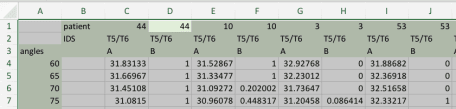
Well, here’s what I was dealing with:



*Exemplar Excel file from collaborator*

Notice that we have 3 header rows, first with patient IDs, second with spine region, and third with variable names (A and B, to protect the innocent).

**Goal**

A dataset that, for each patient and each angle gives us corresponding values of A and B. So this would be a four-column data set with ID, angle, A and B.

**Attempt 1 (readxl)**

d1 <- readxl::read\_excel('spreadsheet1.xlsx')

head(d1)

## # A tibble: 6 x 26  
## X\_\_1 patient `44` `44\_\_1` `10` `10\_\_1` `3` `3\_\_1` `53` `53\_\_1`  
##  
## 1 IDS T5/T6 T5/T6 T5/T6 T5/T6 T5/T6 T5/T6 T5/T6 T5/T6  
## 2 angles A B A B A B A B  
## 3 60 31.83… 1 31.52… 1 32.9… 0 31.8… 0  
## 4 65 31.66… 1 31.33… 1 32.2… 0 32.3… 0  
## 5 70 31.45… 1 31.09… 0.20200… 31.7… 0 32.5… 0  
## 6 75 31.08… 1 30.96… 0.44831… 31.2… 8.641… 32.3… 1  
## # … with 16 more variables: `2` , `2\_\_1` `8` ,  
## # `8\_\_1` , `6` , `6\_\_1` , `43` , `43\_\_1` ,  
## # `48` , `48\_\_1` , `46` , `46\_\_1` , `4` ,  
## # `4\_\_1` , `9` , `9\_\_1`

This strategy gives us funky column names, and pushes two of the headers into data rows. Since the headers are in rows, they’re a little harder to extract and work with. More worrisome is the fact that since the headers leaked into the data rows, the columns are all of type character rather than type numeric, which would now require further careful conversion after cleaning. So I don’t think readxl is the way to go here, if there’s a better solution.

**Attempt 2 (tidyxl)**

d2 <- tidyxl::xlsx\_cells('spreadsheet1.xlsx')

head(d2)

## # A tibble: 6 x 21  
## sheet address row col is\_blank data\_type error logical numeric  
##  
## 1 T5T6 B1 1 2 FALSE character NA NA  
## 2 T5T6 C1 1 3 FALSE numeric NA 44.  
## 3 T5T6 D1 1 4 FALSE numeric NA 44.  
## 4 T5T6 E1 1 5 FALSE numeric NA 10.  
## 5 T5T6 F1 1 6 FALSE numeric NA 10.  
## 6 T5T6 G1 1 7 FALSE numeric NA 3.  
## # … with 12 more variables: date , character ,  
## # character\_formatted , formula , is\_array ,  
## # formula\_ref , formula\_group , comment , height ,  
## # width , style\_format , local\_format\_id

The xlsx\_cells captures the data in a tidy fashion, explicitly calling out rows and columns and other metadata within each cell. We can clean up this data using tidyverse functions:

library(tidyverse)

cleanData1 = function(d) {

angle = d %>% filter(row >= 4, col == 1) %>% pull(numeric)

name = d %>% filter(row %in% c(1,3), col >= 3) %>%

mutate(character = ifelse(is.na(character),

as.character(numeric),

character)) %>%

select(row, col, character) %>%

filter(!is.na(character)) %>%

spread(row, character) %>%

unite(ID, `1`:`3`, sep = '\_') %>%

pull(ID)

data = d %>% filter(row >= 4, col >= 3) %>%

filter(!is.na(numeric)) %>%

select(row, col, numeric) %>%

spread(col, numeric) %>%

select(-row) %>%

set\_names(name) %>%

cbind(angle) %>%

gather(variable, value, -angle) %>%

separate(variable, c('ID','Measure'), sep = '\_') %>%

spread(Measure, value) %>%

select(ID, angle, A, B) %>%

arrange(ID, angle)

return(data)

}

head(cleanData1(d2))

## ID angle A B

## 1 10 60 31.52867 1.000000

## 2 10 65 31.33477 1.000000

## 3 10 70 31.09272 0.202002

## 4 10 75 30.96078 0.448317

## 5 10 80 30.79397 0.670876

## 6 10 85 30.52185 0.461406

This is a lot of data munging, and though dplyr is powerful, it took a lot of trial and error to get the final pipeline done.Nonetheless, I was really psyched about tidyxl, since it automated a job that would have taken manual manipulation (I had 12 spreadsheets like this to process). I was going to write a blog post on this cool package that made my life dealing with messy Excel file a piece of cake. But wait, there’s more…

**Attempt 3 (tidyxl + unpivotr)**

I didn’t know about unpivotr until this post:

So maybe all that complicated munging can be simplfied.

library(unpivotr)

cleanData2 = function(d){

bl = d %>% select(row, col, data\_type, numeric, character) %>%

behead('N', ID) %>%

behead('N', spine) %>%

behead('N', variable)

# Extract the angles column

bl1 = bl %>% filter(variable == 'angles') %>% spatter(variable) %>%

select(row, angles)

# Extract the rest of the columns

bl2 = bl %>% filter(variable %in% c('A','B')) %>% select(-spine, -col) %>%

spatter(ID) %>% # Spread to columns

select(-character) %>% # All my variables are numeric

gather(ID, value, -row, -variable) %>%

spread(variable, value)

final = bl1 %>% left\_join(bl2) %>% # put things back together

arrange(ID, angles) %>%

select(ID, everything(),-row) # re-arrange columns

return(final)

}

cleanData2(d2)

## # A tibble: 588 x 4

## ID angles A B

##

## 1 10 60. 31.5 1.00

## 2 10 65. 31.3 1.00

## 3 10 70. 31.1 0.202

## 4 10 75. 31.0 0.448

## 5 10 80. 30.8 0.671

## 6 10 85. 30.5 0.461

## 7 10 90. 30.3 0.245

## 8 10 95. 30.0 0.159

## 9 10 100. 29.7 0.170

## 10 10 105. 29.2 0.421

## # ... with 578 more rows

In this example, I’m using the behead function (available in the development version of unpivotr on GitHub) to extract out the three rows of headers. Then I’m extracting out the angles column separately and merging it with the rest of the columns.

In case you’re wondering about the “N” in the behead code, unpivotr has a geographic options system as to where the headers are with respect to the main code. This vignette explains this nomenclature:

Code Chunks

This vignette for the unpivotr package demonstrates unpivoting pivot tables of various kinds imported from a spreadsheet via the tidyxl package. It is best read with the spreadsheet open in a spreadsheet program, e.g. Excel, LibreOffice Calc or Gnumeric.

The spreadsheet is in the package directory at system.file("extdata", "purpose.xlsx", package = "unpivotr").

library(dplyr)

##

## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':

##

## filter, lag

## The following objects are masked from 'package:base':

##

## intersect, setdiff, setequal, union

library(tidyxl)

library(unpivotr)

NNW WNW example

(original <- purpose$`NNW WNW`)

## X2 X3 X4 X5 X6 X7

## 1 <NA> <NA> Female <NA> Male <NA>

## 2 <NA> <NA> 0 - 6 7 - 10 0 - 6 7 - 10

## 3 Bachelor's degree 15 - 24 7000 27000 <NA> 13000

## 4 <NA> 25 - 44 12000 137000 9000 81000

## 5 <NA> 45 - 64 10000 64000 7000 66000

## 6 <NA> 65+ <NA> 18000 7000 17000

## 7 Certificate 15 - 24 29000 161000 30000 190000

## 8 <NA> 25 - 44 34000 179000 31000 219000

## 9 <NA> 45 - 64 30000 210000 23000 199000

## 10 <NA> 65+ 12000 77000 8000 107000

## 11 Diploma 15 - 24 <NA> 14000 9000 11000

## 12 <NA> 25 - 44 10000 66000 8000 47000

## 13 <NA> 45 - 64 6000 68000 5000 58000

## 14 <NA> 65+ 5000 41000 1000 34000

## 15 No Qualification 15 - 24 10000 43000 12000 37000

## 16 <NA> 25 - 44 11000 36000 21000 50000

## 17 <NA> 45 - 64 19000 91000 17000 75000

## 18 <NA> 65+ 16000 118000 9000 66000

## 19 Postgraduate qualification 15 - 24 <NA> 6000 <NA> <NA>

## 20 <NA> 25 - 44 5000 86000 7000 60000

## 21 <NA> 45 - 64 6000 55000 6000 68000

## 22 <NA> 65+ <NA> 13000 <NA> 18000

tail(cells <- as\_cells(original))

## # A tibble: 6 x 4

## row col data\_type chr

## <int> <int> <chr> <chr>

## 1 17 6 chr 75000

## 2 18 6 chr 66000

## 3 19 6 chr <NA>

## 4 20 6 chr 60000

## 5 21 6 chr 68000

## 6 22 6 chr 18000

Headers

row\_headers <-

cells %>%

dplyr::filter(col <= 2, !is.na(chr)) %>% # Select all rows of headers at once

select(row, col, header = chr) %>%

split(.$col) # Return each row of headers in its own element of a list

row\_headers

## $`1`

## # A tibble: 5 x 3

## row col header

## <int> <int> <chr>

## 1 3 1 Bachelor's degree

## 2 7 1 Certificate

## 3 11 1 Diploma

## 4 15 1 No Qualification

## 5 19 1 Postgraduate qualification

##

## $`2`

## # A tibble: 20 x 3

## row col header

## <int> <int> <chr>

## 1 3 2 15 - 24

## 2 4 2 25 - 44

## 3 5 2 45 - 64

## 4 6 2 65+

## 5 7 2 15 - 24

## 6 8 2 25 - 44

## 7 9 2 45 - 64

## 8 10 2 65+

## 9 11 2 15 - 24

## 10 12 2 25 - 44

## 11 13 2 45 - 64

## 12 14 2 65+

## 13 15 2 15 - 24

## 14 16 2 25 - 44

## 15 17 2 45 - 64

## 16 18 2 65+

## 17 19 2 15 - 24

## 18 20 2 25 - 44

## 19 21 2 45 - 64

## 20 22 2 65+

col\_headers <-

cells %>%

dplyr::filter(row <= 2, !is.na(chr)) %>%

select(row, col, header = chr) %>%

split(.$row)

col\_headers

## $`1`

## # A tibble: 2 x 3

## row col header

## <int> <int> <chr>

## 1 1 3 Female

## 2 1 5 Male

##

## $`2`

## # A tibble: 4 x 3

## row col header

## <int> <int> <chr>

## 1 2 3 0 - 6

## 2 2 4 7 - 10

## 3 2 5 0 - 6

## 4 2 6 7 - 10

Data

data\_cells <-

cells %>%

dplyr::filter(row >= 3, col >= 3, !is.na(chr)) %>%

mutate(value = as.integer(chr)) %>%

select(row, col, value)

head(data\_cells)

