

# NeuroMex Analysis

Hayden Mountcastle

2025-03-11

## Contents

<b>Purpose</b>	<b>1</b>
<b>Libraries</b>	<b>1</b>
<b>Import data</b>	<b>1</b>
<b>C. Cleaning Data</b>	<b>3</b>
Removing participants . . . . .	3
Demographics . . . . .	4
LEC . . . . .	10
CIDI . . . . .	13
ASSIST . . . . .	22
<b>Filter data to cases</b>	<b>25</b>
<b>Tables</b>	<b>25</b>
<b>Table of LEC x CIDI</b>	<b>38</b>
<b>Models and Figures</b>	<b>47</b>
Regression - update 1-13-25 (current) . . . . .	55
Model: Poisson (old) . . . . .	68
Model Poission 1-13-25 (old) . . . . .	70
NB GLM 01-27-2025 (current) . . . . .	71
<b>Current models with 4+ traumas</b>	<b>76</b>
Regressions 4+ traumas . . . . .	76

## Purpose

The purpose of this document is to run analysis for the NeuroMEX CIDI/LEC paper. The main aim is to determine whether exposure to interpersonal and/or noninterpersonal traumas impact chronic condition outcomes.

We use NeuroMex Freeze 6 data, located in the Koenen Lab dropbox: /Dropbox (Harvard University)/NeuroMex/6. Data Freezes/Data Freeze 6

The latest models and figures start at: **Regression update 1-13-25**, ignore previous. The latest model for chronic count data is **NB GLM 01-27-2025 (current)**, ignore previous.

## Libraries

```
library(tidyverse)
library(ggeasy)
library(table1)
library(confintr)
library(lme4)
library(gtsummary)
library(kableExtra)
library(sandwich)
library(lmtest)
library(AER)
library(MASS)
```

## Import data

We are using the latest NeuroMex Datafreeze 6 – final datafreeze released March 11th, 2025.

```
setwd("/Users/ham593/Dropbox (Harvard University)/NeuroMex/6. Data Freezes/Data Freeze 6")
```

```
nmex <- read.csv("~/Dropbox (Harvard University)/NeuroMex/6. Data Freezes/Data Freeze 6/NeuroMex_Release")
```

## C. Cleaning Data

### Removing participants

Check for participants missing the UBACC

```
no_ubacc <- nmex %>% filter(is.na(ubacc_score_t1) & is.na(ubacc_score_t2)) %>%  
  dplyr::select(partid) %>% pull()  
  
no_ubacc
```

```
## [1] "CAP5649381" "CAP9216055" "GEA7332459" "GEA9480819" "GEA9495241"  
## [6] "INP8439498" "QUP2253899" "QUP5901822" "QUP6859222"
```

Check for missing in CIDI

At least 50% missingness in CIDI

```
cidi_questions <- paste0("cidi_q", 1:18)  
  
# Filter participants who are missing all or at least 50% of the CIDI questions  
no_cidi <- nmex %>%  
  rowwise() %>%  
  mutate(missing_cidi_count = sum(is.na(c_across(all_of(cidi_questions))))) %>%  
  ungroup() %>%  
  filter( missing_cidi_count >= length(cidi_questions) / 2) %>%  
  dplyr::select(partid) %>% pull()  
  
no_cidi
```

```
## [1] "CAP5649381" "CAP9216055" "GEA7332459" "INP1389432" "INP1582155"  
## [6] "INP2366191" "INP4404950" "JAP1320792" "QUP2253899" "QUP5901822"  
## [11] "QUP6859222"
```

We remove participants who did not complete UBACC or left the CIDI prematurely.

```
#These participants are missing the UBACC, did not consent.  
nmex <- nmex %>% filter(!(partid %in%  
  no_ubacc))  
  
#These participants are missing all or 50% of the CIDI,  
#will exclude for simplicity  
  
nmex <- nmex %>% filter(!(partid %in% no_cidi))
```

## Demographics

### Cleaning basic variables

```
#Factorize is_case and is_male

nmex$is_case_f <- factor(nmex$is_case)
nmex$is_case_f <- fct_recode(nmex$is_case_f, "Control" = "0", "Case" = "1")
nmex$is_case_f <- factor(nmex$is_case_f, c("Case", "Control"))

kable(table(nmex$is_case_f), col.names = c("Case Status", "N"), caption = "Frequency of Case Categories")
```

### Case status

Table 1: Frequency of Case Categories

Case Status	N
Case	4681
Control	4622

```
nmex$is_male_f <- factor(nmex$is_male)
nmex$is_male_f <- fct_recode(nmex$is_male_f, "Male" = "1", "Female" = "0")
nmex$is_male_f <- factor(nmex$is_male_f, c("Male", "Female"))

kable(table(nmex$is_male_f), col.names = c("Sex", "N"), caption = "Frequency of Case Categories")
```

### Sex

Table 2: Frequency of Case Categories

Sex	N
Male	4561
Female	4742

**SES status** In our 12-11-24 meeting, we decided to use Low-Middle Low and Middle-High classifications.

```
nmex <- nmex %>%
  mutate(ses_status_f = factor(case_when(
    ses_status %in% c(1,2) ~ "Low-Middle Low",
    ses_status %in% c(3,4, 5) ~ "Middle-High",
    TRUE ~ NA_character_
  )), levels = c("Low-Middle Low", "Middle-High"))

nmex <- nmex %>%
  mutate(ses_status_opt2_f = factor(case_when(
```

```

ses_status == 1 ~ "Low",
ses_status == 2 ~ "Middle Low",
ses_status %in% c(3,4, 5) ~ "Middle-High",
TRUE ~ NA_character_
), levels = c("Low", "Middle Low", "Middle-High"))

kable(table(nmex$ses_status), col.names = c("SES", "N"), caption = "Frequency of SES Status")

```

Table 3: Frequency of SES Status

SES	N
1	2288
2	2848
3	520
4	3418
5	56
777	158
888	14

```

kable(table(nmex$ses_status_f), col.names = c("SES Recoded", "N"), caption = "Frequency of SES Status Recoded")

```

Table 4: Frequency of SES Status Recoded

SES Recoded	N
Low-Middle Low	5136
Middle-High	3994

```

#Age Categorical

nmex <- nmex %>% mutate(age_cat = case_when(

  age_at_interview < 30 ~ "18-29",
  age_at_interview >= 30 & age_at_interview < 40 ~ "30-39",
  age_at_interview >= 40 & age_at_interview < 50 ~ "40-49",
  age_at_interview >= 50 & age_at_interview < 60 ~ "50-59",
  age_at_interview >= 60 ~ "60+"

))

kable(table(nmex$age_cat), col.names = c("Age", "N"), caption = "Frequency of Age Cat.")

```

Age

Table 5: Frequency of Age Cat.

Age	N
18-29	2079
30-39	2204
40-49	2247
50-59	1913
60+	860

**Location** In our 12-11-24 meeting, we decided to use urbanicity (site\_location\_group\_f)

```
#Factorize site location

nmex$site_location_f <- factor(nmex$site_location)
nmex$site_location_f <- fct_recode(nmex$site_location_f,
  "CDMX" = "1",
  "Campeche" = "2",
  "Queretaro" = "3",
  "Leon" = "4",
  "Guadalajara" = "5",
  "Morelia" = "6")

kable(table(nmex$site_location_f), col.names = c("Site", "N"), caption = "Frequency of Site")
```

Table 6: Frequency of Site

Site	N
CDMX	5510
Campeche	132
Queretaro	608
Leon	392
Guadalajara	1931
Morelia	730

```
#Factorize site location

nmex <- nmex %>% mutate(
  site_location_group_f = case_when(
    site_location %in% c(1,3,5) ~ "Urban",
    site_location %in% c(2,4,6) ~ "Rural"
  ))

kable(table(nmex$site_location_group_f), col.names = c("Urbanicity (Site Recoded)", "N"), caption = "Fr
```

Table 7: Frequency of Urbanicity (Site Recoded)

Urbanicity (Site Recoded)	N
Rural	1254
Urban	8049

```
nmex <- nmex %>% mutate(edu_ord = case_when(
  education_gradepassed == 0 ~ "None",
  education_gradepassed == 8 ~ "Literate",
  education_gradepassed %in% c(1,2) ~ "Elementary or middle schol",
  education_gradepassed %in% c(3,4) ~ "High school or technical school",
  education_gradepassed %in% c(5,6,7) ~ "Post secondary school",
  TRUE ~ NA_character_
))

kable(table(nmex$education_gradepassed), col.names = c("Education", "N"), caption = "Frequency of Education")
```

## Education

Table 8: Frequency of Education

Education	N
0	112
1	949
2	2075
3	2177
4	839
5	2336
6	450
7	98
8	267

```
kable(table(nmex$edu_ord), col.names = c("Education (Recoded)", "N"), caption = "Frequency of Education Recoded")
```

Table 9: Frequency of Education Recoded

Education (Recoded)	N
Elementary or middle schol	3024
High school or technical school	3016
Literate	267
None	112
Post secondary school	2884

```
nmex <- nmex %>% mutate(liv_arrange_ord = case_when(
  liv_arrange == 1 ~ "Lives alone",
  liv_arrange %in% c(2, 4) ~ "Lives with parents or relatives",
  liv_arrange == 3 ~ "Lives with nuclear family",
  liv_arrange == 5 ~ "Lives with friends",
  liv_arrange %in% c(6,7,999) ~ "Other",
  liv_arrange == 888 ~ "Does not want to respond",
  TRUE ~ NA_character_
))
```

```
))
```

```
kable(table(nmex$liv_arrange), col.names = c("Living Arrangement", "N"), caption = "Frequency of Living
```

## Living arrangements

Table 10: Frequency of Living Arrangement

Living Arrangement	N
1	828
2	4345
3	3359
4	293
5	161
6	45
7	174
888	46
999	52

```
kable(table(nmex$liv_arrange_ord), col.names = c("Living Arrangement (Recoded)", "N"), caption = "Freque
```

Table 11: Frequency of Living Arrangement Recoded

Living Arrangement (Recoded)	N
Does not want to respond	46
Lives alone	828
Lives with friends	161
Lives with nuclear family	3359
Lives with parents or relatives	4638
Other	271

```
nmex <- nmex %>% mutate(civil_status_ord = case_when(
  civil_status == 1 ~ "Single",
  civil_status %in% c(2,3) ~ "Married/Open Union",
  civil_status == 4 ~ "Divorced/Separated",
  civil_status == 5 ~ "Widowed",
  civil_status %in% c(777,888) ~ "Does not know/Does not want to respond",
  TRUE ~ NA_character_
))
```

```
kable(table(nmex$civil_status), col.names = c("Civil status", "N"), caption = "Frequency of Civil Status
```

## Civil status



Table 12: Frequency of Civil Status

Civil status	N
1	5287
2	2152
3	808
4	836
5	219
999	1

```
kable(table(nmex$civil_status_ord), col.names = c("Civil status (Recoded)", "N"), caption = "Frequency of Civil Status Recoded")
```

Table 13: Frequency of Civil Status Recoded

Civil status (Recoded)	N
Divorced/Separated	836
Married/Open Union	2960
Single	5287
Widowed	219

## LEC

**Trauma Loads** Here, we code for interpersonal trauma and noninterpersonal trauma. We take the sum of each trauma (binary) for “happened to me only”, which will give a sum of traumas experienced. We also calculate the sum for any trauma (both interpersonal and noninterpersonal trauma), but this is not used in the models.

We ignore violent death and accidental death as these can not “happen to me”.

```
#Interpersonal Trauma Load

nmex <- nmex %>% mutate(
  trauma_load_interpersonal_1 = rowSums(dplyr::select(.,
phys_assault_1,
weapon_assault_1,
sexual_assault_1,
other_sexual_1,
captivity_1,
harm_else_1),
na.rm=TRUE),

#removing human suffering
  trauma_load_noninter_1_nhs = rowSums(dplyr::select(.,
  natural_disaster_1,
  fire_exp_1,
  transp_accident_1,
  serious_accident_1,
  toxic_sub_1,
  combat_1,
  illness_injury_1,
  #human_suffering_1,
  # violent_death_2,
  # accidental_death_2
), na.rm=TRUE),

trauma_load_any_1 = rowSums(dplyr::select(.,
  phys_assault_1,
  weapon_assault_1,
  sexual_assault_1,
  other_sexual_1,
  captivity_1,
  harm_else_1,
  natural_disaster_1,
  fire_exp_1,
  transp_accident_1,
  serious_accident_1,
  toxic_sub_1,
  combat_1,
  illness_injury_1),
na.rm=TRUE))
```

**Trauma categorical** Here we used the previously made sums of trauma to create a categorical variable of traumas experienced.

We calculate for 4+ traumas, but decided we do not have the sample size to do this. In the models we use the 3+ trauma variables.

```
nmex <- nmex %>% mutate(
  trauma_load_inter_cat_4 = factor(case_when(
    trauma_load_interpersonal_1 == 0 ~ "0",
    trauma_load_interpersonal_1 == 1 ~ "1",
    trauma_load_interpersonal_1 == 2 ~ "2",
    trauma_load_interpersonal_1 == 3 ~ "3",
    trauma_load_interpersonal_1 >= 4 ~ "4+",
    TRUE ~ NA_character_
  ), levels=c("0", "1", "2", "3", "4+")),

  trauma_load_noninter_nhs_cat_4 = factor(case_when(
    trauma_load_noninter_1_nhs == 0 ~ "0",
    trauma_load_noninter_1_nhs == 1 ~ "1",
    trauma_load_noninter_1_nhs == 2 ~ "2",
    trauma_load_noninter_1_nhs == 3 ~ "3",
    trauma_load_noninter_1_nhs >= 4 ~ "4+",
    TRUE ~ NA_character_), levels=c("0", "1", "2", "3", "4+"))
)

nmex <- nmex %>% mutate(
  trauma_load_inter_cat = factor(case_when(
    trauma_load_interpersonal_1 == 0 ~ "0",
    trauma_load_interpersonal_1 == 1 ~ "1",
    trauma_load_interpersonal_1 == 2 ~ "2",
    trauma_load_interpersonal_1 >= 3 ~ ">=3",
    TRUE ~ NA_character_
  ), levels=c("0", "1", "2", ">=3")),

  trauma_load_noninter_nhs_cat = factor(case_when(
    trauma_load_noninter_1_nhs == 0 ~ "0",
    trauma_load_noninter_1_nhs == 1 ~ "1",
    trauma_load_noninter_1_nhs == 2 ~ "2",
    trauma_load_noninter_1_nhs >= 3 ~ ">=3",
    TRUE ~ NA_character_), levels=c("0", "1", "2", ">=3"))
)

kable(table(nmex$trauma_load_inter_cat), col.names = c("Trauma Load IT Cat.", "N"), caption = "Frequency")
```

Table 14: Frequency Interpersonal Trauma Load

Trauma Load IT Cat.	N
0	4422
1	2304
2	1513
>=3	1064

```
kable(table(nmex$trauma_load_noninter_nhs_cat), col.names = c("Trauma Load NIT Cat.", "N"), caption = "Table 15: Frequency Noninterpersonal Trauma Load")
```

Table 15: Frequency Noninterpersonal Trauma Load

Trauma Load NIT Cat.	N
0	4315
1	2733
2	1397
$\geq 3$	858

## CIDI

Here we make a binary variable for all CIDI variables (1-18). If a participant responded 0 or 777 (unknown) then they are coded as 0, if they responded 1, then they are coded as 0. If the data is NA, it stays NA.

```
#Clean CIDI variables

#Create a list of CIDI variables 1-18
cidi_q <- paste0("cidi_q", seq(1,18))

#Chronic conditions load

#Make all variables 0,1,NA
for (q in cidi_q) {
  new_col <- paste0(q, "_binary")

  nmex <- nmex %>%
    mutate(!new_col := case_when(
      .data[[q]] %in% c(0,777) ~ 0,
      .data[[q]] == 1 ~ 1,
      TRUE ~ NA_real_
    ))

  tab <- kable(table(nmex[[q]]), col.names = c(q, "N"), caption = paste0("Frequency of ", q))
  tab
  tab_binary <- kable(table(nmex[[new_col]]), col.names = c(q, "N"), caption = paste0("Frequency of Recoded: ", q))
  tab_binary
  print(tab)
  print(tab_binary)
}
```

```
##
##
## Table: Frequency of cidi_q1
##
## |cidi_q1 |    N|
## |:-----|----:|
## |0       | 8767|
## |1       |  482|
## |777     |   54|
##
##
## Table: Frequency of Recoded: cidi_q1_binary
##
## |cidi_q1 |    N|
## |:-----|----:|
## |0       | 8821|
## |1       |  482|
##
##
## Table: Frequency of cidi_q2
##
## |cidi_q2 |    N|
```

```

## |:-----|----:|
## |0       | 7797|
## |1       | 1490|
## |777     |   16|
##
##
## Table: Frequency of Recoded: cidi_q2_binary
##
## |cidi_q2 |    N|
## |:-----|----:|
## |0       | 7813|
## |1       | 1490|
##
##
## Table: Frequency of cidi_q3
##
## |cidi_q3 |    N|
## |:-----|----:|
## |0       | 7683|
## |1       | 1610|
## |777     |   10|
##
##
## Table: Frequency of Recoded: cidi_q3_binary
##
## |cidi_q3 |    N|
## |:-----|----:|
## |0       | 7693|
## |1       | 1610|
##
##
## Table: Frequency of cidi_q4
##
## |cidi_q4 |    N|
## |:-----|----:|
## |0       | 8408|
## |1       |  875|
## |777     |   20|
##
##
## Table: Frequency of Recoded: cidi_q4_binary
##
## |cidi_q4 |    N|
## |:-----|----:|
## |0       | 8428|
## |1       |  875|
##
##
## Table: Frequency of cidi_q5
##
## |cidi_q5 |    N|
## |:-----|----:|
## |0       | 7818|
## |1       | 1458|

```

```

## |777      |    27|
##
##
## Table: Frequency of Recoded: cidi_q5_binary
##
## |cidi_q5 |    N|
## |:-----|----:|
## |0       | 7845|
## |1       | 1458|
##
##
## Table: Frequency of cidi_q6
##
## |cidi_q6 |    N|
## |:-----|----:|
## |0       | 9223|
## |1       |   65|
## |777     |   15|
##
##
## Table: Frequency of Recoded: cidi_q6_binary
##
## |cidi_q6 |    N|
## |:-----|----:|
## |0       | 9238|
## |1       |   65|
##
##
## Table: Frequency of cidi_q7
##
## |cidi_q7 |    N|
## |:-----|----:|
## |0       | 9204|
## |1       |   73|
## |777     |   26|
##
##
## Table: Frequency of Recoded: cidi_q7_binary
##
## |cidi_q7 |    N|
## |:-----|----:|
## |0       | 9230|
## |1       |   73|
##
##
## Table: Frequency of cidi_q8
##
## |cidi_q8 |    N|
## |:-----|----:|
## |0       | 8937|
## |1       |   342|
## |777     |   24|
##
##

```

## Table: Frequency of Recoded: cidi\_q8\_binary

```
##
## |cidi_q8 |    N|
## |:-----|----:|
## |0       | 8961|
## |1       |  342|
```

##

##

## Table: Frequency of cidi\_q9

```
##
## |cidi_q9 |    N|
## |:-----|----:|
## |0       | 7947|
## |1       | 1291|
## |777     |   65|
```

##

##

## Table: Frequency of Recoded: cidi\_q9\_binary

```
##
## |cidi_q9 |    N|
## |:-----|----:|
## |0       | 8012|
## |1       | 1291|
```

##

##

## Table: Frequency of cidi\_q10

```
##
## |cidi_q10 |    N|
## |:-----|----:|
## |0       | 8840|
## |1       |  444|
## |777     |   19|
```

##

##

## Table: Frequency of Recoded: cidi\_q10\_binary

```
##
## |cidi_q10 |    N|
## |:-----|----:|
## |0       | 8859|
## |1       |  444|
```

##

##

## Table: Frequency of cidi\_q11

```
##
## |cidi_q11 |    N|
## |:-----|----:|
## |0       | 9241|
## |1       |   45|
## |777     |   17|
```

##

##

## Table: Frequency of Recoded: cidi\_q11\_binary

```
##
## |cidi_q11 |    N|
```



```

## |:-----|----:|
## |0          | 9258|
## |1          |   45|
##
##
## Table: Frequency of cidi_q12
##
## |cidi_q12 |    N|
## |:-----|----:|
## |0          | 9077|
## |1          |   203|
## |777        |    23|
##
##
## Table: Frequency of Recoded: cidi_q12_binary
##
## |cidi_q12 |    N|
## |:-----|----:|
## |0          | 9100|
## |1          |   203|
##
##
## Table: Frequency of cidi_q13
##
## |cidi_q13 |    N|
## |:-----|----:|
## |0          | 7965|
## |1          | 1263|
## |777        |    75|
##
##
## Table: Frequency of Recoded: cidi_q13_binary
##
## |cidi_q13 |    N|
## |:-----|----:|
## |0          | 8040|
## |1          | 1263|
##
##
## Table: Frequency of cidi_q14
##
## |cidi_q14 |    N|
## |:-----|----:|
## |0          | 6370|
## |1          | 2905|
## |777        |    28|
##
##
## Table: Frequency of Recoded: cidi_q14_binary
##
## |cidi_q14 |    N|
## |:-----|----:|
## |0          | 6398|
## |1          | 2905|

```

```

##
##
## Table: Frequency of cidi_q15
##
## |cidi_q15 |    N|
## |:-----|----:|
## |0        | 9198|
## |1        |   53|
## |777      |   52|
##
##
## Table: Frequency of Recoded: cidi_q15_binary
##
## |cidi_q15 |    N|
## |:-----|----:|
## |0        | 9250|
## |1        |   53|
##
##
## Table: Frequency of cidi_q16
##
## |cidi_q16 |    N|
## |:-----|----:|
## |0        | 8979|
## |1        |   298|
## |777      |   26|
##
##
## Table: Frequency of Recoded: cidi_q16_binary
##
## |cidi_q16 |    N|
## |:-----|----:|
## |0        | 9005|
## |1        |   298|
##
##
## Table: Frequency of cidi_q17
##
## |cidi_q17 |    N|
## |:-----|----:|
## |0        | 9135|
## |1        |   146|
## |777      |   22|
##
##
## Table: Frequency of Recoded: cidi_q17_binary
##
## |cidi_q17 |    N|
## |:-----|----:|
## |0        | 9157|
## |1        |   146|
##
##
## Table: Frequency of cidi_q18

```

```
##
## |cidi_q18 |      N|
## |:-----|----:|
## |0        | 8562|
## |1        |  697|
## |777      |   44|
##
##
## Table: Frequency of Recoded: cidi_q18_binary
##
## |cidi_q18 |      N|
## |:-----|----:|
## |0        | 8606|
## |1        |  697|
```

```
nmex <- nmex %>% mutate(cidi_any_binary = case_when(
  (cidi_q1_binary == 1 |
   cidi_q2_binary == 1 |
   cidi_q3_binary == 1 |
   cidi_q4_binary == 1 |
   cidi_q5_binary == 1 |
   cidi_q6_binary == 1 |
   cidi_q7_binary == 1 |
   cidi_q8_binary == 1 |
   cidi_q9_binary == 1 |
   cidi_q10_binary == 1 |
   cidi_q11_binary == 1 |
   cidi_q12_binary == 1 |
   cidi_q13_binary == 1 |
   cidi_q14_binary == 1 |
   #cidi_q15_binary == 1 / Exclude HIV
   cidi_q16_binary == 1 |
   #cidi_q17_binary == 1 / Exclude Cancer
   cidi_q18_binary == 1) ~ 1,

  (cidi_q1_binary %in% c(0,777) |
   cidi_q2_binary %in% c(0,777) |
   cidi_q3_binary %in% c(0,777) |
   cidi_q4_binary %in% c(0,777) |
   cidi_q5_binary %in% c(0,777) |
   cidi_q6_binary %in% c(0,777) |
   cidi_q7_binary %in% c(0,777) |
   cidi_q8_binary %in% c(0,777) |
   cidi_q9_binary %in% c(0,777) |
   cidi_q10_binary %in% c(0,777) |
   cidi_q11_binary %in% c(0,777) |
   cidi_q12_binary %in% c(0,777) |
   cidi_q13_binary %in% c(0,777) |
   cidi_q14_binary %in% c(0,777) |
   cidi_q15_binary %in% c(0,777) |
   cidi_q16_binary %in% c(0,777) |
   cidi_q17_binary %in% c(0,777) |
   cidi_q18_binary %in% c(0,777)) ~ 0,
```

```
TRUE ~ NA_real_
))
```

**CIDI Load** Here we sum all CIDI binary variables to create a count of chronic conditions, similar to above with trauma.

