Final Project Code

Kunwu Lyu and Evan Hart

2025-03-16

Data Wrangling

```r
## Data from ICPSR
survey <- read_tsv("data/ICPSR_37143/DS0001/37143-0001-Data.tsv") %>%
  janitor::clean_names() # To all lower case
receipt <- read_tsv("data/ICPSR_37143/DS0002/37143-0002-Data.tsv") %>%
  janitor::clean_names()
fast_food <- read_tsv("data/ICPSR_37143/DS0003/37143-0003-Data.tsv") %>%
  janitor::clean_names()
grocery <- read_tsv("data/ICPSR_37143/DS0004/37143-0004-Data.tsv") %>%
  janitor::clean_names()
recall <- read_tsv("data/ICPSR_37143/DS0005/37143-0005-Data.tsv") %>%
  janitor::clean_names()

## Combine multiple surveys
full_data <- survey %>%
  full_join(receipt, relationship = "many-to-many") %>%
  full_join(fast_food, relationship = "many-to-many") %>%
  full_join(grocery, relationship = "many-to-many") %>%
  full_join(recall, relationship = "many-to-many")

## Mutating
full_data <- full_data %>%
  mutate(limit = ordered(q75, levels = c("Never", "Seldom", "Sometimes",
                                         "Often", "Always"))) %>% # for ordinal
  mutate(age = as.numeric(q76),
         gender = if_else(q77 == 0, "M", "F"),
         race = case_when(
           !is.na(q79_1) ~ "Native",
           !is.na(q79_2) ~ "Black",
           !is.na(q79_3) ~ "Asian",
           !is.na(q79_4) ~ "White",
           !is.na(q79_a) ~ "Other"
         ),
         edu = as.numeric(q80),
         location = nemslocationindicator,
         city = q1,
         num_kids = q44,
         surveydate = dmy(surveydate)) %>%
  mutate(days_since_ban =
           as.numeric(interval(as.Date("2013-03-12"), surveydate) / days(1))) %>%
  filter(age > 0)
```

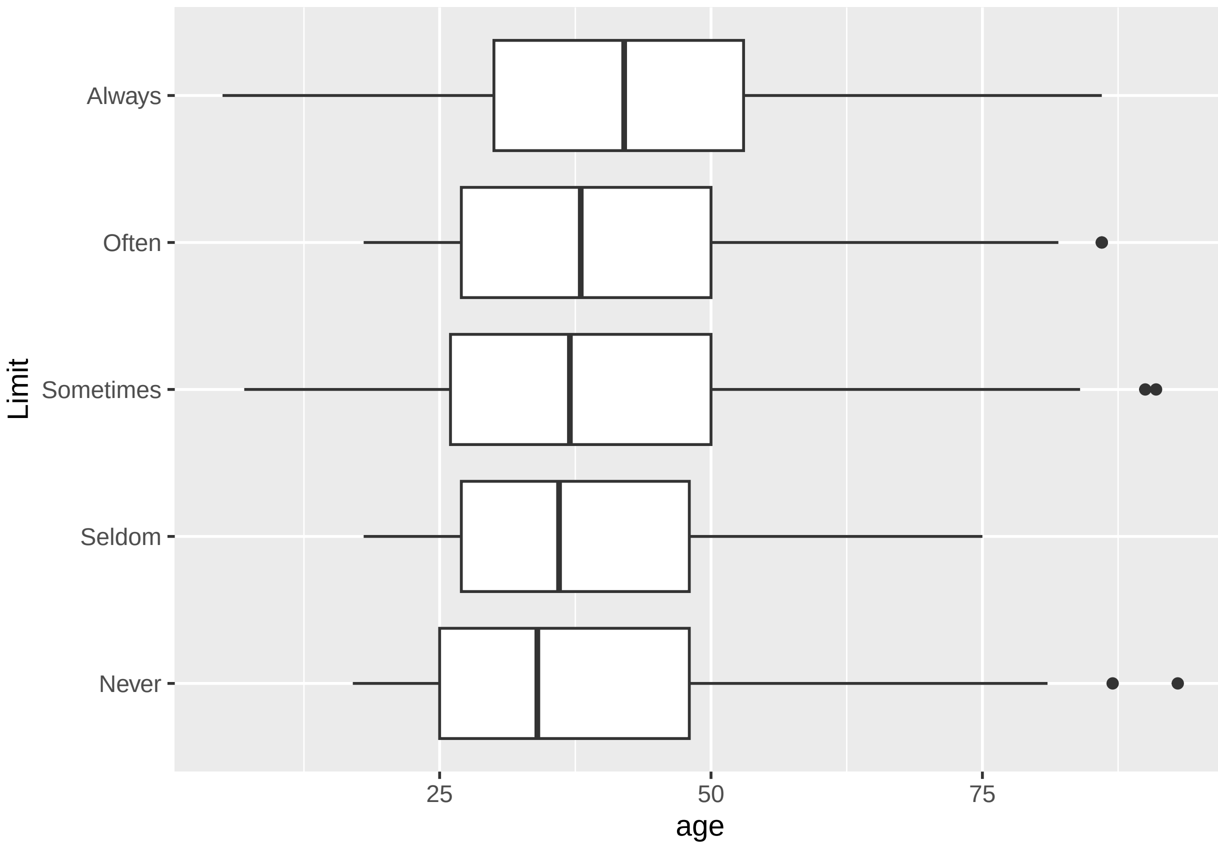
```r
# Standardize numerical for prediction
standardize <- function(x, na.rm = TRUE) {
  (x - mean(x, na.rm = na.rm)) /
    sd(x, na.rm = na.rm)
}

# subset of complete dataset
reduced_data <- full_data %>%
  mutate(age_std = standardize(as.numeric(q76))) %>%
  select(c("receiptid", "person_id", "limit", "age", "age_std", "gender",
           "race", "edu", "city", "caff", "location", "round", "nsigns_ssb",
           "num_kids", "surveydate", "days_since_ban", "caloriescal", "fatg",
           "sugarg")) %>%
  group_by(receiptid) %>%
  mutate(black = if_else(race == "Black", "Black", "non-Black")) %>%
  mutate(caff = sum(caff, na.rm = T), # across each receipt
         caloriescal = sum(caloriescal, na.rm = T),
         fatg = sum(fatg, na.rm = T),
         sugarg = sum(sugarg, na.rm = T)) %>%
  drop_na() %>%
  distinct() %>% # Remove duplicate rows because multiple items are on a receipt
  mutate(receiptid = as.factor(receiptid),
         person_id = as.factor(person_id),
         location = as.factor(location),
         round = as.factor(round),
         edu = case_when(
           edu == 1 ~ "Less than High School",
           edu == 2 ~ "Some High School",
           edu == 3 ~ "High School",
           edu == 4 ~ "Some College",
           edu == 5 ~ "Associates Degree",
           edu == 6 ~ "College Degree",
           edu == 7 ~ "Graduate Degree"
         )) %>%
  ungroup() %>%
  mutate(
         caff_std = standardize(caff),
         nsigns_ssb_std = standardize(nsigns_ssb),
         days_since_ban_std = standardize(days_since_ban),
         caloriescal_std = standardize(caloriescal),
         fatg_std = standardize(fatg),
         sugarg_std = standardize(sugarg)
         )

# Cleaned data
write_csv(reduced_data, "dietControl.csv")

# One receipt can't appear in multiple locations
multi_receipt_locations <- reduced_data %>%
  group_by(receiptid) %>%
  summarize(n_rounds = n_distinct(location)) %>%
  filter(n_rounds > 1) %>%
  pull(receiptid)
```

```r
reduced_data %>%
  filter(receiptid %in% multi_receipt_locations) %>%
  count(receiptid, location)
```

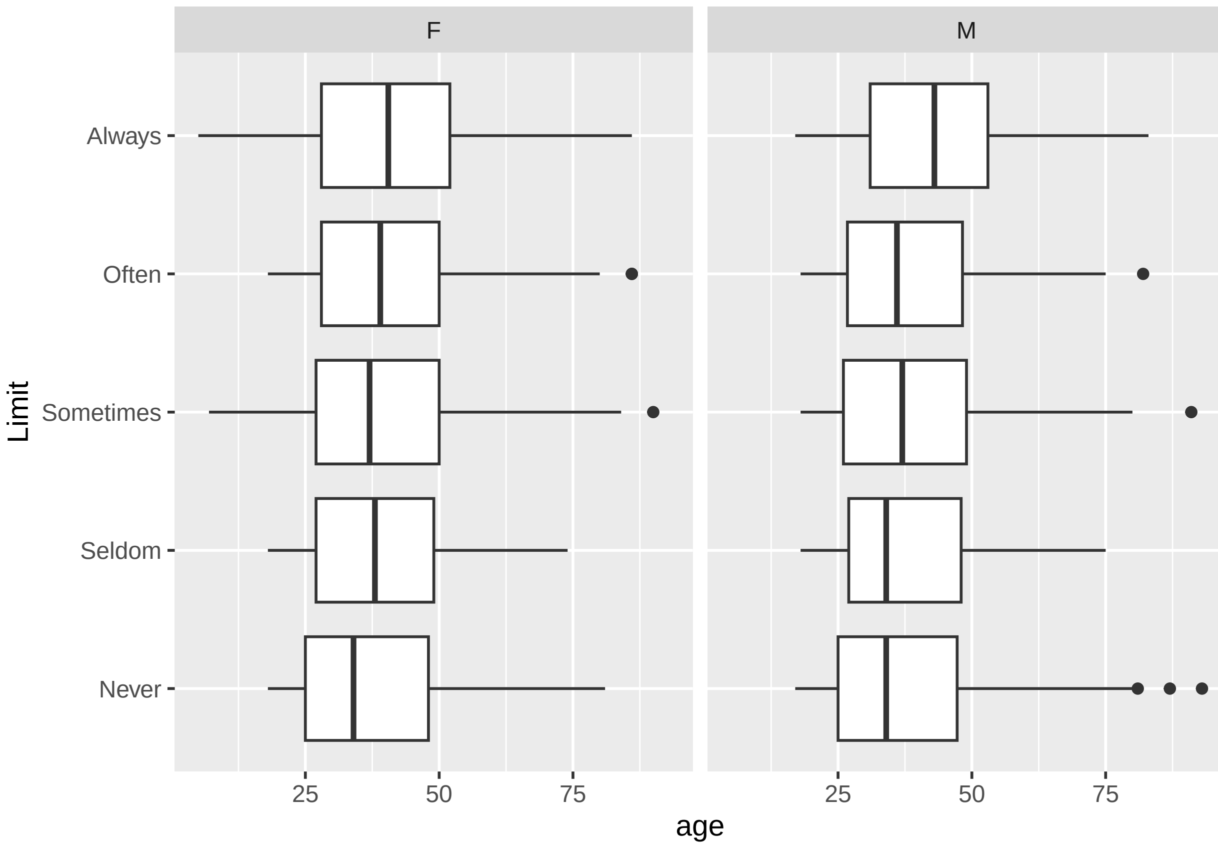
```
## # A tibble: 0 x 3
## # i 3 variables: receiptid <fct>, location <fct>, n <int>
```

EDA

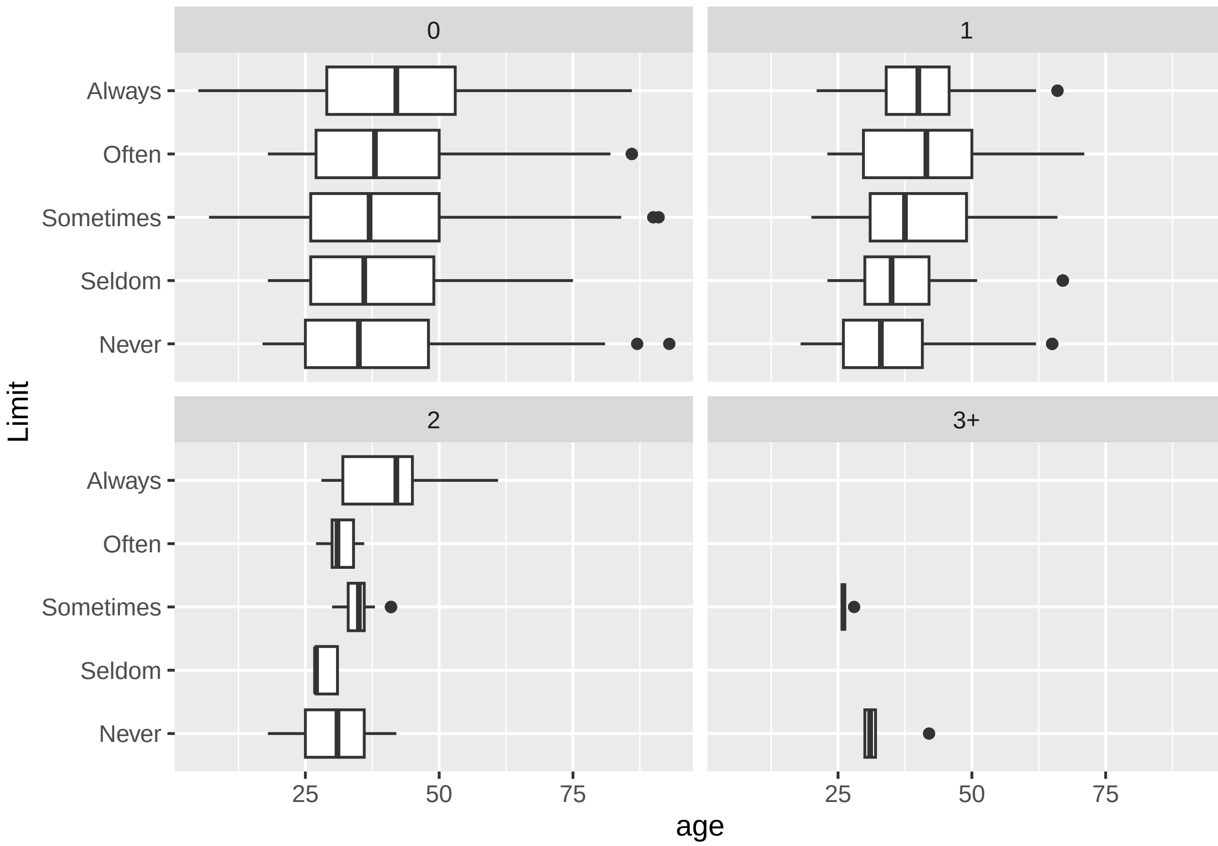
```r
# Single variables, interactions plotted against limit

# Age
ggplot(data = reduced_data, aes(x = age , y = limit)) +
  geom_boxplot() +
  labs(x = "age", y = "Limit")
```



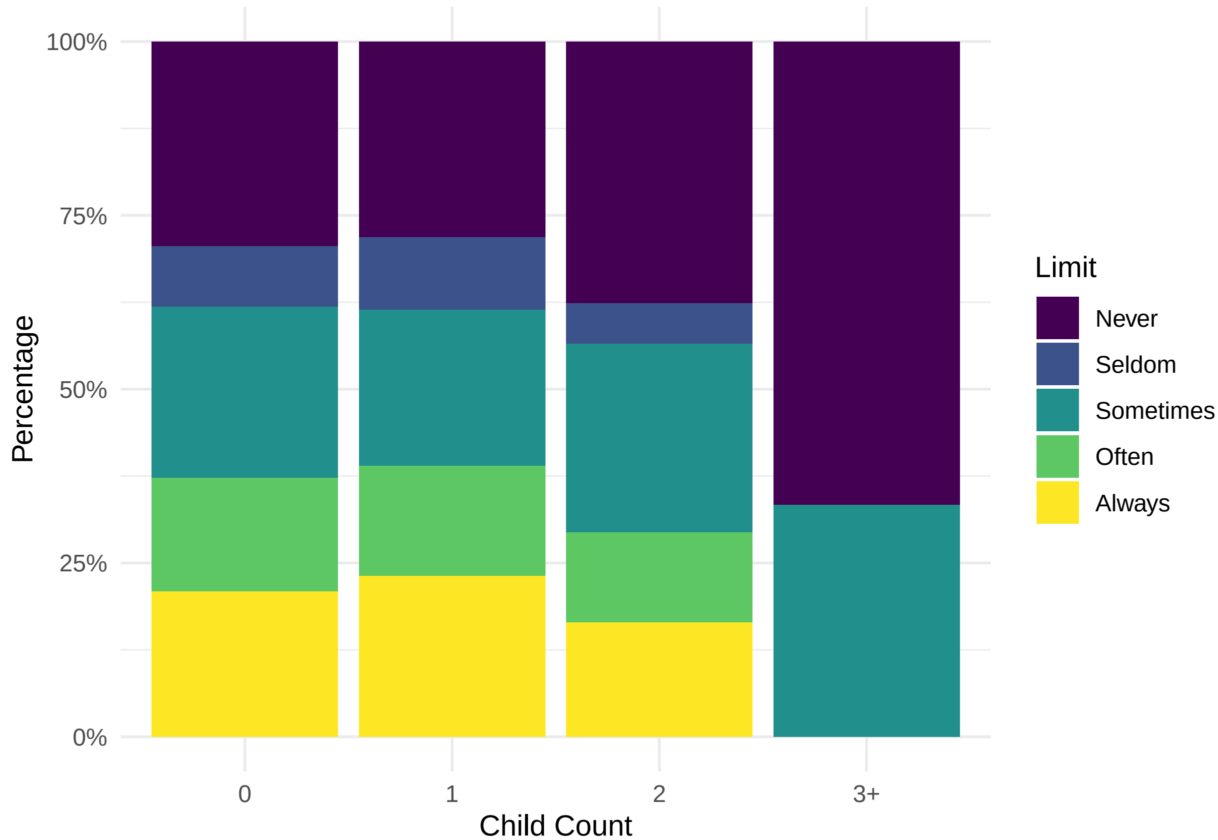

```r
# Age faceted by gender
ggplot(data = reduced_data, aes(x = age , y = limit)) +
  geom_boxplot() +
  facet_wrap(~gender) +
  labs(x = "age", y = "Limit")
```



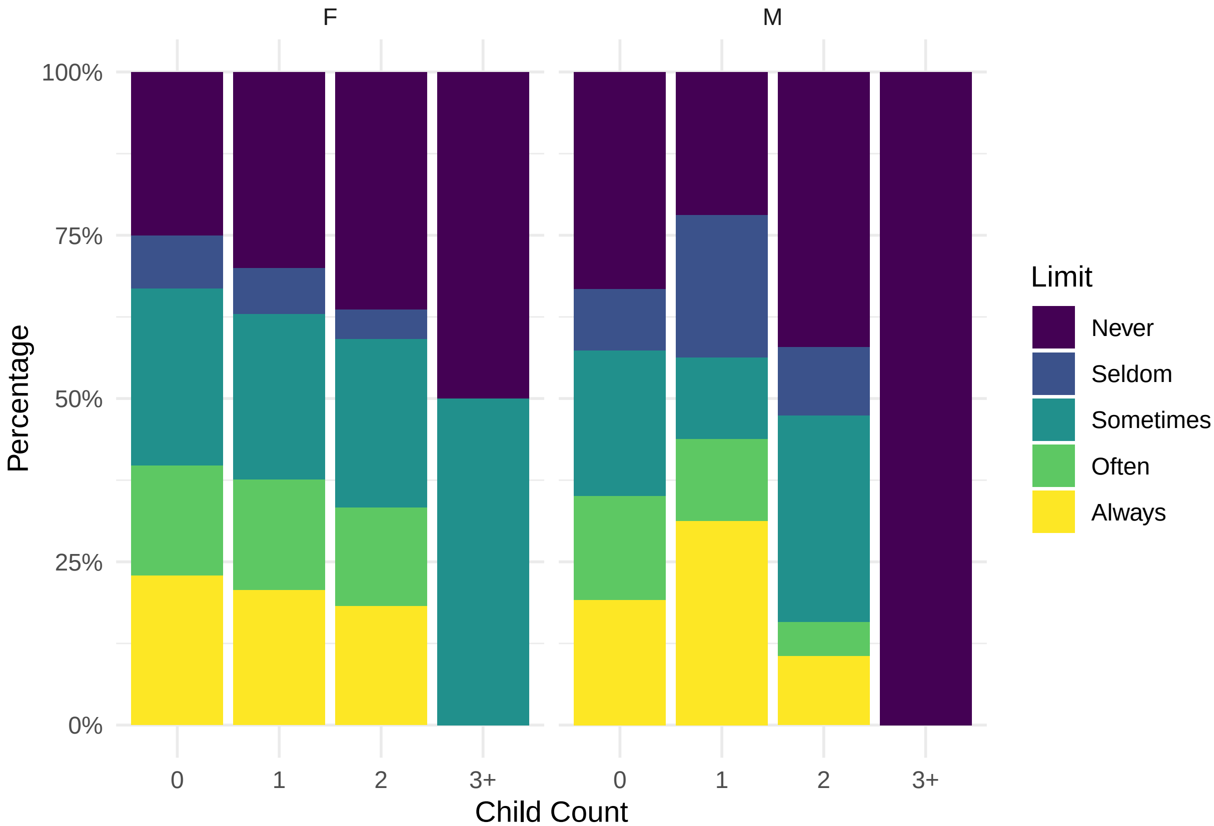

```r
# Age faceted by child count
ggplot(data = reduced_data, aes(x = age , y = limit)) +
  geom_boxplot() +
  facet_wrap(~num_kids) +
  labs(x = "age", y = "Limit")
```



