Analysis of NHS Compliments

November 29, 2019

Table of Contents

1 Set-up

2 Descriptives

3 Complimented practices

3.1 Descriptives

3.2 Chi-square tests

3.3 Figure 1

3.4 Extra-role practices

4 Gratitude aims

4.1 Descriptives

4.2 Chi-square tests

4.3 Figure 2

4.4 Total with gratitude aim

1 Set-up

[25]:
```
library(tidyverse)
library(openxlsx) # to read xlsx file
library(scales) # used for percent
library(stats) # used for chisq.test
library(reshape2) # used for dcast
library(lsr) # used for anova/etaSquared

sessionInfo()
```

```
R version 3.6.1 (2019-07-05)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 10 x64 (build 18362)

Matrix products: default
```


Gillespie A, Reader TW. *BMJ Qual Saf* 2021; 30:484–492. doi: 10.1136/bmjqs-2019-010077

```
locale:
[1] LC_COLLATE=English_United Kingdom.1252
[2] LC_CTYPE=English_United Kingdom.1252
[3] LC_MONETARY=English_United Kingdom.1252
[4] LC_NUMERIC=C
[5] LC_TIME=English_United Kingdom.1252

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
 [1] lsr_0.5         reshape2_1.4.3  scales_1.0.0    openxlsx_4.1.0
 [5] forcats_0.4.0   stringr_1.4.0   dplyr_0.8.0.1   purrr_0.3.2
 [9] readr_1.3.1     tidyr_0.8.3     tibble_2.1.1    ggplot2_3.1.1
[13] tidyverse_1.2.1 jsonlite_1.6    formatR_1.6

loaded via a namespace (and not attached):
 [1] pbdZMQ_0.3-3     tidyselect_0.2.5 repr_0.19.2      haven_2.1.0
 [5] lattice_0.20-38  colorspace_1.4-1 generics_0.0.2   htmltools_0.3.6
 [9] base64enc_0.1-3  rlang_0.3.4      pillar_1.3.1     glue_1.3.1
[13] withr_2.1.2      modelr_0.1.4     readxl_1.3.1     uuid_0.1-2
[17] plyr_1.8.4       munsell_0.5.0    gtable_0.3.0     cellranger_1.1.0
[21] rvest_0.3.3      zip_2.0.1        evaluate_0.13    labeling_0.3
[25] broom_0.5.2      IRdisplay_0.7.0  Rcpp_1.0.1       backports_1.1.4
[29] IRkernel_0.8.15  hms_0.4.2        digest_0.6.18    stringi_1.4.3
[33] grid_3.6.1       cli_1.1.0        tools_3.6.1      magrittr_1.5
[37] lazyeval_0.2.2   crayon_1.3.4     pkgconfig_2.0.2  xml2_1.2.0
[41] lubridate_1.7.4  assertthat_0.2.1 httr_1.4.0       rstudioapi_0.10
[45] R6_2.4.0         nlme_3.1-139     compiler_3.6.1
```

[2]:
```
data <- read.xlsx("Supplementary file 2 data.xlsx")
nrow(data) # total compliments
```

1267

[3]:
```
head(data, n=3L)
```

id	wordcount	to	to_ceo	to_teamunitdept	to_staff	from	from_fofamily	from_other	from_patient
1	144	staff	0	0	1	1	1	0	0
2	392	manager	0	1	0	1	1	0	0
3	380	ceo	1	0	0	1	1	0	0

Gillespie A, Reader TW. *BMJ Qual Saf* 2021; 30:484–492. doi: 10.1136/bmjqs-2019-010077

2 Descriptives

[4]:
```
# wordcount total
sum(data$wordcount)
# wordcount summary
summary(data$wordcount)
# wordcount std deviation
round(sd(data$wordcount), 3)
```

288563

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   12.0   132.0   194.0   227.8   272.0  1477.0
```

157.925

[5]:
```
# function to calculate frequencies and percent
freq_pct <- function(variable) {
        print(deparse(substitute(variable)))
        print(paste0("N = ", sum(variable)))
        print(paste0("% = ", round(sum(variable)/nrow(data), digits = 5)*100))
}
```

[6]:
```
# frequencies & percent for 'from'
freq_pct(data$from_fofamily)
freq_pct(data$from_other)
freq_pct(data$from_patient)
```