We removed HIV (cidi\_q15) and cancer (cidi\_q17) after discussion on 1-8-25. This was to maintain consistency with the other models.

```
#Add all "binary" variables together for a total sum of cidi conditions.
nmex <- nmex %>% mutate(
  cidi_load = rowSums(dplyr::select(.,
cidi_q1_binary, cidi_q2_binary, cidi_q3_binary, cidi_q4_binary,
cidi_q5_binary, cidi_q6_binary, cidi_q7_binary, cidi_q8_binary,
cidi_q9_binary, cidi_q10_binary, cidi_q11_binary, cidi_q12_binary,
cidi_q13_binary, cidi_q14_binary,
#cidi_q15_binary,
cidi_q16_binary,
#cidi_q17_binary,
cidi_q18_binary
), na.rm=TRUE))
```

## CIDI categorical

Similar to above, we create a categorical variable of 0 to 4+ CIDI experienced.

```
#Create variable for 0,1,2,3, or 4+ traumas
nmex <- nmex %>% mutate(
  number_cidi = case_when(
    cidi_load == 0 ~ "0", #If have experienced at least one cidi
    cidi_load == 1 ~ "1",
    cidi_load == 2 ~ "2",
    cidi_load == 3 ~ "3",
    cidi_load >= 4 ~ "4+",
    TRUE ~ NA_character_
  )
)

kable(table(nmex$number_cidi), col.names = c("CIDI Load Cat.", "N"), caption = "Frequency CIDI Loads")
```

Table 16: Frequency CIDI Loads

CIDI Load Cat.	N
0	2913
1	2658
2	1820
3	984
4+	928

**Grouped Chronic conditions** Here we create sums of specific CIDI variables in the following classifications: pain, cardiometabolic, respiratory, neurological,

```
####ALED###
# Create pain load and associated binary variable
nmex <- nmex %>% mutate(
  cidi_pain_load = rowSums(dplyr::select(., cidi_q1_binary,
    cidi_q2_binary, cidi_q3_binary, cidi_q4_binary), na.rm=TRUE)) %>%
  mutate(cidi_pain_binary = ifelse(cidi_pain_load>0, 1, 0))

# Create cardiac load and associated binary variable
nmex <- nmex %>% mutate(
  cidi_cardiac_load = rowSums(dplyr::select(., cidi_q7_binary,
    cidi_q8_binary, cidi_q9_binary, cidi_q13_binary), na.rm=TRUE)) %>%
  mutate(cidi_cardiac_binary = ifelse(cidi_cardiac_load>0, 1, 0))

# Create respiratory load and associated binary variable
nmex <- nmex %>% mutate(
  cidi_resp_load = rowSums(dplyr::select(.,
    cidi_q10_binary, cidi_q11_binary, cidi_q12_binary),
    na.rm=TRUE)) %>%
  mutate(cidi_resp_binary = ifelse(cidi_resp_load>0, 1, 0))

# Create neurological load and associated binary variable
nmex <- nmex %>% mutate(
  cidi_neur_load = rowSums(dplyr::select(.,
    cidi_q6_binary, cidi_q16_binary),
    na.rm=TRUE)) %>%
  mutate(cidi_neur_binary = ifelse(cidi_neur_load>0, 1, 0))

# Create overall CIDI binary variable (any condition)
nmex <- nmex %>% mutate(cidi_binary = ifelse(cidi_load>0, 1, 0))
```

## ASSIST

Here we recode alcohol, tobacco use, and cannabis use as Never/Once or twice”, “Monthly/Weekly”, and “Daily”. We decided to do this to allow for more granularity (relative to a binary “Ever”/“Never”).

```
nmex <- nmex %>% mutate(assist_alcohol_amt_recode =  
  case_when(  
  
    (assist_alcohol == 0 | (assist_alcohol == 1 &  
      assist_alcohol_amt %in% c(1,2))) ~ "Never/Once or twice",  
  
    (assist_alcohol == 1 &  
      assist_alcohol_amt %in% c(3,4)) ~ "Monthly/Weekly",  
  
    (assist_alcohol == 1 &  
      assist_alcohol_amt == 5) ~ "Daily",  
    TRUE ~ NA_character_),  
  
  assist_tobacco_amt_recode =  
    case_when(  
  
      assist_tobacco == 0 | (assist_tobacco == 1 &  
        assist_tobacco_amt %in% c(1,2)) ~ "Never/Once or twice",  
  
      (assist_tobacco == 1 &  
        assist_tobacco_amt %in% c(3,4)) ~ "Monthly/Weekly",  
  
      (assist_tobacco == 1 &  
        assist_tobacco_amt == 5) ~ "Daily",  
  
      TRUE ~ NA_character_),  
  
  assist_cannabis_amt_recode =  
    case_when(  
  
      assist_cannabis == 0 | (assist_cannabis == 1 &  
        assist_cannabis_amt %in% c(1,2)) ~ "Never/Once or twice",  
  
      (assist_cannabis == 1 &  
        assist_cannabis_amt %in% c(3,4)) ~ "Monthly/Weekly",  
  
      (assist_cannabis == 1 &  
        assist_cannabis_amt == 5) ~ "Daily",  
  
      TRUE ~ NA_character_))  
  
kable(table(nmex$assist_alcohol), col.names = c("Alcohol", "N"), caption = "Frequency Ever use alcohol")
```

Table 17: Frequency Ever use alcohol

Alcohol	N
0	1262
1	8038

```
kable(table(nmex$assist_alcohol_amt), col.names = c("Alcohol AMT", "N"), caption = "Frequency alcohol in
```

Table 18: Frequency alcohol in past 3 months

Alcohol AMT	N
1	3823
2	1808
3	1169
4	1161
5	77

```
kable(table(nmex$assist_alcohol_amt_recode), col.names = c("Alcohol Recode", "N"), caption = "Frequency
```

Table 19: Frequency Alcohol Recoded

Alcohol Recode	N
Daily	77
Monthly/Weekly	2330
Never/Once or twice	6893

```
kable(table(nmex$assist_tobacco), col.names = c("Tobacco", "N"), caption = "Frequency Ever use tobacco"
```

Table 20: Frequency Ever use tobacco

Tobacco	N
0	3009
1	6291

```
kable(table(nmex$assist_tobacco_amt), col.names = c("Tobacco AMT", "N"), caption = "Frequency of tobacco
```

Table 21: Frequency of tobacco in past 3 months

Tobacco AMT	N
1	3449
2	533
3	221
4	426
5	1662

```
kable(table(nmex$assist_tobacco_amt_recode), col.names = c("Tobacco Recode", "N"), caption = "Frequency
```

Table 22: Frequency Interpersonal Trauma Load

Tobacco Recode	N
Daily	1662
Monthly/Weekly	647
Never/Once or twice	6991

```
kable(table(nmex$assist_cannabis), col.names = c("Cannabis", "N"), caption = "Frequency Ever use cannab
```

Table 23: Frequency Ever use cannabis

Cannabis	N
0	6269
1	3031

```
kable(table(nmex$assist_cannabis_amt), col.names = c("Cannabis AMT", "N"), caption = "Frequency of cann
```

Table 24: Frequency of cannabis in past 3 months

Cannabis AMT	N
1	2426
2	324
3	110
4	97
5	74

```
kable(table(nmex$assist_cannabis_amt_recode), col.names = c("Cannabis Recode", "N"), caption = "Frequen
```

Table 25: Frequency of cannabis Recoded

Cannabis Recode	N
Daily	74
Monthly/Weekly	207
Never/Once or twice	9019



## Filter data to cases

Here we filter the data to only cases. This is what we will use for the analysis.

```
nmex_cases <- nmex %>% filter(is_case == 1)
```

## Tables

### Demographics

Here we produce Table 1: Demographics

```
label(nmex_cases$age_at_interview) <- "Age at interview"
label(nmex_cases$age_cat) <- "Age Categories"
nmex_cases$psychosis_primary_f <- as.factor(nmex_cases$psychosis_primary)
label(nmex_cases$psychosis_primary_f) <- "Primary Psychosis"
label(nmex_cases$site_location_f) <- "Sites"
label(nmex_cases$site_location_group_f) <- "Urbanicity"
label(nmex_cases$ses_status_f) <- "SES"
label(nmex_cases$edu_ord) <- "Education"

table1(~ age_at_interview + age_cat + psychosis_primary_f +
        site_location_f + site_location_group_f + ses_status_f + factor(ses_status) + edu_ord
      | is_male_f, data= nmex_cases,
      caption = "Table 1: Demographics")
```

```
nmex_cases %>% filter(!is.na(diagnosis_other)) %>%
table1(~ factor(diagnosis_other_cat_1) +
        factor(diagnosis_other_cat_2) +
        factor(diagnosis_other_cat_3) +
        factor(diagnosis_other_cat_4) +
        factor(diagnosis_other_cat_5) +
        factor(diagnosis_other_cat_6) +
        factor(diagnosis_other_cat_7) +
        factor(diagnosis_other_cat_8) +
        factor(diagnosis_other_cat_9) +
        factor(diagnosis_other_cat_10) +
        factor(diagnosis_other_cat_11) +
        factor(diagnosis_other_cat_12) +
        factor(diagnosis_other_cat_13) +
        factor(diagnosis_other_cat_999)
      | diagnosis_other, data= .,
      caption = "Table 1: Demographics")
```

```
## Warning in table1.formula(~factor(diagnosis_other_cat_1) +
## factor(diagnosis_other_cat_2) + : Terms to the right of '|' in formula 'x'
## define table columns and are expected to be factors with meaningful labels.
```

Table 26: Table 1: Demographics

	Male	Female	Overall
	(N=2412)	(N=2269)	(N=4681)
<b>Age at interview</b>			
Mean (SD)	40.4 (12.4)	43.7 (12.6)	42.0 (12.6)
Median [Min, Max]	40.0 [18.0, 84.0]	44.0 [18.0, 82.0]	42.0 [18.0, 84.0]
<b>Age Categories</b>			
18-29	561 (23.3%)	353 (15.6%)	914 (19.5%)
30-39	622 (25.8%)	511 (22.5%)	1133 (24.2%)
40-49	593 (24.6%)	620 (27.3%)	1213 (25.9%)
50-59	487 (20.2%)	563 (24.8%)	1050 (22.4%)
60+	149 (6.2%)	222 (9.8%)	371 (7.9%)
<b>Primary Psychosis</b>			
1	408 (16.9%)	937 (41.3%)	1345 (28.7%)
2	98 (4.1%)	197 (8.7%)	295 (6.3%)
3	52 (2.2%)	90 (4.0%)	142 (3.0%)
4	1777 (73.7%)	930 (41.0%)	2707 (57.8%)
5	53 (2.2%)	81 (3.6%)	134 (2.9%)
7	24 (1.0%)	34 (1.5%)	58 (1.2%)
<b>Sites</b>			
CDMX	1334 (55.3%)	1375 (60.6%)	2709 (57.9%)
Campeche	28 (1.2%)	38 (1.7%)	66 (1.4%)
Queretaro	176 (7.3%)	150 (6.6%)	326 (7.0%)
Leon	103 (4.3%)	111 (4.9%)	214 (4.6%)
Guadalajara	529 (21.9%)	345 (15.2%)	874 (18.7%)
Morelia	242 (10.0%)	250 (11.0%)	492 (10.5%)
<b>Urbanicity</b>			
Rural	373 (15.5%)	399 (17.6%)	772 (16.5%)
Urban	2039 (84.5%)	1870 (82.4%)	3909 (83.5%)
<b>SES</b>			
Low-Middle Low	1636 (67.8%)	1461 (64.4%)	3097 (66.2%)
Middle-High	745 (30.9%)	767 (33.8%)	1512 (32.3%)
Missing	31 (1.3%)	41 (1.8%)	72 (1.5%)
<b>factor(ses_status)</b>			
1	865 (35.9%)	772 (34.0%)	1637 (35.0%)
2	771 (32.0%)	689 (30.4%)	1460 (31.2%)
3	75 (3.1%)	77 (3.4%)	152 (3.2%)
4	652 (27.0%)	678 (29.9%)	1330 (28.4%)
5	18 (0.7%)	12 (0.5%)	30 (0.6%)
777	26 (1.1%)	38 (1.7%)	64 (1.4%)
888	4 (0.2%)	3 (0.1%)	7 (0.1%)
Missing	1 (0.0%)	0 (0%)	1 (0.0%)
<b>Education</b>			
Elementary or middle schol	1156 (47.9%)	911 (40.1%)	2067 (44.2%)
High school or technical school	731 (30.3%)	693 (30.5%)	1424 (30.4%)
Literate	111 (4.6%)	101 (4.5%)	212 (4.5%)
None	31 (1.3%)	58 (2.6%)	89 (1.9%)
Post secondary school	383 (15.9%)	506 (22.3%)	889 (19.0%)

Table 27: Table 1: Demographics

	0	1	777	Overall
	(N=3973)	(N=697)	(N=10)	(N=4680)
<b>factor(diagnosis_other_cat_1)</b>				
0	0 (0%)	531 (76.2%)	0 (0%)	531 (11.3%)
1	0 (0%)	166 (23.8%)	0 (0%)	166 (3.5%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_2)</b>				
0	0 (0%)	566 (81.2%)	0 (0%)	566 (12.1%)
1	0 (0%)	131 (18.8%)	0 (0%)	131 (2.8%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_3)</b>				
0	0 (0%)	610 (87.5%)	0 (0%)	610 (13.0%)
1	0 (0%)	87 (12.5%)	0 (0%)	87 (1.9%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_4)</b>				
0	0 (0%)	670 (96.1%)	0 (0%)	670 (14.3%)
1	0 (0%)	27 (3.9%)	0 (0%)	27 (0.6%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_5)</b>				
0	0 (0%)	677 (97.1%)	0 (0%)	677 (14.5%)
1	0 (0%)	20 (2.9%)	0 (0%)	20 (0.4%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_6)</b>				
0	0 (0%)	670 (96.1%)	0 (0%)	670 (14.3%)
1	0 (0%)	27 (3.9%)	0 (0%)	27 (0.6%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_7)</b>				
0	0 (0%)	618 (88.7%)	0 (0%)	618 (13.2%)
1	0 (0%)	79 (11.3%)	0 (0%)	79 (1.7%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_8)</b>				
0	0 (0%)	562 (80.6%)	0 (0%)	562 (12.0%)
1	0 (0%)	135 (19.4%)	0 (0%)	135 (2.9%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_9)</b>				
0	0 (0%)	653 (93.7%)	0 (0%)	653 (14.0%)
1	0 (0%)	44 (6.3%)	0 (0%)	44 (0.9%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_10)</b>				
0	0 (0%)	644 (92.4%)	0 (0%)	644 (13.8%)
1	0 (0%)	53 (7.6%)	0 (0%)	53 (1.1%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_11)</b>				
0	0 (0%)	666 (95.6%)	0 (0%)	666 (14.2%)
1	0 (0%)	31 (4.4%)	0 (0%)	31 (0.7%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_12)</b>				
0	0 (0%)	664 (95.3%)	0 (0%)	664 (14.2%)
1	0 (0%)	33 (4.7%)	0 (0%)	33 (0.7%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_13)</b>				
0	0 (0%)	692 (99.3%)	0 (0%)	692 (14.8%)
1	0 (0%)	5 (0.7%)	0 (0%)	5 (0.1%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)

```
nmex_cases %>% filter(!is.na(diagnosis_other)) %>%
table1(~ factor(diagnosis_other_cat_1) +
  factor(diagnosis_other_cat_2) +
  factor(diagnosis_other_cat_3) +
  factor(diagnosis_other_cat_4) +
  factor(diagnosis_other_cat_5) +
  factor(diagnosis_other_cat_6) +
  factor(diagnosis_other_cat_7) +
  factor(diagnosis_other_cat_8) +
  factor(diagnosis_other_cat_9) +
  factor(diagnosis_other_cat_10) +
  factor(diagnosis_other_cat_11) +
  factor(diagnosis_other_cat_12) +
  factor(diagnosis_other_cat_13) +
  factor(diagnosis_other_cat_999)
  | diagnosis_other, data= .,
caption = "Table 1: Demographics")
```

```
## Warning in table1.formula(~factor(diagnosis_other_cat_1) +
## factor(diagnosis_other_cat_2) + : Terms to the right of '|' in formula 'x'
## define table columns and are expected to be factors with meaningful labels.
```

### Interpersonal trauma: CIDI and Substance Use

Here we create a table of outcomes by interpersonal trauma.(NOT SHOWN)

### Noninterpersonal trauma: CIDI and Substance Use

Here we create a table of outcomes by noninterpersonal trauma. (NOT SHOWN)

### Table Substance x Site

```
table1(~
  assist_tobacco_amt_recode +
  assist_alcohol_amt_recode +
  assist_cannabis_amt_recode | site_location_f,
  data= nmex_cases)
```

Table 28: Table 1: Demographics

	0	1	777	Overall
	(N=3973)	(N=697)	(N=10)	(N=4680)
<b>factor(diagnosis_other_cat_1)</b>				
0	0 (0%)	531 (76.2%)	0 (0%)	531 (11.3%)
1	0 (0%)	166 (23.8%)	0 (0%)	166 (3.5%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_2)</b>				
0	0 (0%)	566 (81.2%)	0 (0%)	566 (12.1%)
1	0 (0%)	131 (18.8%)	0 (0%)	131 (2.8%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_3)</b>				
0	0 (0%)	610 (87.5%)	0 (0%)	610 (13.0%)
1	0 (0%)	87 (12.5%)	0 (0%)	87 (1.9%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_4)</b>				
0	0 (0%)	670 (96.1%)	0 (0%)	670 (14.3%)
1	0 (0%)	27 (3.9%)	0 (0%)	27 (0.6%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_5)</b>				
0	0 (0%)	677 (97.1%)	0 (0%)	677 (14.5%)
1	0 (0%)	20 (2.9%)	0 (0%)	20 (0.4%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_6)</b>				
0	0 (0%)	670 (96.1%)	0 (0%)	670 (14.3%)
1	0 (0%)	27 (3.9%)	0 (0%)	27 (0.6%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_7)</b>				
0	0 (0%)	618 (88.7%)	0 (0%)	618 (13.2%)
1	0 (0%)	79 (11.3%)	0 (0%)	79 (1.7%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_8)</b>				
0	0 (0%)	562 (80.6%)	0 (0%)	562 (12.0%)
1	0 (0%)	135 (19.4%)	0 (0%)	135 (2.9%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_9)</b>				
0	0 (0%)	653 (93.7%)	0 (0%)	653 (14.0%)
1	0 (0%)	44 (6.3%)	0 (0%)	44 (0.9%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_10)</b>				
0	0 (0%)	644 (92.4%)	0 (0%)	644 (13.8%)
1	0 (0%)	53 (7.6%)	0 (0%)	53 (1.1%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_11)</b>				
0	0 (0%)	666 (95.6%)	0 (0%)	666 (14.2%)
1	0 (0%)	31 (4.4%)	0 (0%)	31 (0.7%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_12)</b>				
0	0 (0%)	664 (95.3%)	0 (0%)	664 (14.2%)
1	0 (0%)	33 (4.7%)	0 (0%)	33 (0.7%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)
<b>factor(diagnosis_other_cat_13)</b>				
0	0 (0%)	692 (99.3%)	0 (0%)	692 (14.8%)
1	0 (0%)	5 (0.7%)	0 (0%)	5 (0.1%)
Missing	3973 (100%)	0 (0%)	10 (100%)	3983 (85.1%)