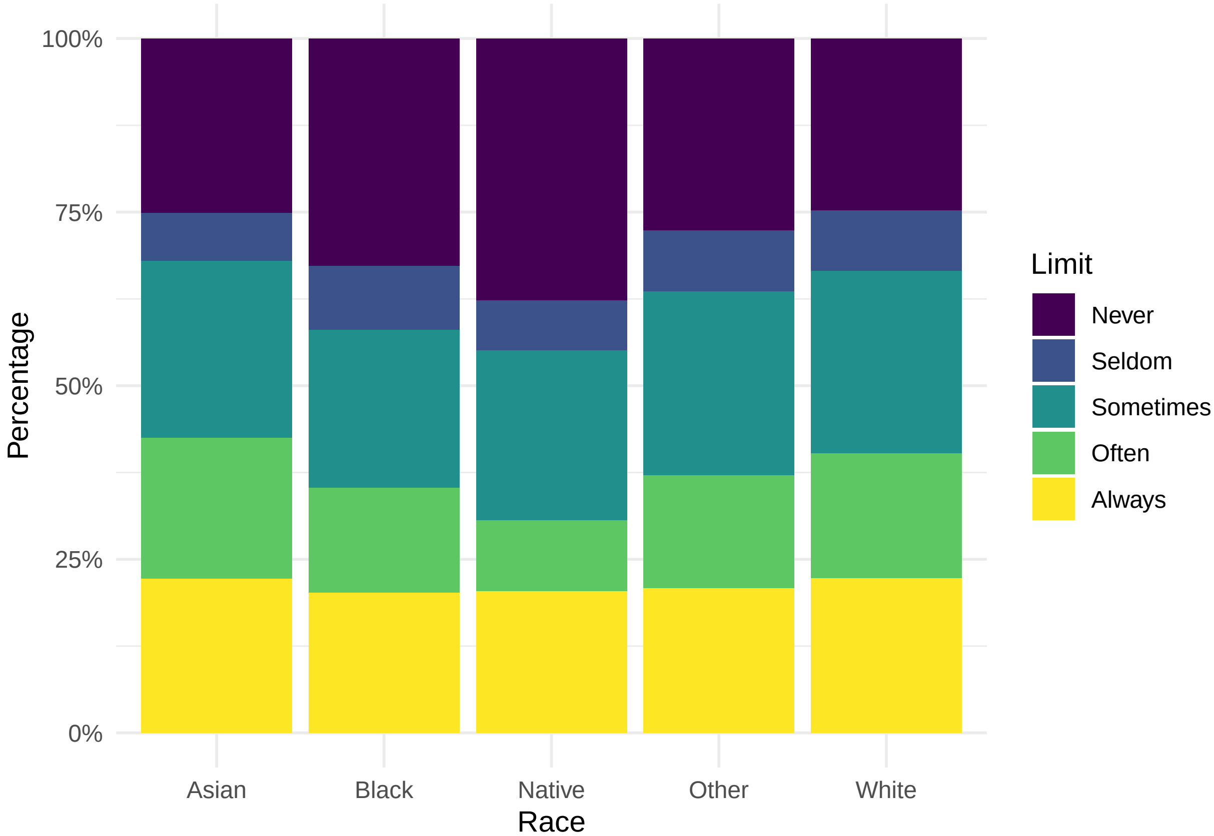

```r
# Child count
ggplot(data = reduced_data, aes(x = num_kids, fill = limit)) +
  geom_bar(position = "fill") +
  scale_y_continuous(labels = scales::percent) +
  labs(x = "Child Count", y = "Percentage", fill = "Limit") +
  theme_minimal()
```



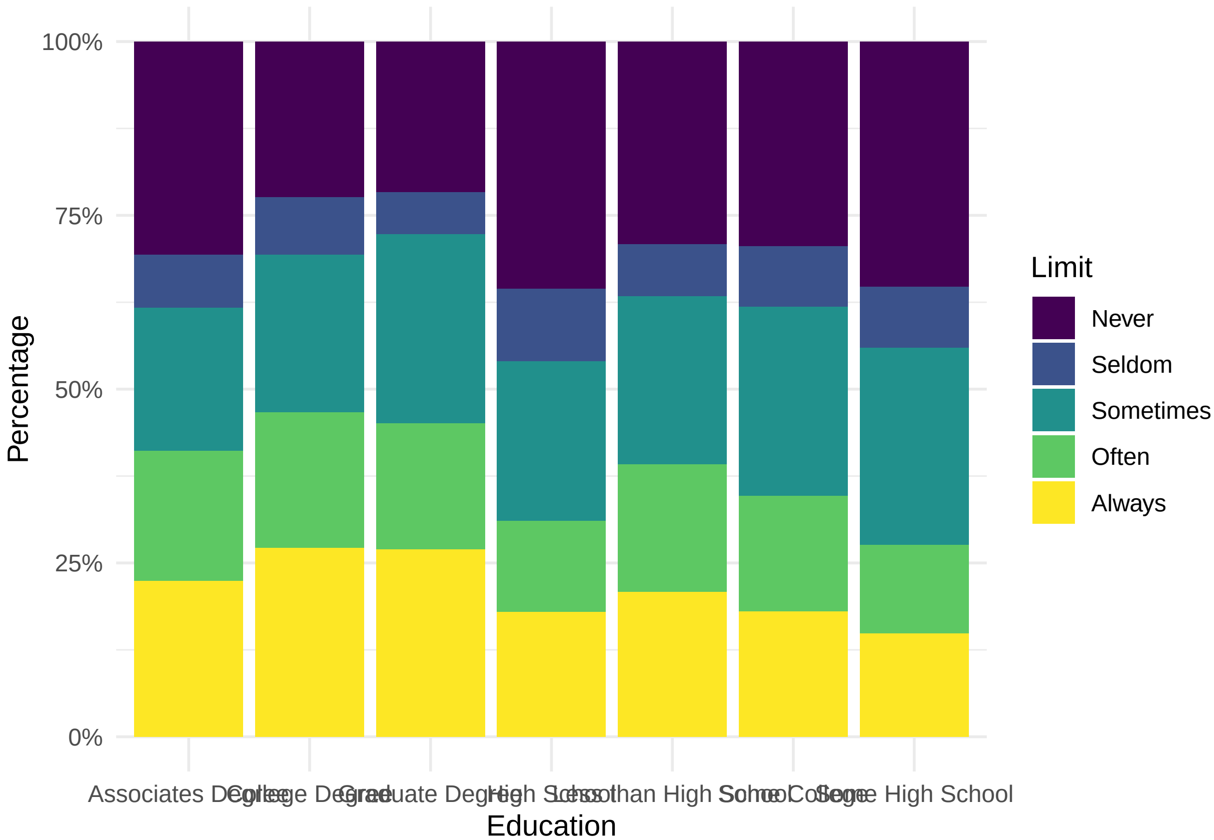

```r
# Child count / gender interaction
ggplot(data = reduced_data, aes(x = num_kids, fill = limit)) +
  geom_bar(position = "fill") +
  scale_y_continuous(labels = scales::percent) +
  labs(x = "Child Count", y = "Percentage", fill = "Limit") +
  facet_wrap(~gender) +
  theme_minimal()
```




```
# Race
ggplot(data = reduced_data, aes(x = race, fill = limit)) +
  geom_bar(position = "fill") +
  scale_y_continuous(labels = scales::percent) +
  labs(x = "Race", y = "Percentage", fill = "Limit") +
  theme_minimal()
```

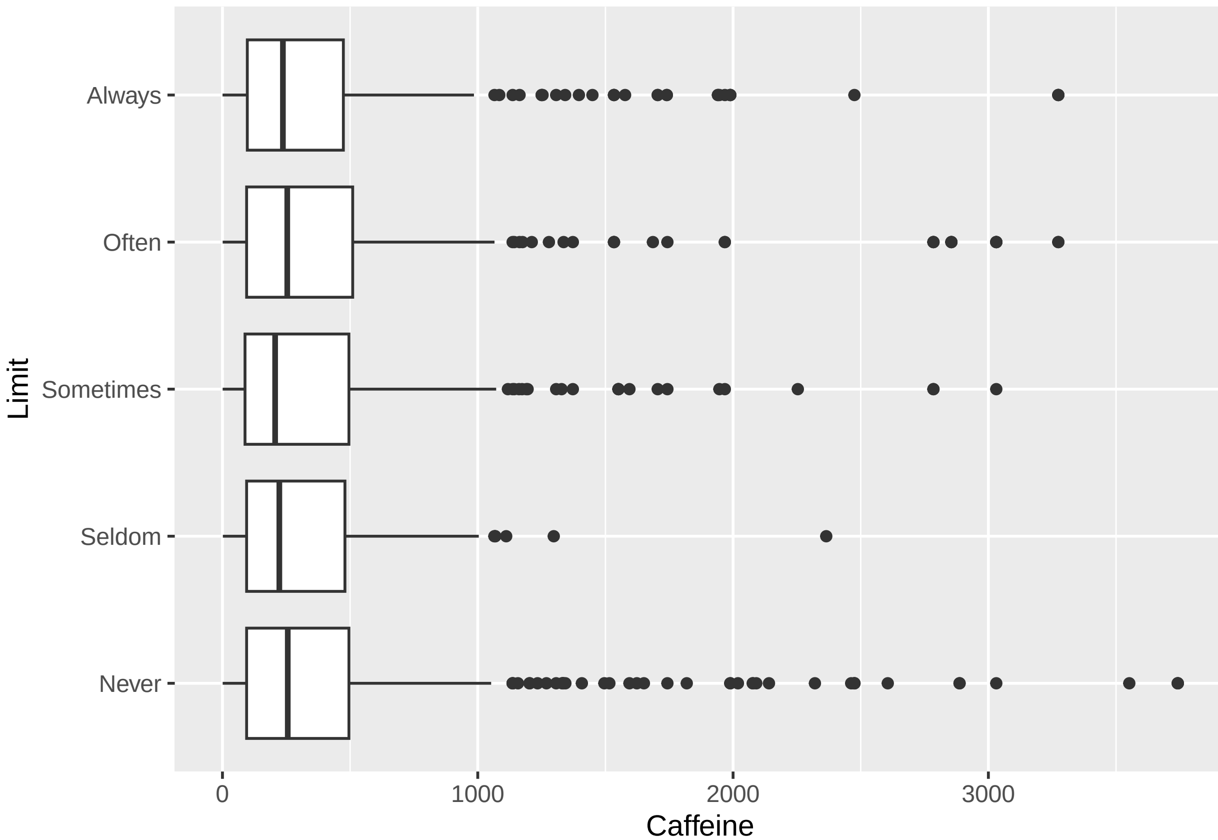



```
# Education
ggplot(data = reduced_data, aes(x = edu, fill = limit)) +
  geom_bar(position = "fill") +
  scale_y_continuous(labels = scales::percent) +
  labs(x = "Education", y = "Percentage", fill = "Limit") +
  theme_minimal()
```

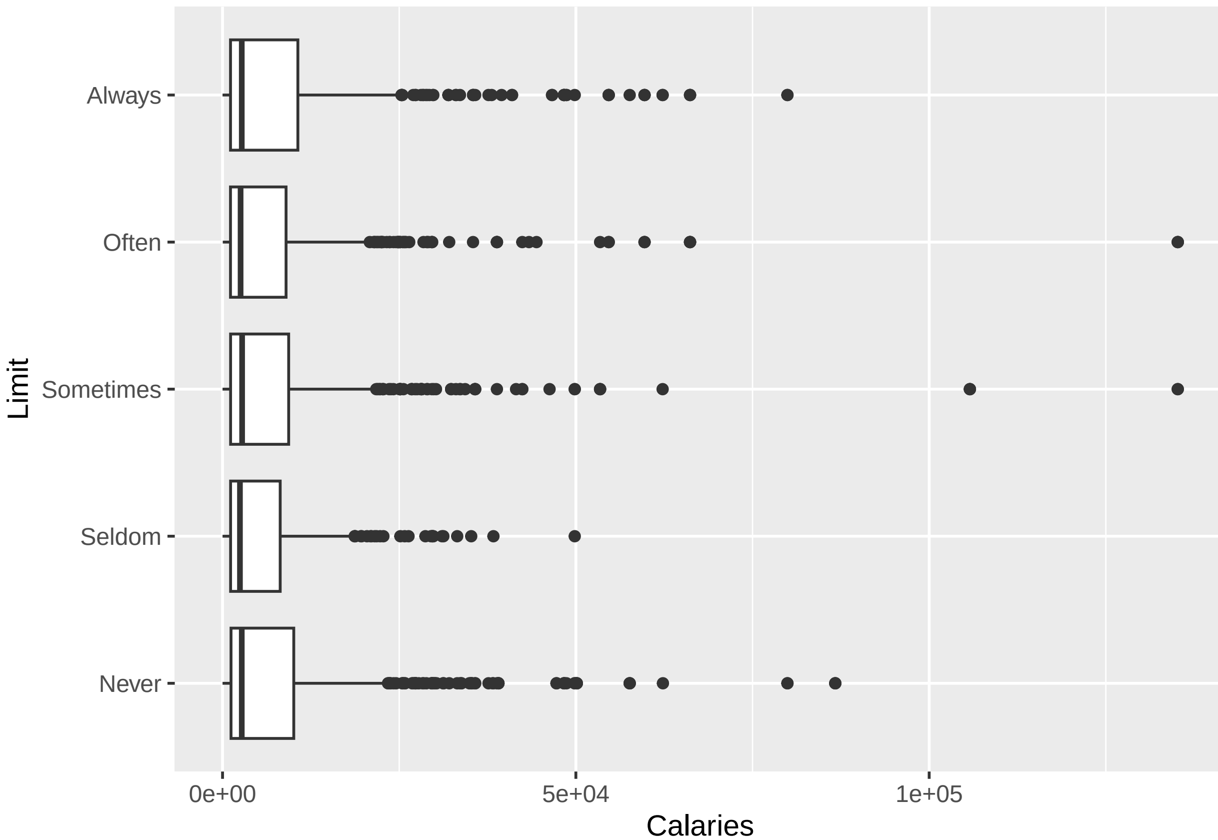



```r
# City
ggplot(data = reduced_data, aes(x = city, fill = limit)) +
  geom_bar(position = "fill") +
  scale_y_continuous(labels = scales::percent) +
  labs(x = "City", y = "Percentage", fill = "Limit") +
  theme_minimal()
```

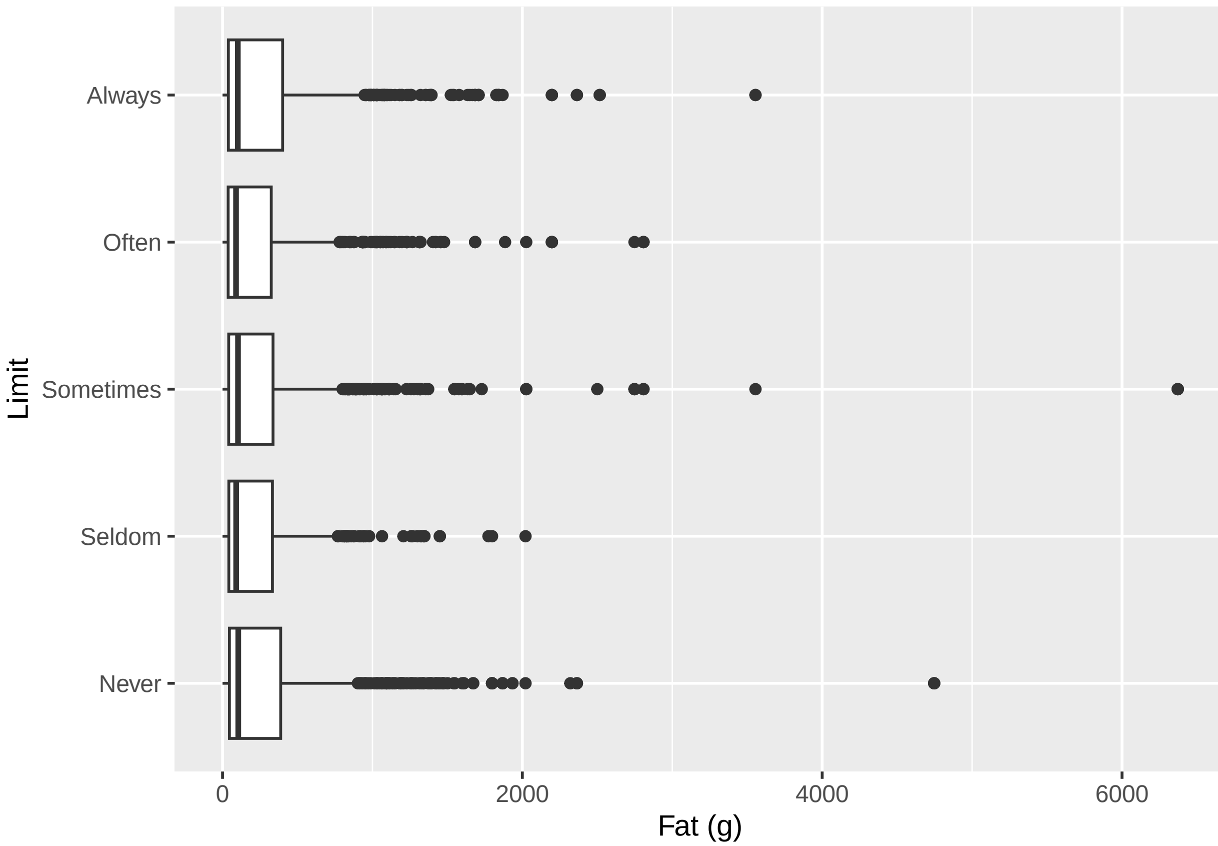

```r
# Caffeine
ggplot(data = reduced_data %>% filter(caff > 0), aes(x = caff , y = limit)) +
  geom_boxplot() +
  labs(x = "Caffeine", y = "Limit")
```



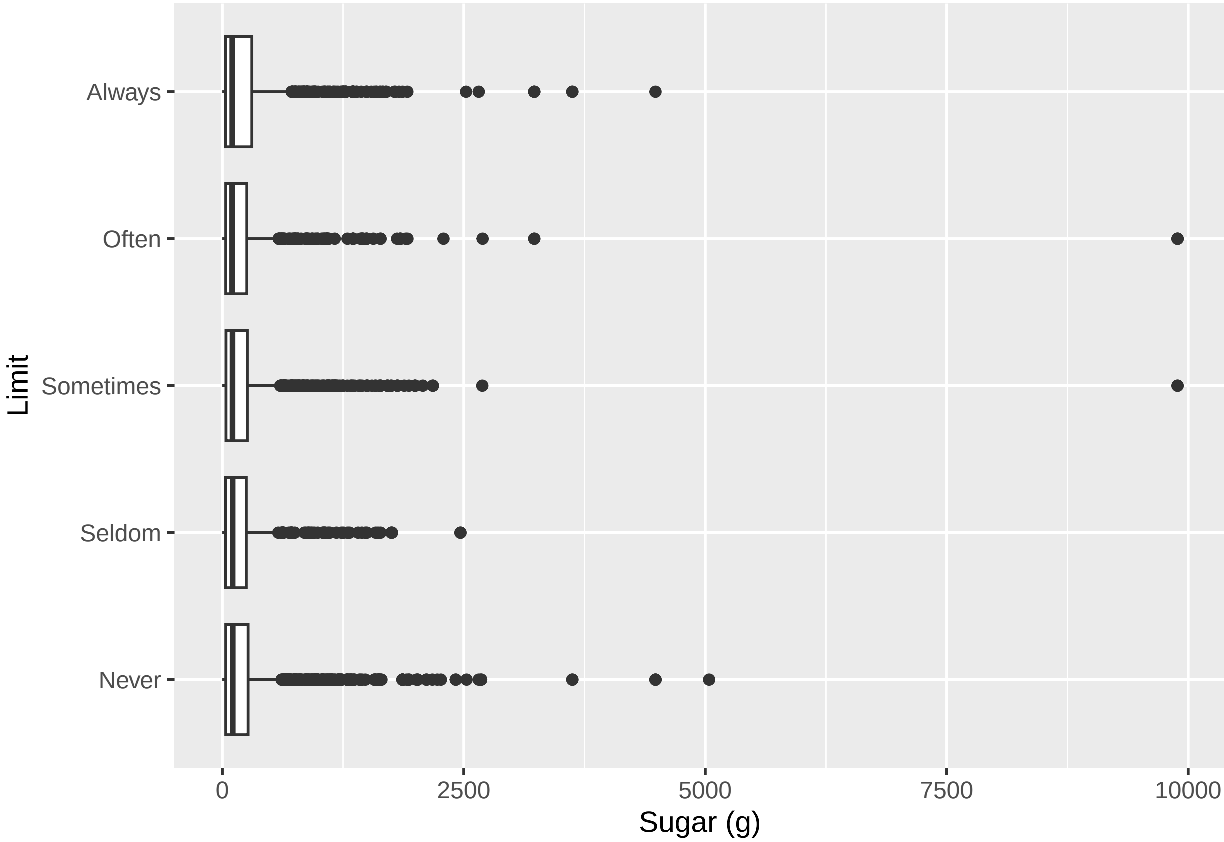

```
# Kcal
ggplot(data = reduced_data, aes(x = caloriescal, y = limit)) +
  geom_boxplot() +
  labs(x = "Calaries", y = "Limit")
```



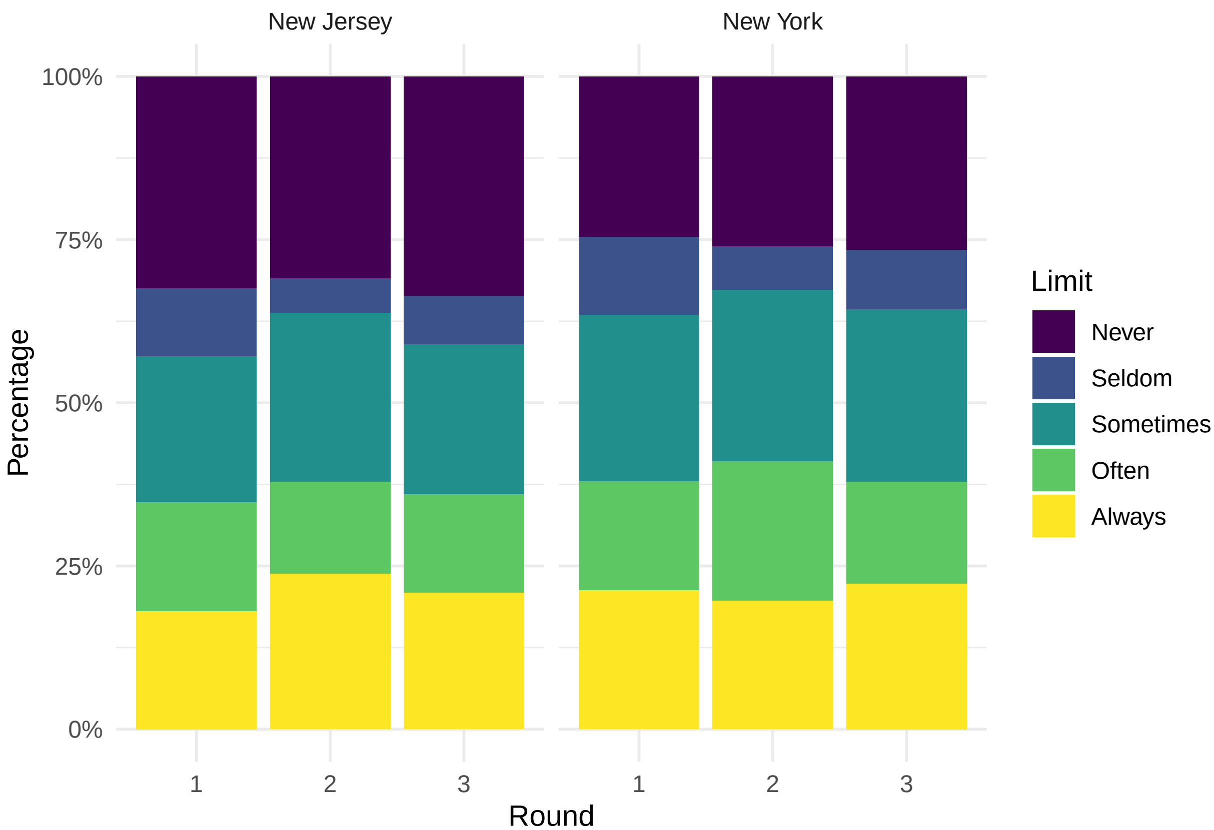

```r
# fat
ggplot(data = reduced_data, aes(x = fatg, y = limit)) +
  geom_boxplot() +
  labs(x = "Fat (g)", y = "Limit")
```