```
[1] "data$from_fofamily"
[1] "N = 459"
[1] "% = 36.227"
[1] "data$from_other"
[1] "N = 77"
[1] "% = 6.077"
[1] "data$from_patient"
[1] "N = 731"
[1] "% = 57.695"
```

[7]:
```
# frequencies & percent for 'age'
freq_pct(data$age_na)
freq_pct(data$age_adult)
freq_pct(data$age_child)
freq_pct(data$age_elderly)
```

```
[1] "data$age_na"
[1] "N = 179"
[1] "% = 14.128"
[1] "data$age_adult"
```


Gillespie A, Reader TW. *BMJ Qual Saf* 2021; 30:484–492. doi: 10.1136/bmjqs-2019-010077

```
[1] "N = 754"
[1] "% = 59.511"
[1] "data$age_child"
[1] "N = 86"
[1] "% = 6.788"
[1] "data$age_elderly"
[1] "N = 248"
[1] "% = 19.574"
```

[8]:
```
# frequencies & percent for 'condition'
freq_pct(data$condition_na)
freq_pct(data$condition_ae)
freq_pct(data$condition_maternity)
freq_pct(data$condition_chronic)
freq_pct(data$condition_planned)
```

```
[1] "data$condition_na"
[1] "N = 173"
[1] "% = 13.654"
[1] "data$condition_ae"
[1] "N = 388"
[1] "% = 30.624"
[1] "data$condition_maternity"
[1] "N = 52"
[1] "% = 4.104"
[1] "data$condition_chronic"
[1] "N = 308"
[1] "% = 24.309"
[1] "data$condition_planned"
[1] "N = 346"
[1] "% = 27.309"
```

[9]:
```
# frequencies & percent for 'outcome'
freq_pct(data$outcome_death)
freq_pct(data$outcome_expected)
freq_pct(data$outcome_na)
freq_pct(data$outcome_negative)
freq_pct(data$outcome_positive)
```

```
[1] "data$outcome_death"
[1] "N = 135"
[1] "% = 10.655"
[1] "data$outcome_expected"
[1] "N = 689"
[1] "% = 54.38"
[1] "data$outcome_na"
[1] "N = 10"
[1] "% = 0.789"
```


Gillespie A, Reader TW. *BMJ Qual Saf* 2021; 30:484–492. doi: 10.1136/bmjqs-2019-010077

```
[1] "data$outcome_negative"
[1] "N = 26"
[1] "% = 2.052"
[1] "data$outcome_positive"
[1] "N = 407"
[1] "% = 32.123"
```

[10]:
```
# frequencies & percent for 'addressee'
table(data$to)
round((table(data$to)/1267)*100, 3)
```

```
   ceo manager   staff
   390     667     210


   ceo manager   staff
30.781  52.644  16.575
```

3 Complimented practices

[11]:
```
# select variables
temp <- data[ , c("to",
                  "practices_clin_qual",
                  "practices_clin_safety",
                  "practices_clin_vague",
                  "practices_clin_extra",
                  "practices_man_env",
                  "practices_man_inst",
                  "practices_man_vague",
                  "practices_man_extra",
                  "practices_rel_list",
                  "practices_rel_com",
                  "practices_rel_respect",
                  "practices_rel_vague",
                  "practices_rel_extra")]
```

3.1 Descriptives

[12]:
```
# frequencies & percent for 'clinical' categories (reusing 'freq_pct' function)
freq_pct(temp$practices_clin_qual)
freq_pct(temp$practices_clin_safety)
freq_pct(temp$practices_clin_extra)
freq_pct(temp$practices_clin_vague)
```

```
[1] "temp$practices_clin_qual"
[1] "N = 147"
```

```
[1] "% = 11.602"
[1] "temp$practices_clin_safety"
[1] "N = 148"
[1] "% = 11.681"
[1] "temp$practices_clin_extra"
[1] "N = 100"
[1] "% = 7.893"
[1] "temp$practices_clin_vague"
[1] "N = 294"
[1] "% = 23.204"
```

[13]:
```
# frequencies & percent for 'relationship' categories
freq_pct(temp$practices_rel_com)
freq_pct(temp$practices_rel_list)
freq_pct(temp$practices_rel_respect)
freq_pct(temp$practices_rel_extra)
freq_pct(temp$practices_rel_vague)
```

