Script 1

Data missingness

Peter Kamerman

06 December 2019

Contents

Objective

To determine whether the degree of missingness or data inconsistencies were associated with any demographic variables (age, sex, ancestry, education, employment status), study variables (study site and group allocation), and clinical variables (active TB, possible psychiatric disorder, CD4, viral load).

Analysis notes

Definitions of missingness

Data were regarded as **missing** when *pain in the last week* data were not present for one or more of weeks 0, 12, 24, 36, 48. Data also were classified as **missing** when there were inconsistencies in the data across the variables collected within a week.

Definition of data inconsistencies

Pain was defined as *pain in the last week* being 'Yes', and *pain at its worst* being > 0. These two measurements were then the "gatekeeper" measurements, such that the two measurements both had to be positive ('Yes' and '> 0', respectively) in order for there to be any entries for *site of pain* and *site of worst pain*. Were the data were inconsistent (e.g., when there was no *pain in the last week* and *pain at its worst* $= 0$, but there were entries for *site of pain* and *site of worst pain*), then the *site of pain* and *site of worst pain* entries were marked as **inconsistent**.

Data also were considered **inconsistent** when *pain in the last week* = 'Yes', but *site of worst pain* = 'None'.

Lastly, data were considered **inconsistent** when *site of worst pain* was not listed as one of the pain locations for a given measurement week. For analysis purposes, missing data in the *site of pain* columns were changed to **'No'** (pain not present in the site). This approach was conservative, but we believed that the approach would have the least effect on the outcome, while still retaining as many participants as possible.

Import data

```
df <- read_rds('data-cleaned/data-ADVANCE.rds')
```

Quick look

```
head(df)
```

```
## # A tibble: 6 x 35
##   ranid interval_name site_name date_of_visit       pain_in_the_las~
##   <chr> <ord>         <chr>     <dttm>              <chr>
## 1 01-0~ 0 weeks       Wits RHI~ 2017-02-01 00:00:00 No
## 2 01-0~ 12 weeks      Wits RHI~ 2017-04-25 00:00:00 No
## 3 01-0~ 24 weeks      Wits RHI~ 2017-07-19 00:00:00 No
## 4 01-0~ 36 weeks      Wits RHI~ 2017-10-11 00:00:00 No
## 5 01-0~ 48 weeks      Wits RHI~ 2018-01-03 00:00:00 No
## 6 01-0~ 0 weeks       Wits RHI~ 2017-02-02 00:00:00 No
## # ... with 30 more variables: where_does_it_hurt_most <chr>,
## #   pain_worst <dbl>, pain_now <dbl>, head_pain <chr>,
## #   cervical_pain <chr>, shoulder_pain <chr>, arm_pain <chr>,
## #   hand_pain <chr>, chest_pain <chr>, abdominal_pain <chr>,
## #   low_back_pain <chr>, buttock_pain <chr>, hip_groin_pain <chr>,
## #   leg_pain <chr>, genital_pain <chr>, foot_pain <chr>, site_worst <chr>,
## #   age <dbl>, sex <chr>, ancestry <chr>, education <chr>,
## #   employment_status <chr>, cd4_cells.ul <dbl>, viral_load_cp.ml <dbl>,
## #   group <chr>, tb_screen <chr>, general_health <dbl>, mms_total <dbl>,
## #   interval_numeric <dbl>, any_missing <chr>
```

```
glimpse(df)
```

```
## Observations: 5,265
## Variables: 35
## $ ranid                   <chr> "01-0001", "01-0001", "01-0001", "01-0...
## $ interval_name           <ord> 0 weeks, 12 weeks, 24 weeks, 36 weeks,...
## $ site_name               <chr> "Wits RHI Yeoville Research Centre", "...
## $ date_of_visit           <dttm> 2017-02-01, 2017-04-25, 2017-07-19, 2...
## $ pain_in_the_last_week   <chr> "No", "No", "No", "No", "No", "No", "Y...
## $ where_does_it_hurt_most <chr> NA, NA, NA, NA, NA, NA, "Hip/groin lef...
## $ pain_worst              <dbl> 0, 0, 0, 0, 0, 0, 3, 3, 5, 0, 0, 0, 0,...
## $ pain_now                <dbl> NA, 0, NA, 0, NA, NA, 0, 2, 4, NA, NA,...
```

```
## $ head_pain           <chr> "No", "No", "No", "No", "No", "No", "N...
## $ cervical_pain       <chr> "No", "No", "No", "No", "No", "No", "N...
## $ shoulder_pain       <chr> "No", "No", "No", "No", "No", "No", "N...
## $ arm_pain            <chr> "No", "No", "No", "No", "No", "No", "N...
## $ hand_pain           <chr> "No", "No", "No", "No", "No", "No", "N...
## $ chest_pain          <chr> "No", "No", "No", "No", "No", "No", "N...
## $ abdominal_pain      <chr> "No", "No", "No", "No", "No", "No", "N...
## $ low_back_pain       <chr> "No", "No", "No", "No", "No", "No", "N...
## $ buttock_pain        <chr> "No", "No", "No", "No", "No", "No", "N...
## $ hip_groin_pain      <chr> "No", "No", "No", "No", "No", "No", "Y...
## $ leg_pain            <chr> "No", "No", "No", "No", "No", "No", "N...
## $ genital_pain        <chr> "No", "No", "No", "No", "No", "No", "N...
## $ foot_pain           <chr> "No", "No", "No", "No", "No", "No", "N...
## $ site_worst          <chr> "None", "None", "None", "None", "None"...
## $ age                 <dbl> 30, 30, 30, 30, 30, 34, 34, 34, 34, 34...
## $ sex                 <chr> "Male", "Male", "Male", "Male", "Male"...
## $ ancestry            <chr> "Black", "Black", "Black", "Black", "B...
## $ education           <chr> "Secondary", "Secondary", "Secondary",...
## $ employment_status   <chr> "Employed", "Employed", "Employed", "E...
## $ cd4_cells.ul        <dbl> 642, NA, 525, NA, 668, 241, NA, 364, N...
## $ viral_load_cp.ml    <dbl> 641, 50, 50, 50, 50, 3851, 50, 50, 50,...
## $ group               <chr> "GROUP 1 (DTG + TAF + FTC)", "GROUP 1 ...
## $ tb_screen           <chr> "Negative", "Negative", "Negative", "N...
## $ general_health      <dbl> 4, 4, 5, 5, 4, 3, 5, 3, 3, 3, 4, 5, 5,...
## $ mms_total           <dbl> 0, 0, 0, 0, 0, 0, 7, 0, 3, 1, 0, 0, 0,...
## $ interval_numeric    <dbl> 0, 12, 24, 36, 48, 0, 12, 24, 36, 48, ...
## $ any_missing         <chr> "No", "No", "No", "No", "No", "No", "N...
```

