportion
no	no	152	281	0.54
no	yes	129	281	0.46
yes	no	18	35	0.51
yes	yes	17	35	0.49

```

## Plot
ggplot(data = data) +
  aes(x = HIV_positive,
      fill = pain_treatment) +
  geom_bar(position = position_fill()) +
  scale_fill_manual(values = pal,
                    na.value = '#000000') +
  labs(title = 'Receiving treatment for pain',
       subtitle = 'By HIV status',
       y = 'Proportion',
       x = 'HIV-positive') +
  theme(legend.title = element_blank())

```



95% confidence intervals for the point estimates

```
# Boot functions
func_tmp <- function(d, i){
  data <- d[i, ]
  data <- data %>%
    filter(!is.na(pain_treatment))
  prop <- mean(data$pain_treatment == 'yes')
  prop
}

# Whole cohort
set.seed(2019)
boot_tmp <- boot(data = data,
  statistic = func_tmp,
  R = 999,
  stype = 'i')
```

```
bootci_tmp <- boot.ci(boot_tmp,
                      type = 'perc')

tibble(`Receiving treatment` = 'yes',
       proportion = round(boot_tmp$t0, 2),
       `lower 95% CI` = round(bootci_tmp$percent[[4]], 2),
       `upper 95% CI` = round(bootci_tmp$percent[[5]], 2)) %>%
  kable(caption = 'Receiving pain treatment - total cohort (95% CI)')
```

Receiving treatment	proportion	lower 95% CI	upper 95% CI
yes	0.46	0.41	0.52

```
# By HIV status (HIV- reported first)
set.seed(2019)
boot_tmp <- data %>%
  group_by(HIV_positive) %>%
  nest() %>%
  mutate(boot = map(.x = data,
                    ~ boot(data = .x,
                           statistic = func_tmp,
                           R = 999,
                           stype = 'i')))) %>%
  mutate(boot_ci = map(.x = boot,
                       ~ boot.ci(.x,
                                 type = 'perc'))))

tibble(`HIV positive` = c('no', 'yes'),
       `Receiving treatment` = c('yes', 'yes'),
       proportion = c(round(boot_tmp$boot[[1]]$t0, 2),
                       round(boot_tmp$boot[[2]]$t0, 2)),
       `lower 95% CI` = c(round(boot_tmp$boot_ci[[1]]$percent[[4]], 2),
                           round(boot_tmp$boot_ci[[2]]$percent[[4]], 2)),
       `upper 95% CI` = c(round(boot_tmp$boot_ci[[1]]$percent[[5]], 2),
                           round(boot_tmp$boot_ci[[2]]$percent[[5]], 2))) %>%
  kable(caption = 'Receiving pain treatment - by HIV status (95% CI)')
```

HIV positive	Receiving treatment	proportion	lower 95% CI	upper 95% CI
no	yes	0.46	0.40	0.52
yes	yes	0.49	0.31	0.66

95% confidence interval of the difference in proportions

The difference in the proportion of HIV-positive participants receiving treatment for their pain vs participants who were HIV-negative (HIV+ minus HIV-).

```
# Boot function
func_tmp <- function(d, i){
  data <- d[i, ]
  data <- data %>%
    filter(!is.na(pain_treatment)) %>%
```

```

    filter(!is.na(HIV_positive))
  data_hiv <- filter(data, HIV_positive == 'yes')
  data_nohiv <- filter(data, HIV_positive == 'no')
  prop_yes <- mean(data_hiv$pain_treatment == 'yes')
  prop_no <- mean(data_nohiv$pain_treatment == 'yes')
  prop_yes - prop_no
}

# Confidence interval of the difference in proportions (HIV+ minus HIV-)
set.seed(2019)
boot_tmp <- boot(data = data,
  statistic = func_tmp,
  R = 999,
  stype = 'i')

bootci_tmp <- boot.ci(boot_tmp,
  type = 'perc')

tibble_tmp <- tibble(`difference in proportion` = round(boot_tmp$t0, 2),
  `lower 95% CI` = round(bootci_tmp$percent[[4]], 2),
  `upper 95% CI` = round(bootci_tmp$percent[[5]], 2))

tibble_tmp %>%
  kable(caption = 'Receiving pain treatment - 95% CI of the difference (HIV+ minus HIV-)')