## # A tibble: 6 x 3

## row col value

## <int> <int> <int>

## 1 3 3 7000

## 2 4 3 12000

## 3 5 3 10000

## 4 7 3 29000

## 5 8 3 34000

## 6 9 3 30000

Join headers to data

data\_cells <-

data\_cells %>%

enhead(col\_headers[[1]], "NNW") %>%

enhead(col\_headers[[2]], "N") %>%

enhead(row\_headers[[1]], "WNW") %>%

enhead(row\_headers[[2]], "W")

Result

data\_cells %>% as.data.frame

## row col value header.data header.header header.data.data

## 1 3 3 7000 Female 0 - 6 Bachelor's degree

## 2 4 3 12000 Female 0 - 6 Bachelor's degree

## 3 5 3 10000 Female 0 - 6 Bachelor's degree

## 4 3 4 27000 Female 7 - 10 Bachelor's degree

## 5 4 4 137000 Female 7 - 10 Bachelor's degree

## 6 5 4 64000 Female 7 - 10 Bachelor's degree

## 7 6 4 18000 Female 7 - 10 Bachelor's degree

## 8 4 5 9000 Male 0 - 6 Bachelor's degree

## 9 5 5 7000 Male 0 - 6 Bachelor's degree

## 10 6 5 7000 Male 0 - 6 Bachelor's degree

## 11 3 6 13000 Male 7 - 10 Bachelor's degree

## 12 4 6 81000 Male 7 - 10 Bachelor's degree

## 13 5 6 66000 Male 7 - 10 Bachelor's degree

## 14 6 6 17000 Male 7 - 10 Bachelor's degree

## 15 7 3 29000 Female 0 - 6 Certificate

## 16 8 3 34000 Female 0 - 6 Certificate

## 17 9 3 30000 Female 0 - 6 Certificate

## 18 10 3 12000 Female 0 - 6 Certificate

## 19 7 4 161000 Female 7 - 10 Certificate

## 20 8 4 179000 Female 7 - 10 Certificate

## 21 9 4 210000 Female 7 - 10 Certificate

## 22 10 4 77000 Female 7 - 10 Certificate

## 23 7 5 30000 Male 0 - 6 Certificate

## 24 8 5 31000 Male 0 - 6 Certificate

## 25 9 5 23000 Male 0 - 6 Certificate

## 26 10 5 8000 Male 0 - 6 Certificate

## 27 7 6 190000 Male 7 - 10 Certificate

## 28 8 6 219000 Male 7 - 10 Certificate

## 29 9 6 199000 Male 7 - 10 Certificate

## 30 10 6 107000 Male 7 - 10 Certificate

## 31 12 3 10000 Female 0 - 6 Diploma

## 32 13 3 6000 Female 0 - 6 Diploma

## 33 14 3 5000 Female 0 - 6 Diploma

## 34 11 4 14000 Female 7 - 10 Diploma

## 35 12 4 66000 Female 7 - 10 Diploma

## 36 13 4 68000 Female 7 - 10 Diploma

## 37 14 4 41000 Female 7 - 10 Diploma

## 38 11 5 9000 Male 0 - 6 Diploma

## 39 12 5 8000 Male 0 - 6 Diploma

## 40 13 5 5000 Male 0 - 6 Diploma

## 41 14 5 1000 Male 0 - 6 Diploma

## 42 11 6 11000 Male 7 - 10 Diploma

## 43 12 6 47000 Male 7 - 10 Diploma

## 44 13 6 58000 Male 7 - 10 Diploma

## 45 14 6 34000 Male 7 - 10 Diploma

## 46 15 3 10000 Female 0 - 6 No Qualification

## 47 16 3 11000 Female 0 - 6 No Qualification

## 48 17 3 19000 Female 0 - 6 No Qualification

## 49 18 3 16000 Female 0 - 6 No Qualification

## 50 15 4 43000 Female 7 - 10 No Qualification

## 51 16 4 36000 Female 7 - 10 No Qualification

## 52 17 4 91000 Female 7 - 10 No Qualification

## 53 18 4 118000 Female 7 - 10 No Qualification

## 54 15 5 12000 Male 0 - 6 No Qualification

## 55 16 5 21000 Male 0 - 6 No Qualification

## 56 17 5 17000 Male 0 - 6 No Qualification

## 57 18 5 9000 Male 0 - 6 No Qualification

## 58 15 6 37000 Male 7 - 10 No Qualification

## 59 16 6 50000 Male 7 - 10 No Qualification

## 60 17 6 75000 Male 7 - 10 No Qualification

## 61 18 6 66000 Male 7 - 10 No Qualification

## 62 20 3 5000 Female 0 - 6 Postgraduate qualification

## 63 21 3 6000 Female 0 - 6 Postgraduate qualification

## 64 19 4 6000 Female 7 - 10 Postgraduate qualification

## 65 20 4 86000 Female 7 - 10 Postgraduate qualification

## 66 21 4 55000 Female 7 - 10 Postgraduate qualification

## 67 22 4 13000 Female 7 - 10 Postgraduate qualification

## 68 20 5 7000 Male 0 - 6 Postgraduate qualification

## 69 21 5 6000 Male 0 - 6 Postgraduate qualification

## 70 20 6 60000 Male 7 - 10 Postgraduate qualification

## 71 21 6 68000 Male 7 - 10 Postgraduate qualification

## 72 22 6 18000 Male 7 - 10 Postgraduate qualification

## header.header.header

## 1 15 - 24

## 2 25 - 44

## 3 45 - 64

## 4 15 - 24

## 5 25 - 44

## 6 45 - 64

## 7 65+

## 8 25 - 44

## 9 45 - 64

## 10 65+

## 11 15 - 24

## 12 25 - 44

## 13 45 - 64

## 14 65+

## 15 15 - 24

## 16 25 - 44

## 17 45 - 64

## 18 65+

## 19 15 - 24

## 20 25 - 44

## 21 45 - 64

## 22 65+

## 23 15 - 24

## 24 25 - 44

## 25 45 - 64

## 26 65+

## 27 15 - 24

## 28 25 - 44

## 29 45 - 64

## 30 65+

## 31 25 - 44

## 32 45 - 64

## 33 65+

## 34 15 - 24

## 35 25 - 44

## 36 45 - 64

## 37 65+

## 38 15 - 24

## 39 25 - 44

## 40 45 - 64

## 41 65+

## 42 15 - 24

## 43 25 - 44

## 44 45 - 64

## 45 65+

## 46 15 - 24

## 47 25 - 44

## 48 45 - 64

## 49 65+

## 50 15 - 24

## 51 25 - 44

## 52 45 - 64

## 53 65+

## 54 15 - 24

## 55 25 - 44

## 56 45 - 64

## 57 65+

## 58 15 - 24

## 59 25 - 44

## 60 45 - 64

## 61 65+

## 62 25 - 44

## 63 45 - 64

## 64 15 - 24

## 65 25 - 44

## 66 45 - 64

## 67 65+

## 68 25 - 44

## 69 45 - 64

## 70 25 - 44

## 71 45 - 64

## 72 65+

NNW\_WNW <- data\_cells %>% arrange(row, col)

NNE WSW example

cells <- as\_cells(purpose$`NNE WSW`)

Headers (same as NNW WNW)

row\_headers <-

cells %>%

dplyr::filter(col <= 2, !is.na(chr)) %>%

select(row, col, header = chr) %>%

split(.$col)

row\_headers

## $`1`

## # A tibble: 5 x 3

## row col header

## <int> <int> <chr>

## 1 6 1 Bachelor's degree

## 2 10 1 Certificate

## 3 14 1 Diploma

## 4 18 1 No Qualification

## 5 22 1 Postgraduate qualification

##

## $`2`

## # A tibble: 20 x 3

## row col header

## <int> <int> <chr>

## 1 3 2 15 - 24

## 2 4 2 25 - 44

## 3 5 2 45 - 64

## 4 6 2 65+

## 5 7 2 15 - 24

## 6 8 2 25 - 44

## 7 9 2 45 - 64

## 8 10 2 65+

## 9 11 2 15 - 24

## 10 12 2 25 - 44

## 11 13 2 45 - 64

## 12 14 2 65+

## 13 15 2 15 - 24

## 14 16 2 25 - 44

## 15 17 2 45 - 64

## 16 18 2 65+

## 17 19 2 15 - 24

## 18 20 2 25 - 44

## 19 21 2 45 - 64

## 20 22 2 65+

col\_headers <-

cells %>%

dplyr::filter(row <= 2, !is.na(chr)) %>%

select(row, col, header = chr) %>%

split(.$row)

col\_headers

## $`1`

## # A tibble: 2 x 3

## row col header

## <int> <int> <chr>

## 1 1 4 Female

## 2 1 6 Male

##

## $`2`

## # A tibble: 4 x 3

## row col header

## <int> <int> <chr>

## 1 2 3 0 - 6

## 2 2 4 7 - 10

## 3 2 5 0 - 6

## 4 2 6 7 - 10

Data (same as NNW WNW)

data\_cells <-

cells %>%

dplyr::filter(row >= 3, col >= 3, !is.na(chr)) %>%

mutate(value = as.integer(chr)) %>%

select(row, col, value)