Number of participants with/without complete pain data

```r
df_pain <- df %>%
    select(ranid, any_missing) %>%
    distinct()

df_pain %>%
    group_by(any_missing) %>%
    summarise(count = n()) %>%
    mutate(n = sum(count),
           proportion = round(count / n, 3)) %>%
    knitr::kable(caption = 'Number of participants with/without complete pain data')
```

Table 1: Number of participants with/without complete pain data

any_missing	count	n	proportion
No	787	1053	0.747
Yes	266	1053	0.253

Demographic variables

Process data

```r
# Extract demographic data
df_demo <- df %>%
    select(ranid, any_missing, age, sex, ancestry,
           education, employment_status) %>%
    distinct()

# Join df_pain and df_demo
df_combined <- df_pain %>%
    left_join(df_demo)
```

```
## Joining, by = c("ranid", "any_missing")
```

Ancestry

```r
# Check counts
df_combined %>%
    group_by(ancestry) %>%
    summarise(count = n()) %>%
    knitr::kable(caption = 'Count within each category of self-identified ancestry')
```

Table 2: Count within each category of self-identified ancestry

ancestry	count
Black	1051
Coloured	2

Only 2 out of 1053 participants did not identify and Black African, and therefore no analysis done on ancestry.

Sex

```r
# Tabulate and print
df_combined %>%
    group_by(sex, any_missing) %>%
    summarise(count = n()) %>%
    group_by(sex) %>%
    mutate(total = sum(count),
           proportion = round(count / total,3)) %>%
    knitr::kable(caption = 'Missing pain data by sex')
```

Table 3: Missing pain data by sex

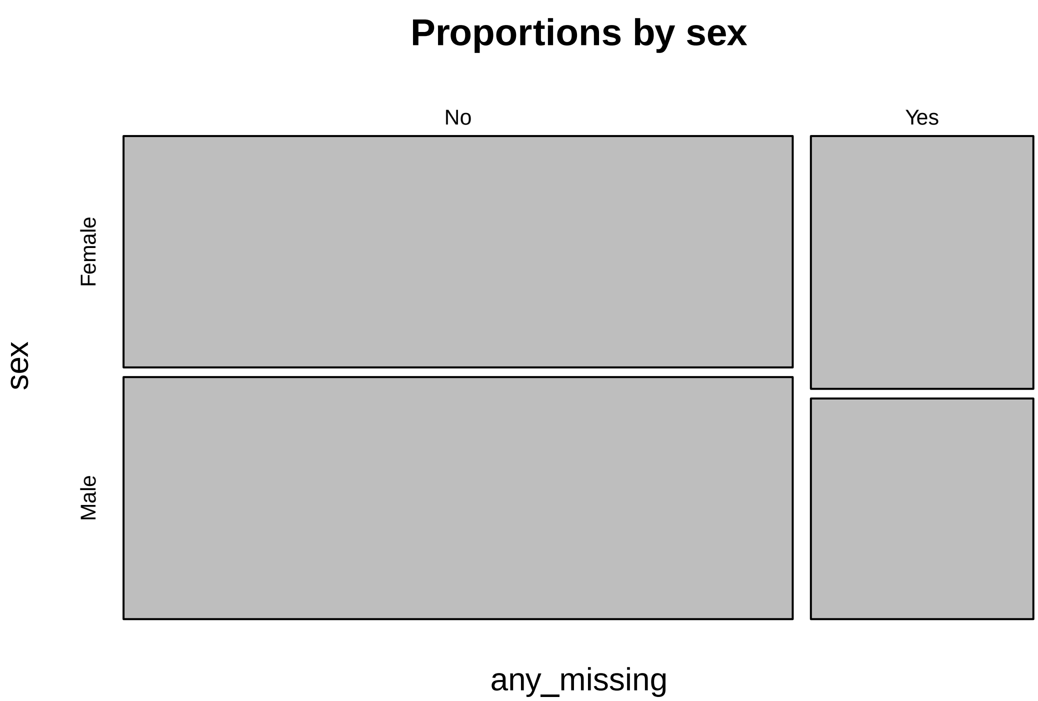
sex	any_missing	count	total	proportion
Female	No	457	623	0.734
Female	Yes	166	623	0.266
Male	No	330	430	0.767
Male	Yes	100	430	0.233

```r
# Tabulate, plot, and test
tab_sex <- xtabs(~any_missing + sex, data = df_combined)

mosaicplot(tab_sex, main = 'Counts by sex')
```


```r
prop_sex <- prop.table(tab_sex, 2)

mosaicplot(prop_sex, main = 'Proportions by sex')
```




```r
chisq.test(tab_sex)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_sex
```

```
## X-squared = 1.3737, df = 1, p-value = 0.2412
```

