Data Cleaning for Skills for Care Data

Meng Ye

3/14/2024

Updated 3/14/2024, use more accurate skills for care variables for analysis

### Setting up

library(tidyverse) # package for data cleaning and plotting  
library(readxl) # package for reading excel file  
library(broom) # extracting model summary as data frame  
library(modelsummary) # deriving model tables  
library(scales) # label percent  
library(lubridate) # working with dates  
library(here) # managing paths

# environment setup to run ordered logit properly  
options(contrasts = rep("contr.treatment", 2))

## Import data

#import the raw data file  
cqc\_skills <- read\_excel(here("data","SfC\_CQC.xlsx")) # CQC data with bridging column of SfC establishment id curated by Skills for Care  
skills\_raw <- read\_csv(here("data","Skills\_raw.csv")) # raw Skills for Care organizational data

## Pivo for CQC\_skills (cqc data with SfC establishmentid)

#select the CQC data that has Skills for Care establishment id  
CQC\_sub <- cqc\_skills %>%   
 filter(!is.na(establishmentid)) %>%   
#restructure to wide data (1 row for each establishment id)  
 pivot\_wider(names\_from = domain, values\_from = rating) %>%   
 mutate(form = ifelse(form == "NA", NA, form))  
  
head(CQC\_sub)

## # A tibble: 6 × 15  
## form cic\_type care\_home primary\_cat region service\_group publication\_date   
## <chr> <chr> <chr> <chr> <chr> <chr> <dttm>   
## 1 FPO NA Y Residential… East … Overall 2020-03-03 00:00:00  
## 2 FPO NA Y Residential… South… Overall 2017-10-31 00:00:00  
## 3 FPO NA Y Residential… East … Overall 2018-12-14 00:00:00  
## 4 FPO NA Y Residential… West … Overall 2019-09-17 00:00:00  
## 5 FPO NA N Community b… Yorks… Overall 2020-02-15 00:00:00  
## 6 NPO NA Y Residential… South… Overall 2018-11-13 00:00:00  
## # ℹ 8 more variables: inherited <chr>, establishmentid <dbl>, Safe <chr>,  
## # Effective <chr>, Caring <chr>, Responsive <chr>, `Well-led` <chr>,  
## # Overall <chr>

CQC\_sub %>%   
 summarize(all\_rows = n(),  
 unique\_rows = n\_distinct(establishmentid))

## # A tibble: 1 × 2  
## all\_rows unique\_rows  
## <int> <int>  
## 1 7258 6799

This results show that some facilities have more than 1 rating group. After inspecting the raw CQC data, it turned out multiple sets of ratings for the same location are for different service groups.

## Data Cleaning for Skills for Care data

### Select Columns

#write variable name list  
columns <- c("ParentSubSingle", "isparent", "estabcreateddate", #basic info  
 "Starters\_Leavers\_FILTER", "totalstarters", "totalleavers", #staff  
 "totalstaff", "totalvacancies","ORGWRKS", "ORGWRKSGP","ESTABWRKS", #staff  
 "esttype", "mainstid", "MAINSERGP1", "MAINSERGP2") #service descriptions

Notes for some variables:

ParentSubSingle: values: 0, 1, 2. Does not say which is parent/sub/single. It is observed 2 corresponds to `isparent = 1”, so should be parent.

Starters\_Leavers\_FILTER: Flag to define if a workplace has reliable starters or leavers information. 1 = Include, 0 = Exclude.

ORGWRKS: Sum of totalstaff **across** the organisation.

ORGWRKSGP: Grouping of ORGWRKS

Value Label -1 Not allocated 991 Micro (1 to 9) 992 Small (10 to 49) 993 Medium (50 to 249) 994 Large (250 or more)

esttype: sector of the workplace

mainstid: The main service provided by the workplace

totalstarters and totalleavers: Total number of permanent and temporary staff that started/left in the previous 12 months.

skills\_selected <- skills\_raw %>%   
 select(establishmentid, all\_of(columns), starts\_with("jr28"), starts\_with("jr29"),  
 matches("^st.\*cap$"), matches("^st.\*util$"))

* “jr” codes

Code Label

28 **Any job role**. This is the sum of all job roles submitted by the establishment.