```
[1] "temp$practices_rel_com"
[1] "N = 218"
[1] "% = 17.206"
[1] "temp$practices_rel_list"
[1] "N = 37"
[1] "% = 2.92"
[1] "temp$practices_rel_respect"
[1] "N = 94"
[1] "% = 7.419"
[1] "temp$practices_rel_extra"
[1] "N = 495"
[1] "% = 39.069"
[1] "temp$practices_rel_vague"
[1] "N = 303"
[1] "% = 23.915"
```

[14]:
```
# frequencies & percent for 'management' categories
freq_pct(temp$practices_man_env)
freq_pct(temp$practices_man_inst)
freq_pct(temp$practices_man_extra)
freq_pct(temp$practices_man_vague)
```

```
[1] "temp$practices_man_env"
[1] "N = 143"
[1] "% = 11.287"
[1] "temp$practices_man_inst"
[1] "N = 199"
[1] "% = 15.706"
[1] "temp$practices_man_extra"
[1] "N = 80"
```


Gillespie A, Reader TW. *BMJ Qual Saf* 2021; 30:484–492. doi: 10.1136/bmjqs-2019-010077

```
[1] "% = 6.314"
[1] "temp$practices_man_vague"
[1] "N = 21"
[1] "% = 1.657"
```

3.2 Chi-square tests

[15]:
```r
# create table
df <- temp %>%
  gather(measure, value, 2:length(temp), factor_key = TRUE) %>%
  group_by(to, measure, value) %>%
  tally

# get variable names to test
vars <- names(temp[2:length(temp)])

# function for chi square test
df_to_chisq <- function(df, grp) {
  filter(df, measure == grp) %>%
    ungroup() %>%
    select(-measure) %>%
    as.data.frame() %>%
    dcast(value ~ to, value.var = "n") %>%
    remove_rownames() %>%
    column_to_rownames(var = "value") %>%
    chisq.test()
}

# test variables
for(i in vars) {
  assign(paste0("chi_", i), df_to_chisq(df, i))
}

# print test results
for(var in vars) {
  print(paste0("Chi-squared test for: ", "chi_", var))
  print(eval(parse(text = paste0("chi_", var))))
}
```

```
Warning message in chisq.test(.):
"Chi-squared approximation may be incorrect"
[1] "Chi-squared test for: chi_practices_clin_qual"

        Pearson's Chi-squared test

data:  .
X-squared = 14.807, df = 2, p-value = 0.0006092
```

```
[1] "Chi-squared test for: chi_practices_clin_safety"

        Pearson's Chi-squared test

data:  .
X-squared = 21.201, df = 2, p-value = 2.49e-05

[1] "Chi-squared test for: chi_practices_clin_vague"

        Pearson's Chi-squared test

data:  .
X-squared = 13.255, df = 2, p-value = 0.001324

[1] "Chi-squared test for: chi_practices_clin_extra"

        Pearson's Chi-squared test

data:  .
X-squared = 1.0038, df = 2, p-value = 0.6054

[1] "Chi-squared test for: chi_practices_man_env"

        Pearson's Chi-squared test

data:  .
X-squared = 9.9462, df = 2, p-value = 0.006922

[1] "Chi-squared test for: chi_practices_man_inst"

        Pearson's Chi-squared test

data:  .
X-squared = 24.824, df = 2, p-value = 4.07e-06

[1] "Chi-squared test for: chi_practices_man_vague"

        Pearson's Chi-squared test

data:  .
X-squared = 5.4228, df = 2, p-value = 0.06644

[1] "Chi-squared test for: chi_practices_man_extra"

        Pearson's Chi-squared test

data:  .
```

```
X-squared = 2.5524, df = 2, p-value = 0.2791

[1] "Chi-squared test for: chi_practices_rel_list"