Education

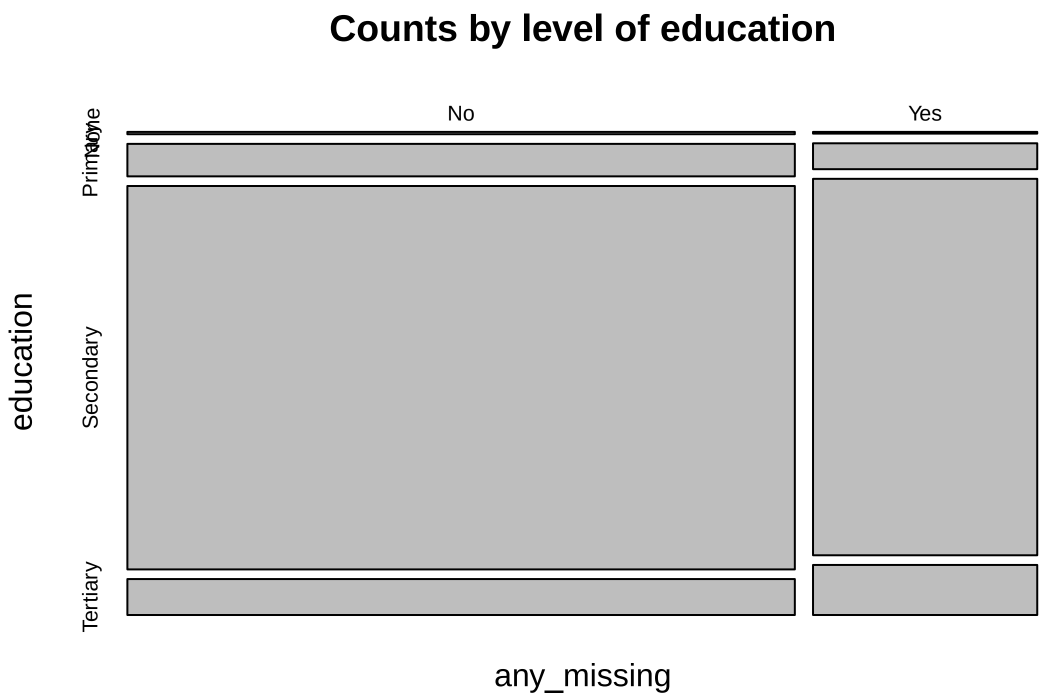
```r
# Tabulate and print
df_combined %>%
    mutate(education = fct_explicit_na(education)) %>%
    group_by(education, any_missing) %>%
    summarise(count = n()) %>%
    group_by(education) %>%
    mutate(total = sum(count),
           proportion = round(count / total,3)) %>%
    knitr::kable(caption = 'Missing pain data by level of education')
```

Table 4: Missing pain data by level of education

education	any_missing	count	total	proportion
No schooling	No	4	5	0.800
No schooling	Yes	1	5	0.200
Primary	No	56	71	0.789
Primary	Yes	15	71	0.211
Secondary	No	661	879	0.752
Secondary	Yes	218	879	0.248
Tertiary	No	62	91	0.681
Tertiary	Yes	29	91	0.319
(Missing)	No	4	7	0.571
(Missing)	Yes	3	7	0.429

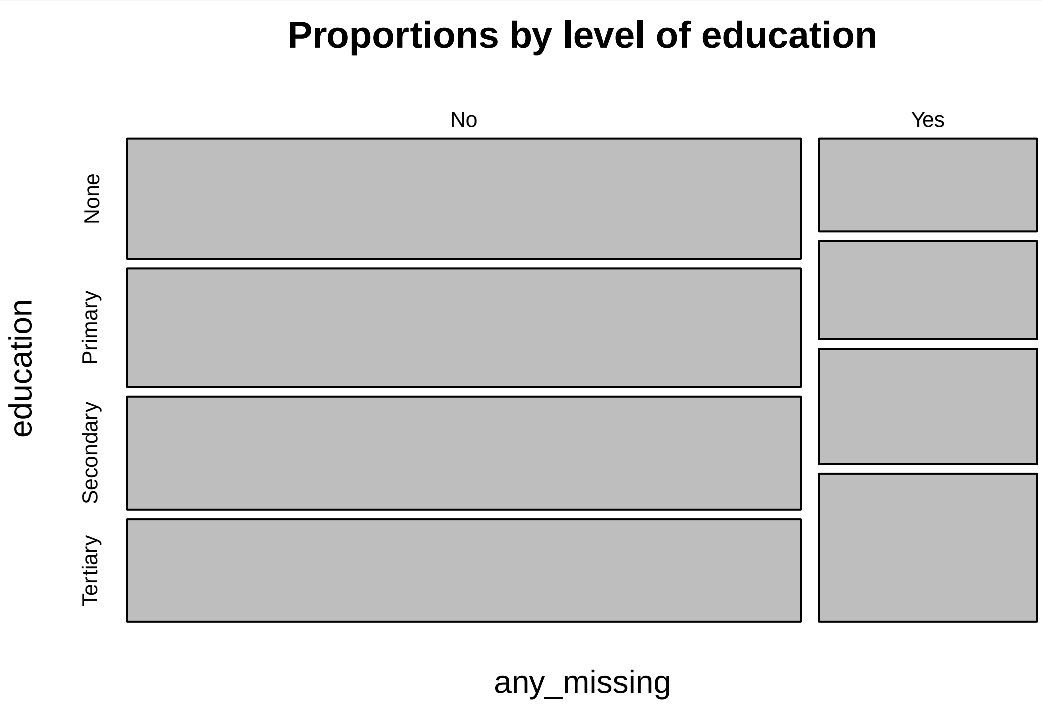
```r
# Tabulate, plot, and test
tab_edu <- df_combined %>%
    mutate(education = ifelse(education == 'No schooling',
                              yes = 'None',
                              no = education)) %>%
    xtabs(~any_missing + education, data = .)

mosaicplot(tab_edu, main = 'Counts by level of education')
```




```r
prop_edu <- prop.table(tab_edu, 2)

mosaicplot(prop_edu, main = 'Proportions by level of education')
```




```r
fisher.test(tab_edu)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  tab_edu
## p-value = 0.4006
## alternative hypothesis: two.sided
```

Employment status

```r
# Tabulate and print
df_combined %>%
    mutate(employment_status = fct_explicit_na(employment_status)) %>%
    group_by(employment_status, any_missing) %>%
    summarise(count = n()) %>%
    group_by(employment_status) %>%
    mutate(total = sum(count),
           proportion = round(count / total,3)) %>%
    knitr::kable(caption = 'Missing pain data by employment status')
```