	CDMX	Campeche	Queretaro	Leon	Guadalajara	Morelia	Overall
	(N=2709)	(N=66)	(N=326)	(N=214)	(N=874)	(N=492)	(N=4681)
<b>assist_tobacco_amt_recode</b>							
Daily	519 (19.2%)	7 (10.6%)	68 (20.9%)	37 (17.3%)	368 (42.1%)	68 (13.8%)	1067 (22.8%)
Monthly/Weekly	139 (5.1%)	3 (4.5%)	19 (5.8%)	7 (3.3%)	50 (5.7%)	23 (4.7%)	241 (5.1%)
Never/Once or twice	2049 (75.6%)	56 (84.8%)	239 (73.3%)	170 (79.4%)	455 (52.1%)	401 (81.5%)	3370 (72.1%)
Missing	2 (0.1%)	0 (0%)	0 (0%)	0 (0%)	1 (0.1%)	0 (0%)	3 (0.1%)
<b>assist_alcohol_amt_recode</b>							
Daily	19 (0.7%)	1 (1.5%)	9 (2.8%)	0 (0%)	8 (0.9%)	2 (0.4%)	39 (0.8%)
Monthly/Weekly	333 (12.3%)	3 (4.5%)	57 (17.5%)	23 (10.7%)	93 (10.6%)	43 (8.7%)	552 (11.8%)
Never/Once or twice	2355 (86.9%)	62 (93.9%)	260 (79.8%)	191 (89.3%)	772 (88.3%)	447 (90.9%)	4087 (87.3%)
Missing	2 (0.1%)	0 (0%)	0 (0%)	0 (0%)	1 (0.1%)	0 (0%)	3 (0.1%)
<b>assist_cannabis_amt_recode</b>							
Daily	25 (0.9%)	0 (0%)	0 (0%)	0 (0%)	11 (1.3%)	1 (0.2%)	37 (0.8%)
Monthly/Weekly	56 (2.1%)	0 (0%)	7 (2.1%)	0 (0%)	38 (4.3%)	5 (1.0%)	106 (2.3%)
Never/Once or twice	2626 (96.9%)	66 (100%)	319 (97.9%)	214 (100%)	824 (94.3%)	486 (98.8%)	4535 (96.9%)
Missing	2 (0.1%)	0 (0%)	0 (0%)	0 (0%)	1 (0.1%)	0 (0%)	3 (0.1%)

**Table Sex X LEC**

Here we make table 2 which shows sex with interpersonal and non-interpersonal chronic conditions.

```
pvalue <- function(x, ...) {
  # Construct vectors of data y, and groups (strata) g
  y <- unlist(x)
  g <- factor(rep(1:length(x), times=apply(x, length)))
  if (is.numeric(y)) {
    # For numeric variables, perform a standard 2-sample t-test
    p <- t.test(y ~ g)$p.value
  } else {
    # For categorical variables, perform a chi-squared test of independence
    p <- chisq.test(table(y, g))$p.value
  }
  # Format the p-value, using an HTML entity for the less-than sign.
  # The initial empty string places the output on the line below the variable label.
  c("", sub("<", "&lt;", format.pval(p, digits=3, eps=0.001)))
}

table1(~ factor(phys_assault_1) + factor(weapon_assault_1) +
factor(sexual_assault_1) + factor(other_sexual_1) + factor(captivity_1) + factor(harm_else_1) +
factor(natural_disaster_1) + factor(fire_exp_1) +
factor(transp_accident_1) + factor(serious_accident_1) + factor(toxic_sub_1) + factor(combat_1) +
factor(illness_injury_1) +
trauma_load_inter_cat +
trauma_load_noninter_nhs_cat +
trauma_load_inter_cat_4 +
trauma_load_noninter_nhs_cat_4 +
trauma_load_interpersonal_1 +
trauma_load_noninter_1_nhs +
trauma_load_any_1 | is_male_f, data=nmex_cases)
```

	Male	Female	Overall
	(N=2412)	(N=2269)	(N=4681)
<b>factor(phys__assault_1)</b>			
0	1422 (59.0%)	1532 (67.5%)	2954 (63.1%)
1	990 (41.0%)	737 (32.5%)	1727 (36.9%)
<b>factor(weapon__assault_1)</b>			
0	1779 (73.8%)	1919 (84.6%)	3698 (79.0%)
1	633 (26.2%)	350 (15.4%)	983 (21.0%)
<b>factor(sexual__assault_1)</b>			
0	2135 (88.5%)	1621 (71.4%)	3756 (80.2%)
1	277 (11.5%)	648 (28.6%)	925 (19.8%)
<b>factor(other__sexual_1)</b>			
0	2202 (91.3%)	1794 (79.1%)	3996 (85.4%)
1	210 (8.7%)	475 (20.9%)	685 (14.6%)
<b>factor(captivity_1)</b>			
0	2260 (93.7%)	2126 (93.7%)	4386 (93.7%)
1	152 (6.3%)	143 (6.3%)	295 (6.3%)
<b>factor(harm__else_1)</b>			
0	2227 (92.3%)	2099 (92.5%)	4326 (92.4%)
1	185 (7.7%)	170 (7.5%)	355 (7.6%)
<b>factor(natural__disaster_1)</b>			
0	1855 (76.9%)	1667 (73.5%)	3522 (75.2%)
1	557 (23.1%)	602 (26.5%)	1159 (24.8%)
<b>factor(fire__exp_1)</b>			
0	2254 (93.4%)	2135 (94.1%)	4389 (93.8%)
1	158 (6.6%)	134 (5.9%)	292 (6.2%)
<b>factor(transp__accident_1)</b>			
0	1890 (78.4%)	1939 (85.5%)	3829 (81.8%)
1	522 (21.6%)	330 (14.5%)	852 (18.2%)
<b>factor(serious__accident_1)</b>			
0	2050 (85.0%)	2024 (89.2%)	4074 (87.0%)
1	362 (15.0%)	245 (10.8%)	607 (13.0%)
<b>factor(toxic__sub_1)</b>			
0	2258 (93.6%)	2197 (96.8%)	4455 (95.2%)
1	154 (6.4%)	72 (3.2%)	226 (4.8%)
<b>factor(combat_1)</b>			
0	2304 (95.5%)	2188 (96.4%)	4492 (96.0%)
1	108 (4.5%)	81 (3.6%)	189 (4.0%)
<b>factor(illness__injury_1)</b>			
0	2031 (84.2%)	1915 (84.4%)	3946 (84.3%)
1	381 (15.8%)	354 (15.6%)	735 (15.7%)
<b>trauma__load__inter__cat</b>			
0	1060 (43.9%)	1005 (44.3%)	2065 (44.1%)
1	644 (26.7%)	545 (24.0%)	1189 (25.4%)
2	440 (18.2%)	356 (15.7%)	796 (17.0%)
>=3	268 (11.1%)	363 (16.0%)	631 (13.5%)
<b>trauma__load__noninter__nhs__cat</b>			
0	1141 (47.3%)	1145 (50.5%)	2286 (48.8%)
1	694 (28.8%)	646 (28.5%)	1340 (28.6%)
2	325 (13.5%)	321 (14.1%)	646 (13.8%)
>=3	252 (10.4%)	157 (6.9%)	409 (8.7%)
<b>trauma__load__inter__cat_4</b>			
0	1060 (43.9%)	1005 (44.3%)	2065 (44.1%)
1	644 (26.7%)	545 (24.0%)	1189 (25.4%)
2	440 (18.2%)	356 (15.7%)	796 (17.0%)

```

# Getting p values
table1(~ factor(phys_assault_1) + factor(weapon_assault_1) +
factor(sexual_assault_1) + factor(other_sexual_1) + factor(captivity_1) + factor(harm_else_1) +
factor(natural_disaster_1) + factor(fire_exp_1) +
factor(transp_accident_1) + factor(serious_accident_1) + factor(toxic_sub_1) + factor(combat_1) +
factor(illness_injury_1) +
trauma_load_inter_cat +
trauma_load_noninter_nhs_cat +
trauma_load_inter_cat_4 +
trauma_load_noninter_nhs_cat_4 +
trauma_load_interpersonal_1 +
trauma_load_noninter_1_nhs +
trauma_load_any_1 | is_male_f, overall=F, data=nmex_cases,
extra.col=list(`P-value`=pvalue))

```



	Male	Female	P-value
	(N=2412)	(N=2269)	
<b>factor(phys__assault_1)</b>			
0	1422 (59.0%)	1532 (67.5%)	<0.001
1	990 (41.0%)	737 (32.5%)	
<b>factor(weapon__assault_1)</b>			
0	1779 (73.8%)	1919 (84.6%)	<0.001
1	633 (26.2%)	350 (15.4%)	
<b>factor(sexual__assault_1)</b>			
0	2135 (88.5%)	1621 (71.4%)	<0.001
1	277 (11.5%)	648 (28.6%)	
<b>factor(other__sexual_1)</b>			
0	2202 (91.3%)	1794 (79.1%)	<0.001
1	210 (8.7%)	475 (20.9%)	
<b>factor(captivity_1)</b>			
0	2260 (93.7%)	2126 (93.7%)	1
1	152 (6.3%)	143 (6.3%)	
<b>factor(harm__else_1)</b>			
0	2227 (92.3%)	2099 (92.5%)	0.862
1	185 (7.7%)	170 (7.5%)	
<b>factor(natural__disaster_1)</b>			
0	1855 (76.9%)	1667 (73.5%)	0.00714
1	557 (23.1%)	602 (26.5%)	
<b>factor(fire__exp_1)</b>			
0	2254 (93.4%)	2135 (94.1%)	0.395
1	158 (6.6%)	134 (5.9%)	
<b>factor(transp__accident_1)</b>			
0	1890 (78.4%)	1939 (85.5%)	<0.001
1	522 (21.6%)	330 (14.5%)	
<b>factor(serious__accident_1)</b>			
0	2050 (85.0%)	2024 (89.2%)	<0.001
1	362 (15.0%)	245 (10.8%)	
<b>factor(toxic__sub_1)</b>			
0	2258 (93.6%)	2197 (96.8%)	<0.001
1	154 (6.4%)	72 (3.2%)	
<b>factor(combat_1)</b>			
0	2304 (95.5%)	2188 (96.4%)	0.133
1	108 (4.5%)	81 (3.6%)	
<b>factor(illness__injury_1)</b>			
0	2031 (84.2%)	1915 (84.4%)	0.887
1	381 (15.8%)	354 (15.6%)	
<b>trauma__load__inter__cat</b>			
0	1060 (43.9%)	1005 (44.3%)	<0.001
1	644 (26.7%)	545 (24.0%)	
2	440 (18.2%)	356 (15.7%)	
>=3	268 (11.1%)	363 (16.0%)	
<b>trauma__load__noninter__nhs__cat</b>			
0	1141 (47.3%)	1145 (50.5%)	<0.001
1	694 (28.8%)	646 (28.5%)	
2	325 (13.5%)	321 (14.1%)	
>=3	252 (10.4%)	157 (6.9%)	
<b>trauma__load__inter__cat_4</b>			
0	1060 (43.9%)	1005 (44.3%)	<0.001
1	644 (26.7%)	545 (24.0%)	
2	440 (18.2%)	356 (15.7%)	

```
chi_cap <- table(as.factor(nmex_cases$captivity_1), nmex_cases$is_male_f)
chi_cap
```

```
##
##      Male Female
##    0 2260    2126
##    1  152     143
```

```
chisq.test(chi_cap, )
```

```
##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data:  chi_cap
## X-squared = 1.7541e-28, df = 1, p-value = 1
```

### Table of chronic conditions

```
table1(~ factor(cidi_any_binary) + #Any

      factor(cidi_pain_binary) + #Pain
      factor(cidi_q1_binary) + # Arthritis
      factor(cidi_q2_binary) + #Back or neck problems
      factor(cidi_q3_binary) + #Headaches
      factor(cidi_q4_binary) + #Other chronic pain

      factor(cidi_cardiac_binary) + #Cardiometabolic
      factor(cidi_q7_binary) + #Heart attack
      factor(cidi_q8_binary) + #Heart disease
      factor(cidi_q9_binary) + #High blood pressure
      factor(cidi_q13_binary) + #Diabetes

      factor(cidi_resp_binary) + #Respiratory
      factor(cidi_q10_binary) + #Asthma
      factor(cidi_q11_binary) + #Tuberculosis
      factor(cidi_q12_binary) + #Other lung

      factor(cidi_neur_binary) + #Neurological
      factor(cidi_q6_binary) + #Stroke
      factor(cidi_q16_binary) + #Epilepsy

      factor(cidi_q14_binary) + #Ulcer
      factor(cidi_q5_binary) + #Allergies
      factor(cidi_q18_binary) + #Hypothyroidism
      cidi_load +
      number_cidi
| is_male_f, data=nmex_cases)
```

	Male	Female	Overall
	(N=2412)	(N=2269)	(N=4681)
<b>factor(cidi_any_binary)</b>			
0	974 (40.4%)	653 (28.8%)	1627 (34.8%)
1	1438 (59.6%)	1616 (71.2%)	3054 (65.2%)
<b>factor(cidi_pain_binary)</b>			
0	1793 (74.3%)	1429 (63.0%)	3222 (68.8%)
1	619 (25.7%)	840 (37.0%)	1459 (31.2%)
<b>factor(cidi_q1_binary)</b>			
0	2347 (97.3%)	2138 (94.2%)	4485 (95.8%)
1	65 (2.7%)	131 (5.8%)	196 (4.2%)
<b>factor(cidi_q2_binary)</b>			
0	2122 (88.0%)	1903 (83.9%)	4025 (86.0%)
1	290 (12.0%)	366 (16.1%)	656 (14.0%)
<b>factor(cidi_q3_binary)</b>			
0	2109 (87.4%)	1809 (79.7%)	3918 (83.7%)
1	303 (12.6%)	460 (20.3%)	763 (16.3%)
<b>factor(cidi_q4_binary)</b>			
0	2256 (93.5%)	2050 (90.3%)	4306 (92.0%)
1	156 (6.5%)	219 (9.7%)	375 (8.0%)
<b>factor(cidi_cardiac_binary)</b>			
0	1927 (79.9%)	1680 (74.0%)	3607 (77.1%)
1	485 (20.1%)	589 (26.0%)	1074 (22.9%)
<b>factor(cidi_q7_binary)</b>			
0	2399 (99.5%)	2255 (99.4%)	4654 (99.4%)
1	13 (0.5%)	14 (0.6%)	27 (0.6%)
<b>factor(cidi_q8_binary)</b>			
0	2332 (96.7%)	2187 (96.4%)	4519 (96.5%)
1	80 (3.3%)	82 (3.6%)	162 (3.5%)
<b>factor(cidi_q9_binary)</b>			
0	2130 (88.3%)	1939 (85.5%)	4069 (86.9%)
1	282 (11.7%)	330 (14.5%)	612 (13.1%)
<b>factor(cidi_q13_binary)</b>			
0	2200 (91.2%)	1931 (85.1%)	4131 (88.3%)
1	212 (8.8%)	338 (14.9%)	550 (11.7%)
<b>factor(cidi_resp_binary)</b>			
0	2233 (92.6%)	2129 (93.8%)	4362 (93.2%)
1	179 (7.4%)	140 (6.2%)	319 (6.8%)
<b>factor(cidi_q10_binary)</b>			
0	2306 (95.6%)	2184 (96.3%)	4490 (95.9%)
1	106 (4.4%)	85 (3.7%)	191 (4.1%)
<b>factor(cidi_q11_binary)</b>			
0	2397 (99.4%)	2259 (99.6%)	4656 (99.5%)
1	15 (0.6%)	10 (0.4%)	25 (0.5%)
<b>factor(cidi_q12_binary)</b>			
0	2343 (97.1%)	2218 (97.8%)	4561 (97.4%)
1	69 (2.9%)	51 (2.2%)	120 (2.6%)
<b>factor(cidi_neur_binary)</b>			
0	2287 (94.8%)	2160 (95.2%)	4447 (95.0%)
1	125 (5.2%)	109 (4.8%)	234 (5.0%)
<b>factor(cidi_q6_binary)</b>		35	
0	2397 (99.4%)	2253 (99.3%)	4650 (99.3%)
1	15 (0.6%)	16 (0.7%)	31 (0.7%)
<b>factor(cidi_q16_binary)</b>			

```

table1(~ factor(cidi_any_binary) + #Any

      factor(cidi_pain_binary) + #Pain
      factor(cidi_q1_binary) + # Arthritis
      factor(cidi_q2_binary) + #Back or neck problems
      factor(cidi_q3_binary) + #Headaches
      factor(cidi_q4_binary) + #Other chronic pain

      factor(cidi_cardiac_binary) + #Cardiometabolic
      factor(cidi_q7_binary) + #Heart attack
      factor(cidi_q8_binary) + #Heart disease
      factor(cidi_q9_binary) + #High blood pressure
      factor(cidi_q13_binary) + #Diabetes

      factor(cidi_resp_binary) + #Respiratory
      factor(cidi_q10_binary) + #Asthma
      factor(cidi_q11_binary) + #Tuberculosis
      factor(cidi_q12_binary) + #Other lung

      factor(cidi_neur_binary) + #Neurological
      factor(cidi_q6_binary) + #Stroke
      factor(cidi_q16_binary) + #Epilepsy

      factor(cidi_q14_binary) + #Ulcer
      factor(cidi_q5_binary) + #Allergies
      factor(cidi_q18_binary) + #Hypothyroidism
      cidi_load +
      number_cidi
| is_male_f, overall=F, data=nmx_cases,
extra.col=list(`P-value`=pvalue))

```

	Male	Female	P-value
	(N=2412)	(N=2269)	
<b>factor(cidi_any_binary)</b>			
0	974 (40.4%)	653 (28.8%)	<0.001
1	1438 (59.6%)	1616 (71.2%)	
<b>factor(cidi_pain_binary)</b>			
0	1793 (74.3%)	1429 (63.0%)	<0.001
1	619 (25.7%)	840 (37.0%)	
<b>factor(cidi_q1_binary)</b>			
0	2347 (97.3%)	2138 (94.2%)	<0.001
1	65 (2.7%)	131 (5.8%)	
<b>factor(cidi_q2_binary)</b>			
0	2122 (88.0%)	1903 (83.9%)	<0.001
1	290 (12.0%)	366 (16.1%)	
<b>factor(cidi_q3_binary)</b>			
0	2109 (87.4%)	1809 (79.7%)	<0.001
1	303 (12.6%)	460 (20.3%)	
<b>factor(cidi_q4_binary)</b>			
0	2256 (93.5%)	2050 (90.3%)	<0.001
1	156 (6.5%)	219 (9.7%)	
<b>factor(cidi_cardiac_binary)</b>			
0	1927 (79.9%)	1680 (74.0%)	<0.001
1	485 (20.1%)	589 (26.0%)	
<b>factor(cidi_q7_binary)</b>			
0	2399 (99.5%)	2255 (99.4%)	0.873
1	13 (0.5%)	14 (0.6%)	
<b>factor(cidi_q8_binary)</b>			
0	2332 (96.7%)	2187 (96.4%)	0.634
1	80 (3.3%)	82 (3.6%)	
<b>factor(cidi_q9_binary)</b>			
0	2130 (88.3%)	1939 (85.5%)	0.00438
1	282 (11.7%)	330 (14.5%)	
<b>factor(cidi_q13_binary)</b>			
0	2200 (91.2%)	1931 (85.1%)	<0.001
1	212 (8.8%)	338 (14.9%)	
<b>factor(cidi_resp_binary)</b>			
0	2233 (92.6%)	2129 (93.8%)	0.101
1	179 (7.4%)	140 (6.2%)	
<b>factor(cidi_q10_binary)</b>			
0	2306 (95.6%)	2184 (96.3%)	0.295
1	106 (4.4%)	85 (3.7%)	
<b>factor(cidi_q11_binary)</b>			
0	2397 (99.4%)	2259 (99.6%)	0.516
1	15 (0.6%)	10 (0.4%)	
<b>factor(cidi_q12_binary)</b>			
0	2343 (97.1%)	2218 (97.8%)	0.217
1	69 (2.9%)	51 (2.2%)	
<b>factor(cidi_neur_binary)</b>			
0	2287 (94.8%)	2160 (95.2%)	0.598
1	125 (5.2%)	109 (4.8%)	
<b>factor(cidi_q6_binary)</b>		37	
0	2397 (99.4%)	2253 (99.3%)	0.864
1	15 (0.6%)	16 (0.7%)	
<b>factor(cidi_q16_binary)</b>			

## Table of LEC x CIDI

```
table1(~ factor(cidi_any_binary) + #Any

      factor(cidi_pain_binary) + #Pain
      factor(cidi_q1_binary) + # Arthritis
      factor(cidi_q2_binary) + #Back or neck problems
      factor(cidi_q3_binary) + #Headaches
      factor(cidi_q4_binary) + #Other chronic pain

      factor(cidi_cardiac_binary) + #Cardiometabolic
      factor(cidi_q7_binary) + #Heart attack
      factor(cidi_q8_binary) + #Heart disease
      factor(cidi_q9_binary) + #High blood pressure
      factor(cidi_q13_binary) + #Diabetes

      factor(cidi_resp_binary) + #Respiratory
      factor(cidi_q10_binary) + #Asthma
      factor(cidi_q11_binary) + #Tuberculosis
      factor(cidi_q12_binary) + #Other lung

      factor(cidi_neur_binary) + #Neurological
      factor(cidi_q6_binary) + #Stroke
      factor(cidi_q16_binary) + #Epilepsy

      factor(cidi_q14_binary) + #Ulcer
      factor(cidi_q5_binary) + #Allergies
      factor(cidi_q18_binary) + #Hypothyroidism
| trauma_load_inter_cat, overall=F, data=nmex_cases)
```