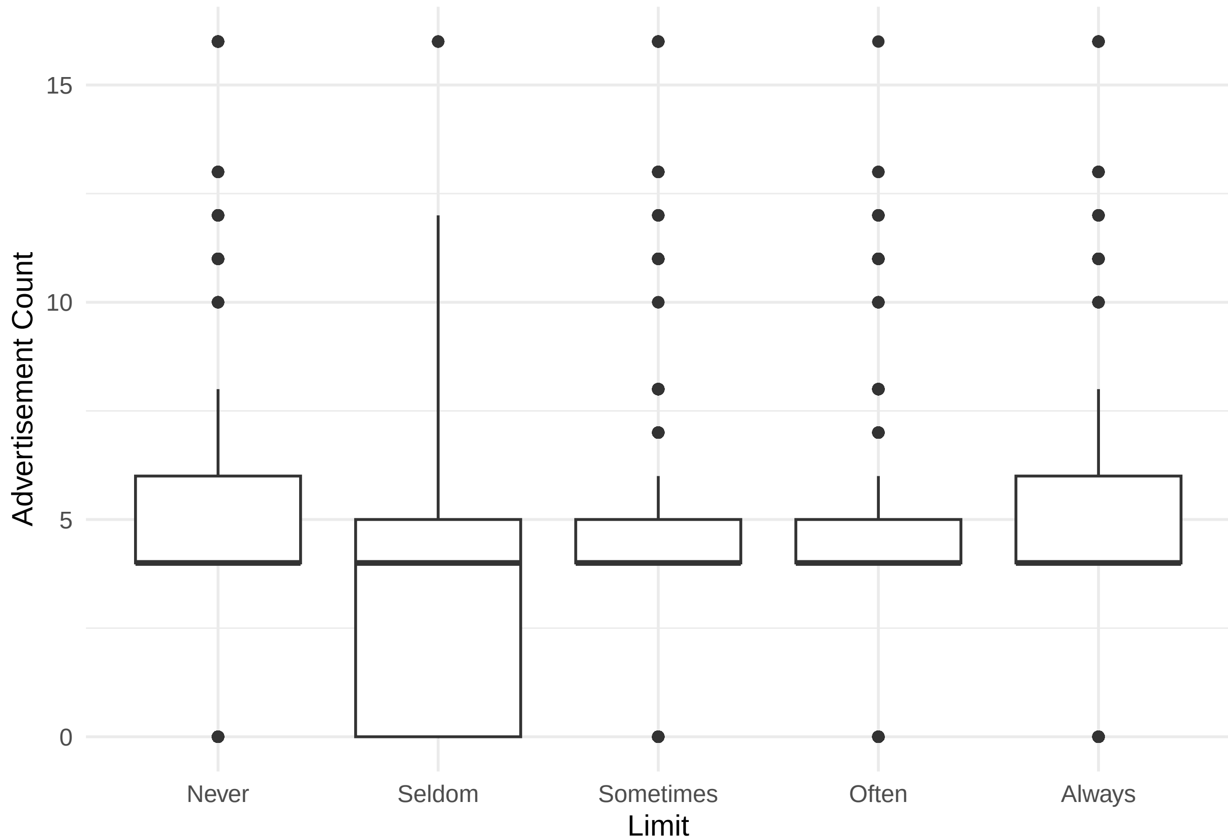

```
# sugar
ggplot(data = reduced_data, aes(x = sugarg, y = limit)) +
  geom_boxplot() +
  labs(x = "Sugar (g)", y = "Limit")
```




```
# Survey round
ggplot(data = reduced_data, aes(x = round, fill = limit)) +
  geom_bar(position = "fill") +
  scale_y_continuous(labels = scales::percent) +
  labs(x = "Round", y = "Percentage", fill = "Limit") +
  facet_wrap(~city) +
  theme_minimal()
```




```r
# Advertisement count
ggplot(data = reduced_data, aes(y = nsigns_ssb, x = limit)) +
  geom_boxplot() +
  labs(y = "Advertisement Count", x = "Limit") +
  theme_minimal()
```



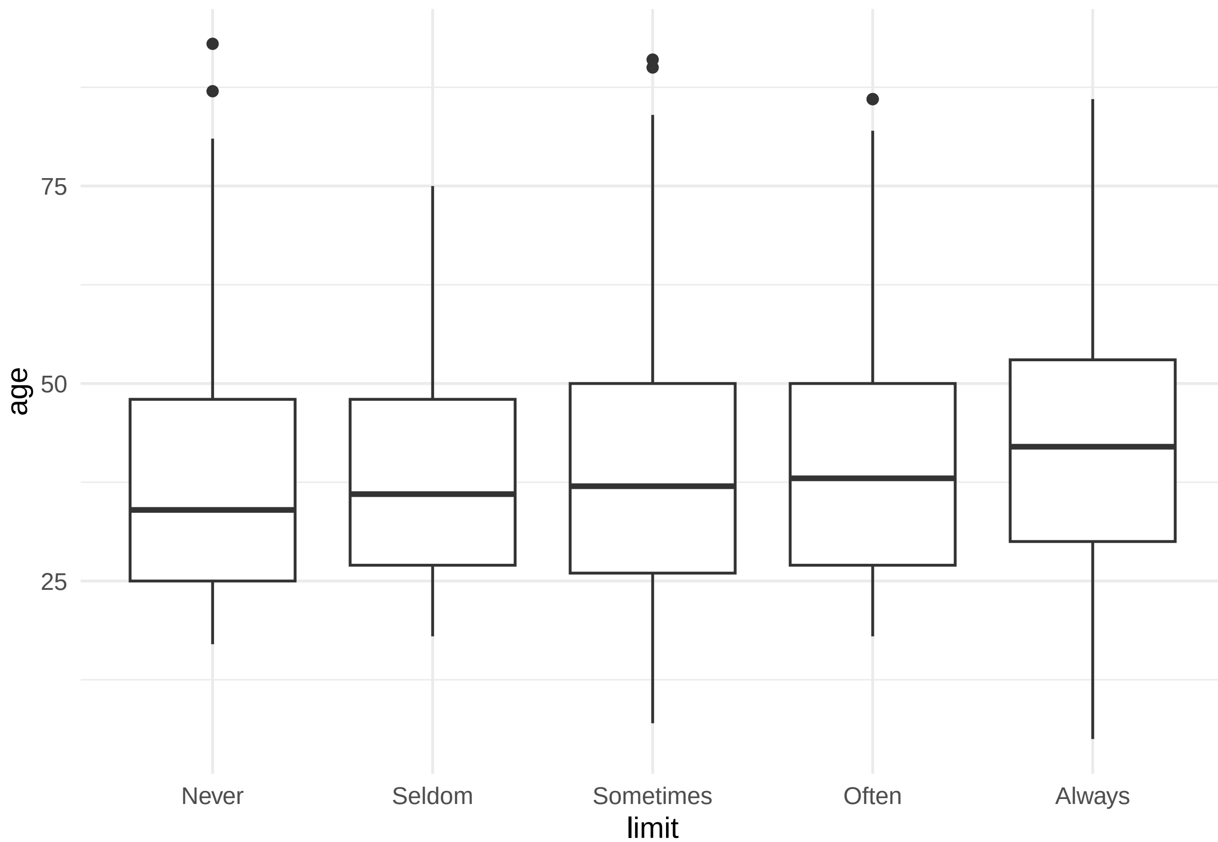

```r
# Plot function for interactions

plot_cats <- c("limit", "gender", "race", "city", "round", "num_kids", "edu")
plot_nums <- c("age", "caff", "nsigns_ssb", "days_since_ban", "caloriescal", "sugarg", "fatg")

library(rlang)
```

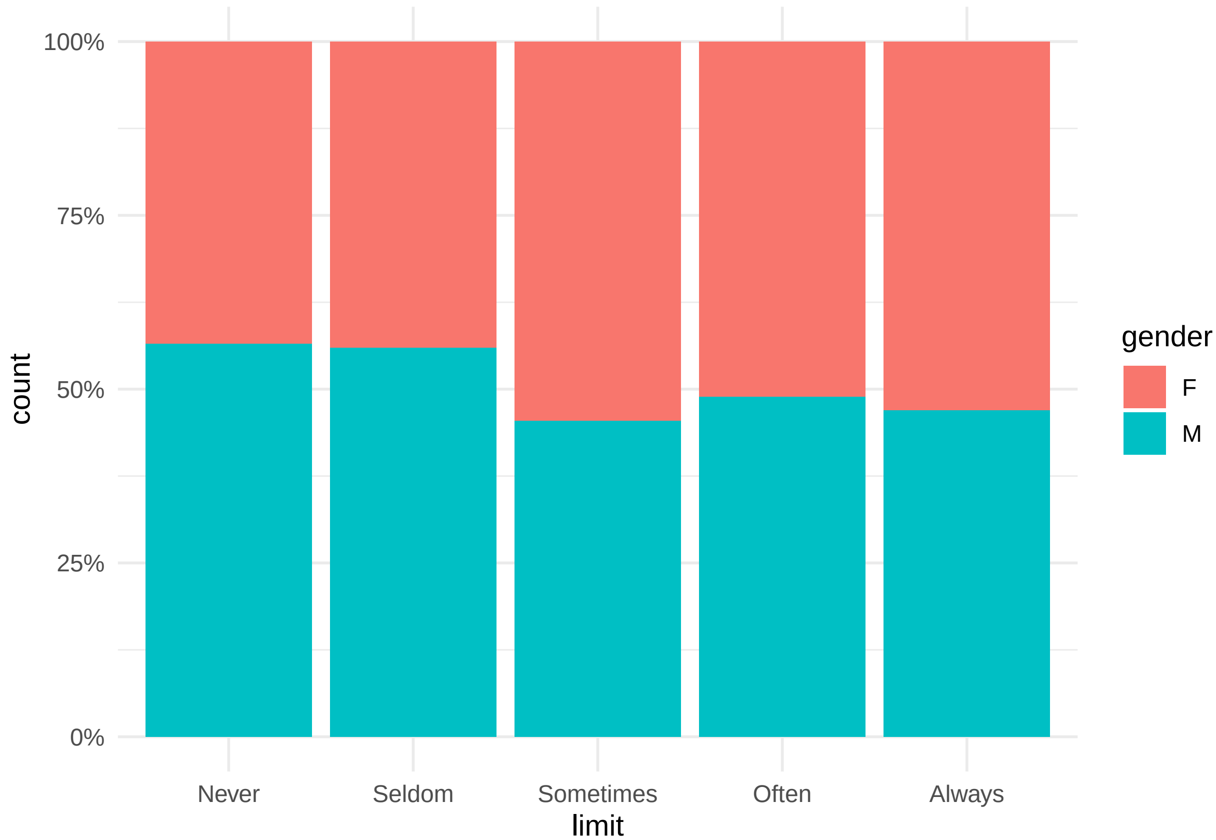
```
##
## Attaching package: 'rlang'

## The following objects are masked from 'package:purrr':
##
##     %@%, flatten, flatten_chr, flatten_dbl, flatten_int, flatten_lgl,
##     flatten_raw, invoke, splice
```

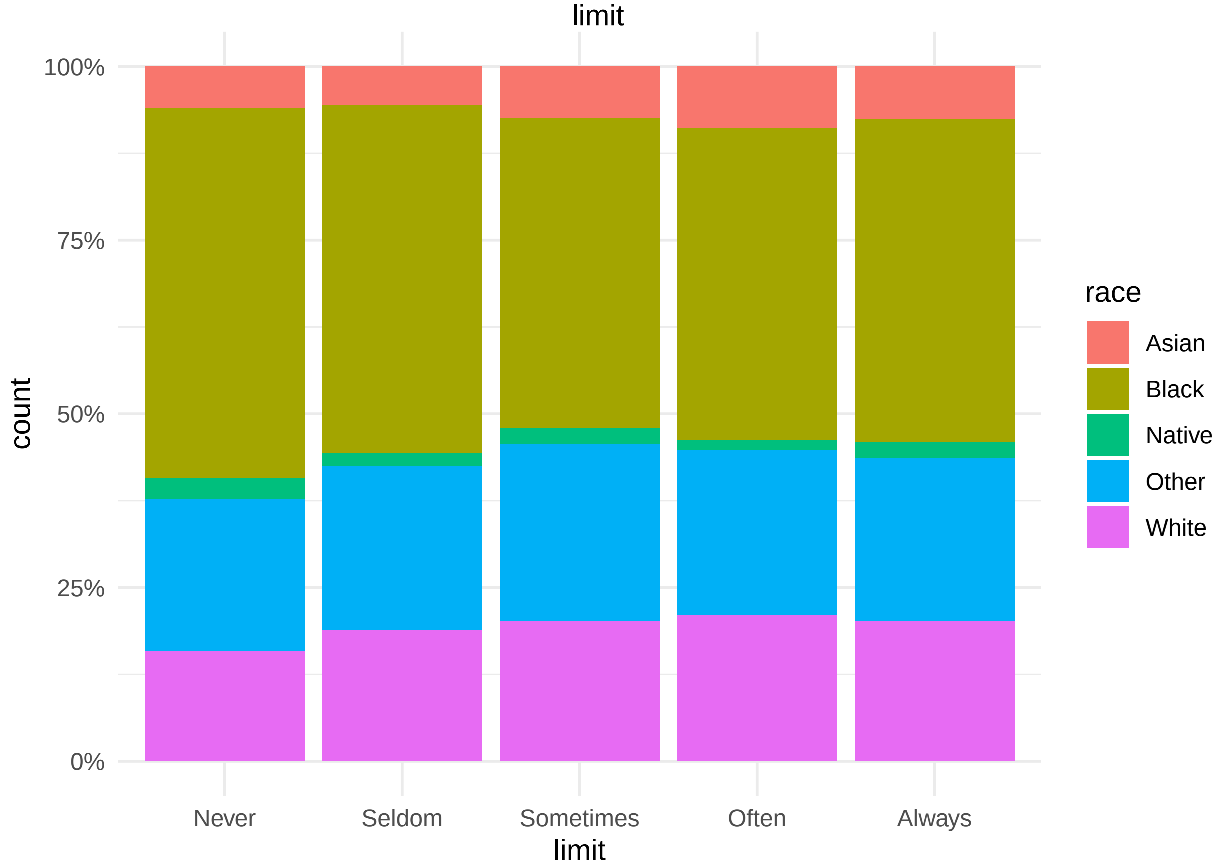
```r
make_plot <- function(var1, var2){
  if(var1 %in% plot_cats & var2 %in% plot_cats){
    print(ret_plot <- ggplot(data =reduced_data, aes(x = !!sym(var1), fill = !!sym(var2))) +
      geom_bar(position = "fill") +
      scale_y_continuous(labels = scales::percent) +
      theme_minimal())
  }

  if(var1 %in% plot_cats & var2 %in% plot_nums){
    print(ret_plot <- ggplot(data = reduced_data, aes(x = !!sym(var1), y = !!sym(var2))) +
      geom_boxplot() +
      theme_minimal())
  }
```

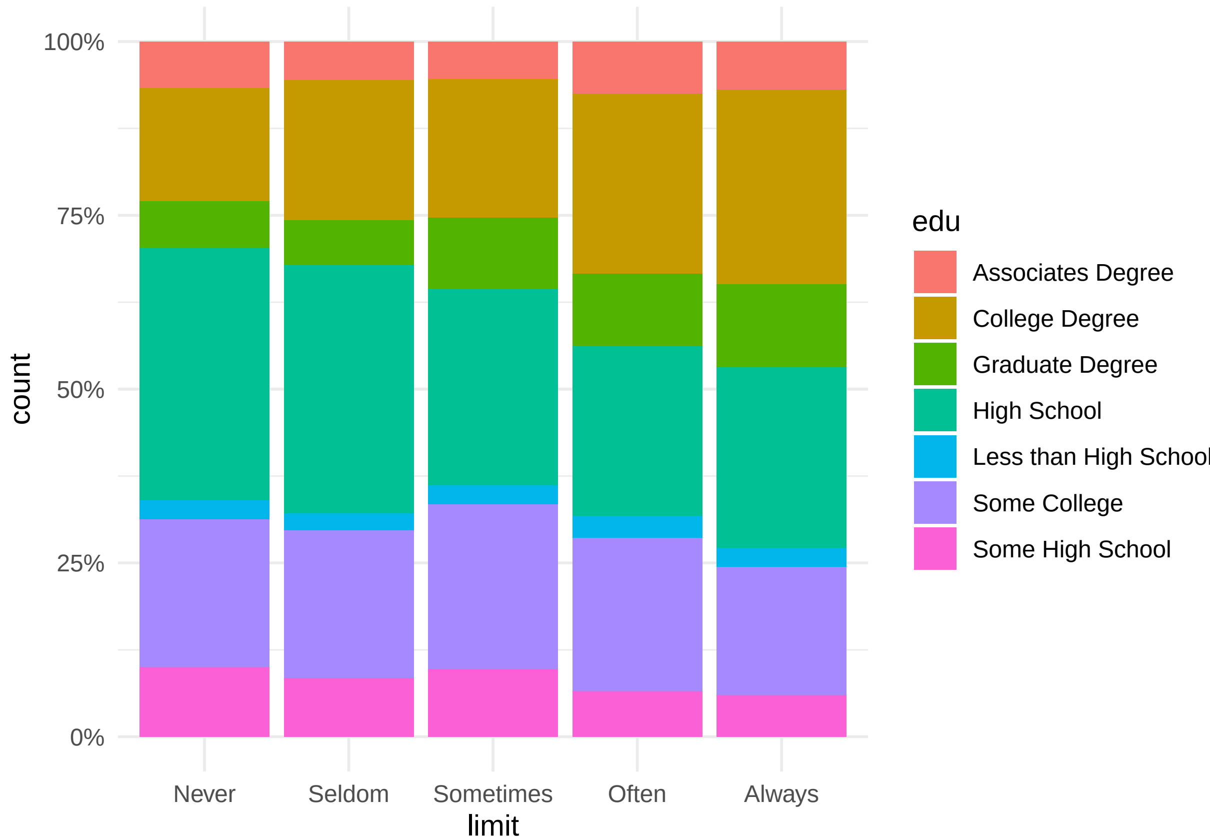
```
  if(var1 %in% plot_nums & var2 %in% plot_cats){
    print(ret_plot <- ggplot(data = reduced_data, aes(x = !!sym(var2), y = !!sym(var1))) +
        geom_boxplot() +
        theme_minimal())
  }

  if(var1 %in% plot_nums & var2 %in% plot_nums){
    print(ggplot(data = reduced_data, aes(x = !!sym(var2), y = !!sym(var1))) +
      geom_point() +
      theme_minimal())
  }
}
```

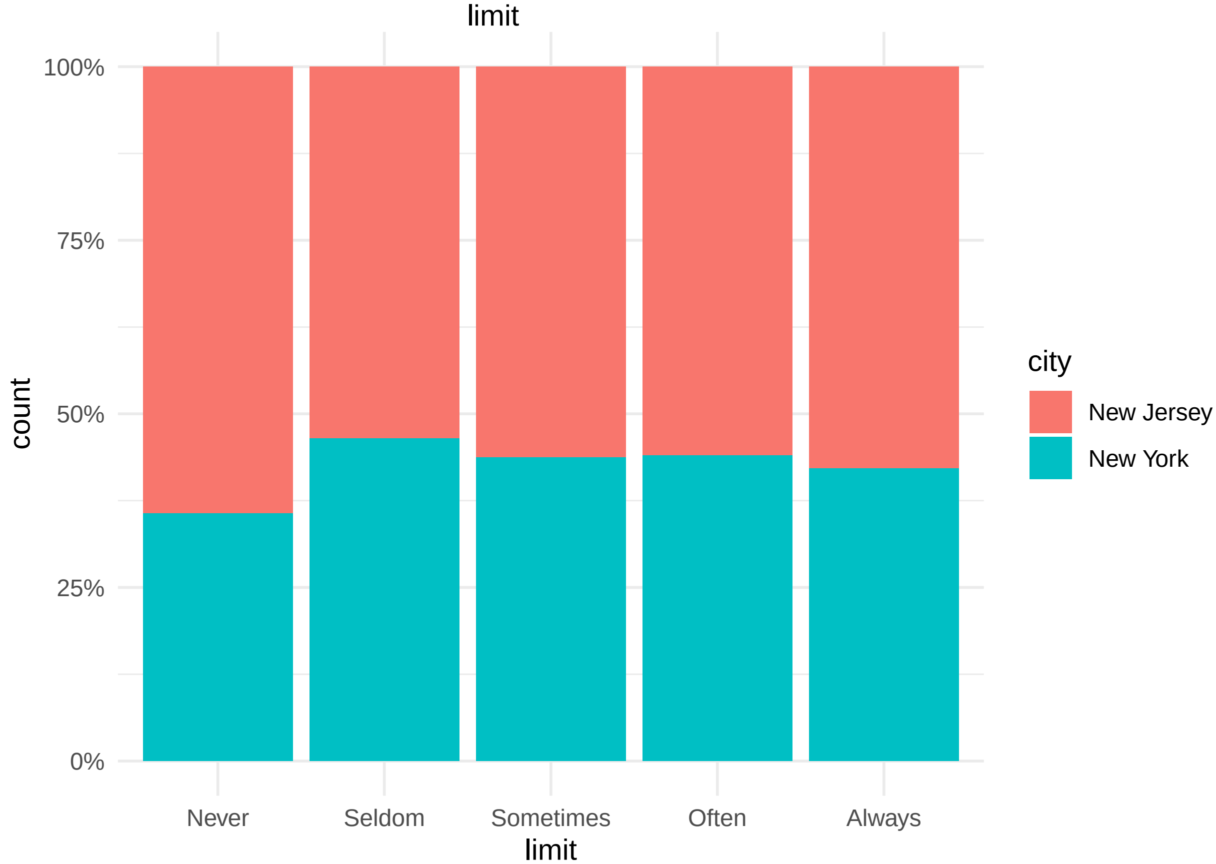
```
for(i in 1:length(names(reduced_data))){
  if(i != length(reduced_data)){
    for(j in (i+1):length(reduced_data)){
      make_plot(names(reduced_data)[i], names(reduced_data)[j])
    }
  }
}
```