        Pearson's Chi-squared test

data:  .
X-squared = 6.3826, df = 2, p-value = 0.04112

[1] "Chi-squared test for: chi_practices_rel_com"

        Pearson's Chi-squared test

data:  .
X-squared = 11.804, df = 2, p-value = 0.002734

[1] "Chi-squared test for: chi_practices_rel_respect"

        Pearson's Chi-squared test

data:  .
X-squared = 0.81746, df = 2, p-value = 0.6645

[1] "Chi-squared test for: chi_practices_rel_vague"

        Pearson's Chi-squared test

data:  .
X-squared = 4.9006, df = 2, p-value = 0.08627

[1] "Chi-squared test for: chi_practices_rel_extra"

        Pearson's Chi-squared test

data:  .
X-squared = 18.074, df = 2, p-value = 0.0001189
```

3.3 Figure 1

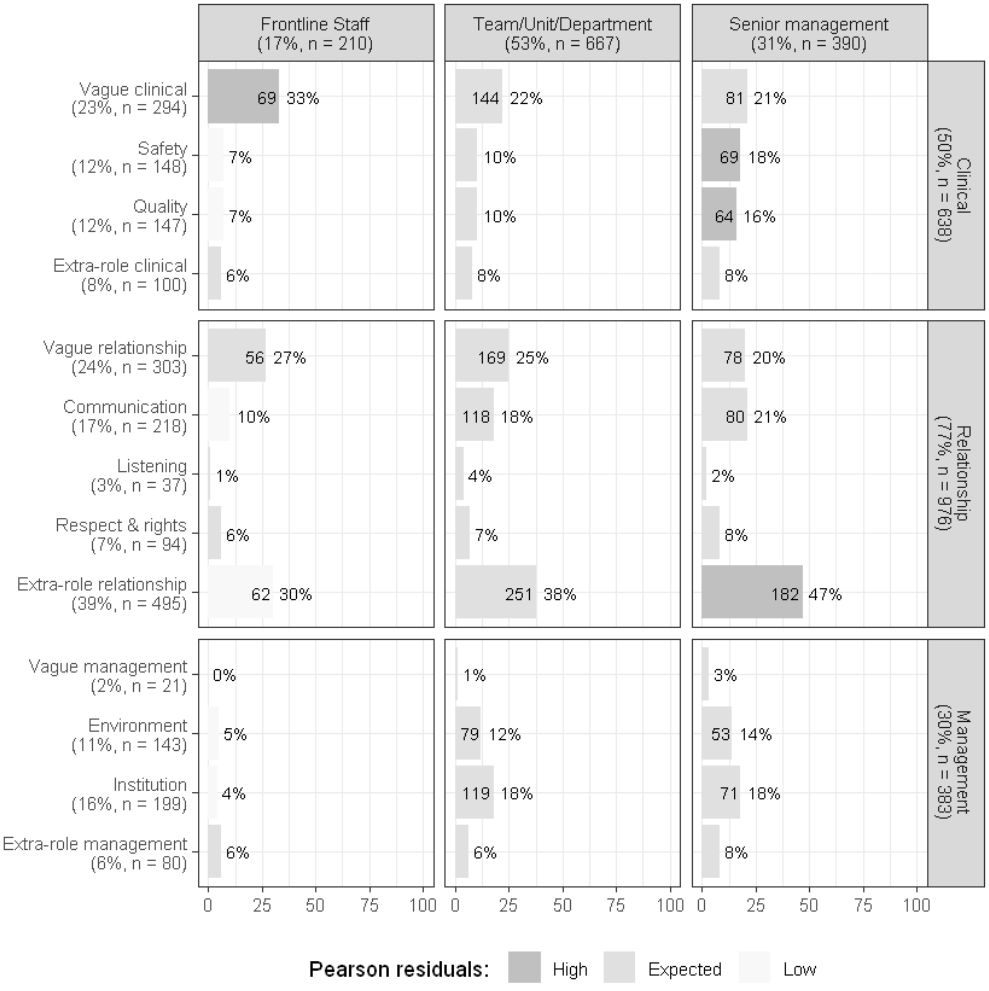
[16]:
```
# function to create dfs with residuals
df_resids <- function(chi, variable1) {
  df_obs <- melt(chi$observed)
  df_resid <- melt(chi$residuals)
  data.frame(measure = variable1, df_obs, df_resid$value)
}
```