```

difference in proportion	lower 95% CI	upper 95% CI
0.03	-0.15	0.2

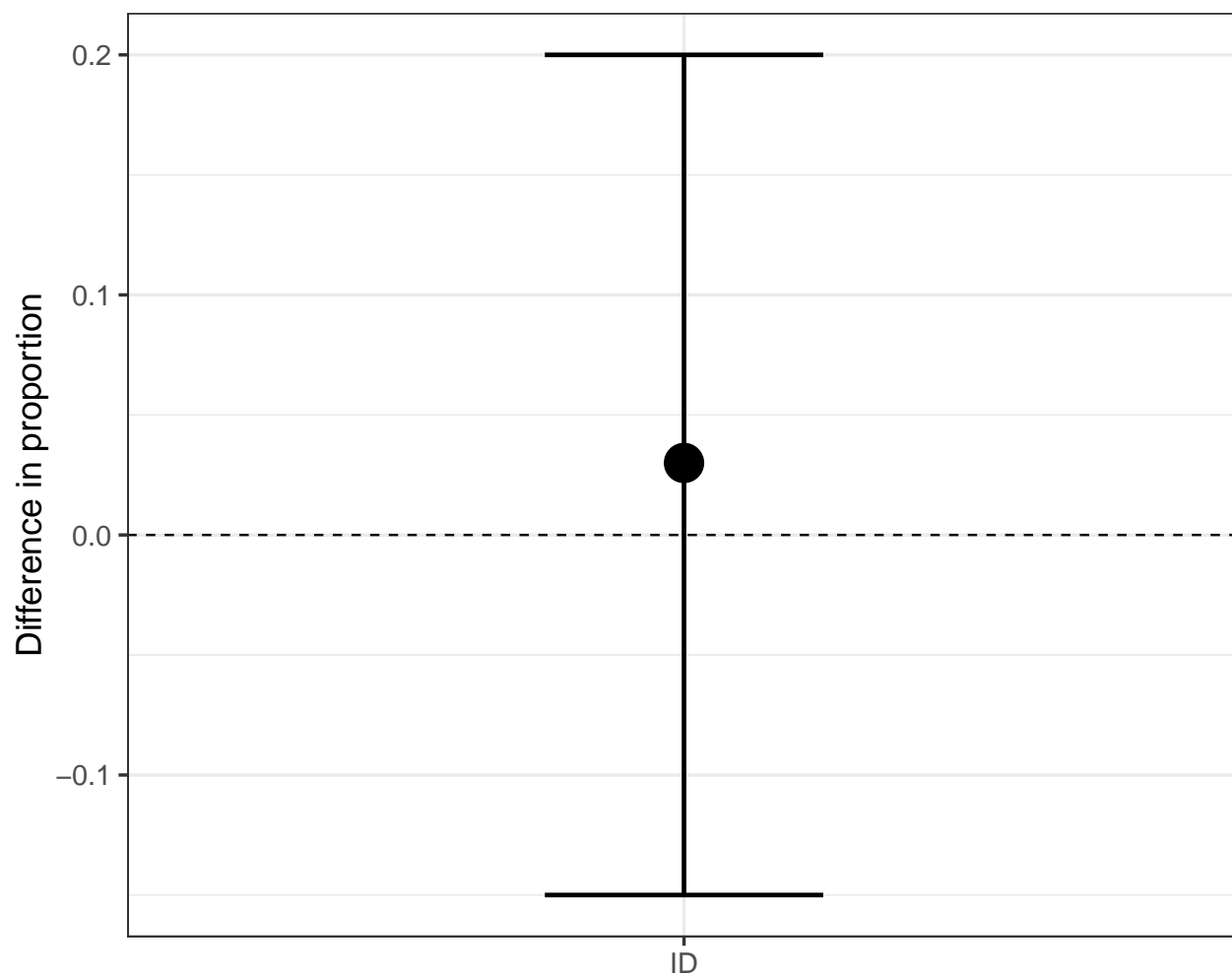
```

# Plot
ggplot(data = tibble_tmp) +
  aes(x = 'ID',
    y = `difference in proportion`,
    ymin = `lower 95% CI`,
    ymax = `upper 95% CI`) +
  geom_point(size = 8) +
  geom_errorbar(size = 1,
    width = 0.3) +
  geom_hline(yintercept = 0,
    linetype = 2) +
  labs(title = 'Receiving pain treatment',
    subtitle = '95% CI of the difference in proportion (HIV+ minus HIV-)',
    y = 'Difference in proportion') +
  theme(axis.title.x = element_blank())