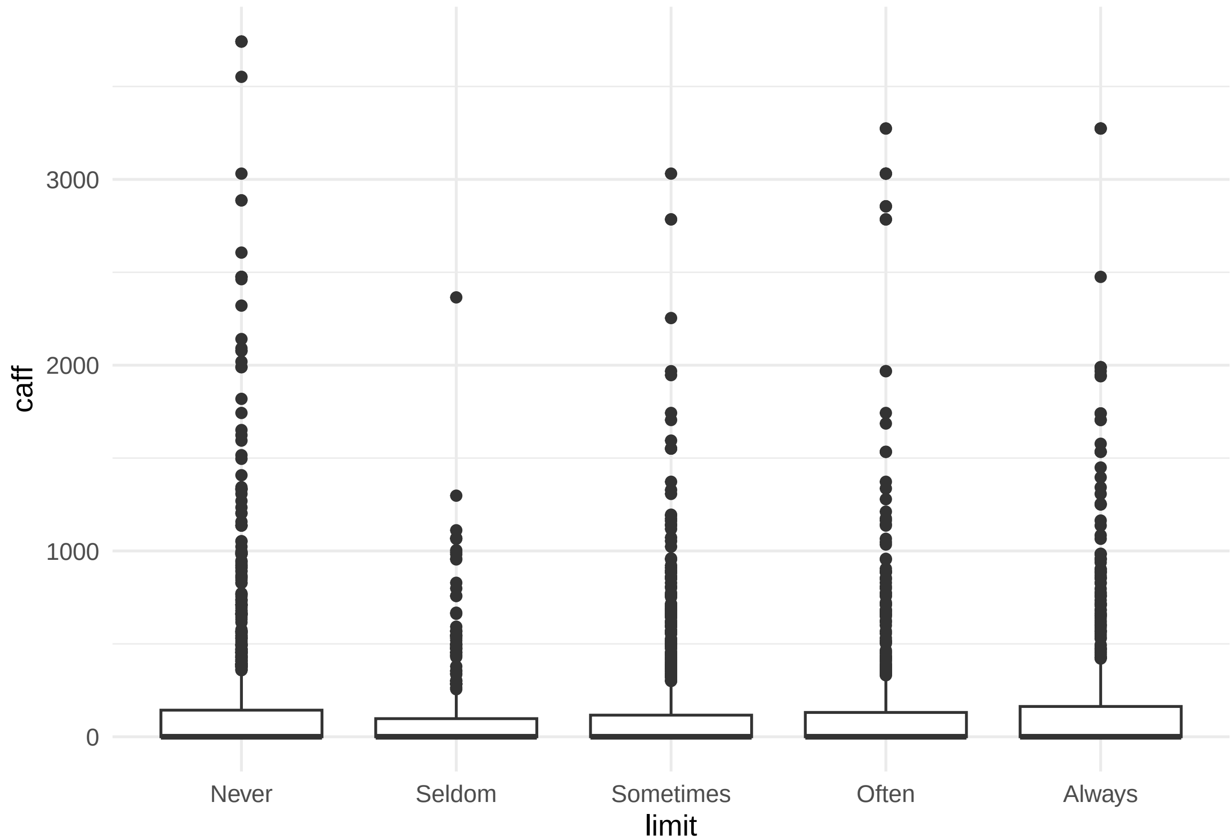



100%
75%
50%
25%
0%
count
Never
Seldom
Sometimes
Often
Always
limit
gender
F
M

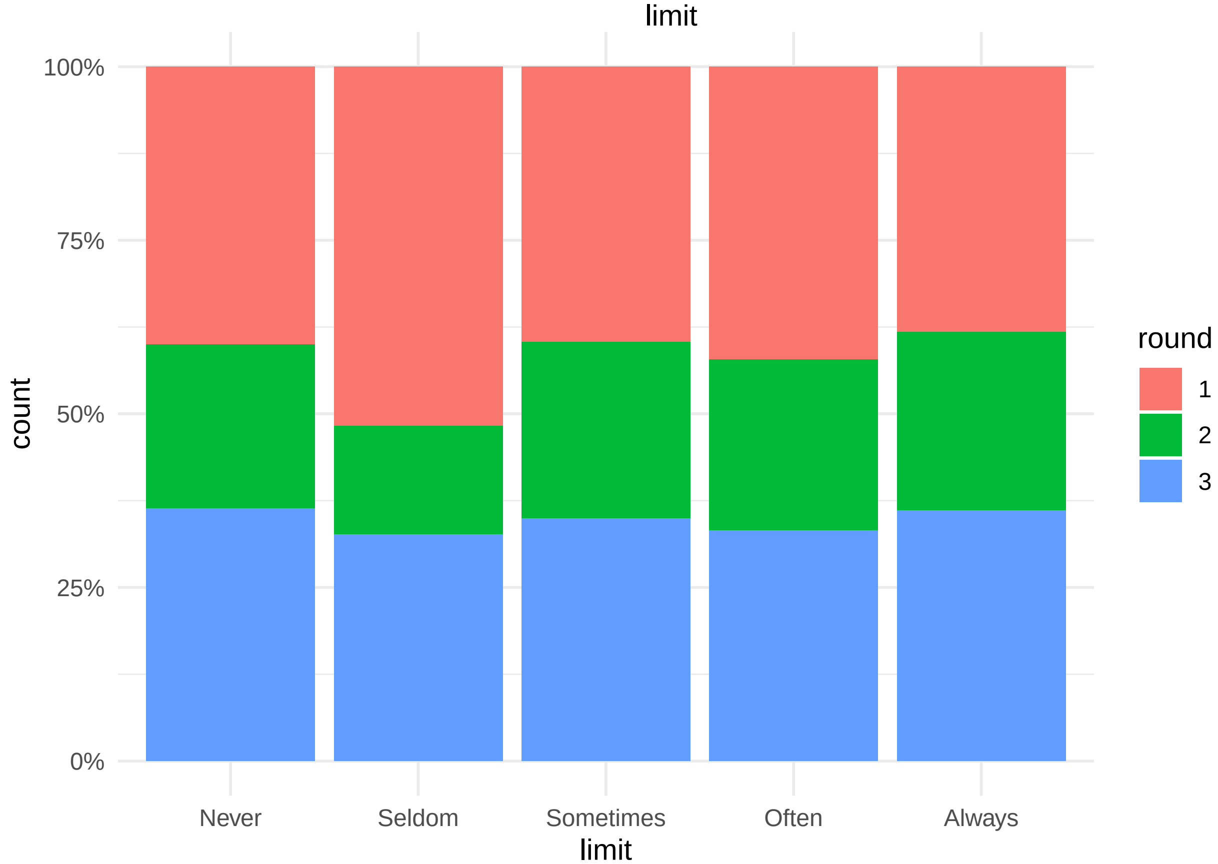

100%
75%
50%
25%
0%
count
Never
Seldom
Sometimes
Often
Always
limit
race
Asian
Black
Native
Other
White


100%
75%
50%
25%
0%
count
Never
Seldom
Sometimes
Often
Always
limit
edu
Associates Degree
College Degree
Graduate Degree
High School
Less than High School
Some College
Some High School

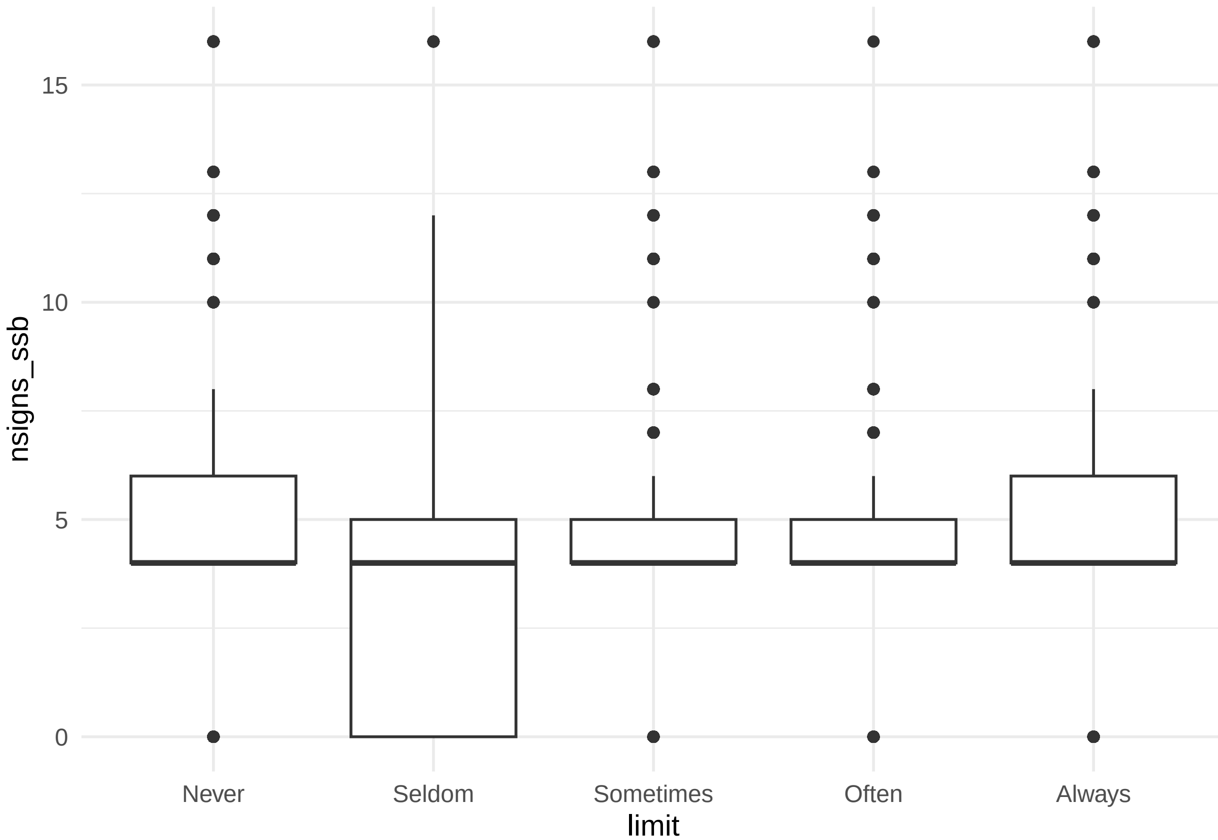

100%
75%
50%
25%
0%
count
Never
Seldom
Sometimes
Often
Always
limit
city
New Jersey
New York

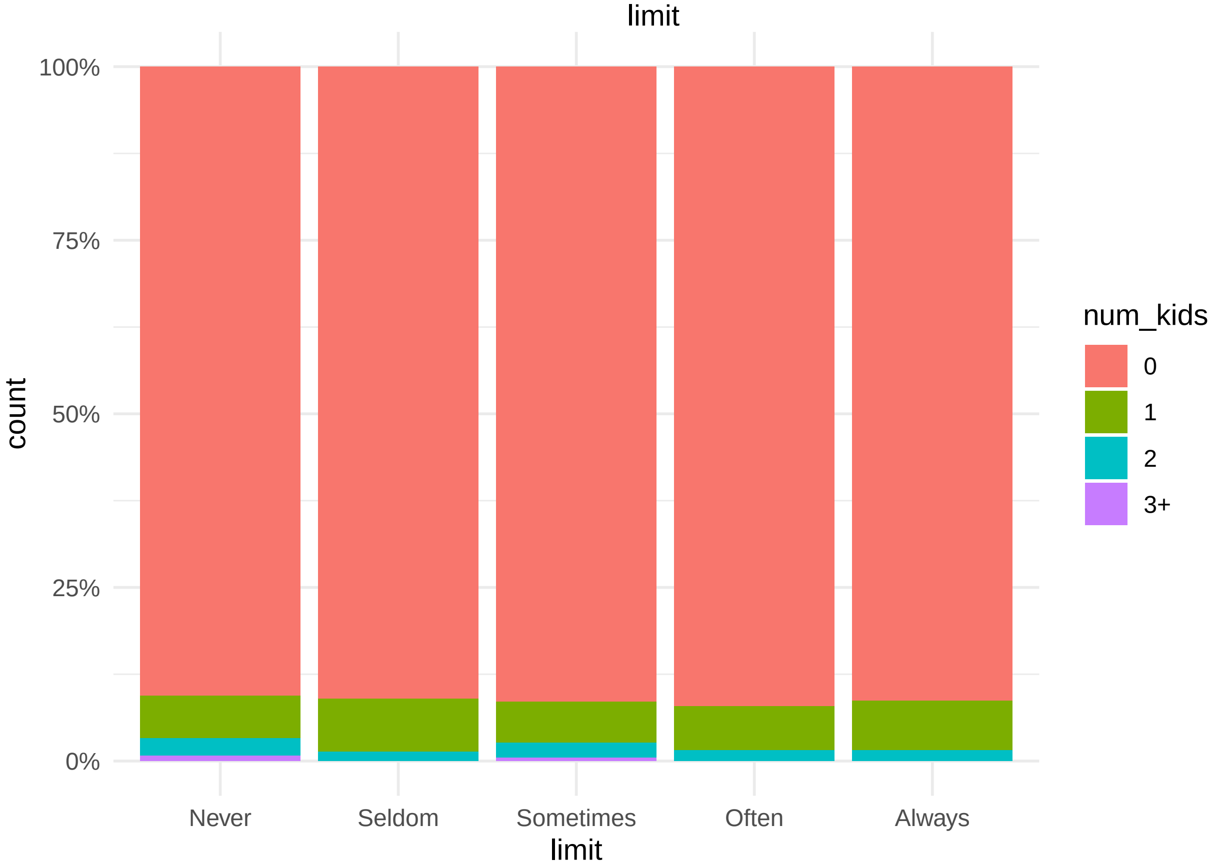


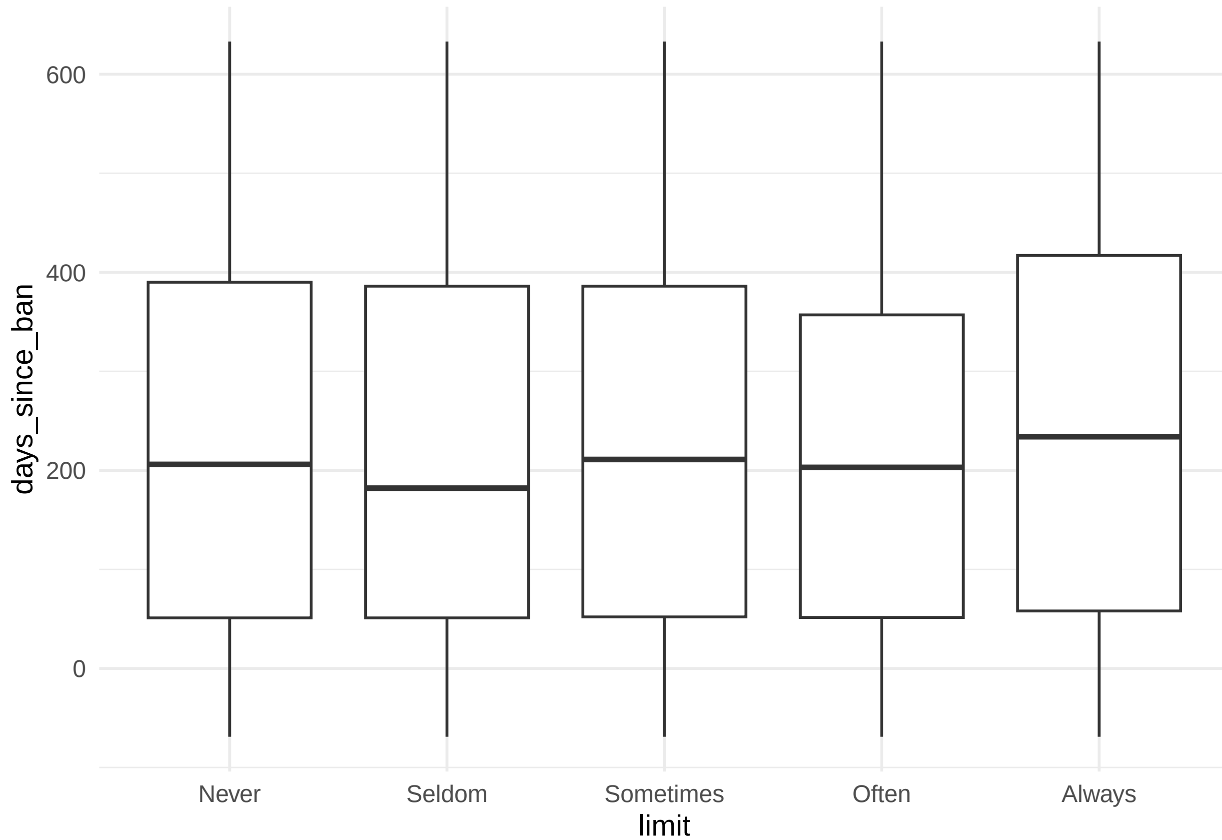
caff
3000
2000
1000
0
Never
Seldom
Sometimes
Often
Always
limit

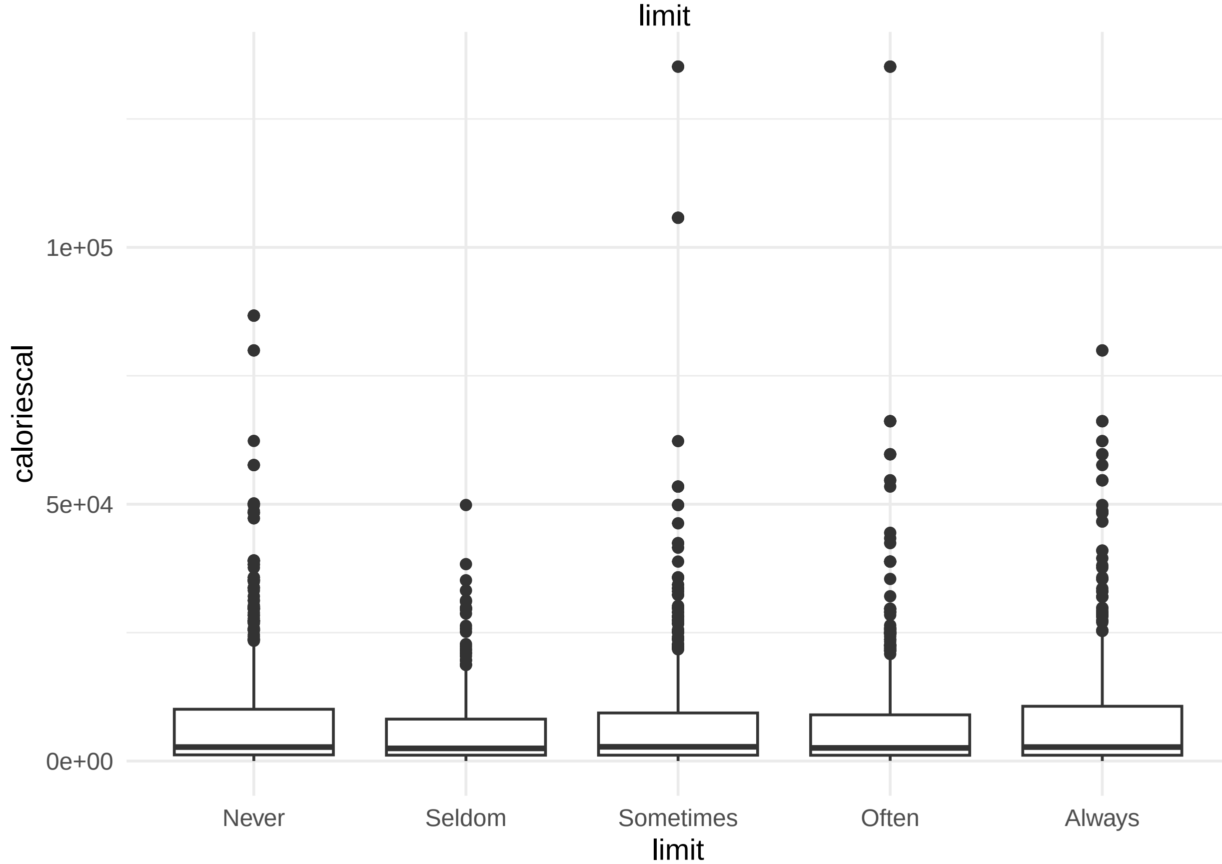


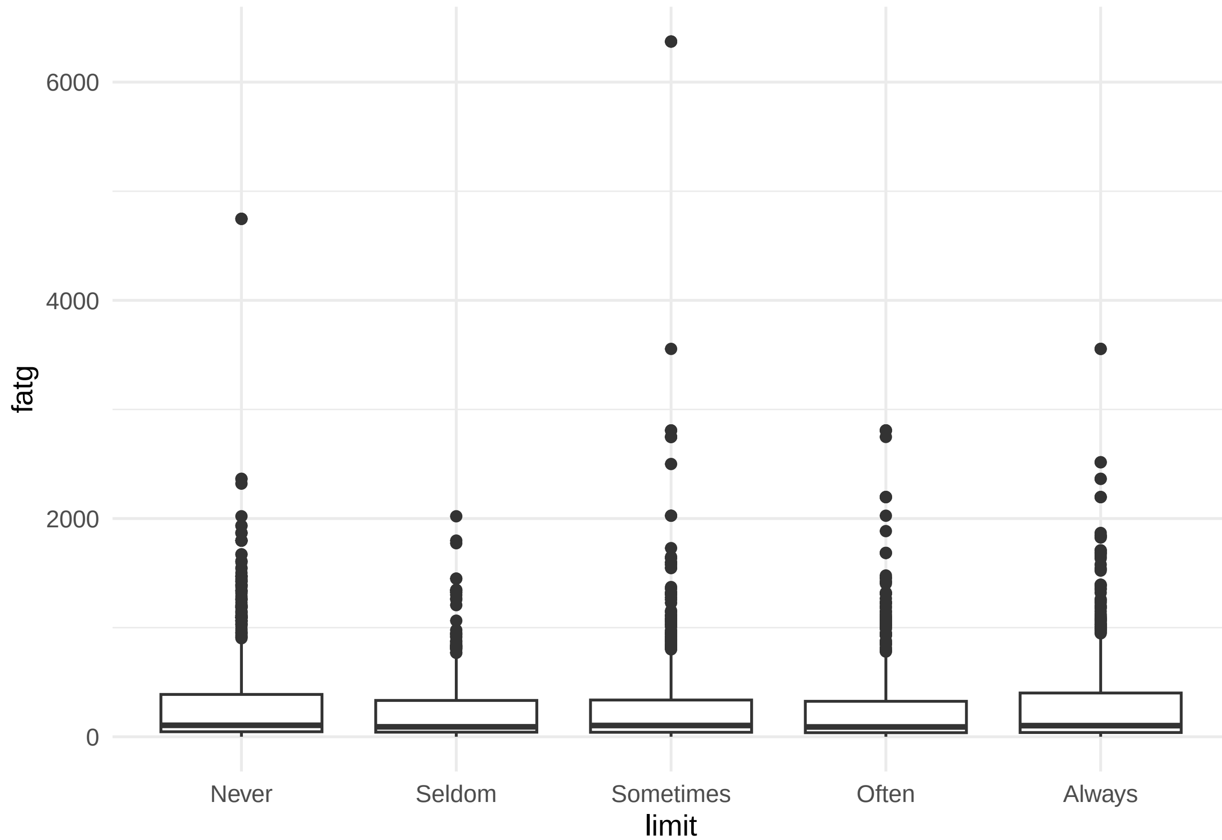
count
100%
75%
50%
25%
0%
Never
Seldom
Sometimes
Often
Always
limit
round
1
2
3