	0	1	2	>=3
	(N=2065)	(N=1189)	(N=796)	(N=631)
<b>factor(cidi_any_binary)</b>				
0	795 (38.5%)	430 (36.2%)	234 (29.4%)	168 (26.6%)
1	1270 (61.5%)	759 (63.8%)	562 (70.6%)	463 (73.4%)
<b>factor(cidi_pain_binary)</b>				
0	1515 (73.4%)	844 (71.0%)	503 (63.2%)	360 (57.1%)
1	550 (26.6%)	345 (29.0%)	293 (36.8%)	271 (42.9%)
<b>factor(cidi_q1_binary)</b>				
0	1991 (96.4%)	1152 (96.9%)	755 (94.8%)	587 (93.0%)
1	74 (3.6%)	37 (3.1%)	41 (5.2%)	44 (7.0%)
<b>factor(cidi_q2_binary)</b>				
0	1834 (88.8%)	1046 (88.0%)	656 (82.4%)	489 (77.5%)
1	231 (11.2%)	143 (12.0%)	140 (17.6%)	142 (22.5%)
<b>factor(cidi_q3_binary)</b>				
0	1787 (86.5%)	1000 (84.1%)	643 (80.8%)	488 (77.3%)
1	278 (13.5%)	189 (15.9%)	153 (19.2%)	143 (22.7%)
<b>factor(cidi_q4_binary)</b>				
0	1923 (93.1%)	1111 (93.4%)	720 (90.5%)	552 (87.5%)
1	142 (6.9%)	78 (6.6%)	76 (9.5%)	79 (12.5%)
<b>factor(cidi_cardiac_binary)</b>				
0	1588 (76.9%)	920 (77.4%)	615 (77.3%)	484 (76.7%)
1	477 (23.1%)	269 (22.6%)	181 (22.7%)	147 (23.3%)
<b>factor(cidi_q7_binary)</b>				
0	2056 (99.6%)	1185 (99.7%)	790 (99.2%)	623 (98.7%)
1	9 (0.4%)	4 (0.3%)	6 (0.8%)	8 (1.3%)
<b>factor(cidi_q8_binary)</b>				
0	2002 (96.9%)	1153 (97.0%)	770 (96.7%)	594 (94.1%)
1	63 (3.1%)	36 (3.0%)	26 (3.3%)	37 (5.9%)
<b>factor(cidi_q9_binary)</b>				
0	1788 (86.6%)	1033 (86.9%)	694 (87.2%)	554 (87.8%)
1	277 (13.4%)	156 (13.1%)	102 (12.8%)	77 (12.2%)
<b>factor(cidi_q13_binary)</b>				
0	1801 (87.2%)	1063 (89.4%)	703 (88.3%)	564 (89.4%)
1	264 (12.8%)	126 (10.6%)	93 (11.7%)	67 (10.6%)
<b>factor(cidi_resp_binary)</b>				
0	1936 (93.8%)	1109 (93.3%)	737 (92.6%)	580 (91.9%)
1	129 (6.2%)	80 (6.7%)	59 (7.4%)	51 (8.1%)
<b>factor(cidi_q10_binary)</b>				
0	1993 (96.5%)	1148 (96.6%)	755 (94.8%)	594 (94.1%)
1	72 (3.5%)	41 (3.4%)	41 (5.2%)	37 (5.9%)
<b>factor(cidi_q11_binary)</b>				
0	2055 (99.5%)	1179 (99.2%)	795 (99.9%)	627 (99.4%)
1	10 (0.5%)	10 (0.8%)	1 (0.1%)	4 (0.6%)
<b>factor(cidi_q12_binary)</b>				
0	2013 (97.5%)	1158 (97.4%)	773 (97.1%)	617 (97.8%)
1	52 (2.5%)	31 (2.6%)	23 (2.9%)	14 (2.2%)
<b>factor(cidi_neur_binary)</b>				
0	1980 (95.9%)	1133 (95.3%)	752 (94.5%)	582 (92.2%)
1	85 (4.1%)	56 (4.7%)	44 (5.5%)	49 (7.8%)
<b>factor(cidi_q6_binary)</b>				39
0	2055 (99.5%)	1183 (99.5%)	787 (98.9%)	625 (99.0%)
1	10 (0.5%)	6 (0.5%)	9 (1.1%)	6 (1.0%)
<b>factor(cidi_q16_binary)</b>				

## Final LECxCIDI

These are the outcomes we are interested in in this paper, rather than all CIDI outcomes.

```
nmex_cases$cidi_pain_binary_f <- factor(nmex_cases$cidi_pain_binary)
nmex_cases$cidi_cardiac_binary_f <- factor(nmex_cases$cidi_cardiac_binary)
nmex_cases$cidi_resp_binary_f <- factor(nmex_cases$cidi_resp_binary)
nmex_cases$cidi_resp_binary_f <- factor(nmex_cases$cidi_neur_binary)
nmex_cases$cidi_q14_binary_f <- factor(nmex_cases$cidi_q14_binary)
nmex_cases$cidi_q5_binary_f <- factor(nmex_cases$cidi_q5_binary)
nmex_cases$cidi_q18_binary_f <- factor(nmex_cases$cidi_q18_binary)
nmex_cases$cidi_neur_binary_f <- factor(nmex_cases$cidi_neur_binary)

label(nmex_cases$cidi_pain_binary_f) <- "Pain"
label(nmex_cases$cidi_cardiac_binary_f) <- "Cardiometabolic"
label(nmex_cases$cidi_resp_binary_f) <- "Respiratory"
label(nmex_cases$cidi_neur_binary_f) <- "Neurological"
label(nmex_cases$cidi_q14_binary_f) <- "Ulcer"
label(nmex_cases$cidi_q5_binary_f) <- "Allergies"
label(nmex_cases$cidi_q18_binary_f) <- "Hypothyroidism"

table1(~
  cidi_pain_binary_f + #Pain
  cidi_cardiac_binary_f + #Cardiometabolic
  cidi_resp_binary_f + #Respiratory
  cidi_resp_binary_f + #Neurological
  cidi_q14_binary_f + #Ulcer
  cidi_q5_binary_f + #Allergies
  cidi_q18_binary_f + #Hypothyroidism
  | trauma_load_inter_cat, overall=F, data=nmex_cases)
```



	0	1	2	>=3
	(N=2065)	(N=1189)	(N=796)	(N=631)
<b>Pain</b>				
0	1515 (73.4%)	844 (71.0%)	503 (63.2%)	360 (57.1%)
1	550 (26.6%)	345 (29.0%)	293 (36.8%)	271 (42.9%)
<b>Cardiometabolic</b>				
0	1588 (76.9%)	920 (77.4%)	615 (77.3%)	484 (76.7%)
1	477 (23.1%)	269 (22.6%)	181 (22.7%)	147 (23.3%)
<b>Respiratory</b>				
0	1980 (95.9%)	1133 (95.3%)	752 (94.5%)	582 (92.2%)
1	85 (4.1%)	56 (4.7%)	44 (5.5%)	49 (7.8%)
<b>Ulcer</b>				
0	1538 (74.5%)	839 (70.6%)	519 (65.2%)	391 (62.0%)
1	527 (25.5%)	350 (29.4%)	277 (34.8%)	240 (38.0%)
<b>Allergies</b>				
0	1880 (91.0%)	1057 (88.9%)	685 (86.1%)	527 (83.5%)
1	185 (9.0%)	132 (11.1%)	111 (13.9%)	104 (16.5%)
<b>Hypothyroidism</b>				
0	1880 (91.0%)	1089 (91.6%)	713 (89.6%)	574 (91.0%)
1	185 (9.0%)	100 (8.4%)	83 (10.4%)	57 (9.0%)

```
table1(~
  cidi_pain_binary_f + #Pain
  cidi_cardiac_binary_f + #Cardiometabolic
  cidi_resp_binary_f + #Respiratory
  cidi_resp_binary_f + #Neurological
  cidi_q14_binary_f + #Ulcer
  cidi_q5_binary_f + #Allergies
  cidi_q18_binary_f + #Hypothyroidism
  | trauma_load_noninter_nhs_cat, overall=F, data=nmex_cases)
```

	0	1	2	>=3
	(N=2286)	(N=1340)	(N=646)	(N=409)
<b>Pain</b>				
0	1628 (71.2%)	925 (69.0%)	410 (63.5%)	259 (63.3%)
1	658 (28.8%)	415 (31.0%)	236 (36.5%)	150 (36.7%)
<b>Cardiometabolic</b>				
0	1782 (78.0%)	1049 (78.3%)	476 (73.7%)	300 (73.3%)
1	504 (22.0%)	291 (21.7%)	170 (26.3%)	109 (26.7%)
<b>Respiratory</b>				
0	2169 (94.9%)	1286 (96.0%)	612 (94.7%)	380 (92.9%)
1	117 (5.1%)	54 (4.0%)	34 (5.3%)	29 (7.1%)
<b>Ulcer</b>				
0	1642 (71.8%)	925 (69.0%)	442 (68.4%)	278 (68.0%)
1	644 (28.2%)	415 (31.0%)	204 (31.6%)	131 (32.0%)
<b>Allergies</b>				
0	2051 (89.7%)	1193 (89.0%)	571 (88.4%)	334 (81.7%)
1	235 (10.3%)	147 (11.0%)	75 (11.6%)	75 (18.3%)
<b>Hypothyroidism</b>				
0	2125 (93.0%)	1216 (90.7%)	551 (85.3%)	364 (89.0%)
1	161 (7.0%)	124 (9.3%)	95 (14.7%)	45 (11.0%)

```
table1(~
  cidi_pain_binary_f + #Pain
  cidi_cardiac_binary_f + #Cardiometabolic
  cidi_resp_binary_f + #Respiratory
  cidi_resp_binary_f + #Neurological
  cidi_q14_binary_f + #Ulcer
  cidi_q5_binary_f + #Allergies
  cidi_q18_binary_f + #Hypothyroidism
  | trauma_load_inter_cat_4, overall=F, data=nmex_cases,
  caption = "Interpersonal trauma")
```

```
table1(~
  cidi_pain_binary_f + #Pain
  cidi_cardiac_binary_f + #Cardiometabolic
  cidi_resp_binary_f + #Respiratory
  cidi_resp_binary_f + #Neurological
  cidi_q14_binary_f + #Ulcer
  cidi_q5_binary_f + #Allergies
  cidi_q18_binary_f + #Hypothyroidism
  | trauma_load_noninter_nhs_cat_4, overall=F, data=nmex_cases,
  caption = "Noninterpersonal trauma")
```

```
table1(~
  factor(cidi_pain_binary) + #Pain

  factor(cidi_cardiac_binary) + #Cardiometabolic
  factor(cidi_resp_binary) + #Respiratory
  factor(cidi_neur_binary) + #Neurological
```

Table 29: Interpersonal trauma

	0	1	2	3	4+
	(N=2065)	(N=1189)	(N=796)	(N=399)	(N=232)
<b>Pain</b>					
0	1515 (73.4%)	844 (71.0%)	503 (63.2%)	235 (58.9%)	125 (53.9%)
1	550 (26.6%)	345 (29.0%)	293 (36.8%)	164 (41.1%)	107 (46.1%)
<b>Cardiometabolic</b>					
0	1588 (76.9%)	920 (77.4%)	615 (77.3%)	311 (77.9%)	173 (74.6%)
1	477 (23.1%)	269 (22.6%)	181 (22.7%)	88 (22.1%)	59 (25.4%)
<b>Respiratory</b>					
0	1980 (95.9%)	1133 (95.3%)	752 (94.5%)	374 (93.7%)	208 (89.7%)
1	85 (4.1%)	56 (4.7%)	44 (5.5%)	25 (6.3%)	24 (10.3%)
<b>Ulcer</b>					
0	1538 (74.5%)	839 (70.6%)	519 (65.2%)	246 (61.7%)	145 (62.5%)
1	527 (25.5%)	350 (29.4%)	277 (34.8%)	153 (38.3%)	87 (37.5%)
<b>Allergies</b>					
0	1880 (91.0%)	1057 (88.9%)	685 (86.1%)	333 (83.5%)	194 (83.6%)
1	185 (9.0%)	132 (11.1%)	111 (13.9%)	66 (16.5%)	38 (16.4%)
<b>Hypothyroidism</b>					
0	1880 (91.0%)	1089 (91.6%)	713 (89.6%)	363 (91.0%)	211 (90.9%)
1	185 (9.0%)	100 (8.4%)	83 (10.4%)	36 (9.0%)	21 (9.1%)

Table 30: Noninterpersonal trauma

	0	1	2	3	4+
	(N=2286)	(N=1340)	(N=646)	(N=259)	(N=150)
<b>Pain</b>					
0	1628 (71.2%)	925 (69.0%)	410 (63.5%)	167 (64.5%)	92 (61.3%)
1	658 (28.8%)	415 (31.0%)	236 (36.5%)	92 (35.5%)	58 (38.7%)
<b>Cardiometabolic</b>					
0	1782 (78.0%)	1049 (78.3%)	476 (73.7%)	192 (74.1%)	108 (72.0%)
1	504 (22.0%)	291 (21.7%)	170 (26.3%)	67 (25.9%)	42 (28.0%)
<b>Respiratory</b>					
0	2169 (94.9%)	1286 (96.0%)	612 (94.7%)	246 (95.0%)	134 (89.3%)
1	117 (5.1%)	54 (4.0%)	34 (5.3%)	13 (5.0%)	16 (10.7%)
<b>Ulcer</b>					
0	1642 (71.8%)	925 (69.0%)	442 (68.4%)	177 (68.3%)	101 (67.3%)
1	644 (28.2%)	415 (31.0%)	204 (31.6%)	82 (31.7%)	49 (32.7%)
<b>Allergies</b>					
0	2051 (89.7%)	1193 (89.0%)	571 (88.4%)	208 (80.3%)	126 (84.0%)
1	235 (10.3%)	147 (11.0%)	75 (11.6%)	51 (19.7%)	24 (16.0%)
<b>Hypothyroidism</b>					
0	2125 (93.0%)	1216 (90.7%)	551 (85.3%)	231 (89.2%)	133 (88.7%)
1	161 (7.0%)	124 (9.3%)	95 (14.7%)	28 (10.8%)	17 (11.3%)

```

factor(cidi_q14_binary) + #Ulcer
factor(cidi_q5_binary) + #Allergies
factor(cidi_q18_binary) + #Hypothyroidism
| trauma_load_inter_cat_4, overall=F, data=nmex_cases)

```

	0	1	2	3	4+
	(N=2065)	(N=1189)	(N=796)	(N=399)	(N=232)
<b>factor(cidi_pain_binary)</b>					
0	1515 (73.4%)	844 (71.0%)	503 (63.2%)	235 (58.9%)	125 (53.9%)
1	550 (26.6%)	345 (29.0%)	293 (36.8%)	164 (41.1%)	107 (46.1%)
<b>factor(cidi_cardiac_binary)</b>					
0	1588 (76.9%)	920 (77.4%)	615 (77.3%)	311 (77.9%)	173 (74.6%)
1	477 (23.1%)	269 (22.6%)	181 (22.7%)	88 (22.1%)	59 (25.4%)
<b>factor(cidi_resp_binary)</b>					
0	1936 (93.8%)	1109 (93.3%)	737 (92.6%)	366 (91.7%)	214 (92.2%)
1	129 (6.2%)	80 (6.7%)	59 (7.4%)	33 (8.3%)	18 (7.8%)
<b>factor(cidi_neur_binary)</b>					
0	1980 (95.9%)	1133 (95.3%)	752 (94.5%)	374 (93.7%)	208 (89.7%)
1	85 (4.1%)	56 (4.7%)	44 (5.5%)	25 (6.3%)	24 (10.3%)
<b>factor(cidi_q14_binary)</b>					
0	1538 (74.5%)	839 (70.6%)	519 (65.2%)	246 (61.7%)	145 (62.5%)
1	527 (25.5%)	350 (29.4%)	277 (34.8%)	153 (38.3%)	87 (37.5%)
<b>factor(cidi_q5_binary)</b>					
0	1880 (91.0%)	1057 (88.9%)	685 (86.1%)	333 (83.5%)	194 (83.6%)
1	185 (9.0%)	132 (11.1%)	111 (13.9%)	66 (16.5%)	38 (16.4%)
<b>factor(cidi_q18_binary)</b>					
0	1880 (91.0%)	1089 (91.6%)	713 (89.6%)	363 (91.0%)	211 (90.9%)
1	185 (9.0%)	100 (8.4%)	83 (10.4%)	36 (9.0%)	21 (9.1%)

```

table1(~
  factor(cidi_pain_binary) + #Pain
  factor(cidi_cardiac_binary) + #Cardiometabolic
  factor(cidi_resp_binary) + #Respiratory
  factor(cidi_neur_binary) + #Neurological
  factor(cidi_q14_binary) + #Ulcer
  factor(cidi_q5_binary) + #Allergies
  factor(cidi_q18_binary) + #Hypothyroidism
| trauma_load_noninter_nhs_cat_4, overall=F, data=nmex_cases)

```

	0	1	2	3	4+
	(N=2286)	(N=1340)	(N=646)	(N=259)	(N=150)
<b>factor(cidi_pain_binary)</b>					
0	1628 (71.2%)	925 (69.0%)	410 (63.5%)	167 (64.5%)	92 (61.3%)
1	658 (28.8%)	415 (31.0%)	236 (36.5%)	92 (35.5%)	58 (38.7%)
<b>factor(cidi_cardiac_binary)</b>					
0	1782 (78.0%)	1049 (78.3%)	476 (73.7%)	192 (74.1%)	108 (72.0%)
1	504 (22.0%)	291 (21.7%)	170 (26.3%)	67 (25.9%)	42 (28.0%)
<b>factor(cidi_resp_binary)</b>					
0	2140 (93.6%)	1256 (93.7%)	601 (93.0%)	233 (90.0%)	132 (88.0%)
1	146 (6.4%)	84 (6.3%)	45 (7.0%)	26 (10.0%)	18 (12.0%)
<b>factor(cidi_neur_binary)</b>					
0	2169 (94.9%)	1286 (96.0%)	612 (94.7%)	246 (95.0%)	134 (89.3%)
1	117 (5.1%)	54 (4.0%)	34 (5.3%)	13 (5.0%)	16 (10.7%)
<b>factor(cidi_q14_binary)</b>					
0	1642 (71.8%)	925 (69.0%)	442 (68.4%)	177 (68.3%)	101 (67.3%)
1	644 (28.2%)	415 (31.0%)	204 (31.6%)	82 (31.7%)	49 (32.7%)
<b>factor(cidi_q5_binary)</b>					
0	2051 (89.7%)	1193 (89.0%)	571 (88.4%)	208 (80.3%)	126 (84.0%)
1	235 (10.3%)	147 (11.0%)	75 (11.6%)	51 (19.7%)	24 (16.0%)
<b>factor(cidi_q18_binary)</b>					
0	2125 (93.0%)	1216 (90.7%)	551 (85.3%)	231 (89.2%)	133 (88.7%)
1	161 (7.0%)	124 (9.3%)	95 (14.7%)	28 (10.8%)	17 (11.3%)

```
table1(~ factor(cidi_any_binary) + #Any

      factor(cidi_pain_binary) + #Pain

      factor(cidi_cardiac_binary) + #Cardiometabolic
      factor(cidi_resp_binary) + #Respiratory
      factor(cidi_q10_binary) + #Asthma
      factor(cidi_neur_binary) + #Neurological
      factor(cidi_q14_binary) + #Ulcer
      factor(cidi_q5_binary) + #Allergies
      factor(cidi_q18_binary) + #Hypothyroidism
| trauma_load_inter_cat_4, overall=F, data=nmex_cases)
```

	0	1	2	3	4+
	(N=2065)	(N=1189)	(N=796)	(N=399)	(N=232)
<b>factor(cidi_any_binary)</b>					
0	795 (38.5%)	430 (36.2%)	234 (29.4%)	110 (27.6%)	58 (25.0%)
1	1270 (61.5%)	759 (63.8%)	562 (70.6%)	289 (72.4%)	174 (75.0%)
<b>factor(cidi_pain_binary)</b>					
0	1515 (73.4%)	844 (71.0%)	503 (63.2%)	235 (58.9%)	125 (53.9%)
1	550 (26.6%)	345 (29.0%)	293 (36.8%)	164 (41.1%)	107 (46.1%)
<b>factor(cidi_cardiac_binary)</b>					
0	1588 (76.9%)	920 (77.4%)	615 (77.3%)	311 (77.9%)	173 (74.6%)
1	477 (23.1%)	269 (22.6%)	181 (22.7%)	88 (22.1%)	59 (25.4%)
<b>factor(cidi_resp_binary)</b>					
0	1936 (93.8%)	1109 (93.3%)	737 (92.6%)	366 (91.7%)	214 (92.2%)
1	129 (6.2%)	80 (6.7%)	59 (7.4%)	33 (8.3%)	18 (7.8%)
<b>factor(cidi_q10_binary)</b>					
0	1993 (96.5%)	1148 (96.6%)	755 (94.8%)	376 (94.2%)	218 (94.0%)
1	72 (3.5%)	41 (3.4%)	41 (5.2%)	23 (5.8%)	14 (6.0%)
<b>factor(cidi_neur_binary)</b>					
0	1980 (95.9%)	1133 (95.3%)	752 (94.5%)	374 (93.7%)	208 (89.7%)
1	85 (4.1%)	56 (4.7%)	44 (5.5%)	25 (6.3%)	24 (10.3%)
<b>factor(cidi_q14_binary)</b>					
0	1538 (74.5%)	839 (70.6%)	519 (65.2%)	246 (61.7%)	145 (62.5%)
1	527 (25.5%)	350 (29.4%)	277 (34.8%)	153 (38.3%)	87 (37.5%)
<b>factor(cidi_q5_binary)</b>					
0	1880 (91.0%)	1057 (88.9%)	685 (86.1%)	333 (83.5%)	194 (83.6%)
1	185 (9.0%)	132 (11.1%)	111 (13.9%)	66 (16.5%)	38 (16.4%)
<b>factor(cidi_q18_binary)</b>					
0	1880 (91.0%)	1089 (91.6%)	713 (89.6%)	363 (91.0%)	211 (90.9%)
1	185 (9.0%)	100 (8.4%)	83 (10.4%)	36 (9.0%)	21 (9.1%)

## Models and Figures

As per: <https://docs.google.com/document/d/1g42GgDfTURCFaz44x83fNtuOiDNBV2NGf1liqdeEl6A/edit?tab=t.0>

1. **Model B:** {Health outcome grouping} ~ trauma\_load\_inter\_categorical (coded as 0, 1, 2, 3+) + trauma\_load\_noninter\_nhs\_categorical (coded as 0, 1, 2, 3+) + age + sex + SES status + urbanicity
2. **Model C:** {Health outcome grouping} ~ trauma\_load\_inter\_continuous + trauma\_load\_noninter\_nhs\_continuous + age + sex + SES status + urbanicity
3. **Model D:** {Health outcome grouping} ~ trauma\_load\_inter\_cat (coded as 0, 1, 2, 3+) + trauma\_load\_noninter\_nhs\_cat (coded as 0, 1, 2, 3+) + age + sex + SES status + urbanicity + alcohol + tobacco + cannabis
4. **Model E:** {Health outcome grouping} ~ trauma\_load\_inter\_continuous + trauma\_load\_noninter\_nhs\_continuous + age + sex + SES status + urbanicity + Alcohol + Cannabis + Tobacco

### Regression (old)

Here I make a df with all data from all tables.