data\_cells

## # A tibble: 72 x 3

## row col value

## <int> <int> <int>

## 1 3 3 7000

## 2 4 3 12000

## 3 5 3 10000

## 4 7 3 29000

## 5 8 3 34000

## 6 9 3 30000

## 7 10 3 12000

## 8 12 3 10000

## 9 13 3 6000

## 10 14 3 5000

## # … with 62 more rows

Join headers to data (different from NNW WNW)

data\_cells <-

data\_cells %>%

enhead(col\_headers[[1]], "NNE") %>% # Different from NNW WNW

enhead(col\_headers[[2]], "N") %>% # Same as NNW WNW

enhead(row\_headers[[1]], "WSW") %>% # Different from NNW WNW

enhead(row\_headers[[2]], "W") # Same as NNW WNW

Result

data\_cells %>% as.data.frame

## row col value header.data header.header header.data.data

## 1 3 3 7000 Female 0 - 6 Bachelor's degree

## 2 4 3 12000 Female 0 - 6 Bachelor's degree

## 3 5 3 10000 Female 0 - 6 Bachelor's degree

## 4 3 4 27000 Female 7 - 10 Bachelor's degree

## 5 4 4 137000 Female 7 - 10 Bachelor's degree

## 6 5 4 64000 Female 7 - 10 Bachelor's degree

## 7 6 4 18000 Female 7 - 10 Bachelor's degree

## 8 4 5 9000 Male 0 - 6 Bachelor's degree

## 9 5 5 7000 Male 0 - 6 Bachelor's degree

## 10 6 5 7000 Male 0 - 6 Bachelor's degree

## 11 3 6 13000 Male 7 - 10 Bachelor's degree

## 12 4 6 81000 Male 7 - 10 Bachelor's degree

## 13 5 6 66000 Male 7 - 10 Bachelor's degree

## 14 6 6 17000 Male 7 - 10 Bachelor's degree

## 15 7 3 29000 Female 0 - 6 Certificate

## 16 8 3 34000 Female 0 - 6 Certificate

## 17 9 3 30000 Female 0 - 6 Certificate

## 18 10 3 12000 Female 0 - 6 Certificate

## 19 7 4 161000 Female 7 - 10 Certificate

## 20 8 4 179000 Female 7 - 10 Certificate

## 21 9 4 210000 Female 7 - 10 Certificate

## 22 10 4 77000 Female 7 - 10 Certificate

## 23 7 5 30000 Male 0 - 6 Certificate

## 24 8 5 31000 Male 0 - 6 Certificate

## 25 9 5 23000 Male 0 - 6 Certificate

## 26 10 5 8000 Male 0 - 6 Certificate

## 27 7 6 190000 Male 7 - 10 Certificate

## 28 8 6 219000 Male 7 - 10 Certificate

## 29 9 6 199000 Male 7 - 10 Certificate

## 30 10 6 107000 Male 7 - 10 Certificate

## 31 12 3 10000 Female 0 - 6 Diploma

## 32 13 3 6000 Female 0 - 6 Diploma

## 33 14 3 5000 Female 0 - 6 Diploma

## 34 11 4 14000 Female 7 - 10 Diploma

## 35 12 4 66000 Female 7 - 10 Diploma

## 36 13 4 68000 Female 7 - 10 Diploma

## 37 14 4 41000 Female 7 - 10 Diploma

## 38 11 5 9000 Male 0 - 6 Diploma

## 39 12 5 8000 Male 0 - 6 Diploma

## 40 13 5 5000 Male 0 - 6 Diploma

## 41 14 5 1000 Male 0 - 6 Diploma

## 42 11 6 11000 Male 7 - 10 Diploma

## 43 12 6 47000 Male 7 - 10 Diploma

## 44 13 6 58000 Male 7 - 10 Diploma

## 45 14 6 34000 Male 7 - 10 Diploma

## 46 15 3 10000 Female 0 - 6 No Qualification

## 47 16 3 11000 Female 0 - 6 No Qualification

## 48 17 3 19000 Female 0 - 6 No Qualification

## 49 18 3 16000 Female 0 - 6 No Qualification

## 50 15 4 43000 Female 7 - 10 No Qualification

## 51 16 4 36000 Female 7 - 10 No Qualification

## 52 17 4 91000 Female 7 - 10 No Qualification

## 53 18 4 118000 Female 7 - 10 No Qualification

## 54 15 5 12000 Male 0 - 6 No Qualification

## 55 16 5 21000 Male 0 - 6 No Qualification

## 56 17 5 17000 Male 0 - 6 No Qualification

## 57 18 5 9000 Male 0 - 6 No Qualification

## 58 15 6 37000 Male 7 - 10 No Qualification

## 59 16 6 50000 Male 7 - 10 No Qualification

## 60 17 6 75000 Male 7 - 10 No Qualification

## 61 18 6 66000 Male 7 - 10 No Qualification

## 62 20 3 5000 Female 0 - 6 Postgraduate qualification

## 63 21 3 6000 Female 0 - 6 Postgraduate qualification

## 64 19 4 6000 Female 7 - 10 Postgraduate qualification

## 65 20 4 86000 Female 7 - 10 Postgraduate qualification

## 66 21 4 55000 Female 7 - 10 Postgraduate qualification

## 67 22 4 13000 Female 7 - 10 Postgraduate qualification

## 68 20 5 7000 Male 0 - 6 Postgraduate qualification

## 69 21 5 6000 Male 0 - 6 Postgraduate qualification

## 70 20 6 60000 Male 7 - 10 Postgraduate qualification

## 71 21 6 68000 Male 7 - 10 Postgraduate qualification

## 72 22 6 18000 Male 7 - 10 Postgraduate qualification

## header.header.header

## 1 15 - 24

## 2 25 - 44

## 3 45 - 64

## 4 15 - 24

## 5 25 - 44

## 6 45 - 64

## 7 65+

## 8 25 - 44

## 9 45 - 64

## 10 65+

## 11 15 - 24

## 12 25 - 44

## 13 45 - 64

## 14 65+

## 15 15 - 24

## 16 25 - 44

## 17 45 - 64

## 18 65+

## 19 15 - 24

## 20 25 - 44

## 21 45 - 64

## 22 65+

## 23 15 - 24

## 24 25 - 44

## 25 45 - 64

## 26 65+

## 27 15 - 24

## 28 25 - 44

## 29 45 - 64

## 30 65+

## 31 25 - 44

## 32 45 - 64

## 33 65+

## 34 15 - 24

## 35 25 - 44

## 36 45 - 64

## 37 65+

## 38 15 - 24

## 39 25 - 44

## 40 45 - 64

## 41 65+

## 42 15 - 24

## 43 25 - 44

## 44 45 - 64

## 45 65+

## 46 15 - 24

## 47 25 - 44

## 48 45 - 64

## 49 65+

## 50 15 - 24

## 51 25 - 44

## 52 45 - 64

## 53 65+

## 54 15 - 24

## 55 25 - 44

## 56 45 - 64

## 57 65+

## 58 15 - 24

## 59 25 - 44

## 60 45 - 64

## 61 65+

## 62 25 - 44

## 63 45 - 64

## 64 15 - 24

## 65 25 - 44

## 66 45 - 64

## 67 65+

## 68 25 - 44

## 69 45 - 64

## 70 25 - 44

## 71 45 - 64

## 72 65+

NNE\_WSW <- data\_cells %>% arrange(row, col)

SSE ESE example

cells <- as\_cells(purpose$`SSE ESE`)

Headers

row\_headers <-

cells %>%

dplyr::filter(col >= 5, !is.na(chr)) %>%

select(row, col, header = chr) %>%

split(.$col)

row\_headers

## $`5`

## # A tibble: 20 x 3

## row col header

## <int> <int> <chr>

## 1 1 5 15 - 24

## 2 2 5 25 - 44

## 3 3 5 45 - 64

## 4 4 5 65+

## 5 5 5 15 - 24

## 6 6 5 25 - 44

## 7 7 5 45 - 64

## 8 8 5 65+

## 9 9 5 15 - 24

## 10 10 5 25 - 44

## 11 11 5 45 - 64

## 12 12 5 65+

## 13 13 5 15 - 24

## 14 14 5 25 - 44

## 15 15 5 45 - 64

## 16 16 5 65+

## 17 17 5 15 - 24

## 18 18 5 25 - 44

## 19 19 5 45 - 64

## 20 20 5 65+

##

## $`6`

## # A tibble: 5 x 3

## row col header

## <int> <int> <chr>

## 1 4 6 Bachelor's degree

## 2 8 6 Certificate

## 3 12 6 Diploma

## 4 16 6 No Qualification

## 5 20 6 Postgraduate qualification

col\_headers <-

cells %>%

dplyr::filter(row >= 21, !is.na(chr)) %>%

select(row, col, header = chr) %>%

split(.$row)

col\_headers

## $`21`

## # A tibble: 4 x 3

## row col header

## <int> <int> <chr>

## 1 21 1 0 - 6

## 2 21 2 7 - 10

## 3 21 3 0 - 6

## 4 21 4 7 - 10

##

## $`22`

## # A tibble: 2 x 3

## row col header

## <int> <int> <chr>

## 1 22 2 Female

## 2 22 4 Male

Data

data\_cells <-

cells %>%

dplyr::filter(row <= 20, col <= 4, !is.na(chr)) %>%

mutate(value = as.integer(chr)) %>%

select(row, col, value)

data\_cells

## # A tibble: 72 x 3

## row col value

## <int> <int> <int>

## 1 1 1 7000

## 2 2 1 12000

## 3 3 1 10000

## 4 5 1 29000

## 5 6 1 34000

## 6 7 1 30000

## 7 8 1 12000

## 8 10 1 10000

## 9 11 1 6000

## 10 12 1 5000

## # … with 62 more rows

Join headers to data

data\_cells <-

data\_cells %>%

enhead(col\_headers[[2]], "SSE") %>%

enhead(col\_headers[[1]], "S") %>%

enhead(row\_headers[[2]], "ESE") %>%

enhead(row\_headers[[1]], "E")