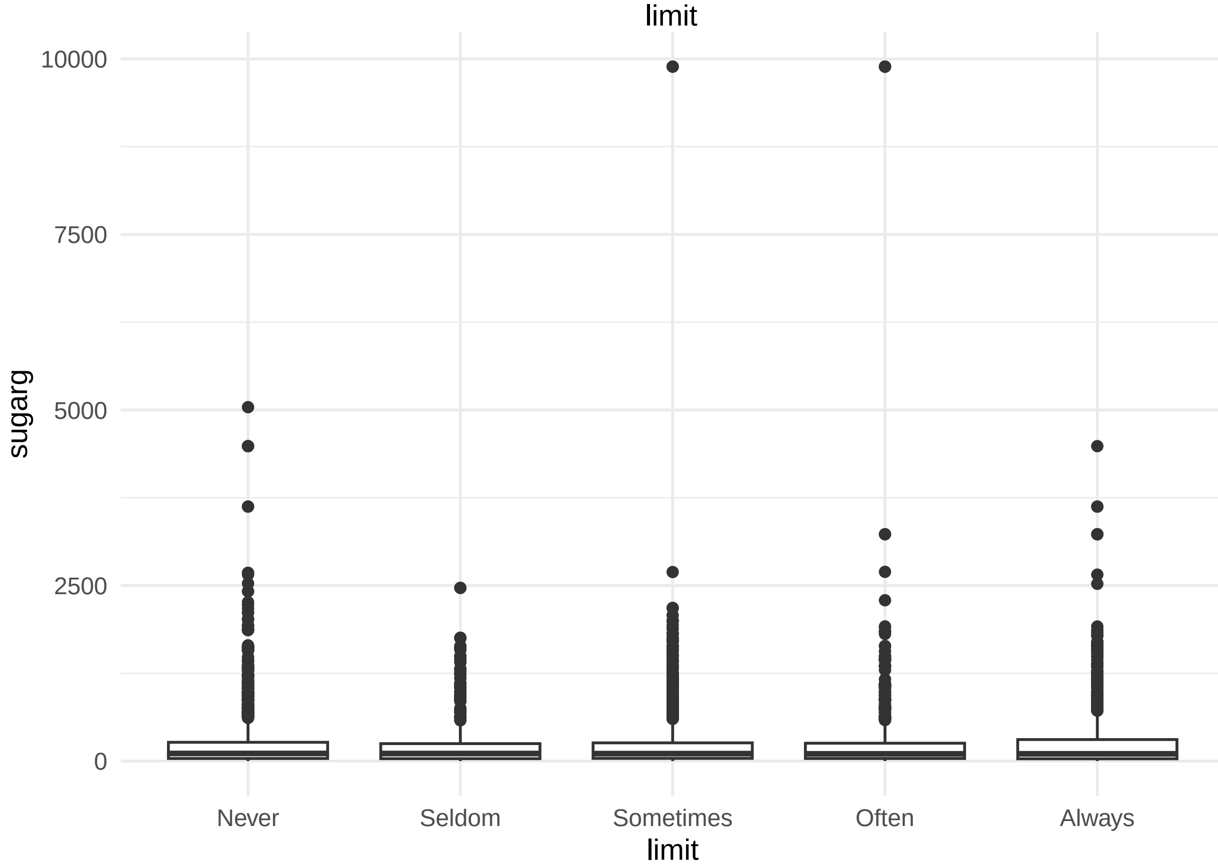

15
10
5
0
nsigns_ssb
Never
Seldom
Sometimes
Often
Always
limit

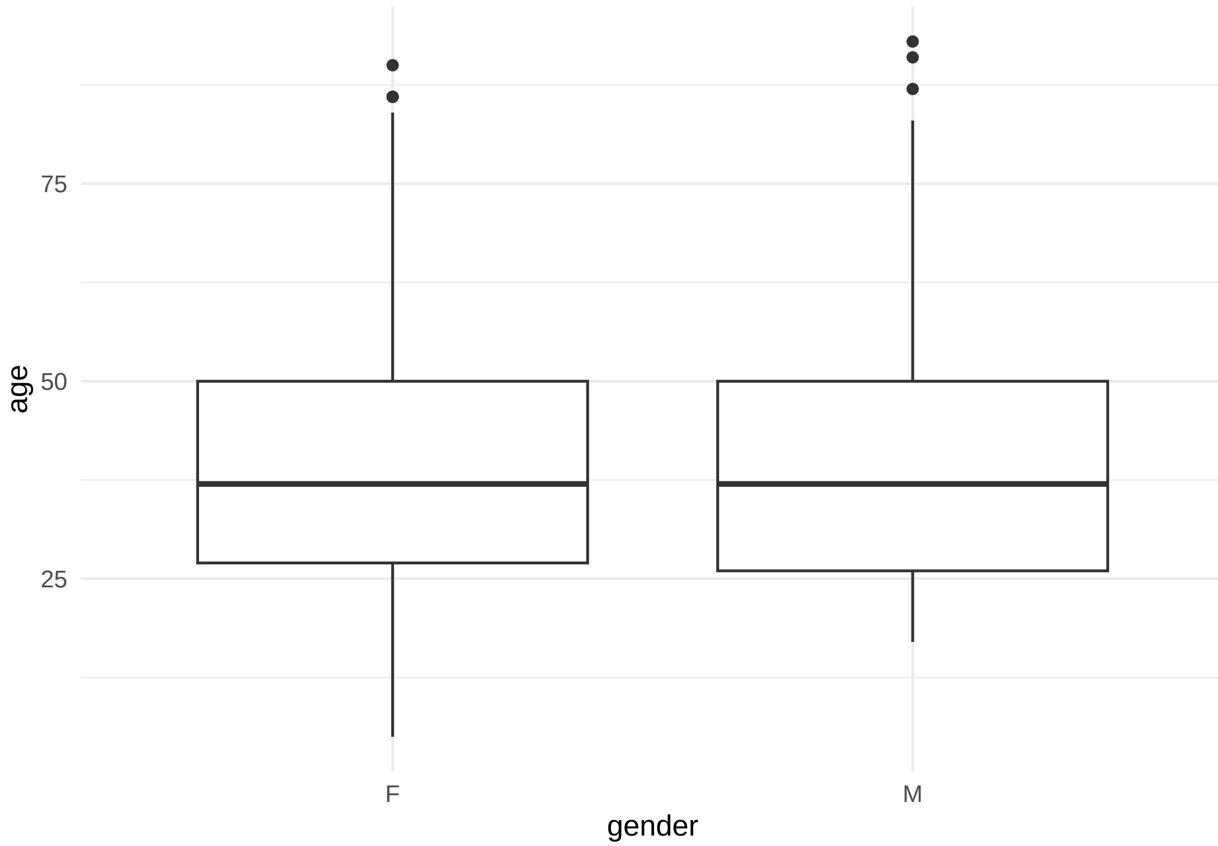

100%
75%
50%
25%
0%
count
Never
Seldom
Sometimes
Often
Always
limit
num_kids
0
1
2
3+


600
400
200
0
days_since_ban
Never
Seldom
Sometimes
Often
Always
limit

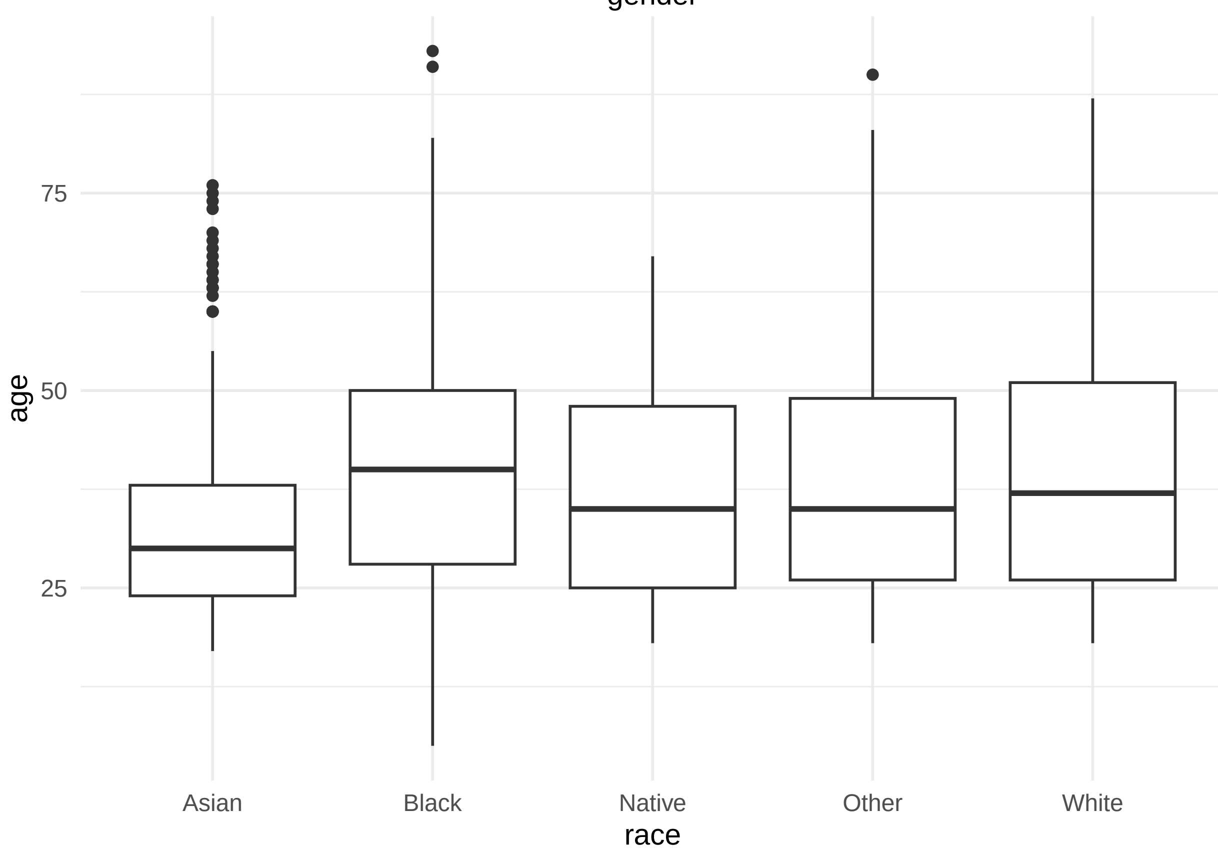

1e+05
5e+04
0e+00
caloriescal
Never
Seldom
Sometimes
Often
Always
limit


6000
4000
2000
0
fatg
Never
Seldom
Sometimes
Often
Always
limit

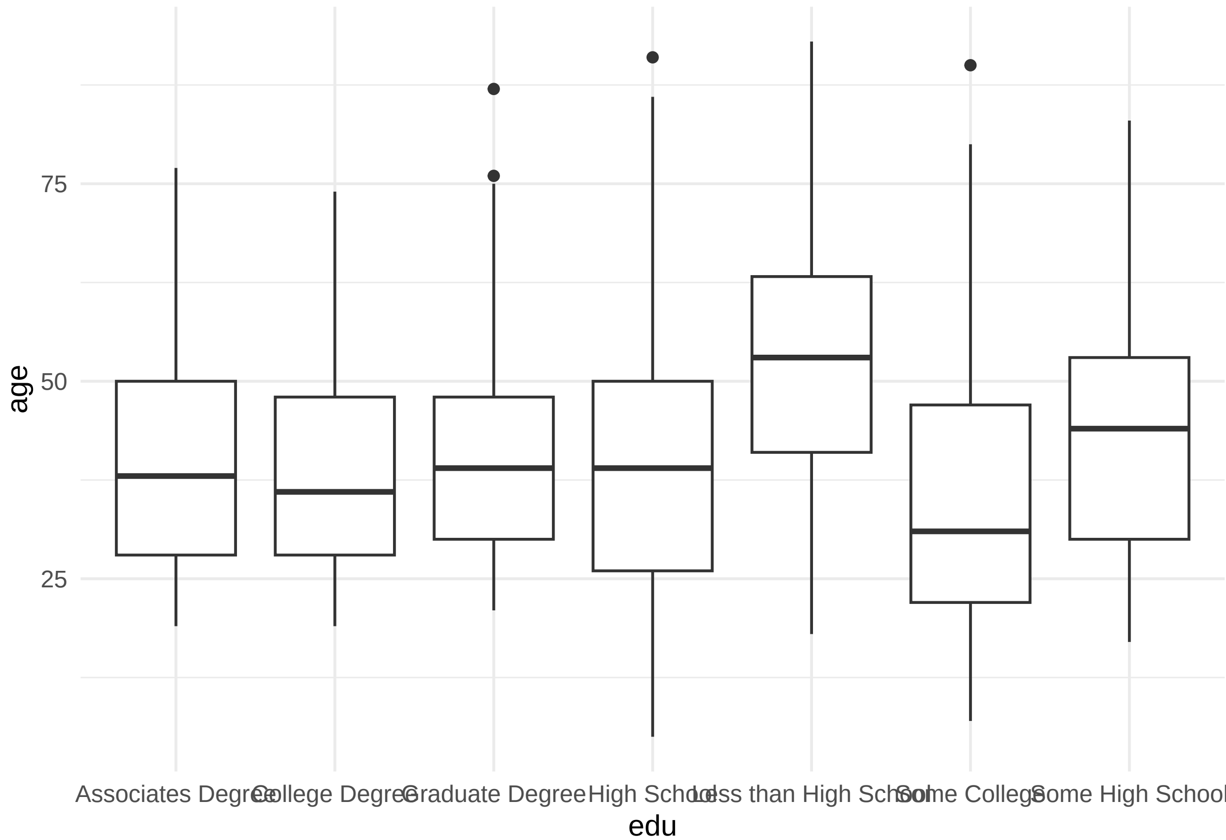

10000
7500
5000
2500
0
sugarg
Never
Seldom
Sometimes
Often
Always
limit



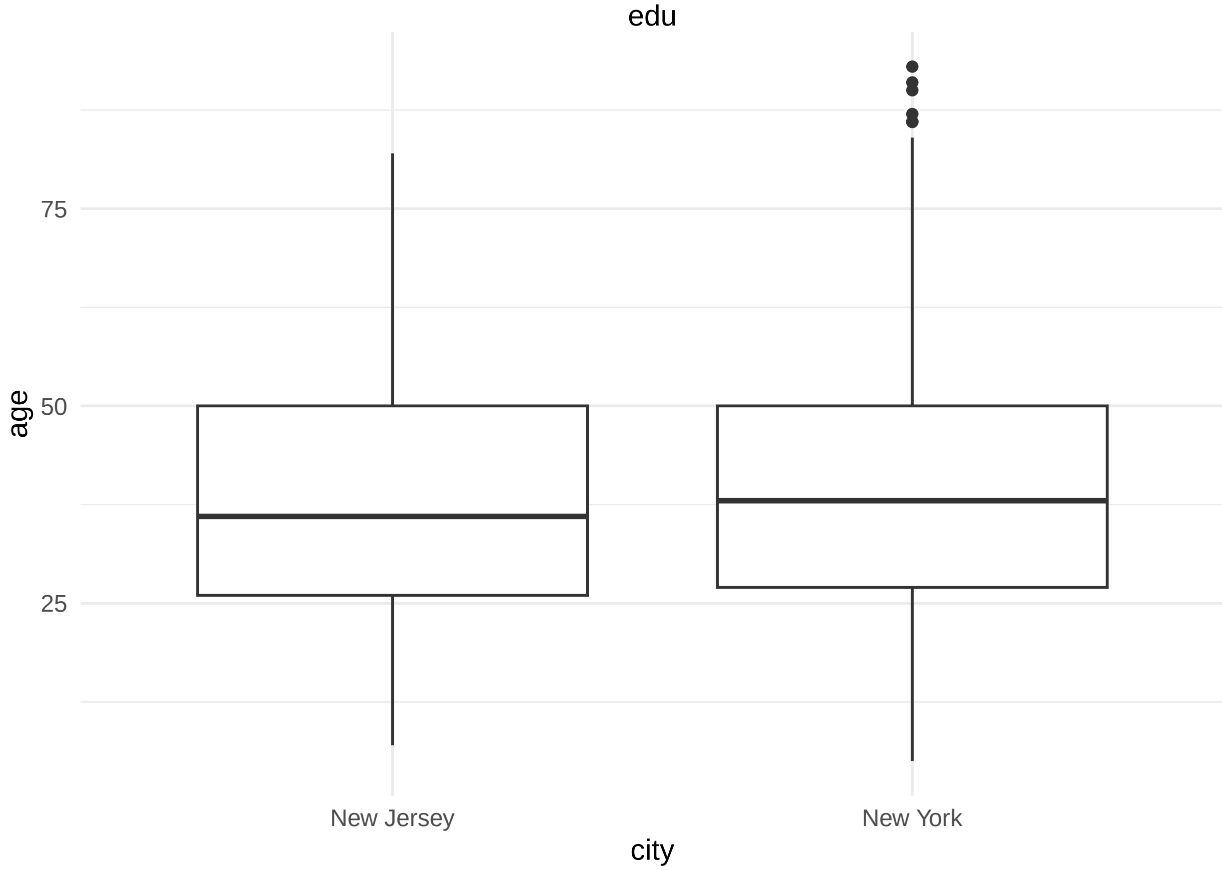
75
50
25
age
F
M
gender



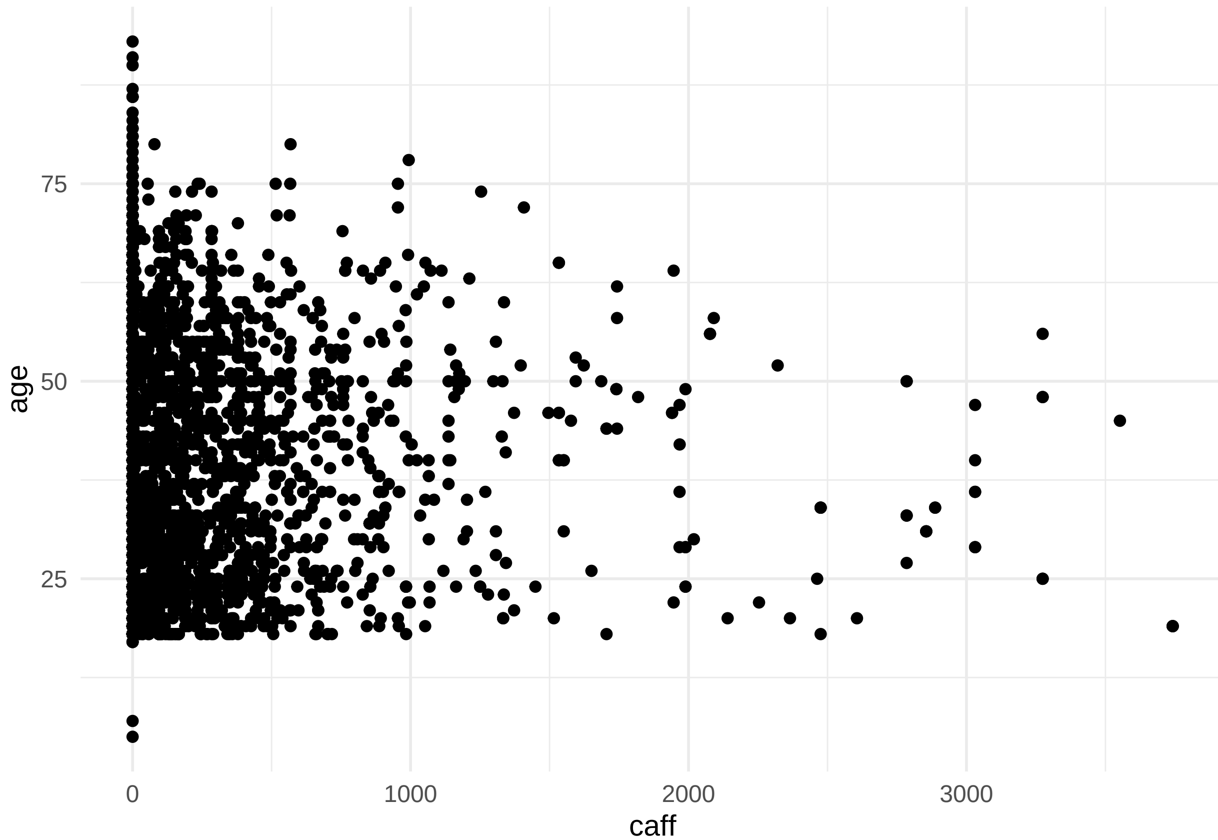
75
50
25
age
Asian
Black
Native
Other
White
race



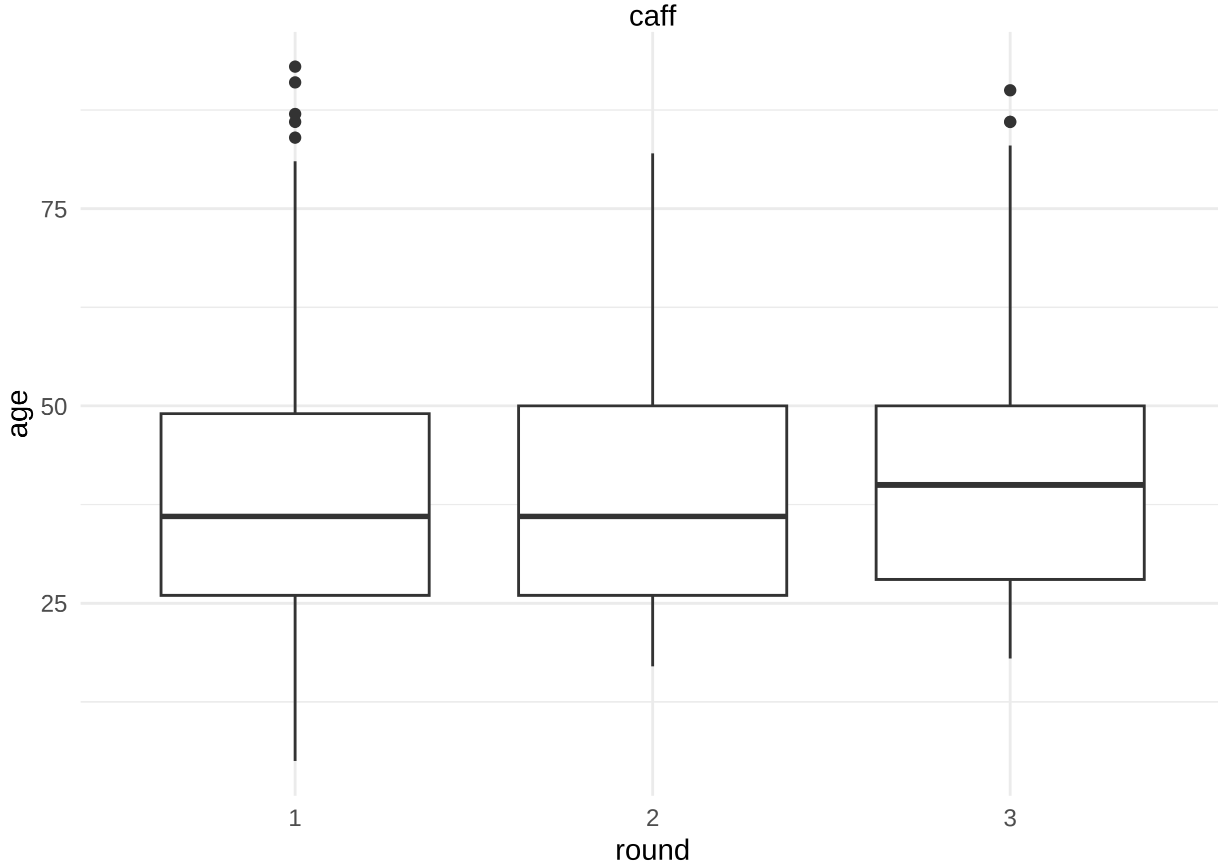
75
50
25
age
Associates Degree
College Degree
Graduate Degree
High School
Less than High School
Some College
Some High School
edu