Table 5: Missing pain data by employment status

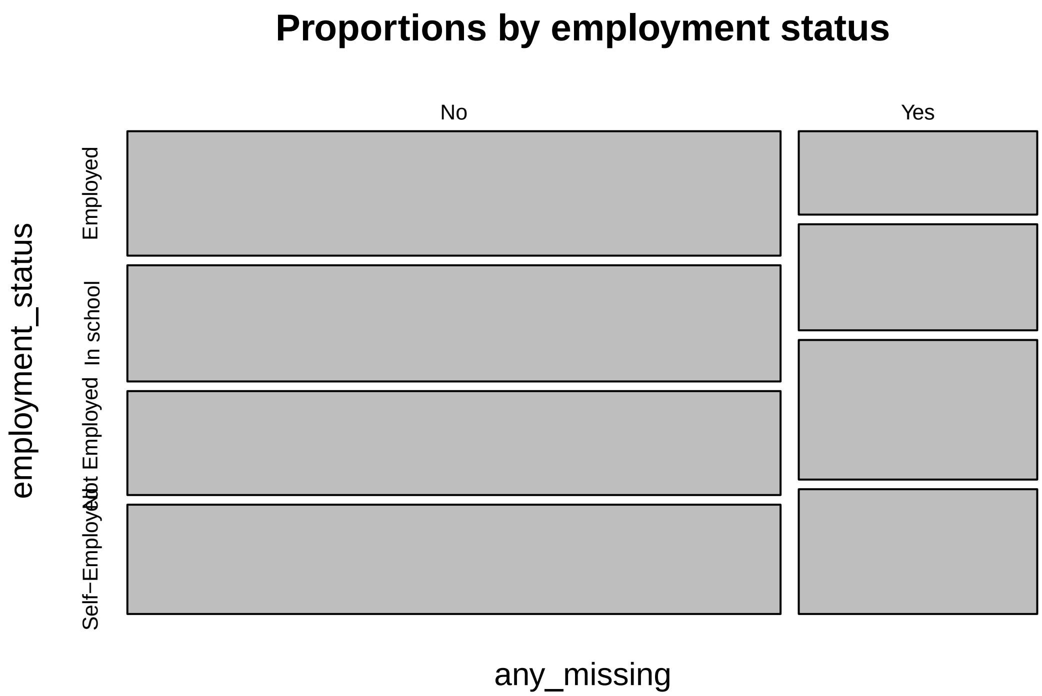
employment_status	any_missing	count	total	proportion
Employed	No	474	590	0.803
Employed	Yes	116	590	0.197
Not Employed	No	261	389	0.671
Not Employed	Yes	128	389	0.329
Schooling	No	6	8	0.750
Schooling	Yes	2	8	0.250
Self-Employed	No	36	51	0.706
Self-Employed	Yes	15	51	0.294
(Missing)	No	10	15	0.667
(Missing)	Yes	5	15	0.333

```r
# Tabulate, plot, and test
tab_employ <- df_combined %>%
    mutate(employment_status = ifelse(employment_status == 'Schooling',
                                      yes = 'In school',
                                      no = employment_status)) %>%
    xtabs(~any_missing + employment_status, data = .)

mosaicplot(tab_employ, main = 'Counts by employment status')
```

```r
prop_employ <- prop.table(tab_employ, 2)

mosaicplot(prop_employ, main = 'Proportions by employment status')
```




```r
fisher.test(tab_employ)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  tab_employ
## p-value = 3.88e-05
## alternative hypothesis: two.sided
```

Those who were unemployed had the greatest proportion of missing values.

Clinical variables

TB

`ever_TB` defined as having developed active TB at any stage over the first 48 weeks of the study.

```r
# Process the TB data and join with missingness data
df_TB <- df %>%
    select(ranid, interval_name, tb_screen) %>%
    # Determine whether active TB developed at any stage of the 48 weeks
    pivot_wider(names_from = interval_name,
                values_from = tb_screen) %>%
    mutate_at(2:6, ~ifelse(. == 'Negative',
                           yes = 0,
                           no = 1)) %>%
    mutate(ever_TB = rowSums(.[2:6], na.rm = TRUE)) %>%
    mutate(ever_TB = ifelse(ever_TB == 0,
                            yes = 'No',
                            no = 'Yes')) %>%
```

```
    select(ranid, ever_TB)

df_TB <- df_combined %>%
    select(ranid, any_missing) %>%
    left_join(df_TB)
```

```
## Joining, by = "ranid"
```

```
# Tabulate and print
df_TB %>%
    mutate(ever_TB = fct_explicit_na(ever_TB)) %>%
    group_by(ever_TB, any_missing) %>%
    summarise(count = n()) %>%
    group_by(ever_TB) %>%
    mutate(total = sum(count),
           proportion = round(count / total,3)) %>%
    knitr::kable(caption = 'Missing pain data by whether there was ever active TB')
```

Table 6: Missing pain data by whether there was ever active TB

ever_TB	any_missing	count	total	proportion
No	No	769	1032	0.745
No	Yes	263	1032	0.255
Yes	No	18	21	0.857
Yes	Yes	3	21	0.143

```
# Tabulate, plot, and test
tab_tb <- xtabs(~any_missing + ever_TB, data = df_TB)

mosaicplot(tab_tb, main = 'Counts by whether there was ever active TB')
```


```
prop_tb <- prop.table(tab_tb, 2)

mosaicplot(prop_tb, main = 'Proportion by whether there was ever active TB')
```


```
chisq.test(tab_tb)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_tb
## X-squared = 0.83832, df = 1, p-value = 0.3599
```