29 **Any direct care role**. This is the sum of all direct care roles submitted by the establishment. (Note: this variable may be more directly related to service quality)

30 Any manager/supervisor role. This is the sum of all manager/supervisor roles submitted by the establishment.

31 Any regulated profession role. This is the sum of all regulated profession roles submitted by the establishment.

32 Any other roles. This is the sum of all other roles, those not included in groups 29,30 and 31, submitted by the establishment.

1 Senior Management

2 Middle Management

3 First Line Manager

4 Registered Manager

5 Supervisor

6 Social Worker

7 Senior Care Worker

8 Care Worker

9 Community Support and Outreach Work

10 Employment Support

11 Advice Guidance and Advocacy

15 Occupational Therapist

16 Registered Nurse

17 Allied Health Professional

22 Technician

23 Other care-providing job role

24 Managers and staff in care-related but not care-providing roles

25 Administrative or office staff not care-providing

26 Ancillary staff not care-providing

27 Other non-care-providing job roles

34 Activities worker or co-ordinator

35 Safeguarding and reviewing officer

36 Occupational therapist assistant

37 Nursing Associate

38 Nursing Assistant

39 Assessment officer

40 Care co-ordinator

41 Care navigator

42 Any Childrens/young peoples job role

## Exploring the strucutre of Skills for Care data

Use this small dataset to check if each establishment ID has multiple rows

skills\_raw %>%   
 select(establishmentid) %>%   
 summarise(count = n(),  
 count\_unqice = n\_distinct(establishmentid))

## # A tibble: 1 × 2  
## count count\_unqice  
## <int> <int>  
## 1 18956 18956

In the Skills for Care data set, each establishmentid only corresponds to 1 row data.

## Merge data and derive new variables

### coding section

# merge and recode data  
merged\_wide <- CQC\_sub %>%   
 left\_join(skills\_selected, by = "establishmentid") %>%   
 mutate(main\_service\_type = factor(mainstid),  
 main\_group1 = factor(MAINSERGP1),  
 main\_group2 = factor(MAINSERGP2),  
   
 main\_service\_group = case\_when(  
 main\_group1 == "1" ~ "Adult residential",  
 main\_group1 == "3" ~ "Adult domicilary",   
 TRUE ~ as.character(NA)),  
   
# derive truncated service types (where CICs exist) see table 1  
 service\_type\_selected = case\_when(  
 main\_service\_type == "1" ~ "care home w/ nursing",  
 main\_service\_type == "2" ~ "care home w/o nursing",  
 main\_service\_type == "8" ~ "Domicilary Care",  
 main\_service\_type == "55" ~ "Supported Living",  
 # dropped 61 -- community based for learning disability  
 TRUE ~ as.character(NA)),  
  
# recode size type to be more readable  
 size\_type = case\_when(  
 ORGWRKSGP == -1 ~ as.character(NA),  
 ORGWRKSGP == 991 ~ "micro",  
 ORGWRKSGP == 992 ~ "small",  
 ORGWRKSGP == 993 ~ "medium",  
 ORGWRKSGP == 994 ~ "large",  
 TRUE ~ as.character(NA)),  
# derive age of the organization  
 start\_date = dmy(estabcreateddate),  
 age = 2024-year(start\_date)  
) %>%   
 mutate(total\_leavers = ifelse(totalleavers < 0, NA, totalleavers),  
 total\_starters = ifelse(totalleavers < 0, NA, totalstarters))  
# recode -1 or -2 to NA for totallevers and totalstarters

Check distribution of the data and potential variables

## check the distribution of the data by service type selected  
merged\_wide %>%   
 group\_by(service\_type\_selected) %>%   
 summarise(count = n())

## # A tibble: 5 × 2  
## service\_type\_selected count  
## <chr> <int>  
## 1 Domicilary Care 1858  
## 2 Supported Living 370  
## 3 care home w/ nursing 1367  
## 4 care home w/o nursing 3400  
## 5 <NA> 263

# checking how much missing values for total starters  
merged\_wide %>%   
 group\_by(!is.na(total\_starters)) %>%   
 summarize(count = n())