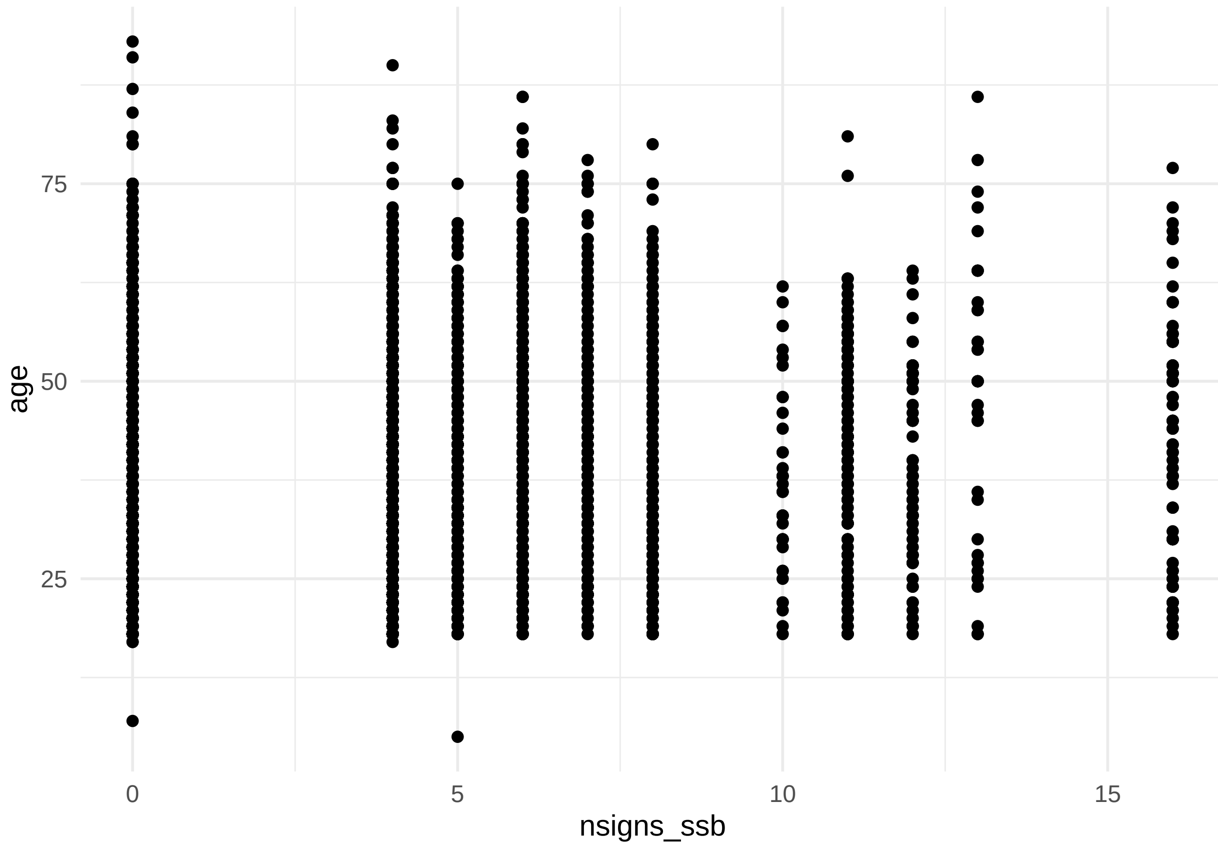
75
50
25
age
New Jersey
New York
city

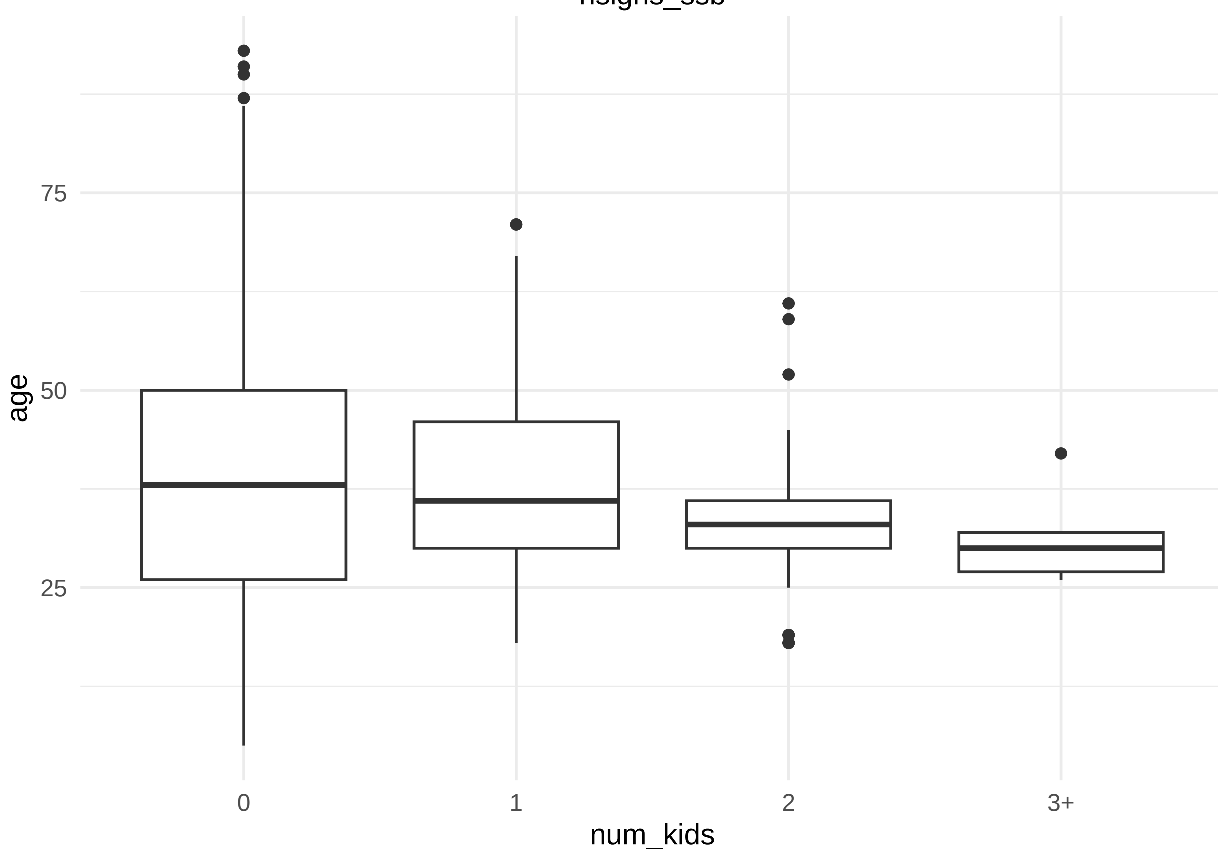


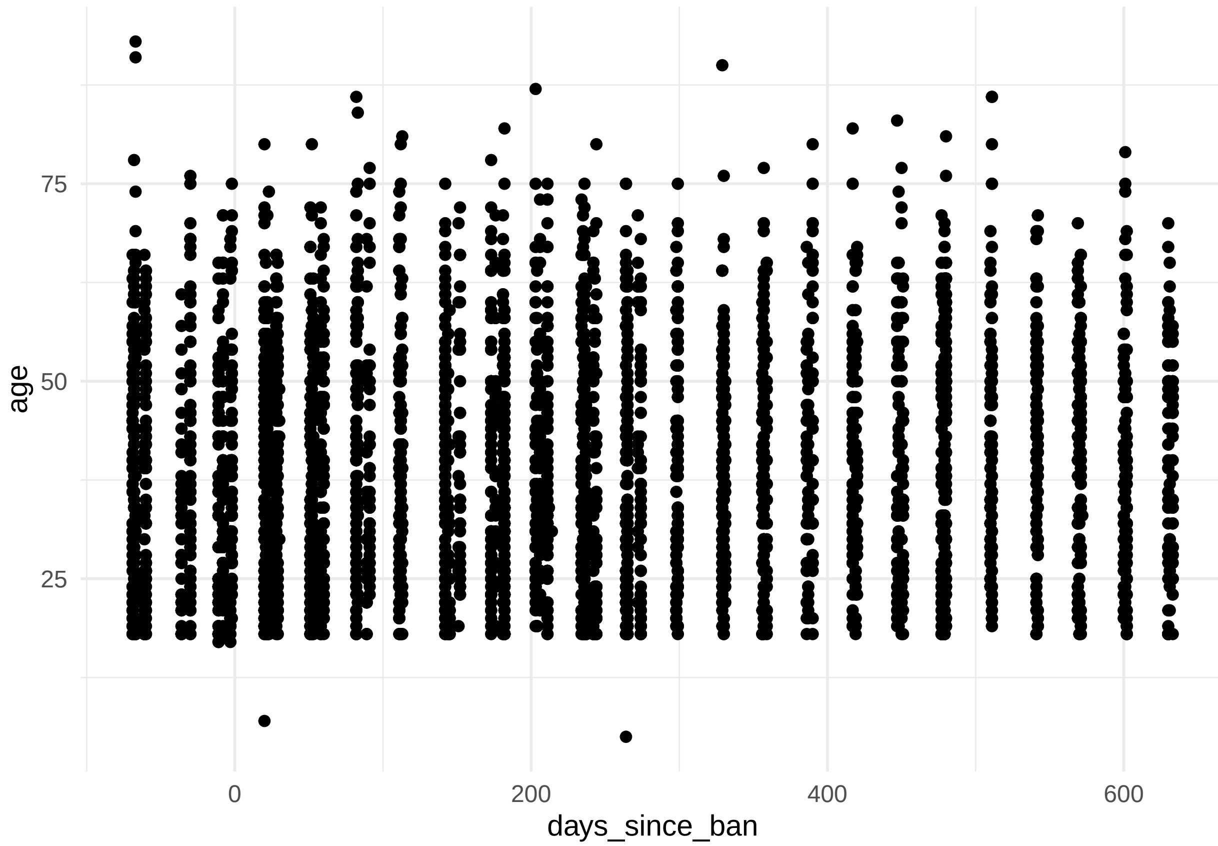
age
75
50
25
0
1000
2000
3000
caff



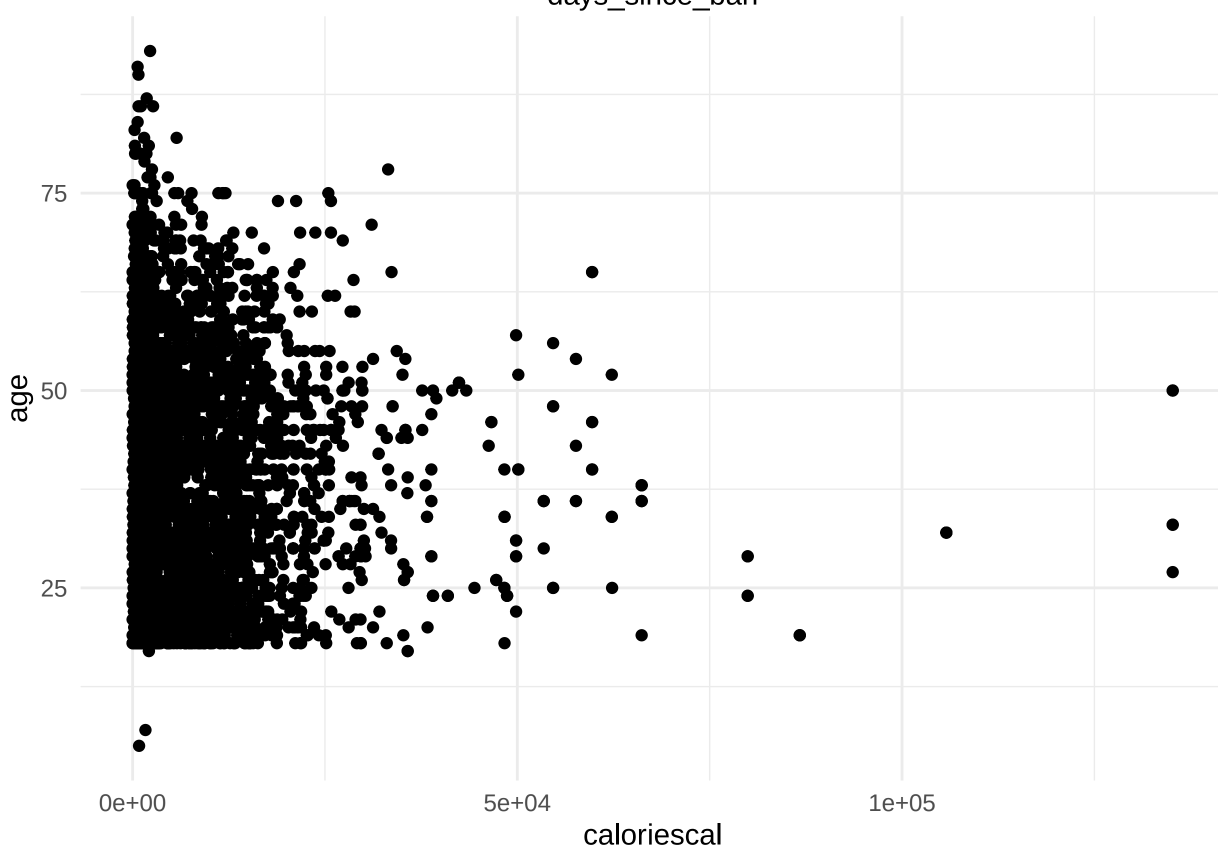
age
75
50
25
1
2
3
round


age
75
50
25
0
5
10
15
nsigns_ssb

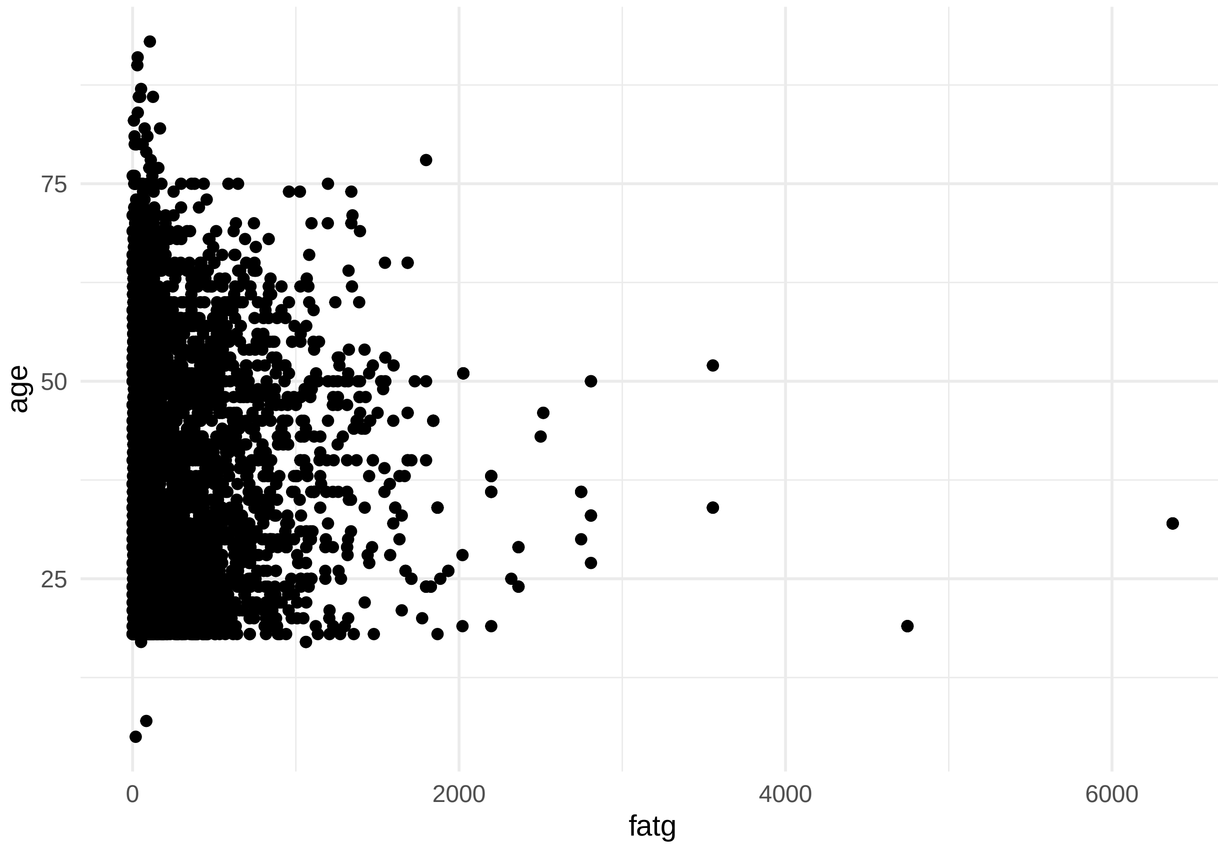

age
75
50
25
0
1
2
3+
num_kids



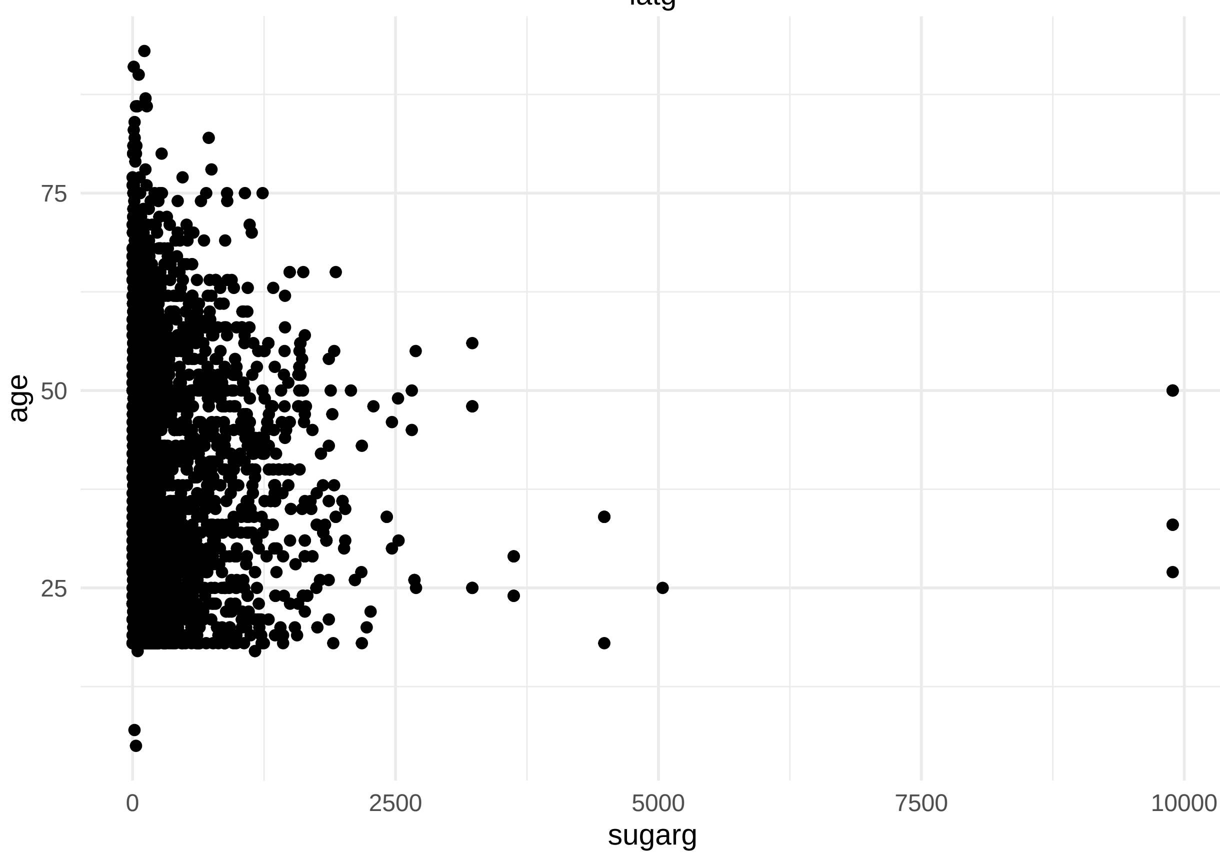
age
75
50
25
0
200
400
600
days_since_ban