Modified Mini Screen (mental health)

ever_MMS defined as having had an MMS score ≥ 6 at any stage over the first 48 weeks of the study. Where, an MSS score ≥ 6 indicates that a person should undergo a formal psychiatric evaluation.

```
# Process the MSS data and join with missingness data
df_MMS <- df %>%
    select(ranid, interval_name, mms_total) %>%
    # Determine whether MMS exceeded 5 at any stage of the 48 weeks
    mutate(mms_high = ifelse(mms_total >= 6,
                             yes = 1,
                             no = 0)) %>%
    select(-mms_total) %>%
    pivot_wider(names_from = interval_name,
                values_from = mms_high) %>%
    mutate(ever_MMS = rowSums(.[2:6], na.rm = TRUE)) %>%
    mutate(ever_MMS = ifelse(ever_MMS == 0,
                        yes = 'No',
                        no = 'Yes')) %>%
    select(ranid, ever_MMS)

df_MMS <- df_combined %>%
    select(ranid, any_missing) %>%
    left_join(df_MMS)
```

```
## Joining, by = "ranid"
```

```
# Tabulate and print
df_MMS %>%
```

```
    mutate(ever_MMS = fct_explicit_na(ever_MMS)) %>%
    group_by(ever_MMS, any_missing) %>%
    summarise(count = n()) %>%
    group_by(ever_MMS) %>%
    mutate(total = sum(count),
           proportion = round(count / total, 3)) %>%
    knitr::kable(caption = 'Proportion missing pain data by whether there was ever a high MMS score')
```

Table 7: Proportion missing pain data by whether there was ever a high MMS score

ever_MMS	any_missing	count	total	proportion
No	No	739	983	0.752
No	Yes	244	983	0.248
Yes	No	48	70	0.686
Yes	Yes	22	70	0.314

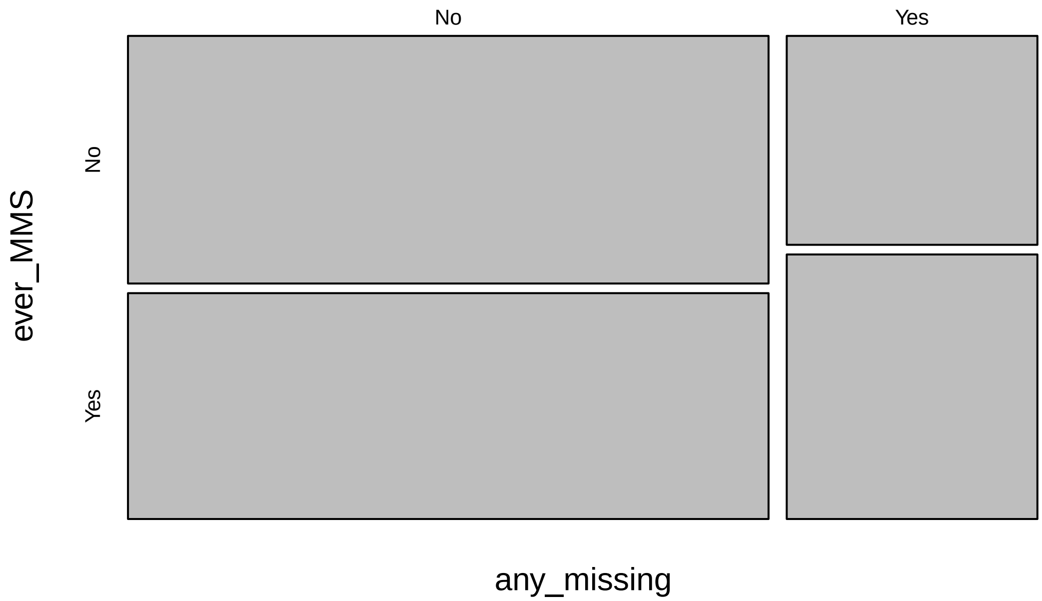
```
# Tabulate, plot, and test
tab_mms <- xtabs(~any_missing + ever_MMS, data = df_MMS)

mosaicplot(tab_mms, main = 'Counts by whether there was ever a high MMS score')
```


```
prop_mms <- prop.table(tab_mms, 2)

mosaicplot(prop_mms, main = 'Proportion of counts by whether there was ever a high MMS score')
```

Proportion of counts by whether there was ever a high MMS score




```r
chisq.test(tab_mms)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_mms
## X-squared = 1.181, df = 1, p-value = 0.2771
```

CD4 T-cell count

`low_CD4` defined as the lowest CD4 T-cell count measured during the course of the first 48 weeks of the study.

```r
# Process the CD4 data and join with missingness data
df_CD4 <- df %>%
    select(ranid, interval_name, cd4_cells.ul) %>%
    # Determine highest VL per participant
    group_by(ranid) %>%
    summarise(low_CD4 = min(cd4_cells.ul, na.rm = TRUE))

df_CD4 <- df_combined %>%
    select(ranid, any_missing) %>%
    left_join(df_CD4)
```

```
## Joining, by = "ranid"
```