## # A tibble: 2 × 2  
## `!is.na(total\_starters)` count  
## <lgl> <int>  
## 1 FALSE 1743  
## 2 TRUE 5515

# checking how much missing values for total leavers  
merged\_wide %>%   
 group\_by(!is.na(total\_leavers)) %>%   
 summarize(count = n())

## # A tibble: 2 × 2  
## `!is.na(total\_leavers)` count  
## <lgl> <int>  
## 1 FALSE 1743  
## 2 TRUE 5515

# checking how much missing values for "jr29"  
merged\_wide %>%   
 group\_by(ifelse(jr29flag == 1, TRUE, FALSE)) %>%   
 summarize(count = n())

## # A tibble: 2 × 2  
## `ifelse(jr29flag == 1, TRUE, FALSE)` count  
## <lgl> <int>  
## 1 FALSE 721  
## 2 TRUE 6537

## Derive new variables

* About the **new** turnover rate

Employee turnover rate is the percent of employees who leave a company within a specific time period. Turnover rate is commonly calculated by month, quarter, or year and includes both voluntary and involuntary losses. Monthly and quarterly turnover rates are commonly expressed as averages, while annual turnover rate is usually cumulative.

$$ = \ = \ = +

$$

### New variables could be derived

count of employees left in last 12 month/latest total staff by service type, total staff/ number of vacancy - not useful due to a lot of missing value

Any other ideas? Please specify **exact** variable rather than vague concept

# derive new variables  
merged\_recoded\_wide <- merged\_wide %>%   
# mutate NA only for jr28flag = 1 and jr29 = 1, respectively  
 mutate(across(starts\_with("jr28"),   
 ~ ifelse(jr28flag == 1, ifelse(is.na(.x), 0, .x), .x))) %>%   
 mutate(across(starts\_with("jr29"),   
 ~ ifelse(jr29flag == 1, ifelse(is.na(.x), 0, .x), .x))) %>%   
 mutate(across(starts\_with("st"),  
 ~ ifelse(is.na(.x), 0, .x))) %>%   
   
 # calculating the old turnover rate, renamed to "turnover1"  
  
 mutate(totalstaff = ifelse(is.na(totalstaff), 0, totalstaff),  
 turnover0 = totalleavers/totalstaff \* 100,  
 turnover0 = ifelse(turnover0 < 0, NA, turnover0),  
 turnover0 = ifelse(is.na(turnover0) | turnover0 == Inf, NA, turnover0)) %>%   
   
 # derive the average totalstaff count for the between T-12month and T0  
 # and calculate the turnover rate with average total staff  
  
 mutate(average\_totalstaff = totalstaff + 1/2 \* (total\_leavers - total\_starters)) %>%   
 mutate(average\_totalstaff = ifelse(is.na(average\_totalstaff), 0, average\_totalstaff),  
 turnover = totalleavers/average\_totalstaff \* 100,  
 turnover = ifelse(turnover < 0, NA, turnover),  
 turnover = ifelse(is.na(turnover) | turnover == Inf, NA, turnover)) %>%   
 rowwise() %>%   
 mutate(capacity = sum(across(matches("^st.\*cap$"))),  
 utility = sum(across(matches("^st.\*util$")))) %>%   
 ungroup() %>%   
 mutate(staff\_level0 = totalstaff/utility,  
 staff\_level0 = ifelse(staff\_level0 == Inf, NA, staff\_level0)) %>%   
 mutate(direct\_staff = jr29work,  
 staff\_level = direct\_staff/utility,  
 staff\_level = ifelse(staff\_level == Inf, NA, staff\_level))

table4 <- datasummary(turnover \* service\_type\_selected \*(Mean + SD)~ form,  
 data = merged\_recoded\_wide, fmt = 1, na.rm = T)  
table4

|  | service\_type\_selected |  | CIC | FPO | GOV | IND | NPO |
| --- | --- | --- | --- | --- | --- | --- | --- |
| turnover | care home w/ nursing | Mean | 34.6 | 39.7 | 12.9 | 18.0 | 25.3 |
|  |  | SD | 23.8 | 159.0 | 14.1 | 15.8 | 21.7 |
|  | care home w/o nursing | Mean | 7.7 | 30.0 | 12.9 | 18.1 | 20.6 |
|  |  | SD | 10.9 | 47.0 | 10.2 | 22.9 | 24.2 |
|  | Domicilary Care | Mean | 30.8 | 43.5 | 11.9 | 29.0 | 29.5 |
|  |  | SD | 30.4 | 131.5 | 8.2 | 32.9 | 50.5 |
|  | Supported Living | Mean | 10.7 | 25.9 | 10.3 |  | 27.7 |
|  |  | SD |  | 27.5 | 10.1 |  | 27.9 |