We loop through each model with each specification and then we loop through each chronic condition in list `cidi_list_binary`.

For variables `cidiq1-q18` (excluding `q15,q17`) we use Bonferonni correction ( $\alpha/16$ ). Else (grouped chronic conditions) we do not.

```
cidi_list_binary <- c(paste0("cidi_q", 1:14, "_binary"), "cidi_q16_binary",
                     "cidi_q18_binary", "cidi_pain_binary",
                     "cidi_cardiac_binary", "cidi_resp_binary",
                     "cidi_neur_binary")

results <- NULL
df <- NULL

models <- c("Model B", "Model C", "Model D", "Model E")

#All Trauma

for (model in models) {

  if (model == "Model B") {
    trauma_var <- "trauma_load_inter_cat + trauma_load_noninter_nhs_cat"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
site_location_group_f"
    type <- "Categorical"
  } else if (model == "Model C") {
    trauma_var <- "trauma_load_interpersonal_1 + trauma_load_noninter_1_nhs"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
site_location_group_f"
    type <- "Numeric"
  } else if (model == "Model D") {
    trauma_var <- "trauma_load_inter_cat + trauma_load_noninter_nhs_cat"
```

```

covar <- "+ age_at_interview + is_male_f + ses_status_f +
site_location_group_f + factor(assist_tobacco_amt_recode) +
factor(assist_alcohol_amt_recode) + factor(assist_cannabis_amt_recode)"
type <- "Numeric"
} else if (model == "Model E") {
trauma_var <- "trauma_load_interpersonal_1 + trauma_load_noninter_1_nhs"
covar <- "+ age_at_interview + is_male_f + ses_status_f +
          factor(assist_tobacco_amt_recode) +
          factor(assist_alcohol_amt_recode) +
          factor(assist_cannabis_amt_recode) + site_location_group_f"
type <- "Numeric"
}

for (ii in 1:length(cidi_list_binary)) {

  fmla <- as.formula(paste(cidi_list_binary[ii], " ~ ",
                           trauma_var, covar))

  # Fit the generalized linear regression model
  fit <- glm(fmla, data = nmex_cases, family = binomial)

  #####
  cidi_var <- cidi_list_binary[ii]
  #print(cidi_list_binary[ii])

  cidi_var_n <- table(nmex_cases[[cidi_var]])[2]

  fixef <- exp(coef(fit))

  #Bonferonni correction for single variables
  if (cidi_list_binary[ii] %in% c("Pain", "Cardiometabolic",
                                "Respiratory", "Neurological")) {
    conf <- exp(confint(fit))
    group <- "Not grouped"
  } else {
    conf <- exp(confint(fit, level = (1 - 0.05 / 16)))
    group <- "Single"
  }

  conf_complete <- conf[complete.cases(conf), ]

  var <- rownames(conf_complete)

  for (j in 1:length(var)) {

    var_temp <- var[j]

    est <- round(exp(summary(fit)[["coefficients"]][var[j],
                                         "Estimate"])), 3)

    #est <- round(exp(summary(fit)[var[j]]), 3)
    ci_ll <- round(conf_complete[var[j], 1], 3)
  }
}

```



```

ci_ul <- round(conf_complete[var[j], 2],3)
ci <- paste0("(", ci_ll, ", ", ci_ul, ")")

p <- summary(fit)$coefficients[var[j], "Pr(>|z|)"]

#Bonferonni correction for single variables
if (!(cidi_list_binary[ii] %in% c("Pain","Cardiometabolic",
                                "Respiratory","Neurological"))) {
  p <- p/16
}

model <- model

if (p < 0.001) {
  pval <- "<0.001"
} else if (p < 0.01) {
  pval <- "<0.01"
} else if (p < 0.05) {
  pval <- "<0.05"
} else {
  pval <- round(p, 3)
}

results <- rbind(results,c(cidi_var, var_temp, est, ci_ll, ci_ul, pval,
                           model, cidi_var_n, type, group))
}
}
}

#Here we edit the results tibble into a dataframe and modify some variables in prep to make the figures

df <- as.data.frame(results)

colnames(df)<-c("cidi_q", "Coefficient", "OR", "ci_ll", "ci_ul", "pval",
               "model", "cidi_var_n", "type","group")

df <- df %>% mutate(cidi_name = case_when(
  cidi_q == "cidi_q1_binary" ~ "Arthritis or rheumatism",
  cidi_q == "cidi_q2_binary" ~ "Chronic back or neck problems",
  cidi_q == "cidi_q3_binary" ~ "Frequent or severe headaches",
  cidi_q == "cidi_q4_binary" ~ "Any other chronic pain",
  cidi_q == "cidi_q5_binary" ~ "Seasonal allergies like hay fever",
  cidi_q == "cidi_q6_binary" ~ "A stroke",
  cidi_q == "cidi_q7_binary" ~ "A heart attack",
  cidi_q == "cidi_q8_binary" ~ "Heart disease",
  cidi_q == "cidi_q9_binary" ~ "High Blood Pressure",
  cidi_q == "cidi_q10_binary" ~ "Asthma",
  cidi_q == "cidi_q11_binary" ~ "Tuberculosis",
  cidi_q == "cidi_q12_binary" ~ "Any other chronic lung disease",
  cidi_q == "cidi_q13_binary" ~ "Diabetes or high blood sugar",
  cidi_q == "cidi_q14_binary" ~ "An ulcer in your stomach or intestine",
  cidi_q == "cidi_q15_binary" ~ "HIV infection or AIDS",

```

```

cidi_q == "cidi_q16_binary" ~ "Epilepsy or seizures",
cidi_q == "cidi_q17_binary" ~ "Cancer",
cidi_q == "cidi_q18_binary" ~ "Other",
cidi_q == "cidi_pain_binary" ~ "Pain",
cidi_q == "cidi_cardiac_binary" ~ "Cardiometabolic",
cidi_q == "cidi_resp_binary" ~ "Respiratory",
cidi_q == "cidi_neur_binary" ~ "Neurological",

),

coef_name = factor(case_when(
  Coefficient == "trauma_load_inter_cat1" ~ "1 Trauma",
  Coefficient == "trauma_load_inter_cat2" ~ "2 Traumas",
  Coefficient == "trauma_load_inter_cat>=3" ~ "3+ Traumas",
  Coefficient == "trauma_load_noninter_nhs_cat1" ~ "1 Trauma",
  Coefficient == "trauma_load_noninter_nhs_cat2" ~ "2 Traumas",
  Coefficient == "trauma_load_noninter_nhs_cat>=3" ~ "3+ Traumas",
  Coefficient == "trauma_load_noninter_nhs_cat>=3" ~ "3+ Traumas",
  Coefficient == "trauma_load_interpersonal_1" ~ "Continuous",
  Coefficient == "trauma_load_noninter_1_nhs" ~ "Continuous"
), levels = c("1 Trauma",
              "2 Traumas",
              "3+ Traumas",
              "Continuous")),

type = factor(type, levels = c("Numeric", "Categorical")),

trauma_type = case_when(
  grepl("noninter", Coefficient) ~ "Non-interpersonal",
  grepl("inter", Coefficient) ~ "Interpersonal"
)

df_trauma_only <- df %>%
  filter(str_starts(Coefficient, "trauma_load"))

```

**Figures: ORs by model by chronic condition** Here we loop through each of the 18 chronic conditions as well as the grouped chronic conditions and pool all results for Model B, C, D, and E.

In the title there will be the number of participants who endorsed the conditions.

```

outs<-unique(df_trauma_only$cidi_name)

pdf("NMEX_modelBCDE_allconditions.pdf")

for (i in outs) {

  tdf<-df_trauma_only[df_trauma_only$cidi_name==i, ]
  tdf$OR <- as.numeric(tdf$OR)
  tdf$ci_ll <- as.numeric(tdf$ci_ll)
  tdf$ci_ul <- as.numeric(tdf$ci_ul)
}

```

```

cidi_var_n <- tdf$cidi_var_n

p <- ggplot(tdf, aes(x = coef_name, y = OR, ymin = ci_ll, ymax = ci_ul,
                    color = trauma_type)) +
  geom_pointrange(position = position_dodge(width = 0.7)) +
  facet_wrap(~ model, nrow = 1, scales = 'free_x') +
  geom_hline(yintercept = 1, lty = 2) +
  theme_bw() +
  theme(
    axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1),
    legend.position = "right"
  ) +
  #scale_fill_manual(name = "Coefficient Type", ) +

  labs(
    y = "OR (99.7% CI) / OR (95% CI) for grouped",
    x = "Model",
    color = "Trauma Type"
  ) +
  ggtitle(paste(i, "; N =", cidi_var_n))

print(p)
}

dev.off()

```

```

## pdf
## 2

```

**Figures: OR Model B and Model C (all CIDI together)** Here we show all chronic conditions in the same figure for easy comparison for all four models. We separate outcomes for interpersonal and noninterpersonal trauma.

```

cidi_name_order = c("Arthritis or rheumatism",
                    "Chronic back or neck problems",
                    "Frequent or severe headaches",
                    "Any other chronic pain",
                    "Seasonal allergies like hay fever",
                    "A stroke",
                    "A heart attack",
                    "Heart disease",
                    "High Blood Pressure",
                    "Asthma",
                    "Tuberculosis",
                    "Any other chronic lung disease",
                    "Diabetes or high blood sugar",
                    "An ulcer in your stomach or intestine",
                    "Epilepsy or seizures",
                    "Other",
                    "Pain",

```

```

        "Cardiometabolic",
        "Respiratory",
        "Neurological"

    )

model_nosub <- df_trauma_only %>% filter(model %in% c("Model C")) %>%
  mutate(OR = as.numeric(OR),
         ci_ll = as.numeric(ci_ll),
         ci_ul = as.numeric(ci_ul),
         cidi_name = factor(cidi_name, levels = cidi_name_order))

outs<-unique(model_nosub$cidi_name)

pdf("Compiled results_modelBCDE.pdf")

# model_nosub <- df_trauma_only %>% filter(model %in% c("Model B")) %>%
# mutate(OR = as.numeric(OR),
#        ci_ll = as.numeric(ci_ll),
#        ci_ul = as.numeric(ci_ul),
#        cidi_name = factor(cidi_name, levels = cidi_name_order))

df_trauma_only %>% filter(model %in% c("Model B")) %>%
mutate(OR = as.numeric(OR),
      ci_ll = as.numeric(ci_ll),
      ci_ul = as.numeric(ci_ul),
      cidi_name = factor(cidi_name, levels = cidi_name_order)) %>%
  ggplot(aes(x = cidi_name, y = OR, ymin = ci_ll, ymax = ci_ul,
            color = coef_name)) +
  facet_wrap(~trauma_type) +
  geom_pointrange(position = position_dodge(width = 0.7)) +
  geom_hline(yintercept = 1, lty = 2) +
  theme_bw() +
  theme(
    legend.position = "right"
  ) +
  labs(
    y = "OR (99.7% CI) / OR (95% CI) for grouped ",
    x = "",
    color = "Trauma Type",
    title = "Model B (Main categorical model)" +
    ylim(0,5) +
    coord_flip()

df_trauma_only %>% filter(model %in% c("Model C")) %>%
mutate(OR = as.numeric(OR),

```

```

    ci_ll = as.numeric(ci_ll),
    ci_ul = as.numeric(ci_ul),
    cidi_name = factor(cidi_name, levels = cidi_name_order)) %>%
  ggplot(aes(x = cidi_name, y = OR, ymin = ci_ll, ymax = ci_ul,
             color = trauma_type)) +
  geom_pointrange(position = position_dodge(width = 0.7)) +
  geom_hline(yintercept = 1, lty = 2) +
  theme_bw() +
  theme(
    legend.position = "right"
  ) +
  labs(
    y = "OR (99.7% CI) / OR (95% CI) for grouped ",
    x = "",
    color = "Trauma Type",
    title = "Model C (Main cont. model)"
  ) + coord_flip()

df_trauma_only %>% filter(model %in% c("Model D")) %>%
mutate(OR = as.numeric(OR),
       ci_ll = as.numeric(ci_ll),
       ci_ul = as.numeric(ci_ul),
       cidi_name = factor(cidi_name, levels = cidi_name_order)) %>%
  ggplot(aes(x = cidi_name, y = OR, ymin = ci_ll, ymax = ci_ul,
             color = coef_name)) +
  facet_wrap(~trauma_type) +
  geom_pointrange(position = position_dodge(width = 0.7)) +
  geom_hline(yintercept = 1, lty = 2) +
  theme_bw() +
  theme(
    legend.position = "right"
  ) +
  labs(
    y = "OR (99.7% CI) / OR (95% CI) for grouped ",
    x = "",
    color = "Trauma Type",
    title = "Model D (Main categorical model)" +
    ylim(0,5) +
    coord_flip()

df_trauma_only %>% filter(model %in% c("Model E")) %>%
mutate(OR = as.numeric(OR),
       ci_ll = as.numeric(ci_ll),
       ci_ul = as.numeric(ci_ul),
       cidi_name = factor(cidi_name, levels = cidi_name_order)) %>%
  ggplot(aes(x = cidi_name, y = OR, ymin = ci_ll, ymax = ci_ul,
             color = trauma_type)) +
  geom_pointrange(position = position_dodge(width = 0.7)) +
  geom_hline(yintercept = 1, lty = 2) +
  theme_bw() +

```

```

facet_wrap(~trauma_type) +
theme(
  legend.position = "right"
) +
labs(
  y = "OR (99.7% CI) / OR (95% CI) for grouped ",
  x = "",
  color = "Trauma Type",
  title = "Model E (Substance use cont. model)"
) + coord_flip()

dev.off()

```

```

## pdf
## 2

```

## Regression - update 1-13-25 (current)

### Update from 1-13-25

Redoing the above model such that:

Here I make a df with all data from all tables.

We loop through each model with each specification and then we loop through each chronic condition in list `cidi_list_binary`.

We look at the following outcomes: Chronic pain grouping, Cardiometabolic conditions grouping, Respiratory conditions grouping, Neurological conditions grouping, Stomach ulcer, Allergies, Hypothyroidism

For these 7 plus the poisson model we do an Bonferonni correction with  $\alpha/8$ . For outcomes with prevalence  $> 10\%$ , we run a relative risk regression. Otherwise we run logistic regression.

We then edit the results tibble into a dataframe and modify some variables in prep to make the figures.

Andrew R Notes on Poisson for Relative Risk Regression - 01-13-25:

Hi Hayden,

Confirmed, you run relative risk regression with the command:

```
fit<-glm(fmla, data=df, family=poisson(link="log"))
```

Note: I think I mentioned in a previous email that when you do relative risk regression using a Poisson model that you need to use robust standard errors. I sent a link previously but there are a couple more at the end of the email. This means you cannot use the `confint()` command to calculate the confidence intervals. You have to manually calculate the standard errors and critical values. For robust standard errors, the specific command is:

```
rse <- coeftest(fit, vcov = vcovHC(fit, type="HC1"))
```

That's going to get you the standard error. To calculate the Bonferroni CI you need to get the right critical value (it is not 1.96; that's the 95% confidence level). The correct critical value will be

```
qnorm(p=.05/(2*# of hypotheses tested), lower.tail=FALSE)
```

I think the # of hypotheses tested is 8, correct? 7 grouped outcomes and then the count? If so, your critical value is 2.734369 and you can calculate the CIs as  $\beta \pm 2.734369 \times \text{robust standard error}$ .

I would not hard code the value 2.734369. As before, make sure you define the number of tests and calculate the alpha based on that in case things change in the future.

Let me know if you have any questions about the above.

Best,

Andrew

<https://cran.r-project.org/web/packages/sandwich/sandwich.pdf>    <https://data.princeton.edu/wws509/r/robust> <https://stats.stackexchange.com/questions/117052/replicating-stata-robust-option-in-r>

```
cidi_list_binary <- c(
  "cidi_q5_binary", "cidi_q14_binary", "cidi_q18_binary",
  "cidi_pain_binary", "cidi_cardiac_binary",
  "cidi_resp_binary", "cidi_neur_binary"
)

bon_cor <- length(cidi_list_binary) + 1
```

```

results <- NULL
df <- NULL

models <- c("Model B", "Model C", "Model D", "Model E")

# All Trauma

for (model in models) {
  if (model == "Model B") {
    trauma_var <- "trauma_load_inter_cat + trauma_load_noninter_nhs_cat"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
site_location_group_f"
    type <- "Categorical"
  } else if (model == "Model C") {
    trauma_var <- "trauma_load_interpersonal_1 + trauma_load_noninter_1_nhs"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
site_location_group_f"
    type <- "Numeric"
  } else if (model == "Model D") {
    trauma_var <- "trauma_load_inter_cat + trauma_load_noninter_nhs_cat"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
site_location_group_f + factor(assist_tobacco_amt_recode) +
factor(assist_alcohol_amt_recode) + factor(assist_cannabis_amt_recode)"
    type <- "Numeric"
  } else if (model == "Model E") {
    trauma_var <- "trauma_load_interpersonal_1 + trauma_load_noninter_1_nhs"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
factor(assist_tobacco_amt_recode) +
factor(assist_alcohol_amt_recode) +
factor(assist_cannabis_amt_recode) + site_location_group_f"
    type <- "Numeric"
  }
}

for (ii in 1:length(cidi_list_binary)) {
  #print(model)
  #print(cidi_list_binary[ii])
  tab <- table(nmex_cases[cidi_list_binary[ii]])
  prev <- tab[2] / (tab[1] + tab[2])
  prev_round <- round(prev,3)
  cidi_var <- cidi_list_binary[ii]
  # Get number of cases
  cidi_var_n <- table(nmex_cases[[cidi_var]])[2]

  if (prev <= 0.1) {
    fmla <- as.formula(paste(
      cidi_list_binary[ii], " ~ ",
      trauma_var, covar
    ))
    # Fit the model for logistic regression
    fit <- glm(fmla, data = nmex_cases, family = binomial)

    # get OR

```



```

fixef <- exp(coef(fit))

# Get confidence interval based on bonferonni
conf <- exp(confint(fit, level = (1 - 0.05 / bon_cor)))

# Extract CI of all values
conf_complete <- conf[complete.cases(conf), ]

# Get all coefficient names and levels
var <- rownames(conf_complete)

# For j in the number of coefficients (var)
for (j in 1:length(var)) {
  reg_type <- "Logistic Regression"
  var_temp <- var[j]

  est <- round(exp(summary(fit)[["coefficients"]][
    var[j],
    "Estimate"
  ]), 3)
  # est <- round(exp(summary(fit)[var[j]]), 3)
  ci_ll <- round(conf_complete[var[j], 1], 3)
  ci_ul <- round(conf_complete[var[j], 2], 3)
  ci <- paste0("(", ci_ll, ", ", ci_ul, ")")

  p <- summary(fit)$coefficients[var[j], "Pr(>|z|)"]
  #print(paste("P val log reg:", p))

  model <- model

  if (p < (0.05 / bon_cor)) {
    pval <- "Significant"
  } else {
    pval <- "Not significant"
  }

  results <- rbind(results, c(
    cidi_var, var_temp, prev_round,
    est, ci_ll, ci_ul, p, pval,
    model, cidi_var_n, type, reg_type
  ))
}
} else {
  # if prev > 10%, do relative risk regression
  reg_type <- "Relative Risk"
  fmla <- as.formula(paste(
    cidi_list_binary[ii], " ~ ",
    trauma_var, covar
  ))
  # Fit the model for relative risk regression (poisson, log link).
  fit <- glm(fmla, data = nmex_cases, family = poisson(link = "log"))

  rse <- coeftest(fit, vcov = vcovHC(fit, type = "HC1"))

```

```

q <- qnorm(p = .05 / (2 * bon_cor), lower.tail = FALSE)

# Extract beta, st_error, and p_val

estimate_complete <- rse[complete.cases(rse), 1]
rob_std_error_complete <- rse[complete.cases(rse), 2]
p_val_rob <- rse[complete.cases(rse), 4]

var <- rownames(rse)

for (j in 1:length(var)) {
  var_temp <- var[j]
  beta <- estimate_complete[var[j]]
  est <- round(exp(beta),3)
  robust_se <- rob_std_error_complete[var[j]]
  p <- p_val_rob[var[j]]
  #print(paste("P val rel risk reg:", p))

  if (p < (0.05 / bon_cor)) {
    pval <- "Significant"
  } else {
    pval <- "Not significant"
  }

  ci_ll <- round(exp(beta - q * robust_se),3)
  ci_ul <- round(exp(beta + q * robust_se),3)

  results <- rbind(results, c(
    cidi_var, var_temp, prev_round,
    est, ci_ll, ci_ul, p, pval,
    model, cidi_var_n, type, reg_type
  ))
}
}
}

#Make results a dataframe and relabel
df <- as.data.frame(results)
colnames(df)<-c("cidi_q", "Coefficient", "prevalence",
              "OR_or_RR", "ci_ll", "ci_ul", "p", "significant",
              "model", "cidi_var_n", "type", "regression_type")

df <- df %>% mutate(cidi_name = case_when(
  cidi_q == "cidi_q1_binary" ~ "Arthritis/Rheumatism",
  cidi_q == "cidi_q2_binary" ~ "Back/neck problems",
  cidi_q == "cidi_q3_binary" ~ "Headaches",
  cidi_q == "cidi_q4_binary" ~ "Other pain",
  cidi_q == "cidi_q5_binary" ~ "Seasonal allergies",

```

```

cidi_q == "cidi_q6_binary" ~ "Stroke",
cidi_q == "cidi_q7_binary" ~ "Heart attack",
cidi_q == "cidi_q8_binary" ~ "Heart disease",
cidi_q == "cidi_q9_binary" ~ "High Blood Pressure",
cidi_q == "cidi_q10_binary" ~ "Asthma",
cidi_q == "cidi_q11_binary" ~ "Tuberculosis",
cidi_q == "cidi_q12_binary" ~ "Lung disease",
cidi_q == "cidi_q13_binary" ~ "Diabetes",
cidi_q == "cidi_q14_binary" ~ "Stomach/intestine ulcer",
cidi_q == "cidi_q15_binary" ~ "HIV/AIDS",
cidi_q == "cidi_q16_binary" ~ "Epilepsy/Seizures",
cidi_q == "cidi_q17_binary" ~ "Cancer",
cidi_q == "cidi_q18_binary" ~ "Hypothyroidism",
cidi_q == "cidi_pain_binary" ~ "Pain",
cidi_q == "cidi_cardiac_binary" ~ "Cardiometabolic",
cidi_q == "cidi_resp_binary" ~ "Respiratory",
cidi_q == "cidi_neur_binary" ~ "Neurological",

),

coef_name = factor(case_when(
  Coefficient == "trauma_load_inter_cat1" ~ "1",
  Coefficient == "trauma_load_inter_cat2" ~ "2",
  Coefficient == "trauma_load_inter_cat>=3" ~ "3+",
  Coefficient == "trauma_load_noninter_nhs_cat1" ~ "1",
  Coefficient == "trauma_load_noninter_nhs_cat2" ~ "2",
  Coefficient == "trauma_load_noninter_nhs_cat>=3" ~ "3+",
  Coefficient == "trauma_load_interpersonal_1" ~ "Continuous",
  Coefficient == "trauma_load_noninter_1_nhs" ~ "Continuous"
), levels = c("1",
              "2",
              "3+",
              "Continuous")),

type = factor(type, levels = c("Numeric", "Categorical")),

trauma_type = factor(case_when(
  grepl("noninter", Coefficient) ~ "Non-interpersonal",
  grepl("inter_|interpersonal", Coefficient) ~ "Interpersonal"
), levels = c("Non-interpersonal", "Interpersonal"))

)

df_trauma_only <- df %>%
  filter(str_starts(Coefficient, "trauma_load"))

```

**Format data for tables** Here I format some of the data for the table which can be found here: [https://docs.google.com/spreadsheets/d/1zhTdSART7JBKzzx\\_rPKCvSLNfKM5NfJTbV29twtnFIk/edit?gid=1299043405#gid=1299043405](https://docs.google.com/spreadsheets/d/1zhTdSART7JBKzzx_rPKCvSLNfKM5NfJTbV29twtnFIk/edit?gid=1299043405#gid=1299043405)

**Figures: ORs by model by chronic condition** Here we loop through each of the 18 chronic conditions as well as the grouped chronic conditions and pool all results for Model B, C, D, and E.