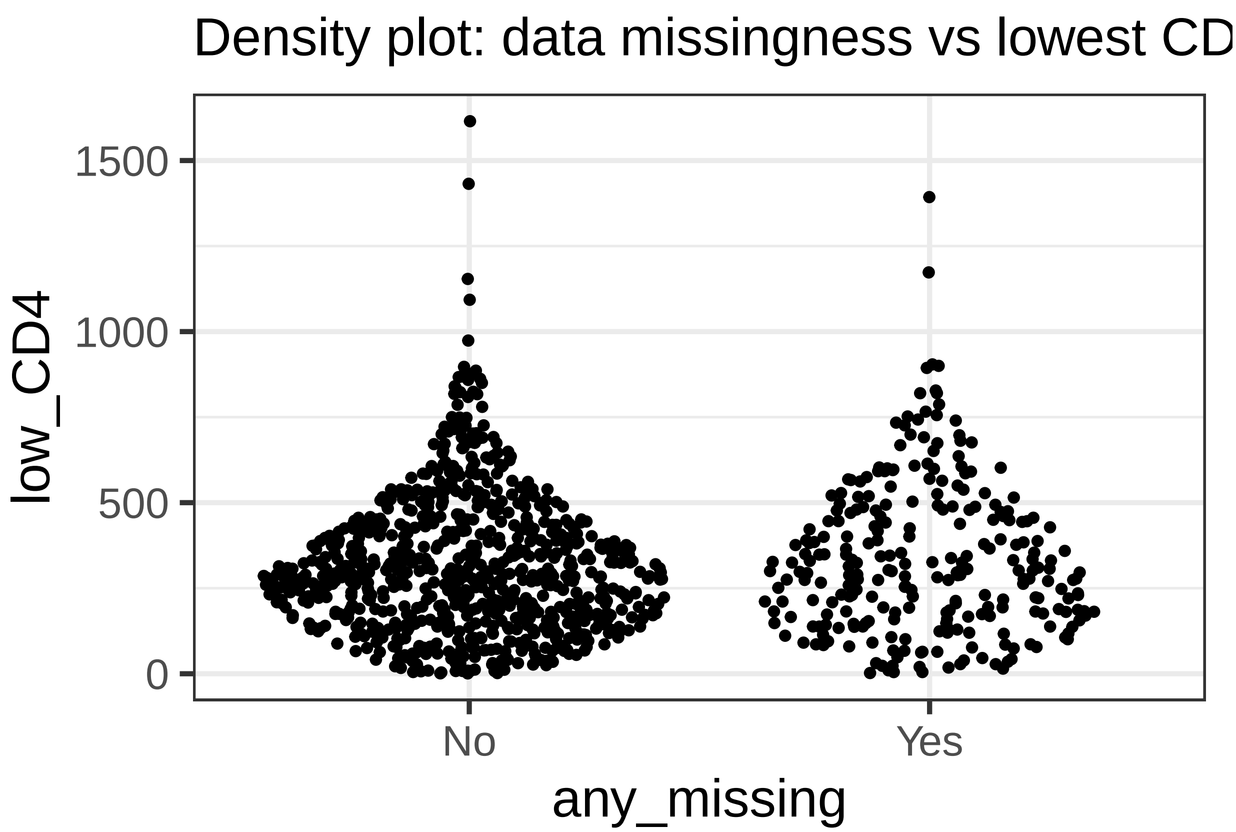
```r
# Tabulate and print
df_CD4 %>%
    group_by(any_missing) %>%
    select(any_missing, low_CD4) %>%
    skim() %>%
    skimr::kable(caption = 'Data missingness by lowest CD4')
```

Skim summary statistics
n obs: 1053
n variables: 2

Variable type: numeric

any_missing	variable	missing	complete	n	mean	sd	p0	p25	p50	p75	p100
No	low_CD4	0	787	787	315.84	201.84	1	171	286	424	1615
Yes	low_CD4	0	266	266	335.17	221.53	2	170.75	300	476.5	1393

```
# Plot, and test
ggplot(data = df_CD4) +
    aes(x = any_missing,
        y = low_CD4) +
    geom_sina() +
    labs(subtitle = 'Density plot: data missingness vs lowest CD4')
```




```
wilcox.test(low_CD4 ~ any_missing, data = df_CD4)
```

```
Wilcoxon rank sum test with continuity correction
```

data: low_CD4 by any_missing W = 100308, p-value = 0.309 alternative hypothesis: true location shift is not equal to 0

Viral load

`high_VL` defined as the highest viral load measured during the course of the first 48 weeks of the study.

```
# Process the VL data and join with missingness data
df_VL <- df %>%
    select(ranid, interval_name, viral_load_cp.ml) %>%
    # Determine highest VL per participant
    group_by(ranid) %>%
    summarise(high_VL = max(viral_load_cp.ml, na.rm = TRUE))

df_VL <- df_combined %>%
    select(ranid, any_missing) %>%
    left_join(df_VL)
```

```
## Joining, by = "ranid"
```

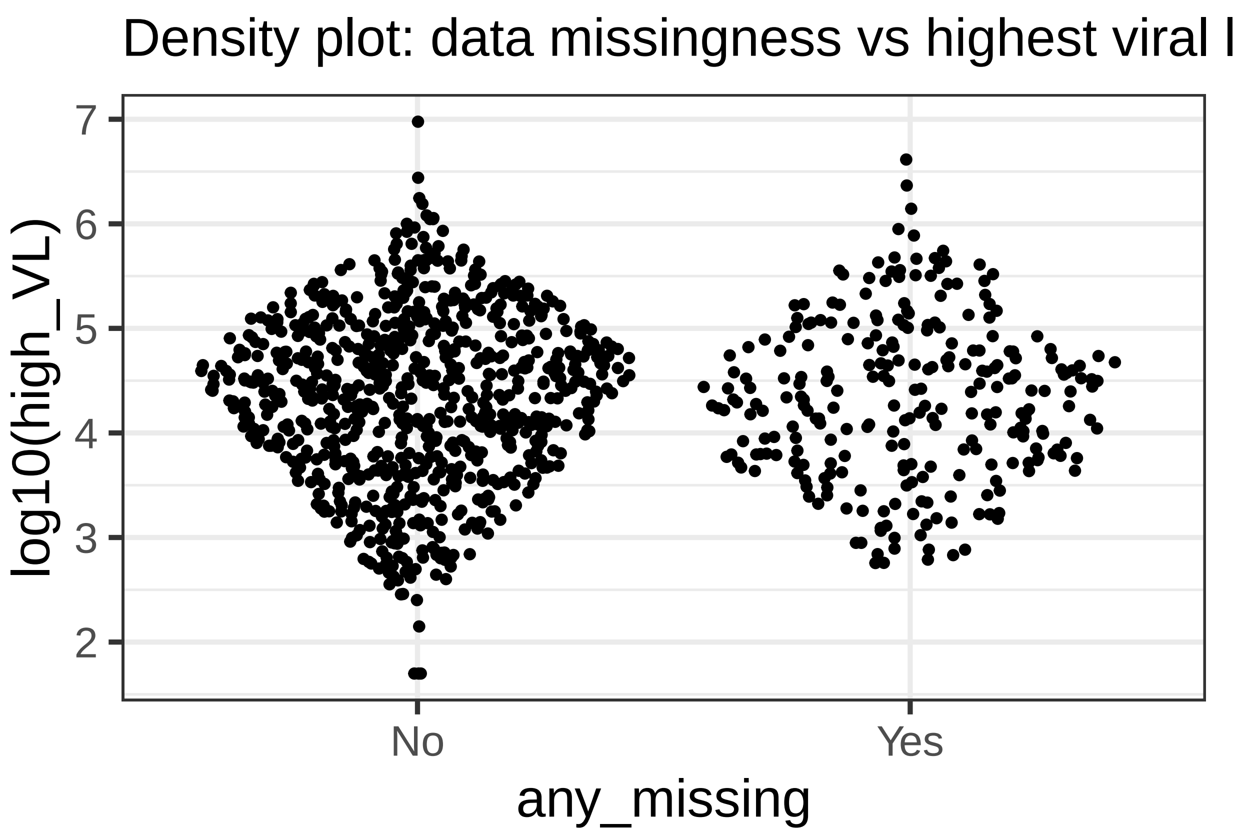
```r
# Tabulate and print
df_VL %>%
    group_by(any_missing) %>%
    select(any_missing, high_VL) %>%
    skim() %>%
    skimr::kable(caption = 'Data missingness by greatest viral load')
```