## Run Toy Models

Prepare data for analysis

#pivot longer back  
merged\_long <- merged\_recoded\_wide %>%   
 pivot\_longer(cols = c(Safe, Effective, Caring, Responsive, `Well-led`, Overall),  
 names\_to = "domain", values\_to = "rating")

#select relevant columns, rename and relabel   
merged\_long\_ordered <- merged\_long %>%   
# recode legal form types to be more readable / easier to present  
 mutate(inherited = ifelse(inherited == "Y", TRUE, FALSE),  
 rating = recode(rating,   
 "Insufficient evidence to rate" = "NA",  
 "Requires improvement" = "Req improv")) %>%   
 # set the order of the values in the factors   
 mutate(form = fct\_relevel(form, "FPO"),  
   
 # assume the order of the ratings as follows but need to double check with the source   
 rating = ordered(rating, levels = c("Inadequate","Req improv", "Good", "Outstanding"))) %>%  
   
 # adding the rating data coded as numerical  
 mutate(rating\_num = case\_when(rating == "Inadequate" ~ 1,  
 rating == "Req improv" ~ 2,  
 rating == "Good" ~ 3,  
 rating == "Outstanding" ~ 4)) %>%   
   
 mutate(category = case\_when(primary\_cat == "Community based adult social care services" ~ "community",  
 primary\_cat == "Residential social care" ~ "residential",  
 TRUE ~ as.character(NA)),  
  
 # deriving year column and dummy variable for before\_covid  
 year = year(publication\_date),  
 during\_covid = ifelse(year >= 2020, TRUE, FALSE),  
 before\_covid = ifelse(year <= 2019, TRUE, FALSE)) %>%  
  
 # converting the ordinal variable to numerical   
 mutate(rating\_num = case\_when(rating == "Inadequate" ~ 1,  
 rating == "Req improv" ~ 2,  
 rating == "Good" ~ 3,  
 rating == "Outstanding" ~ 4)) %>%   
 # derive the rating dummy  
 mutate(rating\_higher = ifelse(rating\_num > 2, 1, 0))  
  
# show first several rows of the data set derived   
head(merged\_long\_ordered)

## # A tibble: 6 × 93  
## form cic\_type care\_home primary\_cat region service\_group publication\_date   
## <fct> <chr> <chr> <chr> <chr> <chr> <dttm>   
## 1 FPO NA Y Residential… East … Overall 2020-03-03 00:00:00  
## 2 FPO NA Y Residential… East … Overall 2020-03-03 00:00:00  
## 3 FPO NA Y Residential… East … Overall 2020-03-03 00:00:00  
## 4 FPO NA Y Residential… East … Overall 2020-03-03 00:00:00  
## 5 FPO NA Y Residential… East … Overall 2020-03-03 00:00:00  
## 6 FPO NA Y Residential… East … Overall 2020-03-03 00:00:00  
## # ℹ 86 more variables: inherited <lgl>, establishmentid <dbl>,  
## # ParentSubSingle <dbl>, isparent <dbl>, estabcreateddate <chr>,  
## # Starters\_Leavers\_FILTER <dbl>, totalstarters <dbl>, totalleavers <dbl>,  
## # totalstaff <dbl>, totalvacancies <dbl>, ORGWRKS <dbl>, ORGWRKSGP <dbl>,  
## # ESTABWRKS <dbl>, esttype <dbl>, mainstid <dbl>, MAINSERGP1 <dbl>,  
## # MAINSERGP2 <dbl>, jr28flag <dbl>, jr28perm <dbl>, jr28temp <dbl>,  
## # jr28pool <dbl>, jr28agcy <dbl>, jr28oth <dbl>, jr28emp <dbl>, …

### Save data

write\_rds(merged\_long\_ordered, file = here("data","cqc\_sfc\_cleaned.rds"))