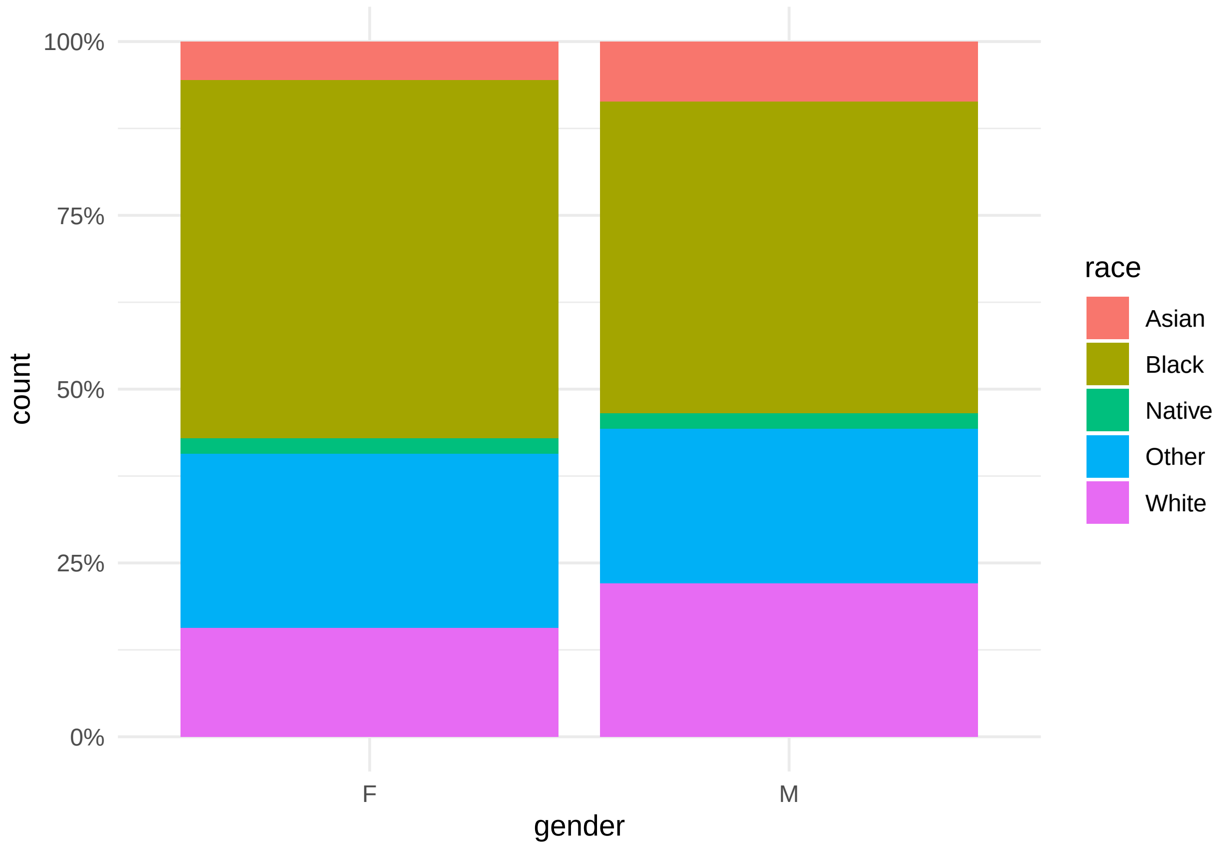
age
75
50
25
0e+00
5e+04
1e+05
caloriescal

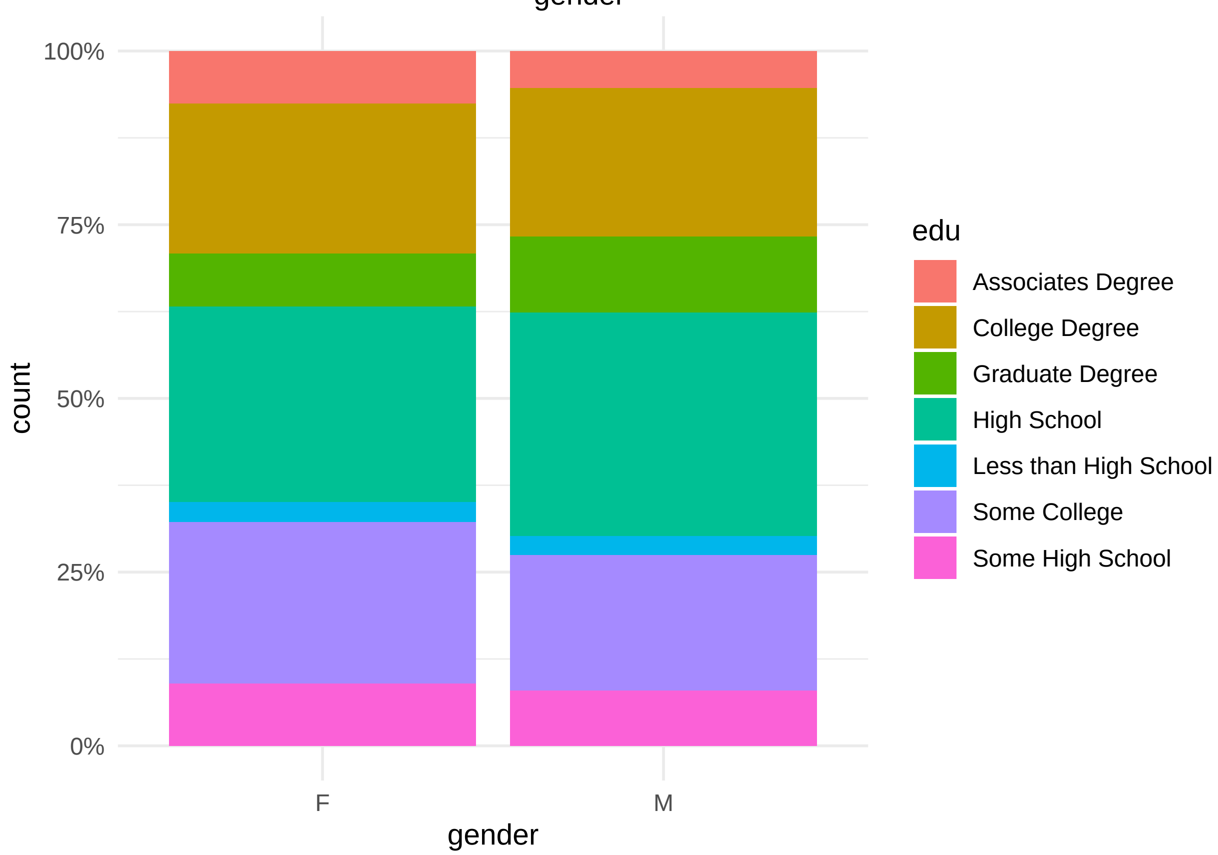


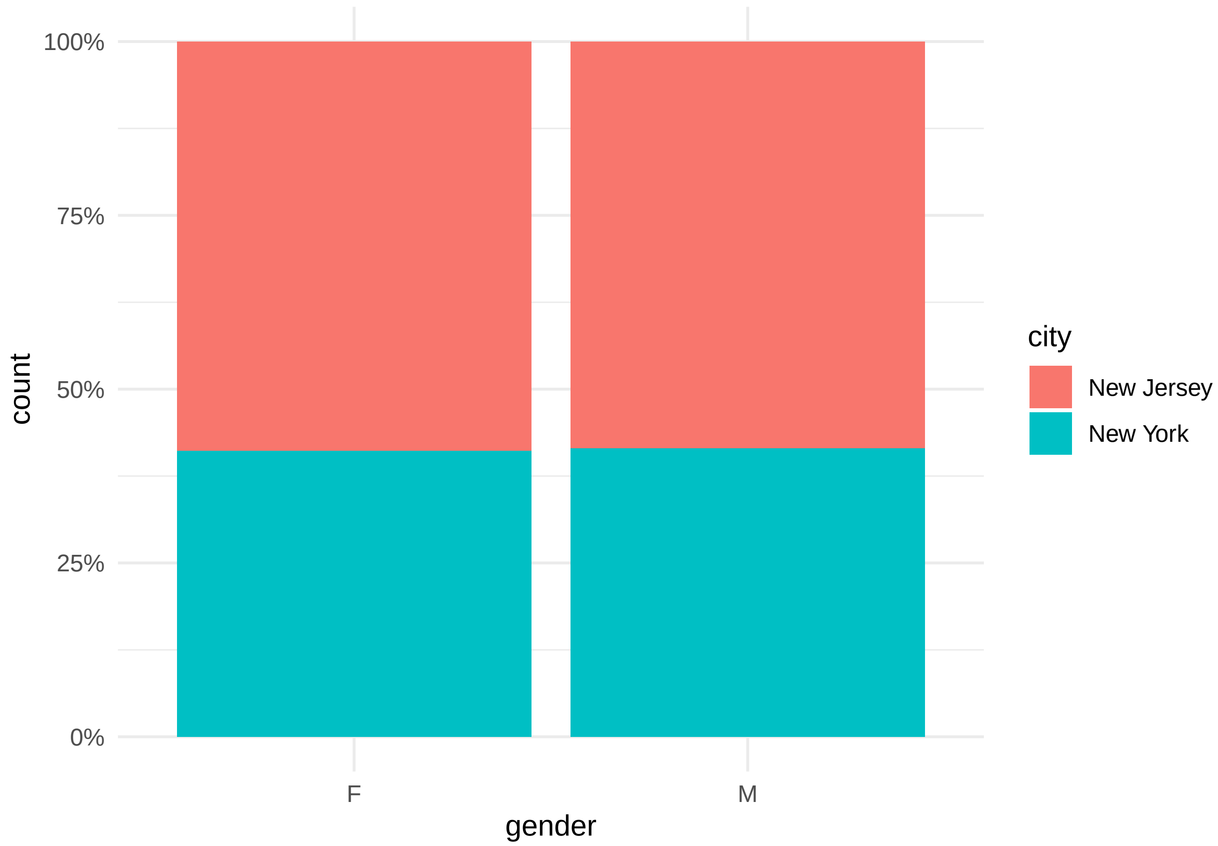
age
75
50
25
0
2000
4000
6000
fatg

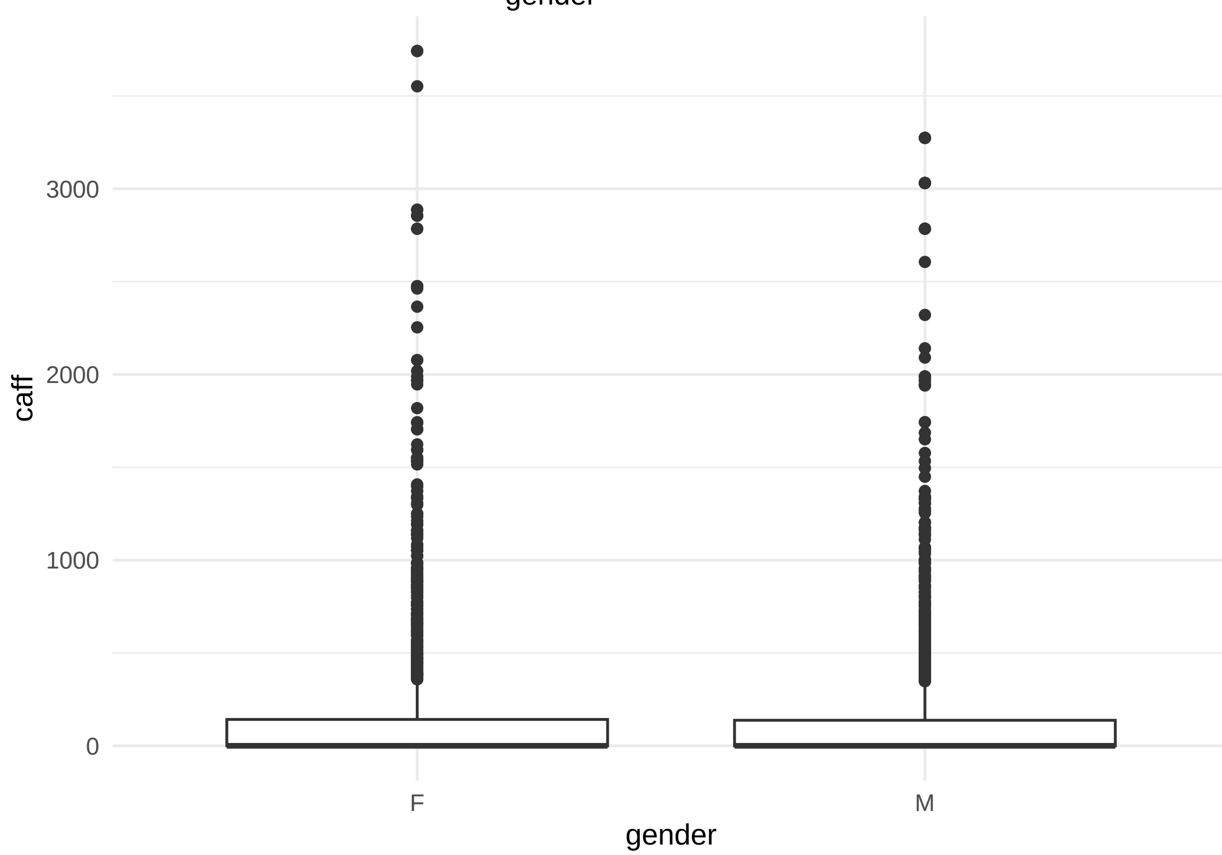


age
75
50
25
0
2500
5000
7500
10000
sugarg

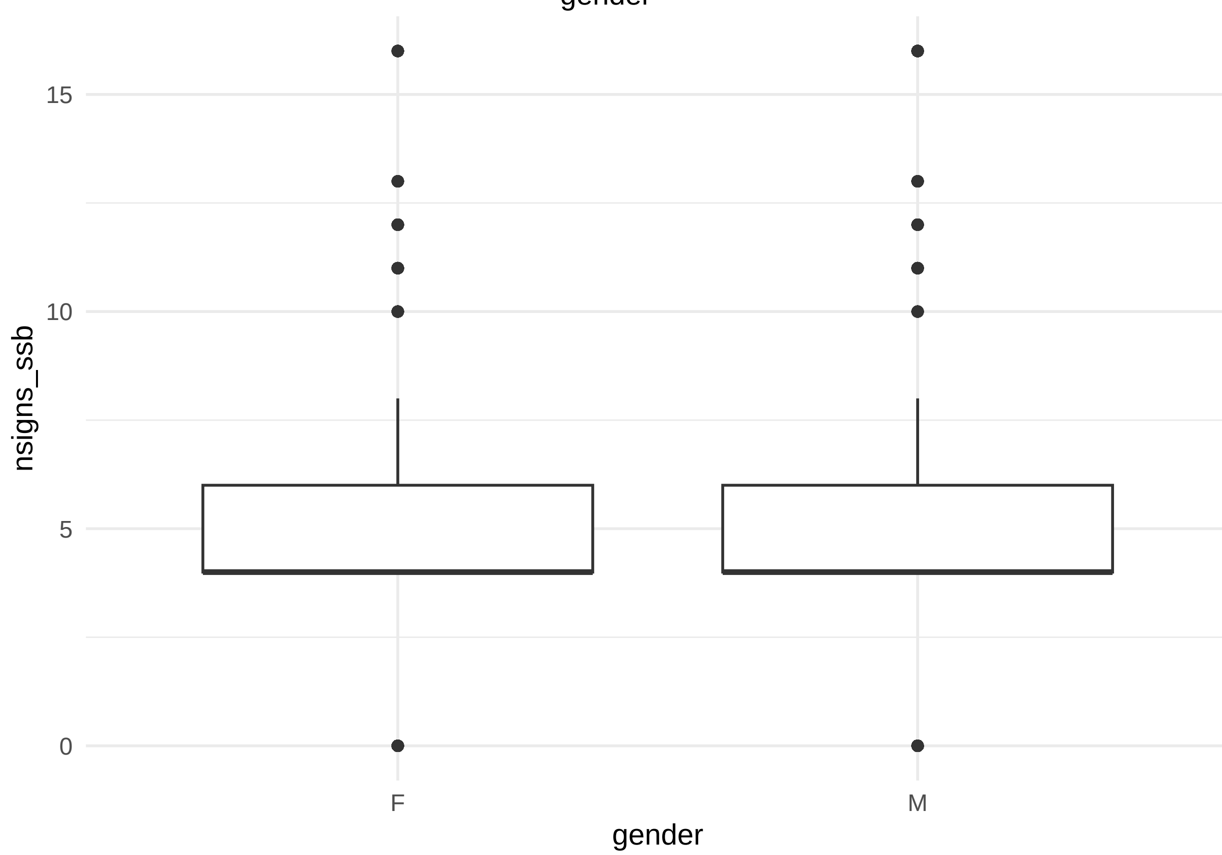

100%
75%
50%
25%
0%
count
F
M
gender
race
Asian
Black
Native
Other
White

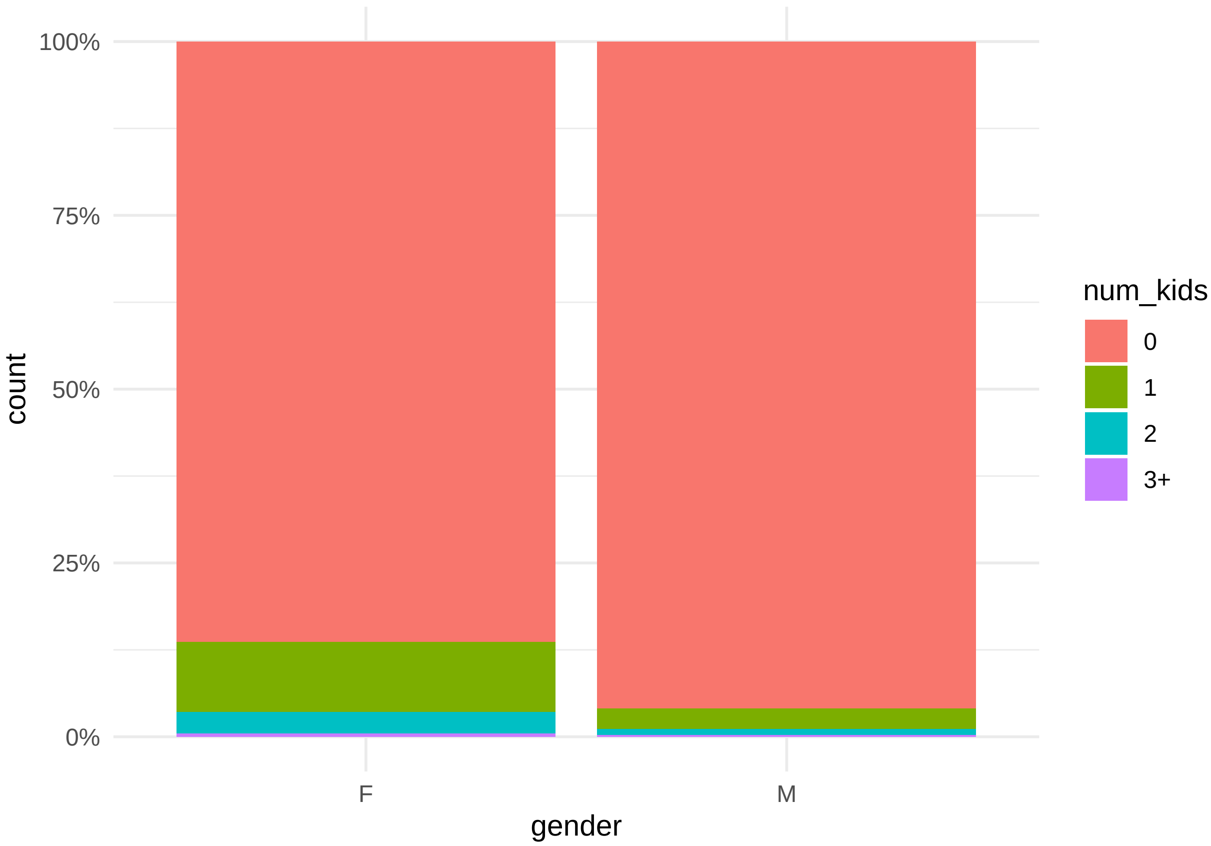

100%
75%
50%
25%
0%
count
F
M
gender
edu
Associates Degree
College Degree
Graduate Degree
High School
Less than High School
Some College
Some High School