Skim summary statistics
n obs: 1053
n variables: 2

Variable type: numeric

any_missing	variable	missing	complete	n	mean	sd	p0	p25	p50	p75	p100
No	high_VL	0	787	787	99267.18	386828.79	50	5791	26333	85912.5	9475772
Yes	high_VL	0	266	266	96926.88	318787.27	570	5935.5	20859.5	66562	4117370

```r
# Plot, and test
ggplot(data = df_VL) +
    aes(x = any_missing,
        y = log10(high_VL)) +
    geom_sina() +
    labs(subtitle = 'Density plot: data missingness vs highest viral load')
```




```r
wilcox.test(high_VL ~ any_missing, data = df_VL)
```

```
Wilcoxon rank sum test with continuity correction
```

data: high_VL by any_missing W = 107520, p-value = 0.5064 alternative hypothesis: true location shift is not equal to 0

Study variables

Proportion missing pain data by study site

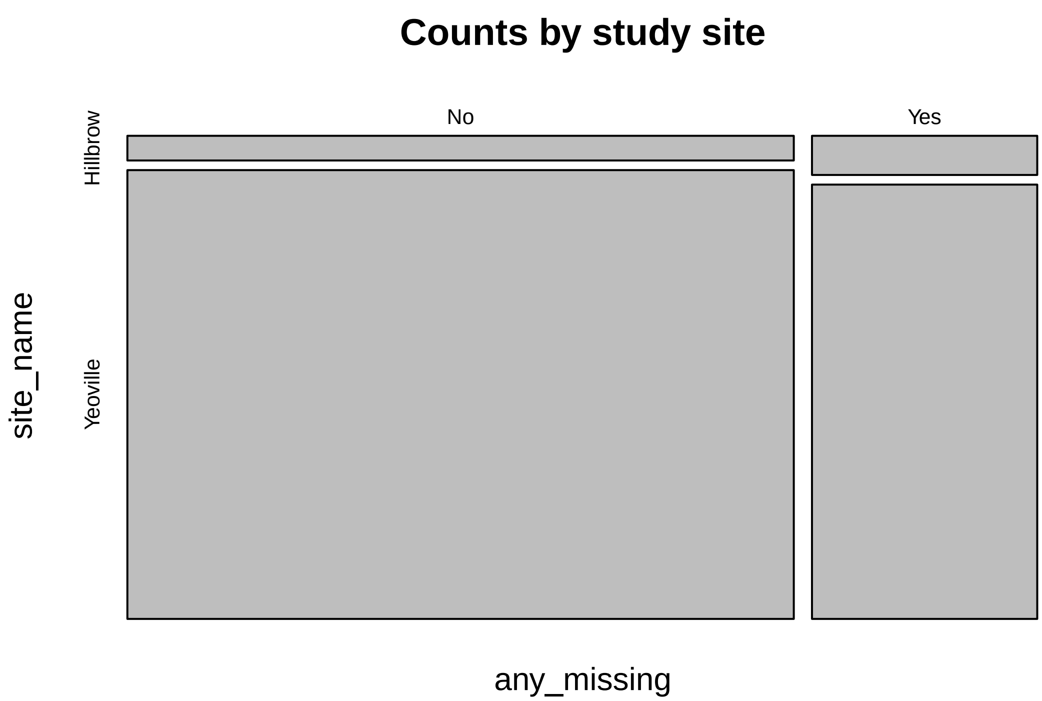
```
# Tabulate and print
df %>%
    filter(interval_name == '0 weeks') %>%
    group_by(site_name, any_missing) %>%
    summarise(count = n()) %>%
    group_by(site_name) %>%
    mutate(total = sum(count),
           proportion = round(count / total,3)) %>%
    knitr::kable(caption = 'Proportion missing pain data by study site')
```

Table 10: Proportion missing pain data by study site

site_name	any_missing	count	total	proportion
Wits RHI Shandukani Hillbrow Johannesburg	No	41	63	0.651
Wits RHI Shandukani Hillbrow Johannesburg	Yes	22	63	0.349
Wits RHI Yeoville Research Centre	No	746	990	0.754
Wits RHI Yeoville Research Centre	Yes	244	990	0.246