In the title there will be the number of participants who endorsed the conditions.

```
#Ensure correct path
setwd("~/Dropbox (Harvard University)/NeuroMex/Data Analysis/Papers/LEC-CIDI Paper/LEC-CIDI Paper/")

outs<-unique(df_trauma_only$cidi_name)

pdf(paste0("NMEX_modelBCDE_allconditions_",today(),".pdf"))

for (i in outs) {

  tdf<-df_trauma_only[df_trauma_only$cidi_name==i, ]
  tdf$OR <- as.numeric(tdf$OR_or_RR)
  tdf$ci_ll <- as.numeric(tdf$ci_ll)
  tdf$ci_ul <- as.numeric(tdf$ci_ul)
  cidi_var_n <- tdf$cidi_var_n

  p <- ggplot(tdf, aes(x = coef_name, y = OR, ymin = ci_ll, ymax = ci_ul,
                      color = trauma_type)) +
    geom_pointrange(position = position_dodge(width = 0.7)) +
    facet_wrap(~ model, nrow = 1, scales = 'free_x') +
    geom_hline(yintercept = 1, lty = 2) +
    theme_bw() +
    theme(
      axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1),
      legend.position = "right"
    ) +
    #scale_fill_manual(name = "Coefficient Type", ) +

    labs(
      y = "OR/RR (99.7% CI)",
      x = "Model",
      color = "Trauma Type"
    ) +
    ggtitle(paste(i, "; N =", cidi_var_n))

  print(p)

}

dev.off()
```

```
## pdf
## 2
```

**Figures: OR Model B and Model C (all CIDI together)** Here we show all chronic conditions in the same figure for easy comparison for all four models. We separate outcomes for interpersonal and nonintepersonal trauma.

```

setwd("~/Dropbox (Harvard University)/NeuroMex/Data Analysis/Papers/LEC-CIDI Paper/LEC-CIDI Paper/")

cidi_name_order = c( "Arthritis/Rheumatism",
  "Back/neck problems",
  "Headaches",
  "Other pain",
  "Seasonal allergies",
  "Stroke",
  "Heart attack",
  "Heart disease",
  "High Blood Pressure",
  "Asthma",
  "Tuberculosis",
  "Lung disease",
  "Diabetes",
  "Stomach/intestine ulcer",
  "HIV/AIDS",
  "Epilepsy/Seizures",
  "Cancer",
  "Hypothyroidism",
  "Pain",
  "Cardiometabolic",
  "Respiratory",
  "Neurological"
)

model_nosub <- df_trauma_only %>% filter(model %in% c("Model C")) %>%
  mutate(OR = as.numeric(OR_or_RR),
    ci_ll = as.numeric(ci_ll),
    ci_ul = as.numeric(ci_ul),
    cidi_name = factor(cidi_name, levels = cidi_name_order))

outs<-unique(model_nosub$cidi_name)

pdf(paste0("Compiled results_modelBCDE_",today(),".pdf"))

# model_nosub <- df_trauma_only %>% filter(model %in% c("Model B")) %>%
# mutate(OR = as.numeric(OR),
#   ci_ll = as.numeric(ci_ll),
#   ci_ul = as.numeric(ci_ul),
#   cidi_name = factor(cidi_name, levels = cidi_name_order))

df_trauma_only %>% filter(model %in% c("Model B")) %>%
mutate(OR = as.numeric(OR_or_RR),
  ci_ll = as.numeric(ci_ll),
  ci_ul = as.numeric(ci_ul),
  cidi_name = factor(cidi_name, levels = cidi_name_order)) %>%
  ggplot(aes(x = cidi_name, y = OR, ymin = ci_ll, ymax = ci_ul,

```

```

        color = coef_name)) +
  facet_wrap(~trauma_type) +
  geom_pointrange(position = position_dodge(width = 0.7)) +
  geom_hline(yintercept = 1, lty = 2) +
  theme_bw() +
  theme(
    legend.position = "right"
  ) +
  labs(
    y = "OR (99.7% CI) / OR (95% CI) for grouped ",
    x = "",
    color = "Trauma Type",
    title = "Model B (Main categorical model)") +
  ylim(0,5) +
  coord_flip()

df_trauma_only %>% filter(model %in% c("Model C")) %>%
mutate(OR = as.numeric(OR_or_RR),
       ci_ll = as.numeric(ci_ll),
       ci_ul = as.numeric(ci_ul),
       cidi_name = factor(cidi_name, levels = cidi_name_order)) %>%
  ggplot(aes(x = cidi_name, y = OR, ymin = ci_ll, ymax = ci_ul,
            color = trauma_type)) +
  geom_pointrange(position = position_dodge(width = 0.7)) +
  geom_hline(yintercept = 1, lty = 2) +
  theme_bw() +
  theme(
    legend.position = "right"
  ) +
  labs(
    y = "OR (99.7% CI) / OR (95% CI) for grouped ",
    x = "",
    color = "Trauma Type",
    title = "Model C (Main cont. model)"
  ) + coord_flip()

df_trauma_only %>% filter(model %in% c("Model D")) %>%
mutate(OR = as.numeric(OR_or_RR),
       ci_ll = as.numeric(ci_ll),
       ci_ul = as.numeric(ci_ul),
       cidi_name = factor(cidi_name, levels = cidi_name_order)) %>%
  ggplot(aes(x = cidi_name, y = OR, ymin = ci_ll, ymax = ci_ul,
            color = coef_name)) +
  facet_wrap(~trauma_type) +
  geom_pointrange(position = position_dodge(width = 0.7)) +
  geom_hline(yintercept = 1, lty = 2) +
  theme_bw() +
  theme(
    legend.position = "right"
  )

```

```

) +
labs(
  y = "OR (99.7% CI) / OR (95% CI) for grouped ",
  x = "",
  color = "Trauma Type",
  title = "Model D (Main categorical model)") +
ylim(0,5) +
coord_flip()

df_trauma_only %>% filter(model %in% c("Model E")) %>%
mutate(OR = as.numeric(OR_or_RR),
  ci_ll = as.numeric(ci_ll),
  ci_ul = as.numeric(ci_ul),
  cidi_name = factor(cidi_name, levels = cidi_name_order)) %>%
ggplot(aes(x = cidi_name, y = OR, ymin = ci_ll, ymax = ci_ul,
  color = trauma_type)) +
geom_pointrange(position = position_dodge(width = 0.7)) +
geom_hline(yintercept = 1, lty = 2) +
theme_bw() +
facet_wrap(~trauma_type) +
theme(
  legend.position = "right"
) +
labs(
  y = "OR (99.7% CI) / OR (95% CI) for grouped ",
  x = "",
  color = "Trauma Type",
  title = "Model E (Substance use cont. model)")
) + coord_flip()

dev.off()

```

```

## pdf
## 2

```

**Figure: OR/RR Model B and C with significance** Here we modify Andrew R's code to create a figure of all outcomes on one page.

```

setwd("~/Dropbox (Harvard University)/NeuroMex/Data Analysis/Papers/LEC-CIDI Paper/LEC-CIDI Paper/")

df_gg <- df_trauma_only %>% filter(model == "Model B")
df_gg$OR_or_RR <- as.numeric(as.character(df_gg$OR_or_RR))
df_gg$ci_ll <- as.numeric(as.character(df_gg$ci_ll))
df_gg$ci_ul <- as.numeric(as.character(df_gg$ci_ul))

pdf(paste0("Compiled results_ModelBCDE_",today(),".pdf"), width=8, height=6)

ggplot(df_gg, aes(x=coef_name, y=OR_or_RR, ymin=ci_ll, ymax=ci_ul,

```

```

        shape=trauma_type, color=significant)) +
geom_pointrange(position=position_dodge(0.5))+
geom_hline(yintercept=1, lty=2)+
coord_flip()+
scale_color_manual(breaks=c("Not significant", "Significant"),
                    values=c("black", "red"),
                    name = "Significance") +
scale_shape_manual(values = c(16, 17),
                   name = "Trauma Type") +
scale_y_continuous(breaks = seq(0, 4, by = 1), limits = c(0, 4)) +
facet_wrap(~cidi_name) +
theme_bw() +
guides(color = guide_legend(reverse=T))+
xlab("Number of Traumatic Events")+
ylab("OR/RR (99.7% CI)") +
theme(legend.position = "bottom") +
ggtitle("Model B: Categorical")

df_gg <- df_trauma_only %>% filter(model == "Model C")
df_gg$OR_or_RR <- as.numeric(as.character(df_gg$OR_or_RR))
df_gg$ci_ll <- as.numeric(as.character(df_gg$ci_ll))
df_gg$ci_ul <- as.numeric(as.character(df_gg$ci_ul))

ggplot(df_gg, aes(x=coef_name, y=OR_or_RR, ymin=ci_ll, ymax=ci_ul,
                  shape=trauma_type, color=significant)) +
  geom_pointrange(position=position_dodge(0.5))+
  geom_hline(yintercept=1, lty=2)+
  coord_flip()+
  scale_color_manual(breaks=c("Not significant", "Significant"),
                    values=c("black", "red"),
                    name = "Significance") +
  scale_shape_manual(values = c(16, 17),
                    name = "Trauma Type") +
  scale_y_continuous(breaks = seq(0, 4, by = 1), limits = c(0, 2)) +
  facet_wrap(~cidi_name) +
  theme_bw() +
  guides(color = guide_legend(reverse=T),
         shape = guide_legend(reverse=T))+
  xlab("Number of Traumatic Events")+
  ylab("OR/RR (99.7% CI)") +
  theme(legend.position = "bottom") +
  ggtitle("Model C: Continuous")

df_gg <- df_trauma_only %>% filter(model == "Model D")
df_gg$OR_or_RR <- as.numeric(as.character(df_gg$OR_or_RR))
df_gg$ci_ll <- as.numeric(as.character(df_gg$ci_ll))
df_gg$ci_ul <- as.numeric(as.character(df_gg$ci_ul))

ggplot(df_gg, aes(x=coef_name, y=OR_or_RR, ymin=ci_ll, ymax=ci_ul,
                  shape=trauma_type, color=significant)) +
  geom_pointrange(position=position_dodge(0.5))+

```



```

geom_hline(yintercept=1, lty=2)+
coord_flip()+
scale_color_manual(breaks=c("Not significant", "Significant"),
                    values=c("black", "red"),
                    name = "Significance") +
scale_shape_manual(values = c(16, 17),
                   name = "Trauma Type") +
scale_y_continuous(breaks = seq(0, 4, by = 1), limits = c(0, 4)) +
facet_wrap(~cidi_name) +
theme_bw() +
  guides(color = guide_legend(reverse=T),
         shape = guide_legend(reverse=T))+
xlab("Number of Traumatic Events")+
ylab("OR/RR (99.7% CI)")+
theme(legend.position = "bottom") +
ggtitle("Model D: Categorical + Substance Use")

df_gg <- df_trauma_only %>% filter(model == "Model E")
df_gg$OR_or_RR <- as.numeric(as.character(df_gg$OR_or_RR))
df_gg$ci_ll <- as.numeric(as.character(df_gg$ci_ll))
df_gg$ci_ul <- as.numeric(as.character(df_gg$ci_ul))

ggplot(df_gg, aes(x=coef_name, y=OR_or_RR, ymin=ci_ll, ymax=ci_ul,
                  shape=trauma_type, color=significant)) +
  geom_pointrange(position=position_dodge(0.5))+
  geom_hline(yintercept=1, lty=2)+
  coord_flip()+
  scale_color_manual(breaks=c("Not significant", "Significant"),
                    values=c("black", "red"),
                    name = "Significance") +
  scale_shape_manual(values = c(16, 17),
                    name = "Trauma Type") +
  scale_y_continuous(breaks = seq(0, 4, by = 1), limits = c(0, 2)) +
  facet_wrap(~cidi_name) +
  theme_bw() +
    guides(color = guide_legend(reverse=T),
         shape = guide_legend(reverse=T))+
  xlab("Number of Traumatic Events")+
  ylab("OR/RR (99.7% CI)")+
  theme(legend.position = "bottom") +
  ggtitle("Model E: Continuous + Substance Use")

dev.off()

```

```

## pdf
## 2

```

**Model B only (current)** Andrew 01-17-25 Model B: I imagine is the main results figure. Model C, D, and E's results will be mentioned in the text. Model B figure needs to be updated in the following ways to be ready for publication: i. Need to add a reference point for 0 traumas ii. Y-axis ticks: Just label 0, 1,

2, 3+. Do NOT label them (“0 Traumas”, “1 Trauma”, “2 Traumas”, etc). iii. Since I think the focus is on Interpersonal trauma, change the order of the trauma types in the plots. Interpersonal trauma should be placed above non-interpersonal trauma. Make sure the order of the names in the legend match the order in the plots so that non-interpersonal is first in the legend. iv. Facet names: “An ulcer in your stomach or intestine” and “Seasonal allergies like hay fever” should be shortened. v. For a final publication version, remove the title (“Model B...”) vi. Change Figure to having 2 rows with 4 columns rather than 3x3 to minimize white space.

Andrew - 1-30-25

You are almost there with the Figure. Please make the following changes:

Remove the Title (“Associations of Interpersonal...” Remove the other text in the bottom right Remake the figure to be 2 rows by 4 columns rather than 4 rows by 2 columns Reorder the outcomes by Pain (row 1, column 1) Ulcers (row 1, column 2) Allergies (row 1, column 3) Neurological (row 1, column 4) Cardiometabolic (row 2, column 1) Respiratory (row 2, column 2) Hypothyroidism (row 2 column 3). Chronic conditions (row 2, column 4) In the Trauma Type legend, make sure Interpersonal comes before Non-Interpersonal since Interpersonal is above Non-interpersonal in the plots (which is the way it should be). Relabel the facet “Chronic Condition Count” to “Chronic condition count” to be consistent with the capitalization

Best,

Andrew

**Note: df\_gg\_temp created here – important for full figure**

```
#Ensure correct path
setwd("~/Dropbox (Harvard University)/NeuroMex/Data Analysis/Papers/LEC-CIDI Paper/LEC-CIDI Paper/")

df_gg <- df_trauma_only %>% filter(model == "Model B")

#Add reference rows for all in cidi_name
reference_row_total <- NULL
for (i in unique(df_gg$cidi_name)) {

  reference_row <- data.frame(cidi_q = NA,
                             Coefficient = NA,
                             prevalence = NA,
                             OR_or_RR = 1,
                             ci_ll = 1, ci_ul = 1,
                             p = NA, significant = "Not significant",
                             model="Model B",
                             cidi_var_n = NA,
                             type = NA,
                             regression_type = NA,
                             cidi_name = i,
                             coef_name = "0",
                             trauma_type = "Interpersonal")

  reference_row2 <- data.frame(cidi_q = NA,
                              Coefficient = NA,
                              prevalence = NA,
                              OR_or_RR = 1,
                              ci_ll = 1, ci_ul = 1,
                              p = NA, significant = "Not significant",
                              model="Model B",
```

```

        cidi_var_n = NA,
        type = NA,
        regression_type = NA,
        cidi_name = i,
        coef_name = "0",
        trauma_type = "Non-interpersonal")

reference_row_3 <- rbind(reference_row, reference_row2)
reference_row_total <- rbind(reference_row_total, reference_row_3)
}

#Ensure data is correct type
#df_gg <- df_trauma_only %>% filter(model == "Model B")
df_gg <- rbind(df_gg, reference_row_total)
df_gg$coef_name <- factor(df_gg$coef_name, levels = c("0", "1", "2", "3+"))
df_gg$OR_or_RR <- as.numeric(as.character(df_gg$OR_or_RR))
df_gg$ci_ll <- as.numeric(as.character(df_gg$ci_ll))
df_gg$ci_ul <- as.numeric(as.character(df_gg$ci_ul))

#Adjust to releavel the data
cidi_name_order <- c("Pain", "Stomach/intestine ulcer",
                    "Seasonal allergies", "Neurological",
                    "Cardiometabolic", "Respiratory", "Hypothyroidism")

df_gg$cidi_name <- factor(df_gg$cidi_name, levels = cidi_name_order)

#Save this data for use later
#Will combine this with negative binomial model later
df_gg_temp <- df_gg

#Create the figure
pdf(paste0("Compiled results_ModelB_",today(),".pdf"), width=10, height=8)

ggplot(df_gg, aes(x=coef_name, y=OR_or_RR, ymin=ci_ll, ymax=ci_ul,
                 shape=trauma_type, color=significant)) +
  geom_pointrange(position=position_dodge(0.5))+
  geom_hline(yintercept=1, lty=2)+
  coord_flip()+
  scale_color_manual(breaks=c("Not significant", "Significant"),
                    values=c("black", "red"),
                    name = "Significance") +
  scale_shape_manual(values = c(16, 17),
                    name = "Trauma Type") +
  scale_y_continuous(breaks = seq(0, 4, by = 1), limits = c(0, 4)) +
  facet_wrap(~cidi_name, nrow=2) +
  theme_bw() +
  guides(color = guide_legend(reverse=T),
         shape = guide_legend(reverse=T))+

```

```
xlab("Number of Traumatic Events Experienced")+
ylab("OR/RR (99.7% CI)")+
theme(legend.position = "bottom")

dev.off()
```

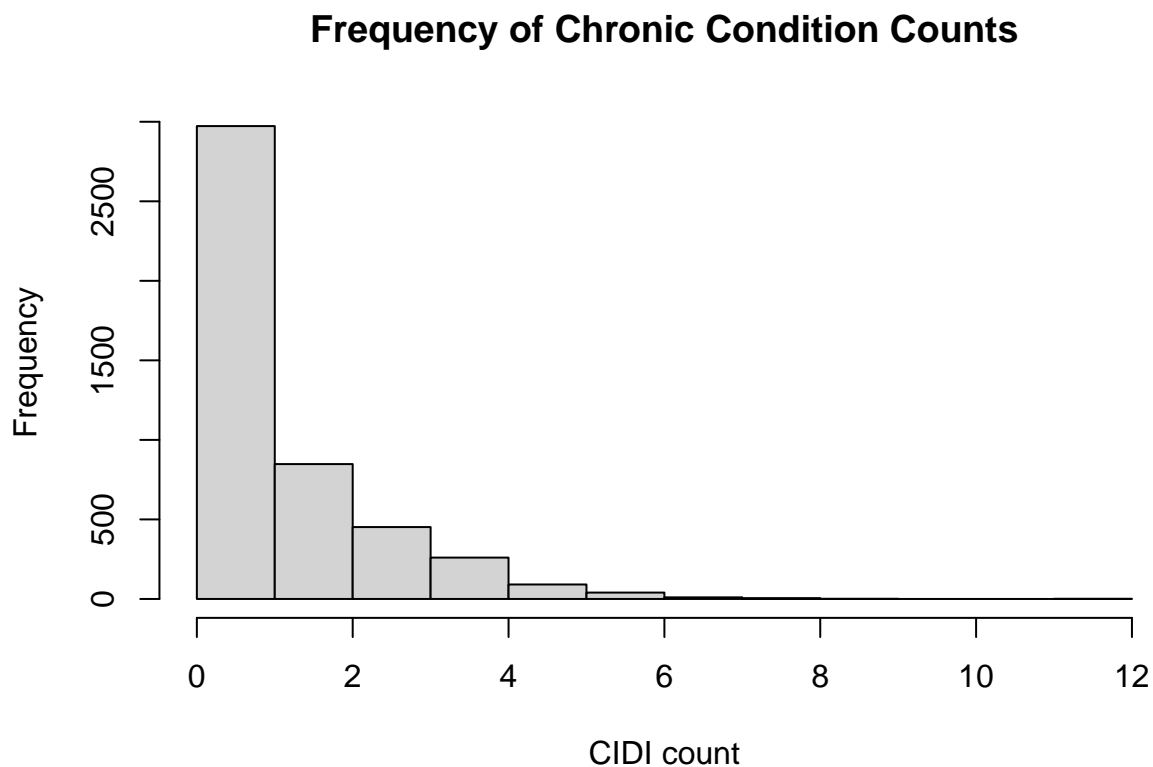
```
## pdf
## 2
```

## Model: Poisson (old)

Here we run a poisson regression on chronic condition count

**Histogram** He we create a quick histogram of chronic condition counts.

```
hist(nmex_cases$cidi_load, main = "Frequency of Chronic Condition Counts",
     xlab = "CIDI count")
```