```
# calculate residuals
resids <- list()
for(var in vars) {
  d <- df_resids(eval(parse(text = paste0("chi_", var))), var)
  resids[[var]] <- d
}

# bind residuals to df
df <- do.call(rbind, resids)
rownames(df) <- NULL

# add n for each addressee
df$n <- ifelse(df$Var2 == "staff", nrow(data[data$to == "staff",]),
               ifelse(df$Var2 == "manager", nrow(data[data$to == "manager",]),
                      ifelse(df$Var2 == "ceo", nrow(data[data$to == "ceo",]),␣
 ↪NA)))

# calculate percent for each addressee
df <- df %>%
  mutate(percent = round(value/n, 2)) %>%
  mutate(percent2 = as.integer((percent)*100))

# remove Var1= 0
df <- df %>%
  filter(Var1 == 1) %>%
  select(-Var1)

# rename variables
df <- df %>%
  rename(to = Var2, Residual = df_resid.value)

# create residual category
df$resid_cat <- ifelse(df$Residual > 2, "High",
                       ifelse(df$Residual < -2, "Low",        "Expected"))

# create y axis facets
df$facet     <- ifelse(grepl("clin", df$measure), "clin",
                       ifelse(grepl("man", df$measure), "man",
                              ifelse(grepl("rel", df$measure), "rel",
                                     NA)))

# order residual categories
df$resid_cat <-
  ordered(df$resid_cat,
          levels = c("High",
                     "Expected",
                     "Low"),
```

```
          labels = c("High",
                     "Expected",
                     "Low"))

# function to give label to variable with % & n
label_variable <- function(name, variable) {
  paste0(name, "\n(",
         round(sum(variable)/nrow(data), digits = 2)*100,
         "%, n = ",
         sum(variable),
         ")")
}

# create labels and order factors
df$resid_cat <-
  ordered(df$resid_cat,
          levels = c("High",
                     "Expected",
                     "Low"),
          labels = c("High",
                     "Expected",
                     "Low"))

df$to <-
  factor(df$to,
         levels = c("staff",
                    "manager",
                    "ceo"),
         labels = c(label_variable("Frontline Staff", data$to_staff),
                    label_variable("Team/Unit/Department",␣
 ↪data$to_teamunitdept),
                    label_variable("Senior management", data$to_ceo)))

df$facet <-
  factor(df$facet,
         levels = c("clin",
                    "rel",
                    "man"),
         labels = c(label_variable("Clinical", data$practices_clin),
                    label_variable("Relationship", data$practices_rel),
                    label_variable("Management", data$practices_man)))

df$measure <-
  factor(df$measure,
         levels = c(
           "practices_clin_extra",
           "practices_clin_qual",
```

```
            "practices_clin_safety",
            "practices_clin_vague",

            "practices_man_extra",
            "practices_man_inst",
            "practices_man_env",
            "practices_man_vague",

            "practices_rel_extra",
            "practices_rel_respect",
            "practices_rel_list",
            "practices_rel_com",
            "practices_rel_vague"
          ),
          labels = c(
            label_variable("Extra-role clinical", data$practices_clin_extra),
            label_variable("Quality", data$practices_clin_qual),
            label_variable("Safety", data$practices_clin_safety),
            label_variable("Vague clinical", data$practices_clin_vague),

            label_variable("Extra-role management", data$practices_man_extra),
            label_variable("Institution", data$practices_man_inst),
            label_variable("Environment", data$practices_man_env),
            label_variable("Vague management", data$practices_man_vague),

            label_variable("Extra-role relationship", data$practices_rel_extra),
            label_variable("Respect & rights", data$practices_rel_respect),
            label_variable("Listening", data$practices_rel_list),
            label_variable("Communication", data$practices_rel_com),
            label_variable("Vague relationship", data$practices_rel_vague)
          ))