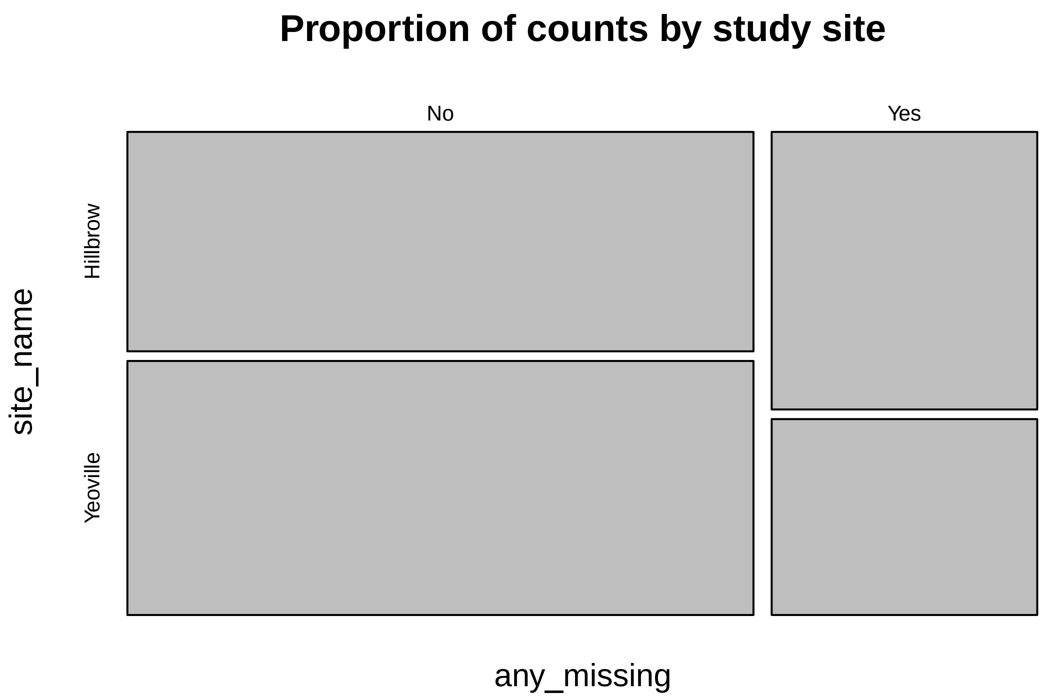
```
# Tabulate, plot, and test
tab_site <- df %>%
    filter(interval_name == '0 weeks') %>%
    mutate(site_name = ifelse(site_name == 'Wits RHI Yeoville Research Centre',
                              yes = 'Yeoville',
                              no = 'Hillbrow')) %>%
    xtabs(~any_missing + site_name, data = .)

mosaicplot(tab_site, main = 'Counts by study site')
```




```
prop_site <- prop.table(tab_site, 2)

mosaicplot(prop_site, main = 'Proportion of counts by study site')
```




```
chisq.test(tab_site)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_site
## X-squared = 2.7898, df = 1, p-value = 0.09487
```

Proportion missing pain data by group allocation

- GROUP 1: DTG + TAF + FTC
- GROUP 2: DTG + TDF + FTC
- GROUP 3: EFV + TDF + FTC

```
# Tabulate and print
df %>%
    filter(interval_name == '0 weeks') %>%
    group_by(group, any_missing) %>%
    summarise(count = n()) %>%
    group_by(group) %>%
    mutate(total = sum(count),
           proportion = round(count / total,3)) %>%
    knitr::kable(caption = 'Proportion missing pain data by group allocation')
```

Table 11: Proportion missing pain data by group allocation

group	any_missing	count	total	proportion
GROUP 1 (DTG + TAF + FTC)	No	262	351	0.746
GROUP 1 (DTG + TAF + FTC)	Yes	89	351	0.254
GROUP 2 (DTG + TDF + FTC)	No	274	351	0.781
GROUP 2 (DTG + TDF + FTC)	Yes	77	351	0.219
GROUP 3 (EFV + TDF + FTC)	No	251	351	0.715
GROUP 3 (EFV + TDF + FTC)	Yes	100	351	0.285

```r
# Tabulate, plot, and test
tab_group <- df %>%
    filter(interval_name == '0 weeks') %>%
    mutate(group = case_when(
        str_detect(group, 'EFV') ~ 'GROUP 3',
        str_detect(group, 'TDF') ~ 'GROUP 2',
        str_detect(group, 'TAF') ~ 'GROUP 1'
    )) %>%
    xtabs(~any_missing + group, data = .)

mosaicplot(tab_group, main = 'Counts by group allocation',
           ylab = 'Allocation')
```


```r
prop_group <- prop.table(tab_group, 2)

mosaicplot(prop_group, main = 'Proportions by group allocation',
           ylab = 'Allocation')
```


```
chisq.test(tab_group)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tab_group
## X-squared = 3.9939, df = 2, p-value = 0.1358
```

Session information

```
sessionInfo()
```

```
## R version 3.6.1 (2019-07-05)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Mojave 10.14.6
##
## 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] ggforce_0.3.1   skimr_1.0.7     forcats_0.4.0   stringr_1.4.0
##  [5] dplyr_0.8.3     purrr_0.3.3     readr_1.3.1     tidyr_1.0.0
##  [9] tibble_2.1.3    ggplot2_3.2.1   tidyverse_1.2.1
##
## loaded via a namespace (and not attached):
##  [1] tidyselect_0.2.5 xfun_0.10        haven_2.1.1      lattice_0.20-38
##  [5] colorspace_1.4-1 vctrs_0.2.0      generics_0.0.2   htmltools_0.4.0
```

```
##  [9] yaml_2.2.0       utf8_1.1.4       rlang_0.4.0      pillar_1.4.2
## [13] glue_1.3.1       withr_2.1.2      tweenr_1.0.1     modelr_0.1.5
## [17] readxl_1.3.1     lifecycle_0.1.0  munsell_0.5.0    gtable_0.3.0
## [21] cellranger_1.1.0 rvest_0.3.4      evaluate_0.14    labeling_0.3
## [25] knitr_1.25       fansi_0.4.0      highr_0.8        broom_0.5.2
## [29] Rcpp_1.0.2       scales_1.0.0     backports_1.1.5  jsonlite_1.6
## [33] farver_1.1.0     hms_0.5.1        digest_0.6.22    stringi_1.4.3
## [37] polyclip_1.10-0  grid_3.6.1       cli_1.1.0        tools_3.6.1
## [41] magrittr_1.5     lazyeval_0.2.2   crayon_1.3.4     pkgconfig_2.0.3
## [45] zeallot_0.1.0    MASS_7.3-51.4    xml2_1.2.2       lubridate_1.7.4
## [49] assertthat_0.2.1 rmarkdown_1.16   httr_1.4.1       rstudioapi_0.10
## [53] R6_2.4.0         nlme_3.1-141     compiler_3.6.1
```