Result

data\_cells %>% as.data.frame

## row col value header.data header.header header.data.data

## 1 1 1 7000 Female 0 - 6 Bachelor's degree

## 2 2 1 12000 Female 0 - 6 Bachelor's degree

## 3 3 1 10000 Female 0 - 6 Bachelor's degree

## 4 1 2 27000 Female 7 - 10 Bachelor's degree

## 5 2 2 137000 Female 7 - 10 Bachelor's degree

## 6 3 2 64000 Female 7 - 10 Bachelor's degree

## 7 4 2 18000 Female 7 - 10 Bachelor's degree

## 8 2 3 9000 Male 0 - 6 Bachelor's degree

## 9 3 3 7000 Male 0 - 6 Bachelor's degree

## 10 4 3 7000 Male 0 - 6 Bachelor's degree

## 11 1 4 13000 Male 7 - 10 Bachelor's degree

## 12 2 4 81000 Male 7 - 10 Bachelor's degree

## 13 3 4 66000 Male 7 - 10 Bachelor's degree

## 14 4 4 17000 Male 7 - 10 Bachelor's degree

## 15 5 1 29000 Female 0 - 6 Certificate

## 16 6 1 34000 Female 0 - 6 Certificate

## 17 7 1 30000 Female 0 - 6 Certificate

## 18 8 1 12000 Female 0 - 6 Certificate

## 19 5 2 161000 Female 7 - 10 Certificate

## 20 6 2 179000 Female 7 - 10 Certificate

## 21 7 2 210000 Female 7 - 10 Certificate

## 22 8 2 77000 Female 7 - 10 Certificate

## 23 5 3 30000 Male 0 - 6 Certificate

## 24 6 3 31000 Male 0 - 6 Certificate

## 25 7 3 23000 Male 0 - 6 Certificate

## 26 8 3 8000 Male 0 - 6 Certificate

## 27 5 4 190000 Male 7 - 10 Certificate

## 28 6 4 219000 Male 7 - 10 Certificate

## 29 7 4 199000 Male 7 - 10 Certificate

## 30 8 4 107000 Male 7 - 10 Certificate

## 31 10 1 10000 Female 0 - 6 Diploma

## 32 11 1 6000 Female 0 - 6 Diploma

## 33 12 1 5000 Female 0 - 6 Diploma

## 34 9 2 14000 Female 7 - 10 Diploma

## 35 10 2 66000 Female 7 - 10 Diploma

## 36 11 2 68000 Female 7 - 10 Diploma

## 37 12 2 41000 Female 7 - 10 Diploma

## 38 9 3 9000 Male 0 - 6 Diploma

## 39 10 3 8000 Male 0 - 6 Diploma

## 40 11 3 5000 Male 0 - 6 Diploma

## 41 12 3 1000 Male 0 - 6 Diploma

## 42 9 4 11000 Male 7 - 10 Diploma

## 43 10 4 47000 Male 7 - 10 Diploma

## 44 11 4 58000 Male 7 - 10 Diploma

## 45 12 4 34000 Male 7 - 10 Diploma

## 46 13 1 10000 Female 0 - 6 No Qualification

## 47 14 1 11000 Female 0 - 6 No Qualification

## 48 15 1 19000 Female 0 - 6 No Qualification

## 49 16 1 16000 Female 0 - 6 No Qualification

## 50 13 2 43000 Female 7 - 10 No Qualification

## 51 14 2 36000 Female 7 - 10 No Qualification

## 52 15 2 91000 Female 7 - 10 No Qualification

## 53 16 2 118000 Female 7 - 10 No Qualification

## 54 13 3 12000 Male 0 - 6 No Qualification

## 55 14 3 21000 Male 0 - 6 No Qualification

## 56 15 3 17000 Male 0 - 6 No Qualification

## 57 16 3 9000 Male 0 - 6 No Qualification

## 58 13 4 37000 Male 7 - 10 No Qualification

## 59 14 4 50000 Male 7 - 10 No Qualification

## 60 15 4 75000 Male 7 - 10 No Qualification

## 61 16 4 66000 Male 7 - 10 No Qualification

## 62 18 1 5000 Female 0 - 6 Postgraduate qualification

## 63 19 1 6000 Female 0 - 6 Postgraduate qualification

## 64 17 2 6000 Female 7 - 10 Postgraduate qualification

## 65 18 2 86000 Female 7 - 10 Postgraduate qualification

## 66 19 2 55000 Female 7 - 10 Postgraduate qualification

## 67 20 2 13000 Female 7 - 10 Postgraduate qualification

## 68 18 3 7000 Male 0 - 6 Postgraduate qualification

## 69 19 3 6000 Male 0 - 6 Postgraduate qualification

## 70 18 4 60000 Male 7 - 10 Postgraduate qualification

## 71 19 4 68000 Male 7 - 10 Postgraduate qualification

## 72 20 4 18000 Male 7 - 10 Postgraduate qualification

## header.header.header

## 1 15 - 24

## 2 25 - 44

## 3 45 - 64

## 4 15 - 24

## 5 25 - 44

## 6 45 - 64

## 7 65+

## 8 25 - 44

## 9 45 - 64

## 10 65+

## 11 15 - 24

## 12 25 - 44

## 13 45 - 64

## 14 65+

## 15 15 - 24

## 16 25 - 44

## 17 45 - 64

## 18 65+

## 19 15 - 24

## 20 25 - 44

## 21 45 - 64

## 22 65+

## 23 15 - 24

## 24 25 - 44

## 25 45 - 64

## 26 65+

## 27 15 - 24

## 28 25 - 44

## 29 45 - 64

## 30 65+

## 31 25 - 44

## 32 45 - 64

## 33 65+

## 34 15 - 24

## 35 25 - 44

## 36 45 - 64

## 37 65+

## 38 15 - 24

## 39 25 - 44

## 40 45 - 64

## 41 65+

## 42 15 - 24

## 43 25 - 44

## 44 45 - 64

## 45 65+

## 46 15 - 24

## 47 25 - 44

## 48 45 - 64

## 49 65+

## 50 15 - 24

## 51 25 - 44

## 52 45 - 64

## 53 65+

## 54 15 - 24

## 55 25 - 44

## 56 45 - 64

## 57 65+

## 58 15 - 24

## 59 25 - 44

## 60 45 - 64

## 61 65+

## 62 25 - 44

## 63 45 - 64

## 64 15 - 24

## 65 25 - 44

## 66 45 - 64

## 67 65+

## 68 25 - 44

## 69 45 - 64

## 70 25 - 44

## 71 45 - 64

## 72 65+

SSE\_ESE <- data\_cells %>% arrange(row, col)

SSW ENE example

cells <- as\_cells(purpose$`SSW ENE`)

Headers (same as SSE ESE)

row\_headers <-

cells %>%

dplyr::filter(col >= 5, !is.na(chr)) %>%

select(row, col, header = chr) %>%

split(.$col)

row\_headers

## $`5`

## # A tibble: 20 x 3

## row col header

## <int> <int> <chr>

## 1 1 5 15 - 24

## 2 2 5 25 - 44

## 3 3 5 45 - 64

## 4 4 5 65+

## 5 5 5 15 - 24

## 6 6 5 25 - 44

## 7 7 5 45 - 64

## 8 8 5 65+

## 9 9 5 15 - 24

## 10 10 5 25 - 44

## 11 11 5 45 - 64

## 12 12 5 65+

## 13 13 5 15 - 24

## 14 14 5 25 - 44

## 15 15 5 45 - 64

## 16 16 5 65+

## 17 17 5 15 - 24

## 18 18 5 25 - 44

## 19 19 5 45 - 64

## 20 20 5 65+

##

## $`6`

## # A tibble: 5 x 3

## row col header

## <int> <int> <chr>

## 1 1 6 Bachelor's degree

## 2 5 6 Certificate

## 3 9 6 Diploma

## 4 13 6 No Qualification

## 5 17 6 Postgraduate qualification

col\_headers <-

cells %>%

dplyr::filter(row >= 21, !is.na(chr)) %>%

select(row, col, header = chr) %>%

split(.$row)

col\_headers

## $`21`

## # A tibble: 4 x 3

## row col header

## <int> <int> <chr>

## 1 21 1 0 - 6

## 2 21 2 7 - 10

## 3 21 3 0 - 6

## 4 21 4 7 - 10

##

## $`22`

## # A tibble: 2 x 3

## row col header

## <int> <int> <chr>

## 1 22 1 Female

## 2 22 3 Male

Data (same as SSE ESE)

data\_cells <-

cells %>%

dplyr::filter(row <= 20, col <= 4, !is.na(chr)) %>%

mutate(value = as.integer(chr)) %>%

select(row, col, value)