# create plotting function (used for Figure 1 & 2)
chi_bar_plot <- function(df) {
  df %>%
    ggplot(aes(x = measure, y = percent2, fill = resid_cat)) +
    scale_fill_manual(values=c("#C0C0C0", "#E0E0E0", "#F8F8F8")) +
    geom_bar(stat = "identity") +
    geom_text(
      data = subset(df, percent2 > 10), # remove n below %
      aes(label = value), hjust = 1.1, size = 3) +
    geom_text(aes(label = percent(percent, accuracy = 1)), hjust = -.2, size =␣
↪3) +
    facet_grid(facet~to, scales = "free", space = "free") +
    scale_y_continuous(limits = c(0, 100)) +
    coord_flip() +
    theme_bw() +
```

```
    theme(strip.text = element_text(size = 9)) +
    theme(plot.subtitle = element_text(size = 9)) +
    theme(legend.position = "bottom",
          legend.text = element_text(size = 9)) +
    theme(axis.title.x = element_blank(),
          axis.title.y = element_blank()) +
    labs(fill = "Pearson residuals: ")
}

# plot Figure 1
chi_bar_plot(df)
```



3.4 Extra-role practices

[17]:
```
# total n and % of compliments reporting extra-role issues
extra <-
  ifelse((data$practices_clin_extra + data$practices_rel_extra +␣
 ↪data$practices_man_extra) > 0, 1, 0)

sum(extra)
round((sum(extra)/1267)*100, 3)
```

601

47.435

4 Gratitude aims

[18]:
```
# select variables
temp <- data[ , c("to",
                  "project_acknowledge_thankyou",
                  "project_acknowledge_thankthem",
                  "project_reward_material",
                  "project_reward_passon",
                  "project_reward_ccd",
                  "project_promote_suggestion",
                  "project_promote_nhs",
                  "project_promote_commend")]
```

4.1 Descriptives

[19]:
```
# frequencies & percent for 'acknowledging' (reusing 'freq_pct' function)
freq_pct(temp$project_acknowledge_thankyou)
freq_pct(temp$project_acknowledge_thankthem)
```

```
[1] "temp$project_acknowledge_thankyou"
[1] "N = 355"
[1] "% = 28.019"
[1] "temp$project_acknowledge_thankthem"
[1] "N = 770"
[1] "% = 60.773"
```

[20]:
```
# frequencies & percent for 'rewarding'
freq_pct(temp$project_reward_passon)
freq_pct(temp$project_reward_material)
freq_pct(temp$project_reward_ccd)
```

```
[1] "temp$project_reward_passon"
[1] "N = 487"
[1] "% = 38.437"
```


Gillespie A, Reader TW. *BMJ Qual Saf* 2021; 30:484–492. doi: 10.1136/bmjqs-2019-010077

```
[1] "temp$project_reward_material"
[1] "N = 48"
[1] "% = 3.788"
[1] "temp$project_reward_ccd"
[1] "N = 58"
[1] "% = 4.578"
```

[21]:
```
# frequencies & percent for 'promoting'
freq_pct(temp$project_promote_commend)
freq_pct(temp$project_promote_nhs)
freq_pct(temp$project_promote_suggestion)
```

```
[1] "temp$project_promote_commend"
[1] "N = 599"
[1] "% = 47.277"
[1] "temp$project_promote_nhs"
[1] "N = 237"
[1] "% = 18.706"
[1] "temp$project_promote_suggestion"
[1] "N = 99"
[1] "% = 7.814"
```

4.2 Chi-square tests

[22]:
```
# create table
df <- temp %>%
  gather(measure, value, 2:length(temp), factor_key = TRUE) %>%
  group_by(to, measure, value) %>%
  tally

# get variable names to test
vars <- names(temp[2:length(temp)])

# test variables (reusing 'df_to_chisq' function)
for(i in vars) {
  assign(paste0("chi_", i), df_to_chisq(df, i))
}

# print test results
for(var in vars) {
  print(paste0("Chi-squared test for: ", "chi_", var))
  print(eval(parse(text = paste0("chi_", var))))
}
```

```
[1] "Chi-squared test for: chi_project_acknowledge_thankyou"