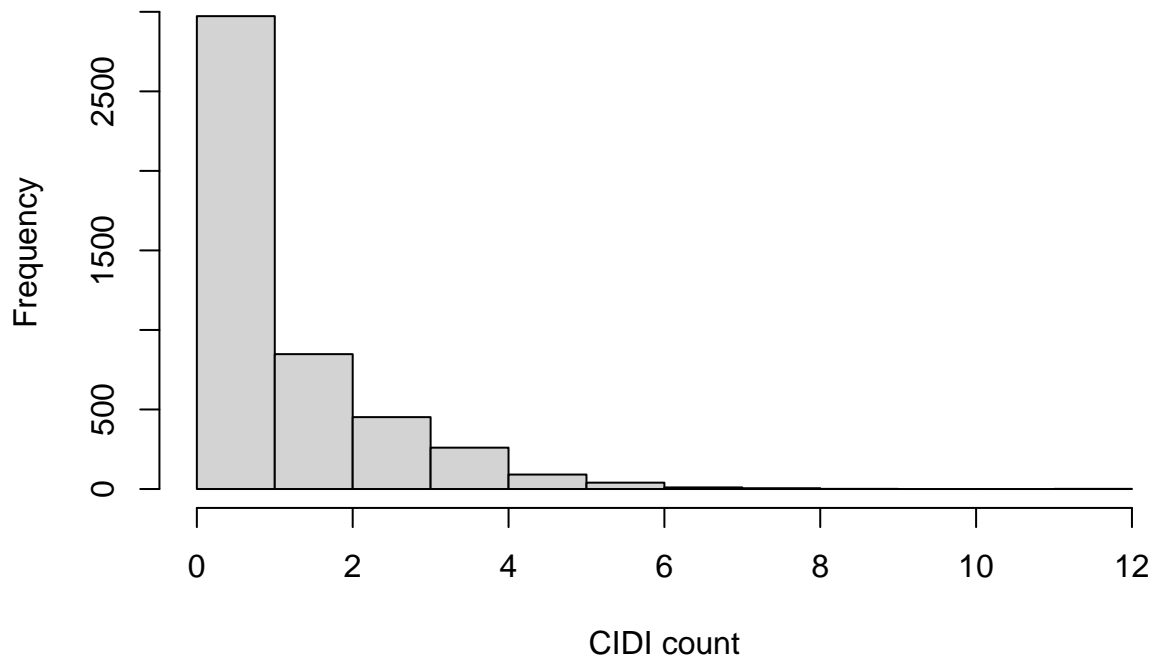


```
hist(nmex_cases$cidi_load, main = "Frequency of Chronic Condition Counts",
     xlab = "CIDI count")
```

Table 31: Distribution of CIDI Count by Interpersonal Trauma Count

	0	1	2	>=3	Overall
	(N=2065)	(N=1189)	(N=796)	(N=631)	(N=4681)
<b>factor(cidi_load)</b>					
0	795 (38.5%)	430 (36.2%)	234 (29.4%)	168 (26.6%)	1627 (34.8%)
1	604 (29.2%)	357 (30.0%)	219 (27.5%)	166 (26.3%)	1346 (28.8%)
2	365 (17.7%)	198 (16.7%)	158 (19.8%)	127 (20.1%)	848 (18.1%)
3	158 (7.7%)	122 (10.3%)	97 (12.2%)	75 (11.9%)	452 (9.7%)
4	98 (4.7%)	53 (4.5%)	62 (7.8%)	47 (7.4%)	260 (5.6%)
5	27 (1.3%)	19 (1.6%)	18 (2.3%)	27 (4.3%)	91 (1.9%)
6	12 (0.6%)	8 (0.7%)	5 (0.6%)	15 (2.4%)	40 (0.9%)
7	4 (0.2%)	1 (0.1%)	1 (0.1%)	4 (0.6%)	10 (0.2%)
8	1 (0.0%)	1 (0.1%)	1 (0.1%)	2 (0.3%)	5 (0.1%)
9	0 (0%)	0 (0%)	1 (0.1%)	0 (0%)	1 (0.0%)
12	1 (0.0%)	0 (0%)	0 (0%)	0 (0%)	1 (0.0%)

### Frequency of Chronic Condition Counts



```
table1(~ factor(cidi_load) | trauma_load_inter_cat, data=nmx_cases, caption = "Distribution of CIDI Co
```

```
table1(~ factor(cidi_load) | trauma_load_noninter_nhs_cat, data=nmx_cases, caption = "Distribution of C
```

Table 32: Distribution of CIDI Count by Non-interpersonal Trauma Count

	0	1	2	>=3	Overall
	(N=2286)	(N=1340)	(N=646)	(N=409)	(N=4681)
<b>factor(cidi_load)</b>					
0	848 (37.1%)	488 (36.4%)	184 (28.5%)	107 (26.2%)	1627 (34.8%)
1	678 (29.7%)	356 (26.6%)	195 (30.2%)	117 (28.6%)	1346 (28.8%)
2	404 (17.7%)	257 (19.2%)	110 (17.0%)	77 (18.8%)	848 (18.1%)
3	201 (8.8%)	127 (9.5%)	78 (12.1%)	46 (11.2%)	452 (9.7%)
4	104 (4.5%)	75 (5.6%)	46 (7.1%)	35 (8.6%)	260 (5.6%)
5	29 (1.3%)	24 (1.8%)	24 (3.7%)	14 (3.4%)	91 (1.9%)
6	13 (0.6%)	10 (0.7%)	7 (1.1%)	10 (2.4%)	40 (0.9%)
7	6 (0.3%)	1 (0.1%)	1 (0.2%)	2 (0.5%)	10 (0.2%)
8	2 (0.1%)	1 (0.1%)	1 (0.2%)	1 (0.2%)	5 (0.1%)
9	0 (0%)	1 (0.1%)	0 (0%)	0 (0%)	1 (0.0%)
12	1 (0.0%)	0 (0%)	0 (0%)	0 (0%)	1 (0.0%)

**Poisson regressions (old)**

He were create using Poisson regression outcomes for each model covariates

We treat chronic conditions (count) as the outcome variable.

**Figure** Here we create a figure compiling the above data.

**Model Poission 1-13-25 (old)**

Here we are updating the above code to match the bonferonni adjustments

**Format data for tables**

**Figure of Poisson regression (old)** Here we create a figure compiling the above data for only Model B.

**Update 01-21-2025 (old)**

Here I will run the Poisson analysis for count data on model B only, as this seems like the most likely main model for our paper. I will run tests on the model for overdispersion and heteroskedacity.

Here we see that there is overdispersion

We can confirm this in this plot of Residuals vs Fitted values by the fan-shaped pattern.

It might be worth trying a quasipoisson model for account for this overdispersion.

Option 1: Keep poisson and use RSE Option 2: Use quasi-poisson to account for overdispersion Option 3: Option 2 plus RSE? Option 4: Zero-inflated regression model

## NB GLM 01-27-2025 (current)

After showing these above tests to Andrew, he confirmed that there does appear to be overdispersion. He suggests running a negative binomial regression instead. I do this below.

### Calculate data NBGLM

Calculate the negative binomial GLM for all models.

```
results <- NULL
df <- NULL
models <- c("Model B", "Model C", "Model D", "Model E")

#All Trauma

for (model in models) {

  if (model == "Model B") {
    trauma_var <- "trauma_load_inter_cat + trauma_load_noninter_nhs_cat"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
site_location_group_f"
    type <- "Categorical"
  } else if (model == "Model C") {
    trauma_var <- "trauma_load_interpersonal_1 + trauma_load_noninter_1_nhs"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
site_location_group_f"
    type <- "Numeric"
  } else if (model == "Model D") {
    trauma_var <- "trauma_load_inter_cat + trauma_load_noninter_nhs_cat"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
site_location_group_f + assist_tobacco_amt_recode +
assist_alcohol_amt_recode + assist_cannabis_amt_recode"
    type <- "Categorical"
  } else if (model == "Model E") {
    trauma_var <- "trauma_load_interpersonal_1 + trauma_load_noninter_1_nhs"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
assist_tobacco_amt_recode + assist_alcohol_amt_recode +
assist_cannabis_amt_recode + site_location_group_f"
    type <- "Numeric"
  }

  fmla <- as.formula(paste("cidi_load", " ~ ",
trauma_var, covar))

  # Fit the negative binomial regression model
  fit <- glm.nb(fmla, data = nmex_cases)

  #####
  print(model)

  fixef <- exp(coef(fit))
```

```

conf <- exp(confint(fit, level = (1 - 0.05 / bon_cor)))
conf <- exp(confint(fit))
conf_complete <- conf[complete.cases(conf), ]

var <- rownames(conf_complete)

for (j in 1:length(var)) {

  var_temp <- var[j]

  est      <- round(exp(summary(fit)[["coefficients"]][var[j],
                                                "Estimate"])),3)

  ci_ll    <- round(conf_complete[var[j], 1],3)
  ci_ul    <- round(conf_complete[var[j], 2],3)
  ci <- paste0("(", ci_ll, ", ", ci_ul, ")")

  p        <- summary(fit)$coefficients[var[j], "Pr(>|z|)"]
  model <- model

  if (p < (0.05 / bon_cor)) {
    significant <- "Significant"
  } else {
    significant <- "Not significant"
  }

  results <- rbind(results,c(model, var_temp, est, ci_ll, ci_ul, p, significant,
                             type))
}
}

```

```
## [1] "Model B"
```

```
## Waiting for profiling to be done...
## Waiting for profiling to be done...
```

```
## [1] "Model C"
```

```
## Waiting for profiling to be done...
## Waiting for profiling to be done...
```

```
## [1] "Model D"
```

```
## Waiting for profiling to be done...
## Waiting for profiling to be done...
```

```
## [1] "Model E"
```

```
## Waiting for profiling to be done...
## Waiting for profiling to be done...
```



```

df <- as.data.frame(results)

colnames(df)<-c("model","Coefficient","RR", "ci_ll", "ci_ul","p", "significant",
              "type")

df <- df %>% mutate(

  coef_name = factor(case_when(
    Coefficient == "trauma_load_inter_cat1" ~ "1",
    Coefficient == "trauma_load_inter_cat2" ~ "2",
    Coefficient == "trauma_load_inter_cat>=3" ~ "3+",
    Coefficient == "trauma_load_noninter_nhs_cat1" ~ "1",
    Coefficient == "trauma_load_noninter_nhs_cat2" ~ "2",
    Coefficient == "trauma_load_noninter_nhs_cat>=3" ~ "3+",
    Coefficient == "trauma_load_noninter_nhs_cat>=3" ~ "3+",
    Coefficient == "trauma_load_interpersonal_1" ~ "Continuous",
    Coefficient == "trauma_load_noninter_1_nhs" ~ "Continuous"
  ), levels = c("1",
                "2",
                "3+",
                "Continuous")),

  type = factor(type, levels = c("Numeric", "Categorical")),

  trauma_type = factor(case_when(
    grepl("noninter", Coefficient) ~ "Non-interpersonal",
    grepl("inter_|interpersonal", Coefficient) ~ "Interpersonal"
  ), levels = c("Non-interpersonal", "Interpersonal"))
)

df_trauma_only <- df %>%
  filter(str_starts(Coefficient, "trauma_load"))

```

Format data

Figure NB GLM Model B (current)

Here I create the figure for Model B's chronic condition count using the output produced above.

```

setwd("~/Dropbox (Harvard University)/NeuroMex/Data Analysis/Papers/LEC-CIDI Paper/LEC-CIDI Paper/")

pdf("NB_ModelB.pdf", width = 8, height = 6)

df_trauma_onlyB <- df_trauma_only %>% filter(model == "Model B")
df_trauma_onlyB$RR <- as.numeric(df_trauma_onlyB$RR)
df_trauma_onlyB$ci_ll <- as.numeric(df_trauma_onlyB$ci_ll)
df_trauma_onlyB$ci_ul <- as.numeric(df_trauma_onlyB$ci_ul)

```

```

ggplot(df_trauma_onlyB, aes(x = coef_name, y = RR, ymin = ci_ll,
                           ymax = ci_ul, shape=trauma_type, color=significant)) +
  geom_pointrange(position = position_dodge(width = 0.7)) +
  scale_color_manual(breaks=c("Not significant", "Significant"),
                    values=c("black", "red"),
                    name = "Significance") +
  scale_shape_manual(values = c(16, 17),
                    name = "Trauma Type") +
  geom_hline(yintercept = 1, lty = 2) +
  theme_bw() +
  theme(
    axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1),
    legend.position = "right"
  ) +
  guides(color = guide_legend(reverse=T),
         shape = guide_legend(reverse=T)) +
  scale_y_continuous(breaks = seq(0, 1.75, by = 0.25), limits = c(0.75, 1.6)) +
  labs(
    y = "RR (99.7% CI)",
    x = "Number of Traumatic Events Experienced",
    color = "Trauma Type"
  ) + coord_flip() +
  theme(legend.position = "bottom")

dev.off()

```

```

## pdf
## 2

```

**Full Figure Model B + NBGLM (current)** Here I combine data from the Logistic Regression/Poisson Regression with the data from the negative binomial regression.

I also manually add reference points of “0” traumas with OR/ci\_ll/ci\_ul of 1. Finally I relevel the data so the facets are in the correct order.

```

df_trauma_onlyB <- df_trauma_only %>% filter(model == "Model B")
df_trauma_onlyB$OR_or_RR <- df_trauma_onlyB$RR
df_trauma_onlyB$prevalence <- NA
df_trauma_onlyB$type <- NA
df_trauma_onlyB$prevalence <- NA
df_trauma_onlyB$regression_type <- NA
df_trauma_onlyB$cidi_name <- "Chronic condition count"
df_trauma_onlyB$cidi_q <- NA
df_trauma_onlyB$cidi_var_n <- NA
df_trauma_onlyB <- df_trauma_onlyB %>% dplyr::select(-RR)

reference_row <- data.frame(cidi_q = NA,
                           Coefficient = NA,
                           prevalence = NA,
                           OR_or_RR = 1,
                           ci_ll = 1, ci_ul = 1,

```

```

        p = NA, significant = "Not significant",
        model="Model B",
        cidi_var_n = NA,
        type = NA,
        regression_type = NA,
        cidi_name = "Chronic condition count",
        coef_name = "0",
        trauma_type = "Non-interpersonal")
df_trauma_onlyB <- rbind(df_trauma_onlyB, reference_row)

reference_row <- data.frame(cidi_q = NA,
        Coefficient = NA,
        prevalence = NA,
        OR_or_RR = 1,
        ci_ll = 1, ci_ul = 1,
        p = NA, significant = "Not significant",
        model="Model B",
        cidi_var_n = NA,
        type = NA,
        regression_type = NA,
        cidi_name = "Chronic condition count",
        coef_name = "0",
        trauma_type = "Interpersonal")
df_trauma_onlyB <- rbind(df_trauma_onlyB, reference_row)

df_gg_temp_nb <- rbind(df_gg_temp, df_trauma_onlyB)

df_gg_temp_nb$coef_name <- factor(df_gg_temp_nb$coef_name, levels = c("0", "1", "2", "3+"))
df_gg_temp_nb$OR_or_RR <- as.numeric(as.character(df_gg_temp_nb$OR_or_RR))
df_gg_temp_nb$ci_ll <- as.numeric(as.character(df_gg_temp_nb$ci_ll))
df_gg_temp_nb$ci_ul <- as.numeric(as.character(df_gg_temp_nb$ci_ul))

#Adjust to relelevel the data
cidi_name_order <- c("Pain", "Stomach/intestine ulcer",
        "Seasonal allergies", "Neurological",
        "Cardiometabolic", "Respiratory", "Hypothyroidism", "Chronic condition count")

df_gg_temp_nb$cidi_name <- factor(df_gg_temp_nb$cidi_name, levels = cidi_name_order)

```

Here I produce the figure. The facets will be in 2 rows, with 4 columns.

```

setwd("~/Dropbox (Harvard University)/NeuroMex/Data Analysis/Papers/LEC-CIDI Paper/LEC-CIDI Paper/")

pdf(paste0("Compiled results_ModelB_withCCcount", today(), ".pdf"), width=10, height=8)

ggplot(df_gg_temp_nb, aes(x=coef_name, y=OR_or_RR, ymin=ci_ll, ymax=ci_ul,
        shape=trauma_type, color=significant)) +

```

```

geom_pointrange(position=position_dodge(0.5))+
geom_hline(yintercept=1, lty=2)+
coord_flip()+
scale_color_manual(breaks=c("Not significant", "Significant"),
                    values=c("black", "red"),
                    name = "Significance") +
scale_shape_manual(values = c(16, 17),
                   name = "Trauma Type") +
scale_y_continuous(breaks = seq(0, 4, by = 1), limits = c(0, 4)) +
facet_wrap(~cidi_name, nrow=2) +
theme_bw() +
guides(color = guide_legend(reverse=T),
       shape = guide_legend(reverse=T))+
xlab("Number of Traumatic Events Experienced")+
ylab("OR/RR (99.7% CI)")+
theme(legend.position = "bottom")

dev.off()

```

```

## pdf
## 2

```

## Current models with 4+ traumas

### Regressions 4+ traumas

These are the same current models as above, just run with 4+ trauma categories.

```

cidi_list_binary <- c(
  "cidi_q5_binary", "cidi_q14_binary", "cidi_q18_binary",
  "cidi_pain_binary", "cidi_cardiac_binary",
  "cidi_resp_binary", "cidi_neur_binary"
)

bon_cor <- length(cidi_list_binary) + 1

results <- NULL
df <- NULL

models <- c("Model B", "Model C", "Model D", "Model E")

# All Trauma

for (model in models) {
  if (model == "Model B") {
    trauma_var <- "trauma_load_inter_cat_4 + trauma_load_noninter_nhs_cat_4"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
    site_location_group_f"

```

```

type <- "Categorical"
} else if (model == "Model C") {
  trauma_var <- "trauma_load_interpersonal_1 + trauma_load_noninter_1_nhs"
  covar <- "+ age_at_interview + is_male_f + ses_status_f +
  site_location_group_f"
  type <- "Numeric"
} else if (model == "Model D") {
  trauma_var <- "trauma_load_inter_cat_4 + trauma_load_noninter_nhs_cat_4"
  covar <- "+ age_at_interview + is_male_f + ses_status_f +
  site_location_group_f + factor(assist_tobacco_amt_recode) +
  factor(assist_alcohol_amt_recode) + factor(assist_cannabis_amt_recode)"
  type <- "Numeric"
} else if (model == "Model E") {
  trauma_var <- "trauma_load_interpersonal_1 + trauma_load_noninter_1_nhs"
  covar <- "+ age_at_interview + is_male_f + ses_status_f +
  factor(assist_tobacco_amt_recode) +
  factor(assist_alcohol_amt_recode) +
  factor(assist_cannabis_amt_recode) + site_location_group_f"
  type <- "Numeric"
}

for (ii in 1:length(cidi_list_binary)) {
  #print(model)
  #print(cidi_list_binary[ii])
  tab <- table(nmex_cases[cidi_list_binary[ii]])
  prev <- tab[2] / (tab[1] + tab[2])
  prev_round <- round(prev,3)
  cidi_var <- cidi_list_binary[ii]
  # Get number of cases
  cidi_var_n <- table(nmex_cases[[cidi_var]])[2]

  if (prev <= 0.1) {
    fmla <- as.formula(paste(
      cidi_list_binary[ii], " ~ ",
      trauma_var, covar
    ))
    # Fit the model for logistic regression
    fit <- glm(fmla, data = nmex_cases, family = binomial)

    # get OR
    fixef <- exp(coef(fit))

    # Get confidence interval based on bonferonni
    conf <- exp(confint(fit, level = (1 - 0.05 / bon_cor)))

    # Extract CI of all values
    conf_complete <- conf[complete.cases(conf), ]

    # Get all coefficient names and levels
    var <- rownames(conf_complete)

    # For j in the number of coefficients (var)
    for (j in 1:length(var)) {

```

```

reg_type <- "Logistic Regression"
var_temp <- var[j]

est <- round(exp(summary(fit)[["coefficients"]][
  var[j],
  "Estimate"
]), 3)
# est <- round(exp(summary(fit)[var[j]]), 3)
ci_ll <- round(conf_complete[var[j], 1], 3)
ci_ul <- round(conf_complete[var[j], 2], 3)
ci <- paste0("(", ci_ll, ", ", ci_ul, ")")

p <- summary(fit)$coefficients[var[j], "Pr(>|z|)"]
#print(paste("P val log reg:", p))

model <- model

if (p < (0.05 / bon_cor)) {
  pval <- "Significant"
} else {
  pval <- "Not significant"
}

results <- rbind(results, c(
  cidi_var, var_temp, prev_round,
  est, ci_ll, ci_ul, p, pval,
  model, cidi_var_n, type, reg_type
))
}
} else {
  # if prev > 10%, do relative risk regression
  reg_type <- "Relative Risk"
  fmla <- as.formula(paste(
    cidi_list_binary[ii], " ~ ",
    trauma_var, covar
  ))
  # Fit the model for relative risk regression (poisson, log link).
  fit <- glm(fmla, data = nmex_cases, family = poisson(link = "log"))

  rse <- coeftest(fit, vcov = vcovHC(fit, type = "HC1"))
  q <- qnorm(p = .05 / (2 * bon_cor), lower.tail = FALSE)

  # Extract beta, st_error, and p_val

  estimate_complete <- rse[complete.cases(rse), 1]
  rob_std_error_complete <- rse[complete.cases(rse), 2]
  p_val_rob <- rse[complete.cases(rse), 4]

  var <- rownames(rse)

  for (j in 1:length(var)) {
    var_temp <- var[j]
    beta <- estimate_complete[var[j]]
  }
}