data\_cells

## # A tibble: 72 x 3

## row col value

## <int> <int> <int>

## 1 1 1 7000

## 2 2 1 12000

## 3 3 1 10000

## 4 5 1 29000

## 5 6 1 34000

## 6 7 1 30000

## 7 8 1 12000

## 8 10 1 10000

## 9 11 1 6000

## 10 12 1 5000

## # … with 62 more rows

Join headers to data

data\_cells <-

data\_cells %>%

enhead(col\_headers[[2]], "SSW") %>% # Different from SSE ESE

enhead(col\_headers[[1]], "S") %>% # Same as SSE ESE

enhead(row\_headers[[2]], "ENE") %>% # Different from SSE ESE

enhead(row\_headers[[1]], "E") # Same as SSE ESE

Result

data\_cells %>% as.data.frame

## row col value header.data header.header header.data.data

## 1 1 1 7000 Female 0 - 6 Bachelor's degree

## 2 2 1 12000 Female 0 - 6 Bachelor's degree

## 3 3 1 10000 Female 0 - 6 Bachelor's degree

## 4 1 2 27000 Female 7 - 10 Bachelor's degree

## 5 2 2 137000 Female 7 - 10 Bachelor's degree

## 6 3 2 64000 Female 7 - 10 Bachelor's degree

## 7 4 2 18000 Female 7 - 10 Bachelor's degree

## 8 2 3 9000 Male 0 - 6 Bachelor's degree

## 9 3 3 7000 Male 0 - 6 Bachelor's degree

## 10 4 3 7000 Male 0 - 6 Bachelor's degree

## 11 1 4 13000 Male 7 - 10 Bachelor's degree

## 12 2 4 81000 Male 7 - 10 Bachelor's degree

## 13 3 4 66000 Male 7 - 10 Bachelor's degree

## 14 4 4 17000 Male 7 - 10 Bachelor's degree

## 15 5 1 29000 Female 0 - 6 Certificate

## 16 6 1 34000 Female 0 - 6 Certificate

## 17 7 1 30000 Female 0 - 6 Certificate

## 18 8 1 12000 Female 0 - 6 Certificate

## 19 5 2 161000 Female 7 - 10 Certificate

## 20 6 2 179000 Female 7 - 10 Certificate

## 21 7 2 210000 Female 7 - 10 Certificate

## 22 8 2 77000 Female 7 - 10 Certificate

## 23 5 3 30000 Male 0 - 6 Certificate

## 24 6 3 31000 Male 0 - 6 Certificate

## 25 7 3 23000 Male 0 - 6 Certificate

## 26 8 3 8000 Male 0 - 6 Certificate

## 27 5 4 190000 Male 7 - 10 Certificate

## 28 6 4 219000 Male 7 - 10 Certificate

## 29 7 4 199000 Male 7 - 10 Certificate

## 30 8 4 107000 Male 7 - 10 Certificate

## 31 10 1 10000 Female 0 - 6 Diploma

## 32 11 1 6000 Female 0 - 6 Diploma

## 33 12 1 5000 Female 0 - 6 Diploma

## 34 9 2 14000 Female 7 - 10 Diploma

## 35 10 2 66000 Female 7 - 10 Diploma

## 36 11 2 68000 Female 7 - 10 Diploma

## 37 12 2 41000 Female 7 - 10 Diploma

## 38 9 3 9000 Male 0 - 6 Diploma

## 39 10 3 8000 Male 0 - 6 Diploma

## 40 11 3 5000 Male 0 - 6 Diploma

## 41 12 3 1000 Male 0 - 6 Diploma

## 42 9 4 11000 Male 7 - 10 Diploma

## 43 10 4 47000 Male 7 - 10 Diploma

## 44 11 4 58000 Male 7 - 10 Diploma

## 45 12 4 34000 Male 7 - 10 Diploma

## 46 13 1 10000 Female 0 - 6 No Qualification

## 47 14 1 11000 Female 0 - 6 No Qualification

## 48 15 1 19000 Female 0 - 6 No Qualification

## 49 16 1 16000 Female 0 - 6 No Qualification

## 50 13 2 43000 Female 7 - 10 No Qualification

## 51 14 2 36000 Female 7 - 10 No Qualification

## 52 15 2 91000 Female 7 - 10 No Qualification

## 53 16 2 118000 Female 7 - 10 No Qualification

## 54 13 3 12000 Male 0 - 6 No Qualification

## 55 14 3 21000 Male 0 - 6 No Qualification

## 56 15 3 17000 Male 0 - 6 No Qualification

## 57 16 3 9000 Male 0 - 6 No Qualification

## 58 13 4 37000 Male 7 - 10 No Qualification

## 59 14 4 50000 Male 7 - 10 No Qualification

## 60 15 4 75000 Male 7 - 10 No Qualification

## 61 16 4 66000 Male 7 - 10 No Qualification

## 62 18 1 5000 Female 0 - 6 Postgraduate qualification

## 63 19 1 6000 Female 0 - 6 Postgraduate qualification

## 64 17 2 6000 Female 7 - 10 Postgraduate qualification

## 65 18 2 86000 Female 7 - 10 Postgraduate qualification

## 66 19 2 55000 Female 7 - 10 Postgraduate qualification

## 67 20 2 13000 Female 7 - 10 Postgraduate qualification

## 68 18 3 7000 Male 0 - 6 Postgraduate qualification

## 69 19 3 6000 Male 0 - 6 Postgraduate qualification

## 70 18 4 60000 Male 7 - 10 Postgraduate qualification

## 71 19 4 68000 Male 7 - 10 Postgraduate qualification

## 72 20 4 18000 Male 7 - 10 Postgraduate qualification

## header.header.header

## 1 15 - 24

## 2 25 - 44

## 3 45 - 64

## 4 15 - 24

## 5 25 - 44

## 6 45 - 64

## 7 65+

## 8 25 - 44

## 9 45 - 64

## 10 65+

## 11 15 - 24

## 12 25 - 44

## 13 45 - 64

## 14 65+

## 15 15 - 24

## 16 25 - 44

## 17 45 - 64

## 18 65+

## 19 15 - 24

## 20 25 - 44

## 21 45 - 64

## 22 65+

## 23 15 - 24

## 24 25 - 44

## 25 45 - 64

## 26 65+

## 27 15 - 24

## 28 25 - 44

## 29 45 - 64

## 30 65+

## 31 25 - 44

## 32 45 - 64

## 33 65+

## 34 15 - 24

## 35 25 - 44

## 36 45 - 64

## 37 65+

## 38 15 - 24

## 39 25 - 44

## 40 45 - 64

## 41 65+

## 42 15 - 24

## 43 25 - 44

## 44 45 - 64

## 45 65+

## 46 15 - 24

## 47 25 - 44

## 48 45 - 64

## 49 65+

## 50 15 - 24

## 51 25 - 44

## 52 45 - 64

## 53 65+

## 54 15 - 24

## 55 25 - 44

## 56 45 - 64

## 57 65+

## 58 15 - 24

## 59 25 - 44

## 60 45 - 64

## 61 65+

## 62 25 - 44

## 63 45 - 64

## 64 15 - 24

## 65 25 - 44

## 66 45 - 64

## 67 65+

## 68 25 - 44

## 69 45 - 64

## 70 25 - 44

## 71 45 - 64

## 72 65+

SSW\_ENE <- data\_cells %>% arrange(row, col)

Check that all sheets are parsed correctly

identical(NNW\_WNW, NNE\_WSW)

## [1] TRUE

identical(SSW\_ENE, SSE\_ESE)

## [1] TRUE

identical(NNW\_WNW[, -1:-2], SSW\_ENE[, -1:-2])

## [1] TRUE

ABOVE LEFT example

cells <- as\_cells(purpose$`ABOVE LEFT`)

Headers

row\_headers <-

cells %>%

dplyr::filter(col <= 2, !is.na(chr)) %>%

select(row, col, header = chr) %>%

split(.$col)

row\_headers

## $`1`

## # A tibble: 2 x 3

## row col header

## <int> <int> <chr>

## 1 5 1 Female

## 2 10 1 Male

##

## $`2`

## # A tibble: 10 x 3

## row col header

## <int> <int> <chr>

## 1 3 2 Bachelor's degree

## 2 4 2 Certificate

## 3 5 2 Diploma

## 4 6 2 No Qualification

## 5 7 2 Postgraduate qualification

## 6 8 2 Bachelor's degree

## 7 9 2 Certificate

## 8 10 2 Diploma

## 9 11 2 No Qualification

## 10 12 2 Postgraduate qualification

col\_headers <-

cells %>%

dplyr::filter(row <= 2, !is.na(chr)) %>%

select(row, col, header = chr) %>%

split(.$row)

col\_headers

## $`1`

## # A tibble: 2 x 3

## row col header

## <int> <int> <chr>

## 1 1 4 0 - 6

## 2 1 7 7 - 10

##

## $`2`

## # A tibble: 6 x 3

## row col header

## <int> <int> <chr>

## 1 2 3 15 - 24

## 2 2 4 25 - 44

## 3 2 5 45 - 64

## 4 2 6 15 - 24

## 5 2 7 25 - 44

## 6 2 8 45 - 64

Data

data\_cells <-

cells %>%

dplyr::filter(row >= 3, col >= 3, !is.na(chr)) %>%

mutate(value = as.integer(chr)) %>%

select(row, col, value)