        Pearson's Chi-squared test
```

```
data:  .
X-squared = 362.53, df = 2, p-value < 2.2e-16

[1] "Chi-squared test for: chi_project_acknowledge_thankthem"

        Pearson's Chi-squared test

data:  .
X-squared = 140.58, df = 2, p-value < 2.2e-16

[1] "Chi-squared test for: chi_project_reward_material"

        Pearson's Chi-squared test

data:  .
X-squared = 3.3843, df = 2, p-value = 0.1841

[1] "Chi-squared test for: chi_project_reward_passon"

        Pearson's Chi-squared test

data:  .
X-squared = 96.81, df = 2, p-value < 2.2e-16

[1] "Chi-squared test for: chi_project_reward_ccd"

        Pearson's Chi-squared test

data:  .
X-squared = 28.892, df = 2, p-value = 5.322e-07

[1] "Chi-squared test for: chi_project_promote_suggestion"

        Pearson's Chi-squared test

data:  .
X-squared = 7.0654, df = 2, p-value = 0.02923

[1] "Chi-squared test for: chi_project_promote_nhs"

        Pearson's Chi-squared test

data:  .
X-squared = 43.005, df = 2, p-value = 4.588e-10

[1] "Chi-squared test for: chi_project_promote_commend"

        Pearson's Chi-squared test
```

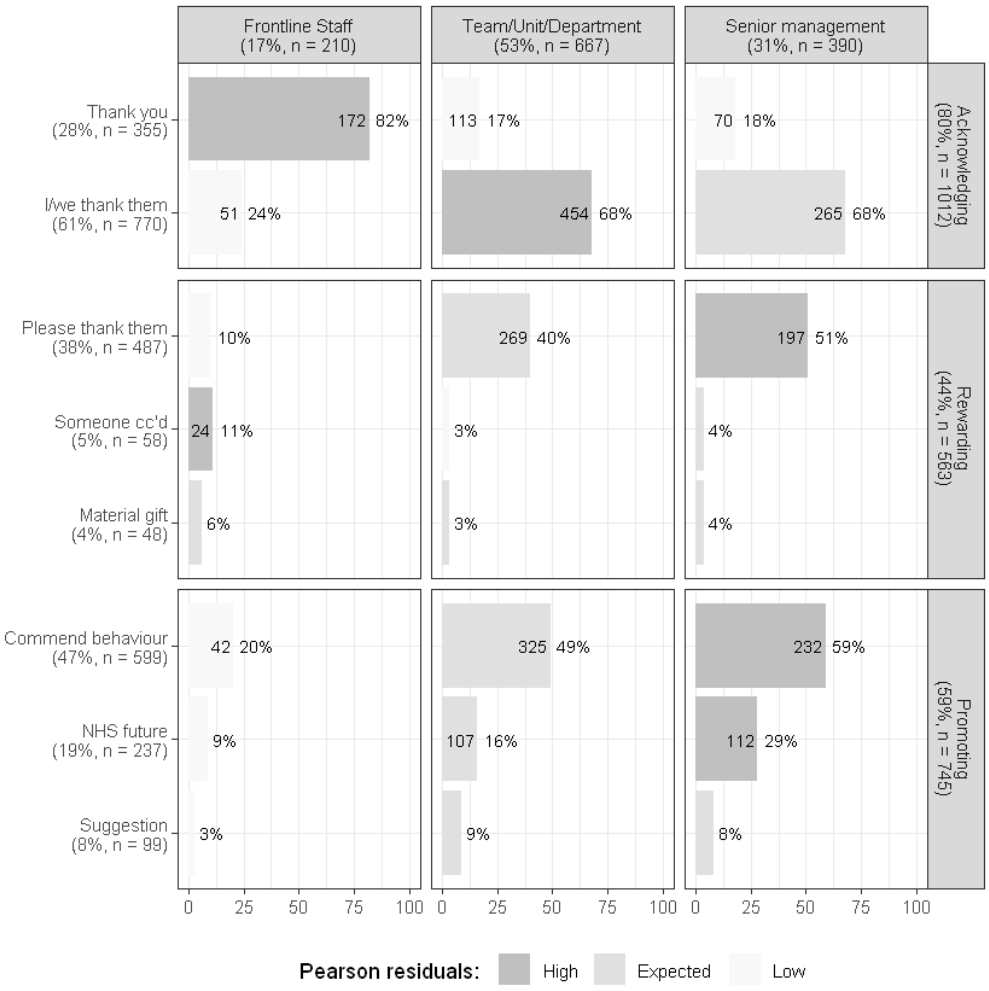
```
data:  .
X-squared = 86.573, df = 2, p-value < 2.2e-16
```

4.3 Figure 2

[23]:
```r
# calculate residuals (reusing 'df_resids' function)
resids <- list()
for(var in vars) {
  d <- df_resids(eval(parse(text = paste0("chi_", var))), var)
  resids[[var]] <- d
}