```

```

    est <- round(exp(beta),3)
    robust_se <- rob_std_error_complete[var[j]]
    p <- p_val_rob[var[j]]
    #print(paste("P val rel risk reg:", p))

    if (p < (0.05 / bon_cor)) {
      pval <- "Significant"
    } else {
      pval <- "Not significant"
    }

    ci_ll <- round(exp(beta - q * robust_se),3)
    ci_ul <- round(exp(beta + q * robust_se),3)

    results <- rbind(results, c(
      cidi_var, var_temp, prev_round,
      est, ci_ll, ci_ul, p, pval,
      model, cidi_var_n, type, reg_type
    ))
  }
}
}

#Make results a dataframe and relabel
df <- as.data.frame(results)
colnames(df)<-c("cidi_q", "Coefficient", "prevalence",
  "OR_or_RR", "ci_ll", "ci_ul", "p", "significant",
  "model", "cidi_var_n", "type", "regression_type")

df <- df %>% mutate(cidi_name = case_when(
  cidi_q == "cidi_q1_binary" ~ "Arthritis/Rheumatism",
  cidi_q == "cidi_q2_binary" ~ "Back/neck problems",
  cidi_q == "cidi_q3_binary" ~ "Headaches",
  cidi_q == "cidi_q4_binary" ~ "Other pain",
  cidi_q == "cidi_q5_binary" ~ "Seasonal allergies",
  cidi_q == "cidi_q6_binary" ~ "Stroke",
  cidi_q == "cidi_q7_binary" ~ "Heart attack",
  cidi_q == "cidi_q8_binary" ~ "Heart disease",
  cidi_q == "cidi_q9_binary" ~ "High Blood Pressure",
  cidi_q == "cidi_q10_binary" ~ "Asthma",
  cidi_q == "cidi_q11_binary" ~ "Tuberculosis",
  cidi_q == "cidi_q12_binary" ~ "Lung disease",
  cidi_q == "cidi_q13_binary" ~ "Diabetes",
  cidi_q == "cidi_q14_binary" ~ "Stomach/intestine ulcer",
  cidi_q == "cidi_q15_binary" ~ "HIV/AIDS",
  cidi_q == "cidi_q16_binary" ~ "Epilepsy/Seizures",
  cidi_q == "cidi_q17_binary" ~ "Cancer",
  cidi_q == "cidi_q18_binary" ~ "Hypothyroidism",

```

```

cidi_q == "cidi_pain_binary" ~ "Pain",
cidi_q == "cidi_cardiac_binary" ~ "Cardiometabolic",
cidi_q == "cidi_resp_binary" ~ "Respiratory",
cidi_q == "cidi_neur_binary" ~ "Neurological",

),

coef_name = factor(case_when(
  Coefficient == "trauma_load_inter_cat_41" ~ "1",
  Coefficient == "trauma_load_inter_cat_42" ~ "2",
  Coefficient == "trauma_load_inter_cat_43" ~ "3",
  Coefficient == "trauma_load_inter_cat_44+" ~ "4+",
  Coefficient == "trauma_load_noninter_nhs_cat_41" ~ "1",
  Coefficient == "trauma_load_noninter_nhs_cat_42" ~ "2",
  Coefficient == "trauma_load_noninter_nhs_cat_43" ~ "3",
  Coefficient == "trauma_load_noninter_nhs_cat_44+" ~ "4+",
  Coefficient == "trauma_load_interpersonal_1" ~ "Continuous",
  Coefficient == "trauma_load_noninter_1_nhs" ~ "Continuous"
), levels = c("1",
              "2",
              "3",
              "4+",
              "Continuous")),

type = factor(type, levels = c("Numeric", "Categorical")),

trauma_type = factor(case_when(
  grepl("noninter", Coefficient) ~ "Non-interpersonal",
  grepl("inter_|interpersonal", Coefficient) ~ "Interpersonal"
), levels = c("Non-interpersonal", "Interpersonal"))

)

df_trauma_only <- df %>%
  filter(str_starts(Coefficient, "trauma_load"))

setwd("~/Dropbox (Harvard University)/NeuroMex/Data Analysis/Papers/LEC-CIDI Paper/LEC-CIDI Paper/")

pdf(paste0("Compiled results_ModelBCDE_4cat_", today(), ".pdf"), width=8, height=6)

df_gg <- df_trauma_only %>% filter(model == "Model B")
df_gg$OR_or_RR <- as.numeric(as.character(df_gg$OR_or_RR))
df_gg$ci_ll <- as.numeric(as.character(df_gg$ci_ll))
df_gg$ci_ul <- as.numeric(as.character(df_gg$ci_ul))

ggplot(df_gg, aes(x=coef_name, y=OR_or_RR, ymin=ci_ll, ymax=ci_ul,
                  shape=trauma_type, color=significant)) +
  geom_pointrange(position=position_dodge(0.5))+
  geom_hline(yintercept=1, lty=2)+
  coord_flip()+

```



```

scale_color_manual(breaks=c("Not significant", "Significant"),
                    values=c("black", "red"),
                    name = "Significance") +
scale_shape_manual(values = c(16, 17),
                   name = "Trauma Type") +
scale_y_continuous(breaks = seq(0, 5.5, by = 1), limits = c(0, 5.5)) +
facet_wrap(~cidi_name) +
theme_bw() +
guides(color = guide_legend(reverse=T))+
xlab("Number of Traumatic Events")+
ylab("OR/RR (99.7% CI)")+
theme(legend.position = "bottom") +
ggtitle("Model B: Categorical")

df_gg <- df_trauma_only %>% filter(model == "Model C")
df_gg$OR_or_RR <- as.numeric(as.character(df_gg$OR_or_RR))
df_gg$ci_ll <- as.numeric(as.character(df_gg$ci_ll))
df_gg$ci_ul <- as.numeric(as.character(df_gg$ci_ul))

ggplot(df_gg, aes(x=coef_name, y=OR_or_RR, ymin=ci_ll, ymax=ci_ul,
                  shape=trauma_type, color=significant)) +
  geom_pointrange(position=position_dodge(0.5))+
  geom_hline(yintercept=1, lty=2)+
  coord_flip()+
  scale_color_manual(breaks=c("Not significant", "Significant"),
                    values=c("black", "red"),
                    name = "Significance") +
  scale_shape_manual(values = c(16, 17),
                    name = "Trauma Type") +
  scale_y_continuous(breaks = seq(0, 5.5, by = 1), limits = c(0, 5.5)) +
  facet_wrap(~cidi_name) +
  theme_bw() +
  guides(color = guide_legend(reverse=T),
         shape = guide_legend(reverse=T))+
  xlab("Number of Traumatic Events")+
  ylab("OR/RR (99.7% CI)")+
  theme(legend.position = "bottom") +
  ggtitle("Model C: Continuous")

df_gg <- df_trauma_only %>% filter(model == "Model D")
df_gg$OR_or_RR <- as.numeric(as.character(df_gg$OR_or_RR))
df_gg$ci_ll <- as.numeric(as.character(df_gg$ci_ll))
df_gg$ci_ul <- as.numeric(as.character(df_gg$ci_ul))

ggplot(df_gg, aes(x=coef_name, y=OR_or_RR, ymin=ci_ll, ymax=ci_ul,
                  shape=trauma_type, color=significant)) +
  geom_pointrange(position=position_dodge(0.5))+
  geom_hline(yintercept=1, lty=2)+
  coord_flip()+
  scale_color_manual(breaks=c("Not significant", "Significant"),
                    values=c("black", "red"),

```

```

        name = "Significance") +
  scale_shape_manual(values = c(16, 17),
    name = "Trauma Type") +
scale_y_continuous(breaks = seq(0, 5.5, by = 1), limits = c(0, 5.5)) +
facet_wrap(~cidi_name) +
theme_bw() +
  guides(color = guide_legend(reverse=T),
    shape = guide_legend(reverse=T))+
xlab("Number of Traumatic Events")+
ylab("OR/RR (99.7% CI)")+
theme(legend.position = "bottom") +
ggtitle("Model D: Categorical + Substance Use")

df_gg <- df_trauma_only %>% filter(model == "Model E")
df_gg$OR_or_RR <- as.numeric(as.character(df_gg$OR_or_RR))
df_gg$ci_ll <- as.numeric(as.character(df_gg$ci_ll))
df_gg$ci_ul <- as.numeric(as.character(df_gg$ci_ul))

ggplot(df_gg, aes(x=coef_name, y=OR_or_RR, ymin=ci_ll, ymax=ci_ul,
  shape=trauma_type, color=significant)) +
  geom_pointrange(position=position_dodge(0.5))+
  geom_hline(yintercept=1, lty=2)+
  coord_flip()+
  scale_color_manual(breaks=c("Not significant", "Significant"),
    values=c("black", "red"),
    name = "Significance") +
  scale_shape_manual(values = c(16, 17),
    name = "Trauma Type") +
  scale_y_continuous(breaks = seq(0, 5.5, by = 1), limits = c(0, 5.5)) +
  facet_wrap(~cidi_name) +
  theme_bw() +
    guides(color = guide_legend(reverse=T),
      shape = guide_legend(reverse=T))+
  xlab("Number of Traumatic Events")+
  ylab("OR/RR (99.7% CI)")+
  theme(legend.position = "bottom") +
  ggtitle("Model E: Continuous + Substance Use")

dev.off()

```

```

## pdf
## 2

```

Temporary df for models and figure

```

df_gg <- df_trauma_only %>% filter(model == "Model B")

#Add reference rows for all in cidi_name
reference_row_total <- NULL
for (i in unique(df_gg$cidi_name)) {

```

```

reference_row <- data.frame(cidi_q = NA,
                           Coefficient = NA,
                           prevalence = NA,
                           OR_or_RR = 1,
                           ci_ll = 1, ci_ul = 1,
                           p = NA, significant = "Not significant",
                           model="Model B",
                           cidi_var_n = NA,
                           type = NA,
                           regression_type = NA,
                           cidi_name = i,
                           coef_name = "0",
                           trauma_type = "Interpersonal")

reference_row2 <- data.frame(cidi_q = NA,
                            Coefficient = NA,
                            prevalence = NA,
                            OR_or_RR = 1,
                            ci_ll = 1, ci_ul = 1,
                            p = NA, significant = "Not significant",
                            model="Model B",
                            cidi_var_n = NA,
                            type = NA,
                            regression_type = NA,
                            cidi_name = i,
                            coef_name = "0",
                            trauma_type = "Non-interpersonal")

reference_row_3 <- rbind(reference_row, reference_row2)
reference_row_total <- rbind(reference_row_total, reference_row_3)
}

#Ensure data is correct type
df_gg <- df_trauma_only %>% filter(model == "Model B")
df_gg <- rbind(df_gg, reference_row_total)
df_gg$coef_name <- factor(df_gg$coef_name, levels = c("0", "1", "2", "3", "4+"))
df_gg$OR_or_RR <- as.numeric(as.character(df_gg$OR_or_RR))
df_gg$ci_ll <- as.numeric(as.character(df_gg$ci_ll))
df_gg$ci_ul <- as.numeric(as.character(df_gg$ci_ul))

#Adjust to relevele the data
cidi_name_order <- c("Pain", "Stomach/intestine ulcer",
                    "Seasonal allergies", "Neurological",
                    "Cardiometabolic", "Respiratory", "Hypothyroidism")

df_gg$cidi_name <- factor(df_gg$cidi_name, levels = cidi_name_order)

#Save this data for use later
#Will combine this with negative binomial model later

```

```
df_gg_temp <- df_gg
```

## NB GLM 01-27-2025 (current)

After showing these above tests to Andrew, he confirmed that there does appear to be overdispersion. He suggests running a negative binomial regression instead. I do this below.

## Calculate data NBGLM

Calculate the negative binomial GLM for all models.

```
results <- NULL
df <- NULL
models <- c("Model B", "Model C", "Model D", "Model E")

#All Trauma

for (model in models) {

  if (model == "Model B") {
    trauma_var <- "trauma_load_inter_cat_4 + trauma_load_noninter_nhs_cat_4"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
    site_location_group_f"
    type <- "Categorical"
  } else if (model == "Model C") {
    trauma_var <- "trauma_load_interpersonal_1 + trauma_load_noninter_1_nhs"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
    site_location_group_f"
    type <- "Numeric"
  } else if (model == "Model D") {
    trauma_var <- "trauma_load_inter_cat_4 + trauma_load_noninter_nhs_cat_4"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
    site_location_group_f + assist_tobacco_amt_recode +
    assist_alcohol_amt_recode + assist_cannabis_amt_recode"
    type <- "Categorical"
  } else if (model == "Model E") {
    trauma_var <- "trauma_load_interpersonal_1 + trauma_load_noninter_1_nhs"
    covar <- "+ age_at_interview + is_male_f + ses_status_f +
    assist_tobacco_amt_recode + assist_alcohol_amt_recode +
    assist_cannabis_amt_recode + site_location_group_f"
    type <- "Numeric"
  }

  fmla <- as.formula(paste("cidi_load", " ~ ",
                           trauma_var, covar))

  # Fit the negative binomial regression model
  fit <- glm.nb(fmla, data = nmex_cases)
```

```
#####
print(model)

fixef <- exp(coef(fit))
conf <- exp(confint(fit, level = (1 - 0.05 / bon_cor)))
conf <- exp(confint(fit))
conf_complete <- conf[complete.cases(conf), ]

var <- rownames(conf_complete)

for (j in 1:length(var)) {

  var_temp <- var[j]

  est      <- round(exp(summary(fit)[["coefficients"]][var[j],
                                                                    "Estimate"]),3)

  ci_ll    <- round(conf_complete[var[j], 1],3)
  ci_ul    <- round(conf_complete[var[j], 2],3)
  ci <- paste0("(", ci_ll, ", ", ci_ul, ")")

  p        <- summary(fit)$coefficients[var[j], "Pr(>|z|)"]
  model <- model

  if (p < (0.05 / bon_cor)) {
    significant <- "Significant"
  } else {
    significant <- "Not significant"
  }

  results <- rbind(results,c(model, var_temp, est, ci_ll, ci_ul, p, significant,
                             type))
}
}
```

```
## [1] "Model B"
```

```
## Waiting for profiling to be done...
```

```
## Waiting for profiling to be done...
```

```
## [1] "Model C"
```

```
## Waiting for profiling to be done...
```

```
## Waiting for profiling to be done...
```

```
## [1] "Model D"
```

```
## Waiting for profiling to be done...
```

```
## Waiting for profiling to be done...
```

```
## [1] "Model E"
```

```
## Waiting for profiling to be done...
## Waiting for profiling to be done...
```

```
df <- as.data.frame(results)

colnames(df)<-c("model","Coefficient","RR", "ci_ll", "ci_ul","p", "significant",
              "type")

df <- df %>% mutate(

  coef_name = factor(case_when(
    Coefficient == "trauma_load_inter_cat_41" ~ "1",
    Coefficient == "trauma_load_inter_cat_42" ~ "2",
    Coefficient == "trauma_load_inter_cat_43" ~ "3",
    Coefficient == "trauma_load_inter_cat_44+" ~ "4+",
    Coefficient == "trauma_load_noninter_nhs_cat_41" ~ "1",
    Coefficient == "trauma_load_noninter_nhs_cat_42" ~ "2",
    Coefficient == "trauma_load_noninter_nhs_cat_43" ~ "3",
    Coefficient == "trauma_load_noninter_nhs_cat_44+" ~ "4+",
    Coefficient == "trauma_load_interpersonal_1" ~ "Continuous",
    Coefficient == "trauma_load_noninter_1_nhs" ~ "Continuous"
  ), levels = c("1",
                "2",
                "3",
                "4+",
                "Continuous")),

  type = factor(type, levels = c("Numeric", "Categorical")),

  trauma_type = factor(case_when(
    grepl("noninter", Coefficient) ~ "Non-interpersonal",
    grepl("inter_|interpersonal", Coefficient) ~ "Interpersonal"
  ), levels = c("Non-interpersonal", "Interpersonal"))
)

df_trauma_only <- df %>%
  filter(str_starts(Coefficient, "trauma_load"))
```

### Figure NB GLM Model B (current)

Here I create the figure for Model B's chronic condition count using the output produced above.

```
setwd("~/Dropbox (Harvard University)/NeuroMex/Data Analysis/Papers/LEC-CIDI Paper/LEC-CIDI Paper/")

pdf("NB_ModelB.pdf", width = 8, height = 6)

df_trauma_onlyB <- df_trauma_only %>% filter(model == "Model B")
df_trauma_onlyB$RR <- as.numeric(df_trauma_onlyB$RR)
df_trauma_onlyB$ci_ll <- as.numeric(df_trauma_onlyB$ci_ll)
```

```

df_trauma_onlyB$ci_ul <- as.numeric(df_trauma_onlyB$ci_ul)

ggplot(df_trauma_onlyB, aes(x = coef_name, y = RR, ymin = ci_ll,
                           ymax = ci_ul, shape=trauma_type, color=significant)) +
  geom_pointrange(position = position_dodge(width = 0.7)) +
  scale_color_manual(breaks=c("Not significant", "Significant"),
                    values=c("black", "red"),
                    name = "Significance") +
  scale_shape_manual(values = c(16, 17),
                    name = "Trauma Type") +
  geom_hline(yintercept = 1, lty = 2) +
  theme_bw() +
  theme(
    axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1),
    legend.position = "right"
  ) +
  guides(color = guide_legend(reverse=T),
         shape = guide_legend(reverse=T)) +
  scale_y_continuous(breaks = seq(0, 1.75, by = 0.25), limits = c(0.75, 1.7)) +
  labs(
    y = "RR (99.7% CI)",
    x = "Number of Traumatic Events Experienced",
    color = "Trauma Type"
  ) + coord_flip() +
  theme(legend.position = "bottom")

dev.off()

```

```

## pdf
## 2

```

**Full Figure Model B + NBGLM (current)** Here I combine data from the Logistic Regression/Poisson Regression with the data from the negative binomial regression.

I also manually add reference points of “0” traumas with OR/ci\_ll/ci\_ul of 1. Finally I relevel the data so the facets are in the correct order.

```

df_trauma_onlyB <- df_trauma_only %>% filter(model == "Model B")
df_trauma_onlyB$OR_or_RR <- df_trauma_onlyB$RR
df_trauma_onlyB$prevalence <- NA
df_trauma_onlyB$type <- NA
df_trauma_onlyB$prevalence <- NA
df_trauma_onlyB$regression_type <- NA
df_trauma_onlyB$cidi_name <- "Chronic condition count"
df_trauma_onlyB$cidi_q <- NA
df_trauma_onlyB$cidi_var_n <- NA
df_trauma_onlyB <- df_trauma_onlyB %>% dplyr::select(-RR)

reference_row <- data.frame(cidi_q = NA,
                           Coefficient = NA,
                           prevalence = NA,

```

```

        OR_or_RR = 1,
        ci_ll = 1, ci_ul = 1,
        p = NA, significant = "Not significant",
        model="Model B",
        cidi_var_n = NA,
        type = NA,
        regression_type = NA,
        cidi_name = "Chronic condition count",
        coef_name = "0",
        trauma_type = "Non-interpersonal")
df_trauma_onlyB <- rbind(df_trauma_onlyB, reference_row)

reference_row <- data.frame(cidi_q = NA,
        Coefficient = NA,
        prevalence = NA,
        OR_or_RR = 1,
        ci_ll = 1, ci_ul = 1,
        p = NA, significant = "Not significant",
        model="Model B",
        cidi_var_n = NA,
        type = NA,
        regression_type = NA,
        cidi_name = "Chronic condition count",
        coef_name = "0",
        trauma_type = "Interpersonal")
df_trauma_onlyB <- rbind(df_trauma_onlyB, reference_row)

df_gg_temp_nb <- rbind(df_gg_temp,df_trauma_onlyB)

df_gg_temp_nb$coef_name <- factor(df_gg_temp_nb$coef_name, levels = c("0", "1", "2", "3", "4+"))
df_gg_temp_nb$OR_or_RR <- as.numeric(as.character(df_gg_temp_nb$OR_or_RR))
df_gg_temp_nb$ci_ll <- as.numeric(as.character(df_gg_temp_nb$ci_ll))
df_gg_temp_nb$ci_ul <- as.numeric(as.character(df_gg_temp_nb$ci_ul))

#Adjust to relevel the data
cidi_name_order <- c("Pain", "Stomach/intestine ulcer",
        "Seasonal allergies", "Neurological",
        "Cardiometabolic", "Respiratory", "Hypothyroidism", "Chronic condition count")

df_gg_temp_nb$cidi_name <- factor(df_gg_temp_nb$cidi_name, levels = cidi_name_order)

```

Here I produce the figure. The facets will be in 2 rows, with 4 columns.

```

setwd("~/Dropbox (Harvard University)/NeuroMex/Data Analysis/Papers/LEC-CIDI Paper/LEC-CIDI Paper/")

pdf(paste0("Compiled results_ModelB_withCCcount",today(),".pdf"), width=10, height=8)

```



```

ggplot(df_gg_temp_nb, aes(x=coef_name, y=OR_or_RR, ymin=ci_ll, ymax=ci_ul,
                          shape=trauma_type, color=significant)) +
  geom_pointrange(position=position_dodge(0.5))+
  geom_hline(yintercept=1, lty=2)+
  coord_flip()+
  scale_color_manual(breaks=c("Not significant", "Significant"),
                    values=c("black", "red"),
                    name = "Significance") +
  scale_shape_manual(values = c(16, 17),
                    name = "Trauma Type") +
  scale_y_continuous(breaks = seq(0, 4, by = 1), limits = c(0, 5.5)) +
  facet_wrap(~cidi_name, nrow=2) +
  theme_bw() +
  guides(color = guide_legend(reverse=T),
         shape = guide_legend(reverse=T))+
  xlab("Number of Traumatic Events Experienced")+
  ylab("OR/RR (99.7% CI)")+
  theme(legend.position = "bottom")

dev.off()

```

```

## pdf
## 2

```