data\_cells

## # A tibble: 55 x 3

## row col value

## <int> <int> <int>

## 1 3 3 7000

## 2 4 3 29000

## 3 6 3 10000

## 4 9 3 30000

## 5 10 3 9000

## 6 11 3 12000

## 7 3 4 12000

## 8 4 4 34000

## 9 5 4 10000

## 10 6 4 11000

## # … with 45 more rows

Join headers to data

data\_cells <-

data\_cells %>%

enhead(col\_headers[[1]], "ABOVE") %>% # Different from SSE ESE

enhead(col\_headers[[2]], "N") %>% # Same as SSE ESE

enhead(row\_headers[[1]], "LEFT") %>% # Different from SSE ESE

enhead(row\_headers[[2]], "W") # Same as SSE ESE

Result

data\_cells %>% as.data.frame

## row col value header.data header.header header.data.data

## 1 3 3 7000 0 - 6 15 - 24 Female

## 2 4 3 29000 0 - 6 15 - 24 Female

## 3 6 3 10000 0 - 6 15 - 24 Female

## 4 3 4 12000 0 - 6 25 - 44 Female

## 5 4 4 34000 0 - 6 25 - 44 Female

## 6 5 4 10000 0 - 6 25 - 44 Female

## 7 6 4 11000 0 - 6 25 - 44 Female

## 8 7 4 5000 0 - 6 25 - 44 Female

## 9 3 5 10000 0 - 6 45 - 64 Female

## 10 4 5 30000 0 - 6 45 - 64 Female

## 11 5 5 6000 0 - 6 45 - 64 Female

## 12 6 5 19000 0 - 6 45 - 64 Female

## 13 7 5 6000 0 - 6 45 - 64 Female

## 14 3 6 27000 7 - 10 15 - 24 Female

## 15 4 6 161000 7 - 10 15 - 24 Female

## 16 5 6 14000 7 - 10 15 - 24 Female

## 17 6 6 43000 7 - 10 15 - 24 Female

## 18 7 6 6000 7 - 10 15 - 24 Female

## 19 3 7 137000 7 - 10 25 - 44 Female

## 20 4 7 179000 7 - 10 25 - 44 Female

## 21 5 7 66000 7 - 10 25 - 44 Female

## 22 6 7 36000 7 - 10 25 - 44 Female

## 23 7 7 86000 7 - 10 25 - 44 Female

## 24 3 8 64000 7 - 10 45 - 64 Female

## 25 4 8 210000 7 - 10 45 - 64 Female

## 26 5 8 68000 7 - 10 45 - 64 Female

## 27 6 8 91000 7 - 10 45 - 64 Female

## 28 7 8 55000 7 - 10 45 - 64 Female

## 29 9 3 30000 0 - 6 15 - 24 Male

## 30 10 3 9000 0 - 6 15 - 24 Male

## 31 11 3 12000 0 - 6 15 - 24 Male

## 32 8 4 9000 0 - 6 25 - 44 Male

## 33 9 4 31000 0 - 6 25 - 44 Male

## 34 10 4 8000 0 - 6 25 - 44 Male

## 35 11 4 21000 0 - 6 25 - 44 Male

## 36 12 4 7000 0 - 6 25 - 44 Male

## 37 8 5 7000 0 - 6 45 - 64 Male

## 38 9 5 23000 0 - 6 45 - 64 Male

## 39 10 5 5000 0 - 6 45 - 64 Male

## 40 11 5 17000 0 - 6 45 - 64 Male

## 41 12 5 6000 0 - 6 45 - 64 Male

## 42 8 6 13000 7 - 10 15 - 24 Male

## 43 9 6 190000 7 - 10 15 - 24 Male

## 44 10 6 11000 7 - 10 15 - 24 Male

## 45 11 6 37000 7 - 10 15 - 24 Male

## 46 8 7 81000 7 - 10 25 - 44 Male

## 47 9 7 219000 7 - 10 25 - 44 Male

## 48 10 7 47000 7 - 10 25 - 44 Male

## 49 11 7 50000 7 - 10 25 - 44 Male

## 50 12 7 60000 7 - 10 25 - 44 Male

## 51 8 8 66000 7 - 10 45 - 64 Male

## 52 9 8 199000 7 - 10 45 - 64 Male

## 53 10 8 58000 7 - 10 45 - 64 Male

## 54 11 8 75000 7 - 10 45 - 64 Male

## 55 12 8 68000 7 - 10 45 - 64 Male

## header.header.header

## 1 Bachelor's degree

## 2 Certificate

## 3 No Qualification

## 4 Bachelor's degree

## 5 Certificate

## 6 Diploma

## 7 No Qualification

## 8 Postgraduate qualification

## 9 Bachelor's degree

## 10 Certificate

## 11 Diploma

## 12 No Qualification

## 13 Postgraduate qualification

## 14 Bachelor's degree

## 15 Certificate

## 16 Diploma

## 17 No Qualification

## 18 Postgraduate qualification

## 19 Bachelor's degree

## 20 Certificate

## 21 Diploma

## 22 No Qualification

## 23 Postgraduate qualification

## 24 Bachelor's degree

## 25 Certificate

## 26 Diploma

## 27 No Qualification

## 28 Postgraduate qualification

## 29 Certificate

## 30 Diploma

## 31 No Qualification

## 32 Bachelor's degree

## 33 Certificate

## 34 Diploma

## 35 No Qualification

## 36 Postgraduate qualification

## 37 Bachelor's degree

## 38 Certificate

## 39 Diploma

## 40 No Qualification

## 41 Postgraduate qualification

## 42 Bachelor's degree

## 43 Certificate

## 44 Diploma

## 45 No Qualification

## 46 Bachelor's degree

## 47 Certificate

## 48 Diploma

## 49 No Qualification

## 50 Postgraduate qualification

## 51 Bachelor's degree

## 52 Certificate

## 53 Diploma

## 54 No Qualification

## 55 Postgraduate qualification

ABOVE\_LEFT <- data\_cells %>% arrange(row, col)

BELOW RIGHT example

cells <- as\_cells(purpose$`BELOW RIGHT`)

Headers

row\_headers <-

cells %>%

dplyr::filter(col >= 7, !is.na(chr)) %>%

select(row, col, header = chr) %>%

split(.$col)

row\_headers

## $`7`

## # A tibble: 10 x 3

## row col header

## <int> <int> <chr>

## 1 1 7 Bachelor's degree

## 2 2 7 Certificate

## 3 3 7 Diploma

## 4 4 7 No Qualification

## 5 5 7 Postgraduate qualification

## 6 6 7 Bachelor's degree

## 7 7 7 Certificate

## 8 8 7 Diploma

## 9 9 7 No Qualification

## 10 10 7 Postgraduate qualification

##

## $`8`

## # A tibble: 2 x 3

## row col header

## <int> <int> <chr>

## 1 3 8 Female

## 2 8 8 Male

col\_headers <-

cells %>%

dplyr::filter(row >= 11, !is.na(chr)) %>%

select(row, col, header = chr) %>%

split(.$row)

col\_headers

## $`11`

## # A tibble: 6 x 3

## row col header

## <int> <int> <chr>

## 1 11 1 15 - 24

## 2 11 2 25 - 44

## 3 11 3 45 - 64

## 4 11 4 15 - 24

## 5 11 5 25 - 44

## 6 11 6 45 - 64

##

## $`12`

## # A tibble: 2 x 3

## row col header

## <int> <int> <chr>

## 1 12 2 0 - 6

## 2 12 5 7 - 10

Data

data\_cells <-

cells %>%

dplyr::filter(row <= 10, col <= 6, !is.na(chr)) %>%

mutate(value = as.integer(chr)) %>%

select(row, col, value)

data\_cells

## # A tibble: 55 x 3

## row col value

## <int> <int> <int>

## 1 1 1 7000

## 2 2 1 29000

## 3 4 1 10000

## 4 7 1 30000

## 5 8 1 9000

## 6 9 1 12000

## 7 1 2 12000

## 8 2 2 34000

## 9 3 2 10000

## 10 4 2 11000

## # … with 45 more rows

Join headers to data

data\_cells <-

data\_cells %>%

enhead(col\_headers[[2]], "BELOW") %>%

enhead(col\_headers[[1]], "S") %>%

enhead(row\_headers[[2]], "RIGHT") %>%

enhead(row\_headers[[1]], "E")

Result

data\_cells %>% as.data.frame

## row col value header.data header.header header.data.data

## 1 1 1 7000 0 - 6 15 - 24 Female

## 2 2 1 29000 0 - 6 15 - 24 Female

## 3 4 1 10000 0 - 6 15 - 24 Female

## 4 1 2 12000 0 - 6 25 - 44 Female

## 5 2 2 34000 0 - 6 25 - 44 Female

## 6 3 2 10000 0 - 6 25 - 44 Female

## 7 4 2 11000 0 - 6 25 - 44 Female

## 8 5 2 5000 0 - 6 25 - 44 Female

## 9 1 3 10000 0 - 6 45 - 64 Female

## 10 2 3 30000 0 - 6 45 - 64 Female

## 11 3 3 6000 0 - 6 45 - 64 Female

## 12 4 3 19000 0 - 6 45 - 64 Female

## 13 5 3 6000 0 - 6 45 - 64 Female

## 14 1 4 27000 7 - 10 15 - 24 Female

## 15 2 4 161000 7 - 10 15 - 24 Female

## 16 3 4 14000 7 - 10 15 - 24 Female

## 17 4 4 43000 7 - 10 15 - 24 Female

## 18 5 4 6000 7 - 10 15 - 24 Female

## 19 1 5 137000 7 - 10 25 - 44 Female

## 20 2 5 179000 7 - 10 25 - 44 Female

## 21 3 5 66000 7 - 10 25 - 44 Female

## 22 4 5 36000 7 - 10 25 - 44 Female

## 23 5 5 86000 7 - 10 25 - 44 Female

## 24 1 6 64000 7 - 10 45 - 64 Female

## 25 2 6 210000 7 - 10 45 - 64 Female

## 26 3 6 68000 7 - 10 45 - 64 Female

## 27 4 6 91000 7 - 10 45 - 64 Female

## 28 5 6 55000 7 - 10 45 - 64 Female

## 29 7 1 30000 0 - 6 15 - 24 Male

## 30 8 1 9000 0 - 6 15 - 24 Male

## 31 9 1 12000 0 - 6 15 - 24 Male

## 32 6 2 9000 0 - 6 25 - 44 Male

## 33 7 2 31000 0 - 6 25 - 44 Male

## 34 8 2 8000 0 - 6 25 - 44 Male

## 35 9 2 21000 0 - 6 25 - 44 Male

## 36 10 2 7000 0 - 6 25 - 44 Male

## 37 6 3 7000 0 - 6 45 - 64 Male

## 38 7 3 23000 0 - 6 45 - 64 Male

## 39 8 3 5000 0 - 6 45 - 64 Male

## 40 9 3 17000 0 - 6 45 - 64 Male

## 41 10 3 6000 0 - 6 45 - 64 Male

## 42 6 4 13000 7 - 10 15 - 24 Male

## 43 7 4 190000 7 - 10 15 - 24 Male

## 44 8 4 11000 7 - 10 15 - 24 Male

## 45 9 4 37000 7 - 10 15 - 24 Male

## 46 6 5 81000 7 - 10 25 - 44 Male

## 47 7 5 219000 7 - 10 25 - 44 Male

## 48 8 5 47000 7 - 10 25 - 44 Male

## 49 9 5 50000 7 - 10 25 - 44 Male

## 50 10 5 60000 7 - 10 25 - 44 Male

## 51 6 6 66000 7 - 10 45 - 64 Male

## 52 7 6 199000 7 - 10 45 - 64 Male

## 53 8 6 58000 7 - 10 45 - 64 Male

## 54 9 6 75000 7 - 10 45 - 64 Male

## 55 10 6 68000 7 - 10 45 - 64 Male

## header.header.header

## 1 Bachelor's degree

## 2 Certificate

## 3 No Qualification

## 4 Bachelor's degree

## 5 Certificate

## 6 Diploma

## 7 No Qualification

## 8 Postgraduate qualification

## 9 Bachelor's degree

## 10 Certificate

## 11 Diploma

## 12 No Qualification

## 13 Postgraduate qualification

## 14 Bachelor's degree

## 15 Certificate

## 16 Diploma

## 17 No Qualification

## 18 Postgraduate qualification

## 19 Bachelor's degree

## 20 Certificate

## 21 Diploma

## 22 No Qualification

## 23 Postgraduate qualification

## 24 Bachelor's degree

## 25 Certificate

## 26 Diploma

## 27 No Qualification

## 28 Postgraduate qualification

## 29 Certificate

## 30 Diploma

## 31 No Qualification

## 32 Bachelor's degree

## 33 Certificate

## 34 Diploma

## 35 No Qualification

## 36 Postgraduate qualification

## 37 Bachelor's degree

## 38 Certificate

## 39 Diploma

## 40 No Qualification

## 41 Postgraduate qualification

## 42 Bachelor's degree

## 43 Certificate

## 44 Diploma

## 45 No Qualification

## 46 Bachelor's degree

## 47 Certificate

## 48 Diploma

## 49 No Qualification

## 50 Postgraduate qualification

## 51 Bachelor's degree

## 52 Certificate

## 53 Diploma

## 54 No Qualification

## 55 Postgraduate qualification

BELOW\_RIGHT <- data\_cells %>% arrange(row, col)