# bind residuals to df
df <- do.call(rbind, resids)
rownames(df) <- NULL

# add n for each addressee
df$n <- ifelse(df$Var2 == "staff", nrow(data[data$to == "staff",]),
        ifelse(df$Var2 == "manager", nrow(data[data$to == "manager",]),
        ifelse(df$Var2 == "ceo", nrow(data[data$to == "ceo",]), NA)))

# calculate percent for each addressee
df <- df %>%
  mutate(percent = round(value/n, 2)) %>%
  mutate(percent2 = as.integer((percent)*100))

# remove Var1= 0
df <- df %>%
  filter(Var1 == 1) %>%
  select(-Var1)

# rename variables
df <- df %>%
  rename(to = Var2, Residual = df_resid.value)

# create residual category
df$resid_cat <- ifelse(df$Residual > 2, "High",
                ifelse(df$Residual < -2, "Low",        "Expected"))

# create y axis facets
df$facet <- ifelse(grepl("acknowledge", df$measure), "acknowledge",
            ifelse(grepl("reward", df$measure), "reward",
            ifelse(grepl("promote", df$measure), "promote",
            NA)))
```

```
# create labels and order factors
df$resid_cat <-
  ordered(df$resid_cat,
          levels = c("High",
                     "Expected",
                     "Low"),
          labels = c("High",
                     "Expected",
                     "Low"))

df$to <-
  factor(df$to,
         levels = c("staff",
                    "manager",
                    "ceo"),
         labels = c(label_variable("Frontline Staff", data$to_staff),
                    label_variable("Team/Unit/Department",␣
 ↪data$to_teamunitdept),
                    label_variable("Senior management", data$to_ceo)))

df$facet <-
  factor(df$facet,
         levels = c("acknowledge",
                    "reward",
                    "promote"),
         labels = c(label_variable("Acknowledging", data$project_acknowledge),
                    label_variable("Rewarding", data$project_reward),
                    label_variable("Promoting", data$project_promote)))

df$measure <-
  factor(df$measure,
         levels = c(
           "project_acknowledge_thankthem",
           "project_acknowledge_thankyou",

           "project_reward_material",
           "project_reward_ccd",
           "project_reward_passon",

           "project_promote_suggestion",
           "project_promote_nhs",
           "project_promote_commend"
           ),
         labels = c(
           label_variable("I/we thank them",␣
 ↪data$project_acknowledge_thankthem),
           label_variable("Thank you", data$project_acknowledge_thankyou),
```

```
          label_variable("Material gift", data$project_reward_material),
          label_variable("Someone cc'd", data$project_reward_ccd),
          label_variable("Please thank them", data$project_reward_passon),

          label_variable("Suggestion", data$project_promote_suggestion),
          label_variable("NHS future", data$project_promote_nhs),
          label_variable("Commend behaviour", data$project_promote_commend)
          ))

# plot Figure 2 (reusing 'chi_bar_plot' function)
chi_bar_plot(df)
```



4.4 Total with gratitude aim

[24]:
```
# total n and % of compliments with a gratitude aim
project <-
  ifelse((data$project_acknowledge + data$project_reward +
 ↪data$project_promote) > 0, 1, 0)

sum(project)
round((sum(project)/1267)*100, 3)
```

1227

96.843

Gillespie A, Reader TW. *BMJ Qual Saf* 2021; 30:484–492. doi: 10.1136/bmjqs-2019-010077