```


Receiving pain treatment

95% CI of the difference in proportion (HIV+ minus HIV–)



Session information

```
sessionInfo()
```

```
## R version 3.6.0 (2019-04-26)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Mojave 10.14.5
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.6/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] DataExplorer_0.8.0 rcompanion_2.1.7 boot_1.3-22
## [4] skimr_1.0.5 magrittr_1.5 forcats_0.4.0
## [7] stringr_1.4.0 dplyr_0.8.0.1 purrr_0.3.2
## [10] readr_1.3.1 tidyr_0.8.3 tibble_2.1.1
## [13] ggplot2_3.1.1 tidyverse_1.2.1
##
## loaded via a namespace (and not attached):
## [1] httr_1.4.0 jsonlite_1.6 splines_3.6.0
## [4] modelr_0.1.4 assertthat_0.2.1 expm_0.999-4
## [7] highr_0.8 stats4_3.6.0 coin_1.3-0
## [10] cellranger_1.1.0 yaml_2.2.0 pillar_1.3.1
## [13] backports_1.1.4 lattice_0.20-38 glue_1.3.1
## [16] digest_0.6.18 rvest_0.3.3 colorspace_1.4-1
## [19] sandwich_2.5-1 htmltools_0.3.6 Matrix_1.2-17
## [22] plyr_1.8.4 pkgconfig_2.0.2 broom_0.5.2
## [25] haven_2.1.0 EMT_1.1 mvtnorm_1.0-10
## [28] scales_1.0.0 manipulate_1.0.1 generics_0.0.2
## [31] ellipsis_0.1.0 TH.data_1.0-10 withr_2.1.2.9000
## [34] lazyeval_0.2.2 cli_1.1.0 survival_2.44-1.1
## [37] crayon_1.3.4 readxl_1.3.1 evaluate_0.13
## [40] fansi_0.4.0 nlme_3.1-139 MASS_7.3-51.4
## [43] xml2_1.2.0 foreign_0.8-71 data.table_1.12.2
## [46] tools_3.6.0 hms_0.4.2 matrixStats_0.54.0
## [49] multcomp_1.4-10 munsell_0.5.0 networkD3_0.4
## [52] compiler_3.6.0 multcompView_0.1-7 rlang_0.3.4
## [55] grid_3.6.0 rstudioapi_0.10 htmlwidgets_1.3
## [58] igraph_1.2.4.1 labeling_0.3 rmarkdown_1.12
## [61] DescTools_0.99.28 gtable_0.3.0 codetools_0.2-16
## [64] R6_2.4.0 gridExtra_2.3 zoo_1.8-5
## [67] lubridate_1.7.4 knitr_1.22 utf8_1.1.4
## [70] nortest_1.0-4 libcoin_1.0-4 modeltools_0.2-22
## [73] stringi_1.4.3 parallel_3.6.0 Rcpp_1.0.1
## [76] tidyselect_0.2.5 xfun_0.6 lmtest_0.9-37

```