Check that all sheets are parsed correctly

identical(ABOVE\_LEFT[, -1:-2], BELOW\_RIGHT[, -1:-2])

## [1] TRUE

ABOVE LEFT border example

spreadsheet <- system.file("extdata/purpose.xlsx", package = "unpivotr")

cells <- tidy\_xlsx(spreadsheet, "ABOVE LEFT border")$data[[1]]

## Warning: 'tidy\_xlsx()' is deprecated.

## Use 'xlsx\_cells()' or 'xlsx\_formats()' instead.

Headers

# Same as ABOVE LEFT without borders

row\_headers <-

cells %>%

dplyr::filter(col <= 3, !is\_blank) %>%

select(row, col, header = character) %>%

split(.$col)

row\_headers

## $`2`

## # A tibble: 2 x 3

## row col header

## <int> <int> <chr>

## 1 5 2 Female

## 2 10 2 Male

##

## $`3`

## # A tibble: 10 x 3

## row col header

## <int> <int> <chr>

## 1 4 3 Bachelor's degree

## 2 5 3 Certificate

## 3 6 3 Diploma

## 4 7 3 No Qualification

## 5 8 3 Postgraduate qualification

## 6 9 3 Bachelor's degree

## 7 10 3 Certificate

## 8 11 3 Diploma

## 9 12 3 No Qualification

## 10 13 3 Postgraduate qualification

col\_headers <-

cells %>%

dplyr::filter(row <= 3, !is\_blank) %>%

select(row, col, header = character) %>%

split(.$row)

col\_headers

## $`2`

## # A tibble: 2 x 3

## row col header

## <int> <int> <chr>

## 1 2 6 0 - 6

## 2 2 8 7 - 10

##

## $`3`

## # A tibble: 6 x 3

## row col header

## <int> <int> <chr>

## 1 3 4 15 - 24

## 2 3 5 25 - 44

## 3 3 6 45 - 64

## 4 3 7 15 - 24

## 5 3 8 25 - 44

## 6 3 9 45 - 64

Data

# Same as ABOVE LEFT without borders

data\_cells <-

cells %>%

dplyr::filter(row >= 4, col >= 4, !is\_blank) %>%

mutate(content = ifelse(is.na(character), numeric, NA)) %>%

mutate(value = as.integer(content)) %>%

select(row, col, value)

data\_cells

## # A tibble: 55 x 3

## row col value

## <int> <int> <int>

## 1 4 4 7000

## 2 4 5 12000

## 3 4 6 10000

## 4 4 7 27000

## 5 4 8 137000

## 6 4 9 64000

## 7 5 4 29000

## 8 5 5 34000

## 9 5 6 30000

## 10 5 7 161000

## # … with 45 more rows

Borders

# Different from ABOVE LEFT without borders

# Find cells with borders on the bottom, and ones with borders on the left

formatting <- tidy\_xlsx(spreadsheet)$formats

## Warning: 'tidy\_xlsx()' is deprecated.

## Use 'xlsx\_cells()' or 'xlsx\_formats()' instead.

left\_borders <- which(!is.na(formatting$local$border$left$style))

top\_borders <- which(!is.na(formatting$local$border$top$style))

left\_border\_cells <-

cells %>%

dplyr::filter(row == 2, local\_format\_id %in% left\_borders) %>%

select(row, col)

top\_border\_cells <-

cells %>%

dplyr::filter(col == 2, local\_format\_id %in% top\_borders) %>%

select(row, col)

Join headers to data

data\_cells <-

data\_cells %>%

enhead(col\_headers[[1]], "ABOVE", left\_border\_cells) %>% # Different from ABOVE LEFT

enhead(col\_headers[[2]], "N") %>% # Same as ABOVE LEFT

enhead(row\_headers[[1]], "LEFT", top\_border\_cells) %>% # Different from ABOVE LEFT

enhead(row\_headers[[2]], "W") # Same as ABOVE LEFT

Result

data\_cells %>% as.data.frame

## row col value header.data header.header header.data.data

## 1 4 4 7000 0 - 6 15 - 24 Female

## 2 4 5 12000 0 - 6 25 - 44 Female

## 3 4 6 10000 0 - 6 45 - 64 Female

## 4 4 7 27000 0 - 6 15 - 24 Female

## 5 5 4 29000 0 - 6 15 - 24 Female

## 6 5 5 34000 0 - 6 25 - 44 Female

## 7 5 6 30000 0 - 6 45 - 64 Female

## 8 5 7 161000 0 - 6 15 - 24 Female

## 9 6 5 10000 0 - 6 25 - 44 Female

## 10 6 6 6000 0 - 6 45 - 64 Female

## 11 6 7 14000 0 - 6 15 - 24 Female

## 12 7 4 10000 0 - 6 15 - 24 Female

## 13 7 5 11000 0 - 6 25 - 44 Female

## 14 7 6 19000 0 - 6 45 - 64 Female

## 15 7 7 43000 0 - 6 15 - 24 Female

## 16 4 8 137000 7 - 10 25 - 44 Female

## 17 4 9 64000 7 - 10 45 - 64 Female

## 18 5 8 179000 7 - 10 25 - 44 Female

## 19 5 9 210000 7 - 10 45 - 64 Female

## 20 6 8 66000 7 - 10 25 - 44 Female

## 21 6 9 68000 7 - 10 45 - 64 Female

## 22 7 8 36000 7 - 10 25 - 44 Female

## 23 7 9 91000 7 - 10 45 - 64 Female

## 24 8 5 5000 0 - 6 25 - 44 Male

## 25 8 6 6000 0 - 6 45 - 64 Male

## 26 8 7 6000 0 - 6 15 - 24 Male

## 27 9 5 9000 0 - 6 25 - 44 Male

## 28 9 6 7000 0 - 6 45 - 64 Male

## 29 9 7 13000 0 - 6 15 - 24 Male

## 30 10 4 30000 0 - 6 15 - 24 Male

## 31 10 5 31000 0 - 6 25 - 44 Male

## 32 10 6 23000 0 - 6 45 - 64 Male

## 33 10 7 190000 0 - 6 15 - 24 Male

## 34 11 4 9000 0 - 6 15 - 24 Male

## 35 11 5 8000 0 - 6 25 - 44 Male

## 36 11 6 5000 0 - 6 45 - 64 Male

## 37 11 7 11000 0 - 6 15 - 24 Male

## 38 12 4 12000 0 - 6 15 - 24 Male

## 39 12 5 21000 0 - 6 25 - 44 Male

## 40 12 6 17000 0 - 6 45 - 64 Male

## 41 12 7 37000 0 - 6 15 - 24 Male

## 42 13 5 7000 0 - 6 25 - 44 Male

## 43 13 6 6000 0 - 6 45 - 64 Male

## 44 8 8 86000 7 - 10 25 - 44 Male

## 45 8 9 55000 7 - 10 45 - 64 Male

## 46 9 8 81000 7 - 10 25 - 44 Male

## 47 9 9 66000 7 - 10 45 - 64 Male

## 48 10 8 219000 7 - 10 25 - 44 Male

## 49 10 9 199000 7 - 10 45 - 64 Male

## 50 11 8 47000 7 - 10 25 - 44 Male

## 51 11 9 58000 7 - 10 45 - 64 Male

## 52 12 8 50000 7 - 10 25 - 44 Male

## 53 12 9 75000 7 - 10 45 - 64 Male

## 54 13 8 60000 7 - 10 25 - 44 Male

## 55 13 9 68000 7 - 10 45 - 64 Male

## header.header.header

## 1 Bachelor's degree

## 2 Bachelor's degree

## 3 Bachelor's degree

## 4 Bachelor's degree

## 5 Certificate

## 6 Certificate

## 7 Certificate

## 8 Certificate

## 9 Diploma

## 10 Diploma

## 11 Diploma

## 12 No Qualification

## 13 No Qualification

## 14 No Qualification

## 15 No Qualification

## 16 Bachelor's degree

## 17 Bachelor's degree

## 18 Certificate

## 19 Certificate

## 20 Diploma

## 21 Diploma

## 22 No Qualification

## 23 No Qualification

## 24 Postgraduate qualification

## 25 Postgraduate qualification

## 26 Postgraduate qualification

## 27 Bachelor's degree

## 28 Bachelor's degree

## 29 Bachelor's degree

## 30 Certificate

## 31 Certificate

## 32 Certificate

## 33 Certificate

## 34 Diploma

## 35 Diploma

## 36 Diploma

## 37 Diploma

## 38 No Qualification

## 39 No Qualification

## 40 No Qualification

## 41 No Qualification

## 42 Postgraduate qualification

## 43 Postgraduate qualification

## 44 Postgraduate qualification

## 45 Postgraduate qualification

## 46 Bachelor's degree

## 47 Bachelor's degree

## 48 Certificate

## 49 Certificate

## 50 Diploma

## 51 Diploma

## 52 No Qualification

## 53 No Qualification

## 54 Postgraduate qualification

## 55 Postgraduate qualification

ABOVE\_LEFT\_borders <- data\_cells %>% arrange(row, col)

BELOW RIGHT border example

cells <- tidy\_xlsx(spreadsheet, "BELOW RIGHT border")$data[[1]]

## Warning: 'tidy\_xlsx()' is deprecated.

## Use 'xlsx\_cells()' or 'xlsx\_formats()' instead.

Headers

# Same as BELOW RIGHT without borders

row\_headers <-

cells %>%

dplyr::filter(col >= 10, !is\_blank) %>%

select(row, col, header = character) %>%

split(.$col)

row\_headers

## $`10`

## # A tibble: 10 x 3

## row col header

## <int> <int> <chr>

## 1 4 10 Bachelor's degree

## 2 5 10 Certificate

## 3 6 10 Diploma

## 4 7 10 No Qualification

## 5 8 10 Postgraduate qualification

## 6 9 10 Bachelor's degree

## 7 10 10 Certificate

## 8 11 10 Diploma

## 9 12 10 No Qualification

## 10 13 10 Postgraduate qualification

##

## $`11`

## # A tibble: 2 x 3

## row col header

## <int> <int> <chr>

## 1 7 11 Female

## 2 10 11 Male

col\_headers <-

cells %>%

dplyr::filter(row >= 14, !is\_blank) %>%

select(row, col, header = character) %>%

split(.$row)

col\_headers

## $`14`

## # A tibble: 6 x 3

## row col header

## <int> <int> <chr>

## 1 14 4 15 - 24

## 2 14 5 25 - 44

## 3 14 6 45 - 64

## 4 14 7 15 - 24

## 5 14 8 25 - 44

## 6 14 9 45 - 64

##

## $`15`

## # A tibble: 2 x 3

## row col header

## <int> <int> <chr>

## 1 15 6 0 - 6

## 2 15 7 7 - 10

Data

# Same as BELOW RIGHT without borders

data\_cells <-

cells %>%

dplyr::filter(row <= 13, col <= 9, !is\_blank) %>%

mutate(content = ifelse(is.na(character), numeric, NA)) %>%

mutate(value = as.integer(content)) %>%

select(row, col, value)

data\_cells

## # A tibble: 55 x 3

## row col value

## <int> <int> <int>

## 1 4 4 7000

## 2 4 5 12000

## 3 4 6 10000

## 4 4 7 27000

## 5 4 8 137000

## 6 4 9 64000

## 7 5 4 29000

## 8 5 5 34000

## 9 5 6 30000

## 10 5 7 161000

## # … with 45 more rows

Borders

# Different from BELOW RIGHT without borders

# Find cells with borders on the bottom, and ones with borders on the left

formatting <- tidy\_xlsx(spreadsheet)$formats

## Warning: 'tidy\_xlsx()' is deprecated.

## Use 'xlsx\_cells()' or 'xlsx\_formats()' instead.

left\_borders <- which(!is.na(formatting$local$border$left$style))

top\_borders <- which(!is.na(formatting$local$border$top$style))

left\_border\_cells <-

cells %>%

dplyr::filter(row == 15, local\_format\_id %in% left\_borders) %>%

select(row, col)

top\_border\_cells <-

cells %>%

dplyr::filter(col == 11, local\_format\_id %in% top\_borders) %>%

select(row, col)

Join headers to data

data\_cells <-

data\_cells %>%

enhead(col\_headers[[2]], "BELOW", left\_border\_cells) %>% # Different from BELOW RIGHT

enhead(col\_headers[[1]], "S") %>% # Same as BELOW RIGHT

enhead(row\_headers[[2]], "RIGHT", top\_border\_cells) %>% # Different from BELOW RIGHT

enhead(row\_headers[[1]], "E") # Same as BELOW RIGHT

Result

data\_cells %>% as.data.frame

## row col value header.data header.header header.data.data

## 1 4 4 7000 0 - 6 15 - 24 Female

## 2 4 5 12000 0 - 6 25 - 44 Female

## 3 4 6 10000 0 - 6 45 - 64 Female

## 4 5 4 29000 0 - 6 15 - 24 Female

## 5 5 5 34000 0 - 6 25 - 44 Female

## 6 5 6 30000 0 - 6 45 - 64 Female

## 7 6 5 10000 0 - 6 25 - 44 Female

## 8 6 6 6000 0 - 6 45 - 64 Female

## 9 7 4 10000 0 - 6 15 - 24 Female

## 10 7 5 11000 0 - 6 25 - 44 Female

## 11 7 6 19000 0 - 6 45 - 64 Female

## 12 8 5 5000 0 - 6 25 - 44 Female

## 13 8 6 6000 0 - 6 45 - 64 Female

## 14 4 7 27000 7 - 10 15 - 24 Female

## 15 4 8 137000 7 - 10 25 - 44 Female

## 16 4 9 64000 7 - 10 45 - 64 Female

## 17 5 7 161000 7 - 10 15 - 24 Female

## 18 5 8 179000 7 - 10 25 - 44 Female

## 19 5 9 210000 7 - 10 45 - 64 Female

## 20 6 7 14000 7 - 10 15 - 24 Female

## 21 6 8 66000 7 - 10 25 - 44 Female

## 22 6 9 68000 7 - 10 45 - 64 Female

## 23 7 7 43000 7 - 10 15 - 24 Female

## 24 7 8 36000 7 - 10 25 - 44 Female

## 25 7 9 91000 7 - 10 45 - 64 Female

## 26 8 7 6000 7 - 10 15 - 24 Female

## 27 8 8 86000 7 - 10 25 - 44 Female

## 28 8 9 55000 7 - 10 45 - 64 Female

## 29 9 5 9000 0 - 6 25 - 44 Male

## 30 9 6 7000 0 - 6 45 - 64 Male

## 31 10 4 30000 0 - 6 15 - 24 Male

## 32 10 5 31000 0 - 6 25 - 44 Male

## 33 10 6 23000 0 - 6 45 - 64 Male

## 34 11 4 9000 0 - 6 15 - 24 Male

## 35 11 5 8000 0 - 6 25 - 44 Male

## 36 11 6 5000 0 - 6 45 - 64 Male

## 37 12 4 12000 0 - 6 15 - 24 Male

## 38 12 5 21000 0 - 6 25 - 44 Male

## 39 12 6 17000 0 - 6 45 - 64 Male

## 40 13 5 7000 0 - 6 25 - 44 Male

## 41 13 6 6000 0 - 6 45 - 64 Male

## 42 9 7 13000 7 - 10 15 - 24 Male

## 43 9 8 81000 7 - 10 25 - 44 Male

## 44 9 9 66000 7 - 10 45 - 64 Male

## 45 10 7 190000 7 - 10 15 - 24 Male

## 46 10 8 219000 7 - 10 25 - 44 Male

## 47 10 9 199000 7 - 10 45 - 64 Male

## 48 11 7 11000 7 - 10 15 - 24 Male

## 49 11 8 47000 7 - 10 25 - 44 Male

## 50 11 9 58000 7 - 10 45 - 64 Male

## 51 12 7 37000 7 - 10 15 - 24 Male

## 52 12 8 50000 7 - 10 25 - 44 Male

## 53 12 9 75000 7 - 10 45 - 64 Male

## 54 13 8 60000 7 - 10 25 - 44 Male

## 55 13 9 68000 7 - 10 45 - 64 Male

## header.header.header

## 1 Bachelor's degree

## 2 Bachelor's degree

## 3 Bachelor's degree

## 4 Certificate

## 5 Certificate

## 6 Certificate

## 7 Diploma

## 8 Diploma

## 9 No Qualification

## 10 No Qualification

## 11 No Qualification

## 12 Postgraduate qualification

## 13 Postgraduate qualification

## 14 Bachelor's degree

## 15 Bachelor's degree

## 16 Bachelor's degree

## 17 Certificate

## 18 Certificate

## 19 Certificate

## 20 Diploma

## 21 Diploma

## 22 Diploma

## 23 No Qualification

## 24 No Qualification

## 25 No Qualification

## 26 Postgraduate qualification

## 27 Postgraduate qualification

## 28 Postgraduate qualification

## 29 Bachelor's degree

## 30 Bachelor's degree

## 31 Certificate

## 32 Certificate

## 33 Certificate

## 34 Diploma

## 35 Diploma

## 36 Diploma

## 37 No Qualification

## 38 No Qualification

## 39 No Qualification

## 40 Postgraduate qualification

## 41 Postgraduate qualification

## 42 Bachelor's degree

## 43 Bachelor's degree

## 44 Bachelor's degree

## 45 Certificate

## 46 Certificate

## 47 Certificate

## 48 Diploma

## 49 Diploma

## 50 Diploma

## 51 No Qualification

## 52 No Qualification

## 53 No Qualification

## 54 Postgraduate qualification

## 55 Postgraduate qualification

BELOW\_RIGHT\_borders <- data\_cells %>% arrange(row, col)

Check that all sheets are parsed correctly

identical(ABOVE\_LEFT[, -1:-2], BELOW\_RIGHT[, -1:-2])

## [1] TRUE

identical(ABOVE\_LEFT[, -1:-2], ABOVE\_LEFT\_borders[, -1:-2])

## [1] FALSE

identical(ABOVE\_LEFT[, -1:-2], BELOW\_RIGHT\_borders[, -1:-2])

## [1] TRUE

**Attempt 4 (tidyxl + unpivotr)**

After re-reading the unpivotr documentation, I realized that the angles column could be treated as a row header in the unpivotr code. So I further modified the function:

cleanData3 = function(d) {

final = d %>%

select(row, col, data\_type, numeric, character) %>%

behead('N', ID) %>% # Extract column headers

behead('N', spine) %>%

behead('N', variable) %>%

behead('W', angles) %>% # angles as row header

select(numeric, ID:angles, data\_type, -spine) %>% # all vars are numeric

filter(variable %in% c'A','B')) %>% # Kills off some extra columns

spatter(variable) # Spreads, using data\_type, numeric

return(final)

}

cleanData3(d2)

## A tibble: 588 x 4

## ID angles A B

##

## 1 10 60. 31.5 1.00

## 2 10 65. 31.3 1.00

## 3 10 70. 31.1 0.202

## 4 10 75. 31.0 0.448

## 5 10 80. 30.8 0.671

## 6 10 85. 30.5 0.461

## 7 10 90. 30.3 0.245

## 8 10 95. 30.0 0.159

## 9 10 100. 29.7 0.170

##10 10 105. 29.2 0.421

## ... with 578 more rows

We get to the same output, but with much cleaner code. This is cool!!I’m going to go deeper into the unpivotr documentation and see what else can be in my regular pipeline. A big thank you to the tool-makers that create these tools that make everyday activies easier and make us stay saner.