100%
75%
50%
25%
0%
count
F
M
gender
city
New Jersey
New York

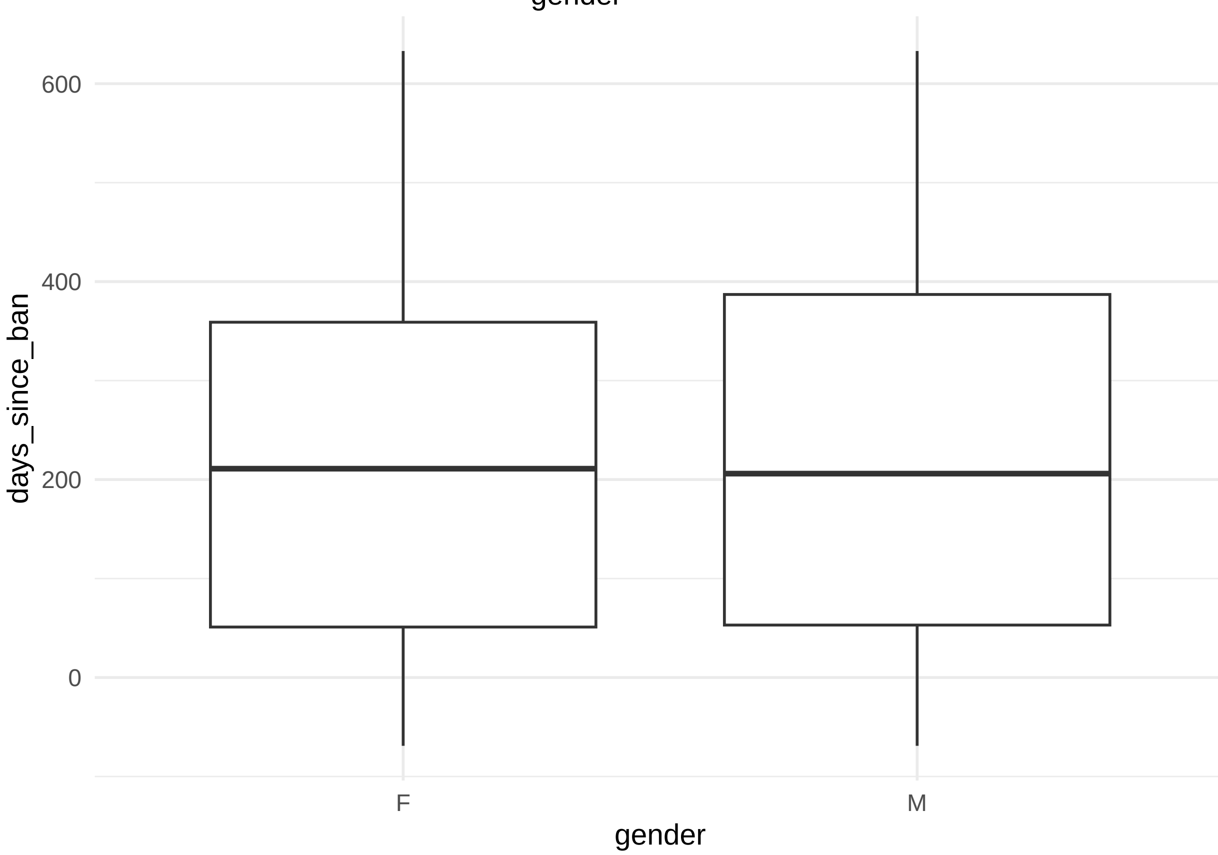

3000
2000
1000
0
caff
F
M
gender


100%
75%
50%
25%
0%
count
F
M
gender
round
1
2
3

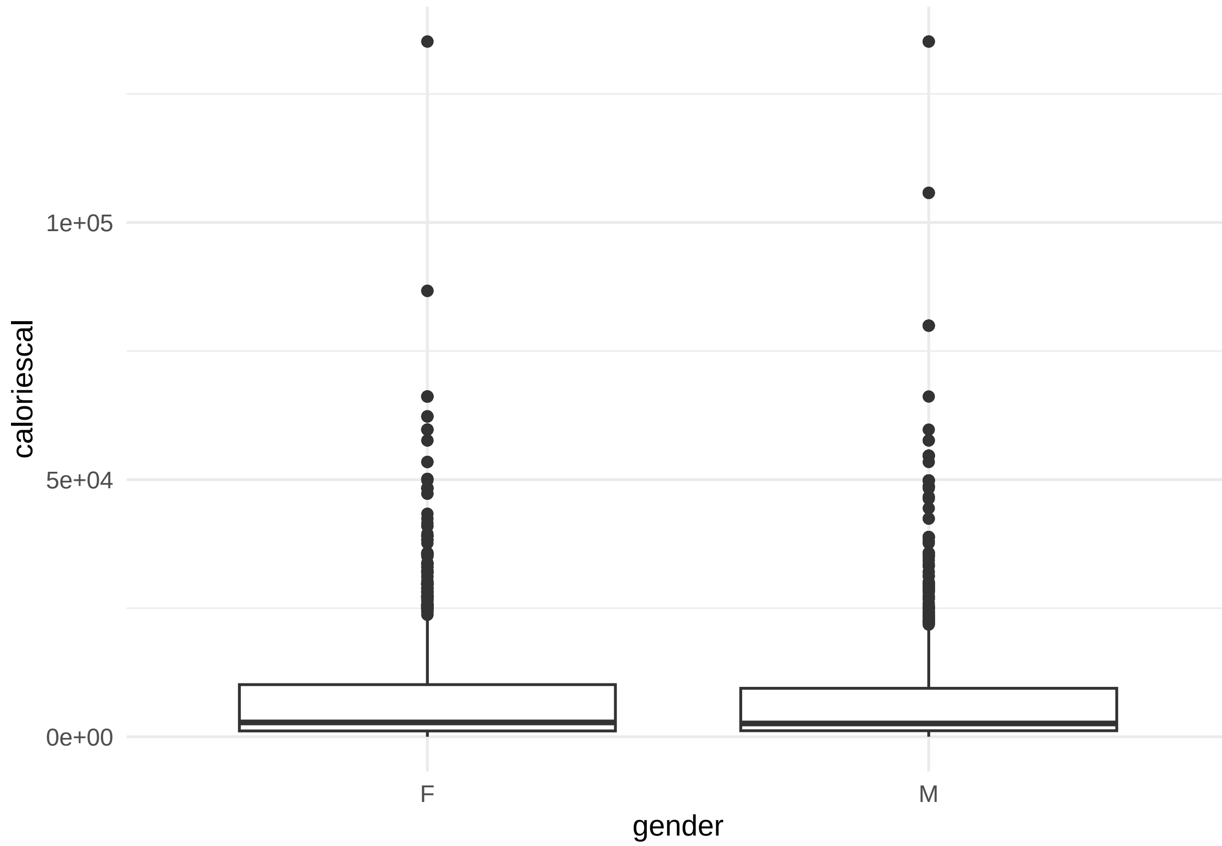

15
10
5
0
nsigns_ssb
F
M
gender

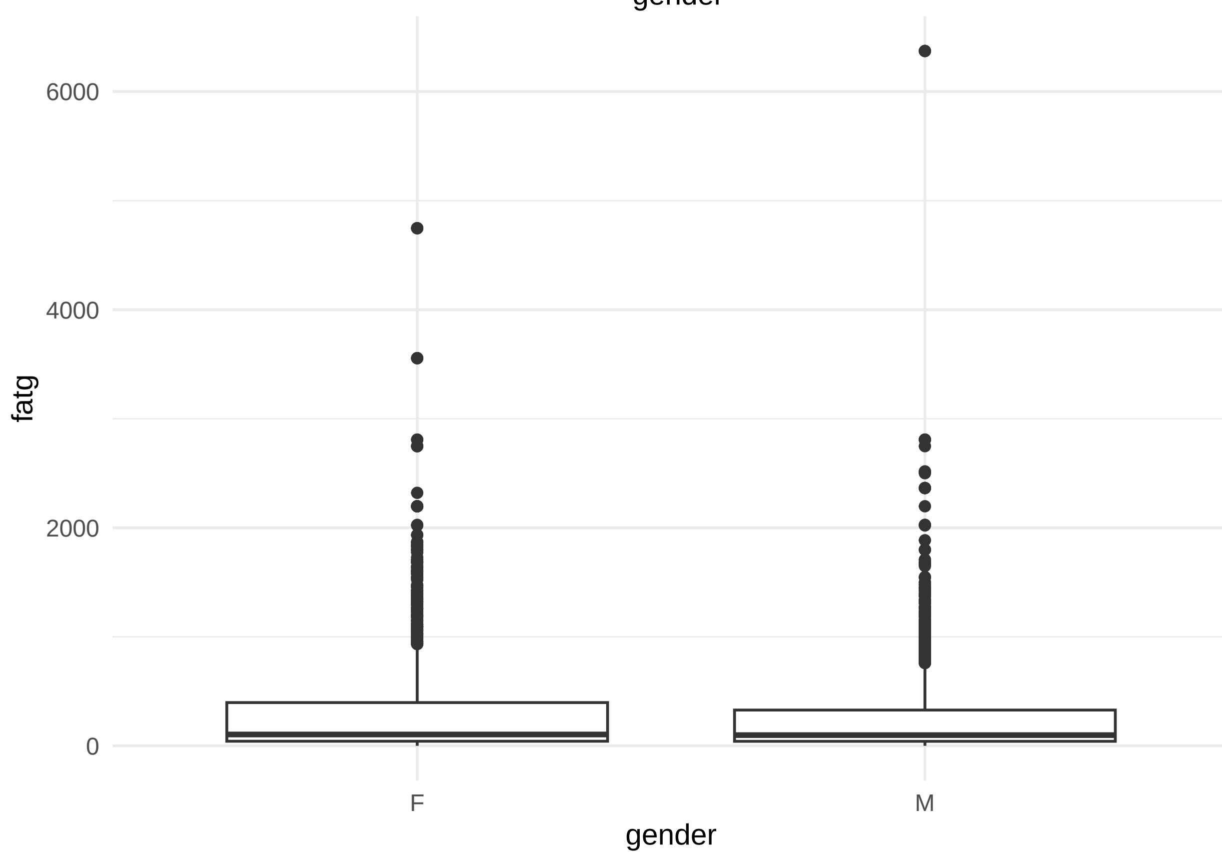


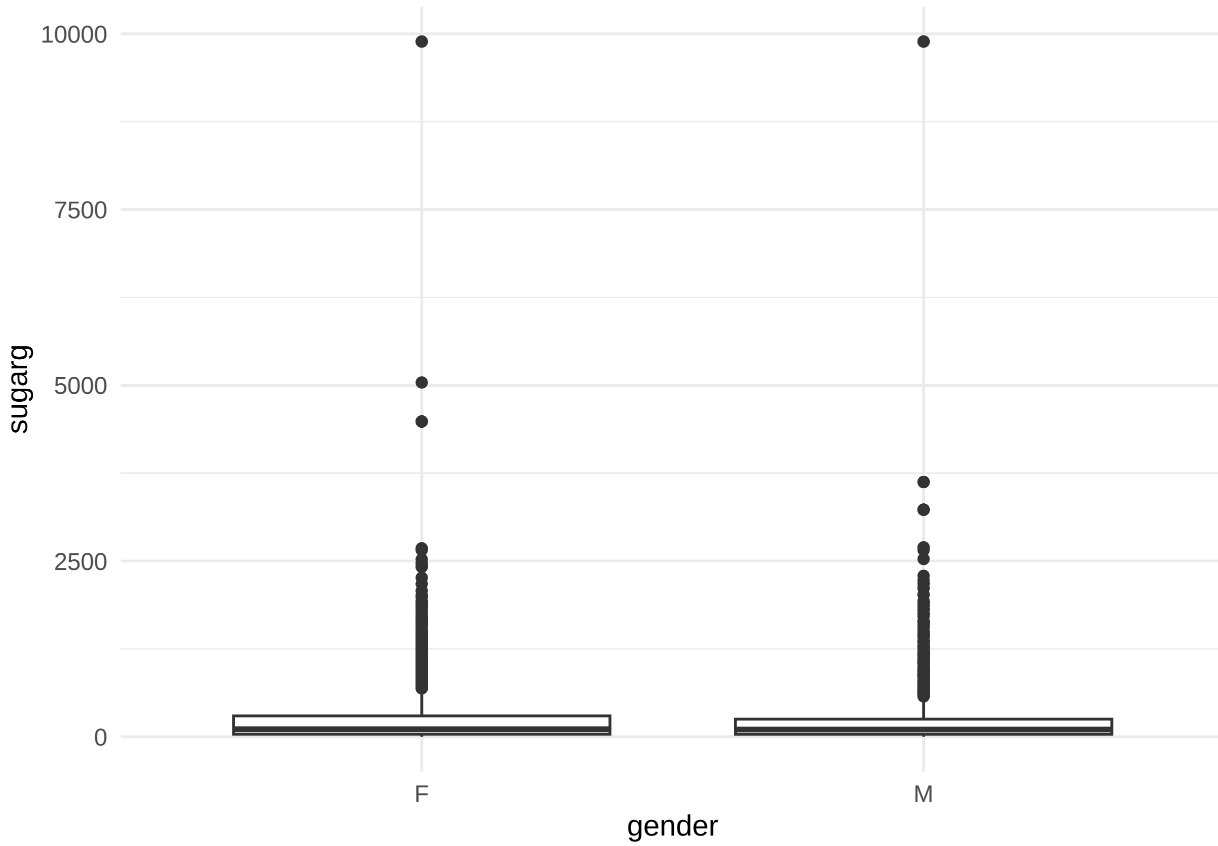
100%
75%
50%
25%
0%
count
F
M
gender
num_kids
0
1
2
3+



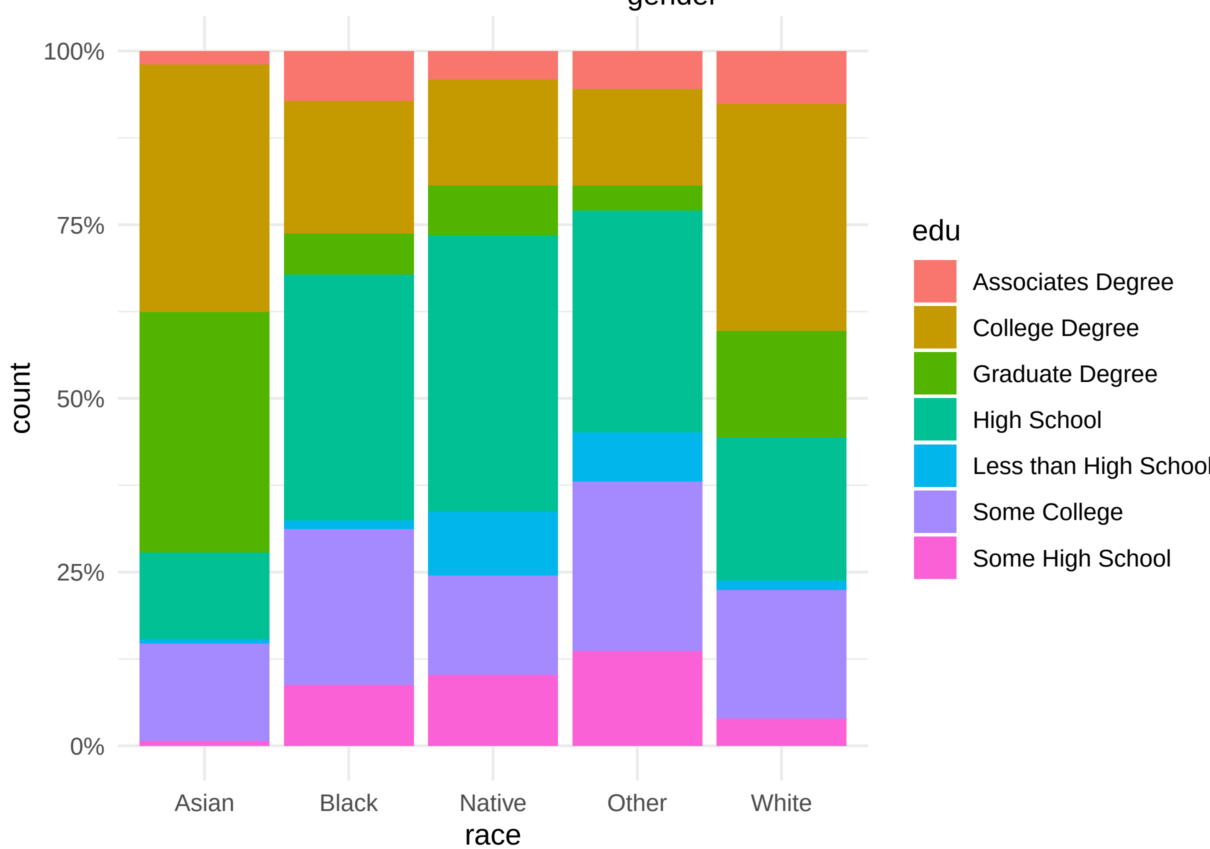
600
400
200
0
days_since_ban
F
M
gender


caloriescal
1e+05
5e+04
0e+00
F
M
gender

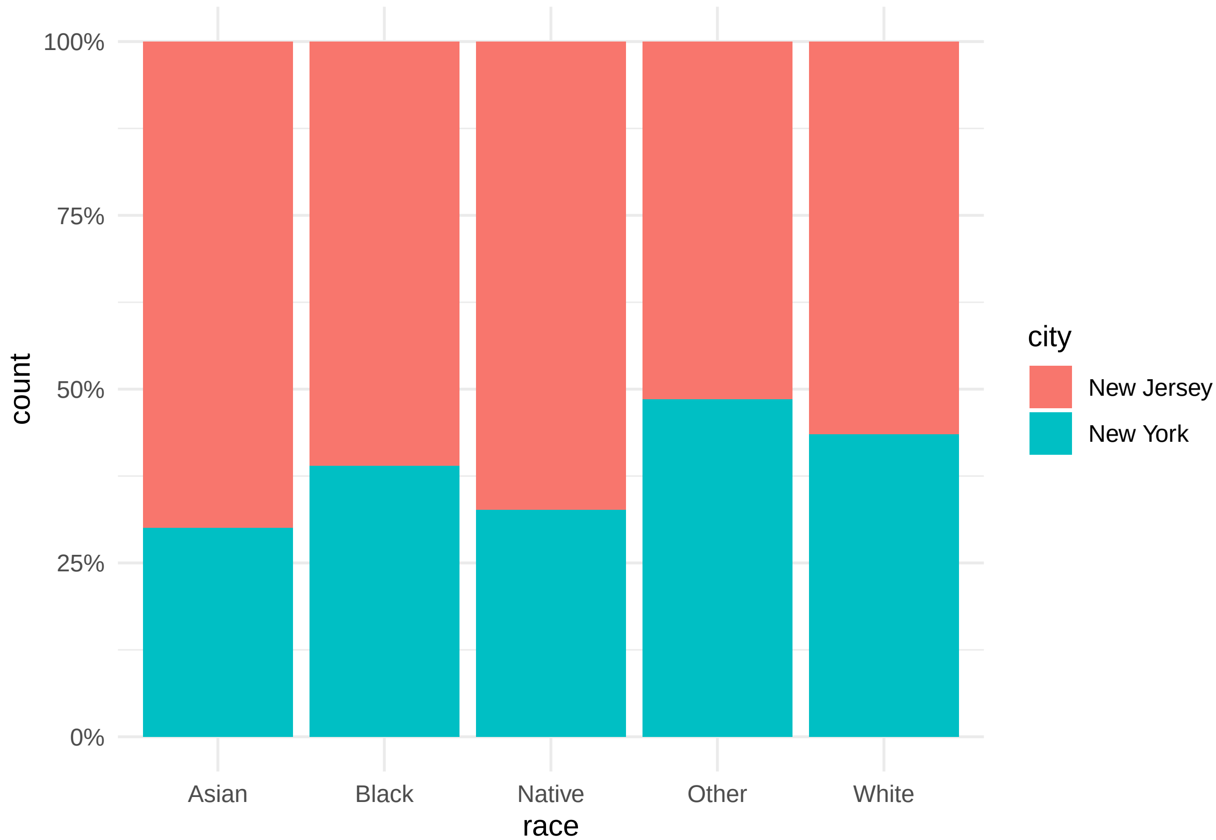

fatg
6000
4000
2000
0
F
M
gender



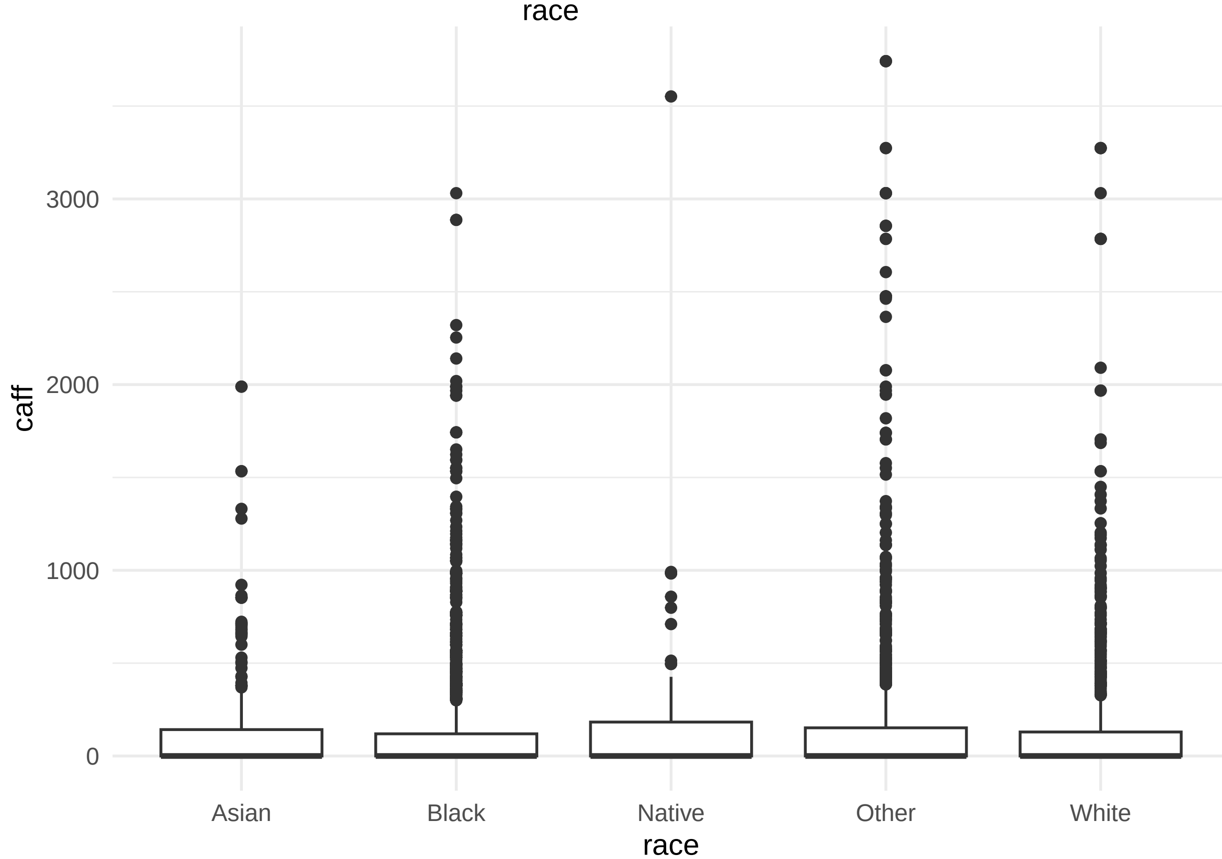
10000
7500
5000
2500
0
sugarg
F
M
gender



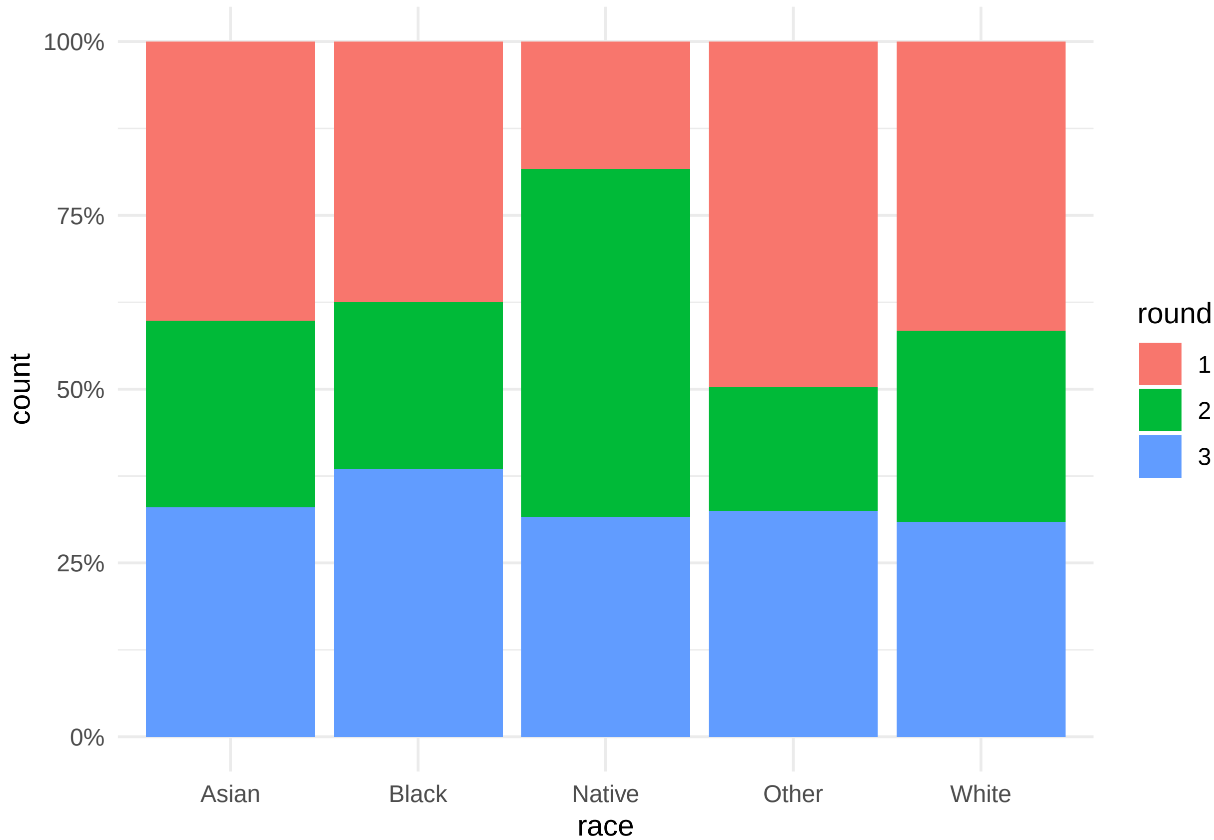
100%
75%
50%
25%
0%
count
Asian
Black
Native
Other
White
race
edu
Associates Degree
College Degree
Graduate Degree
High School
Less than High School
Some College
Some High School

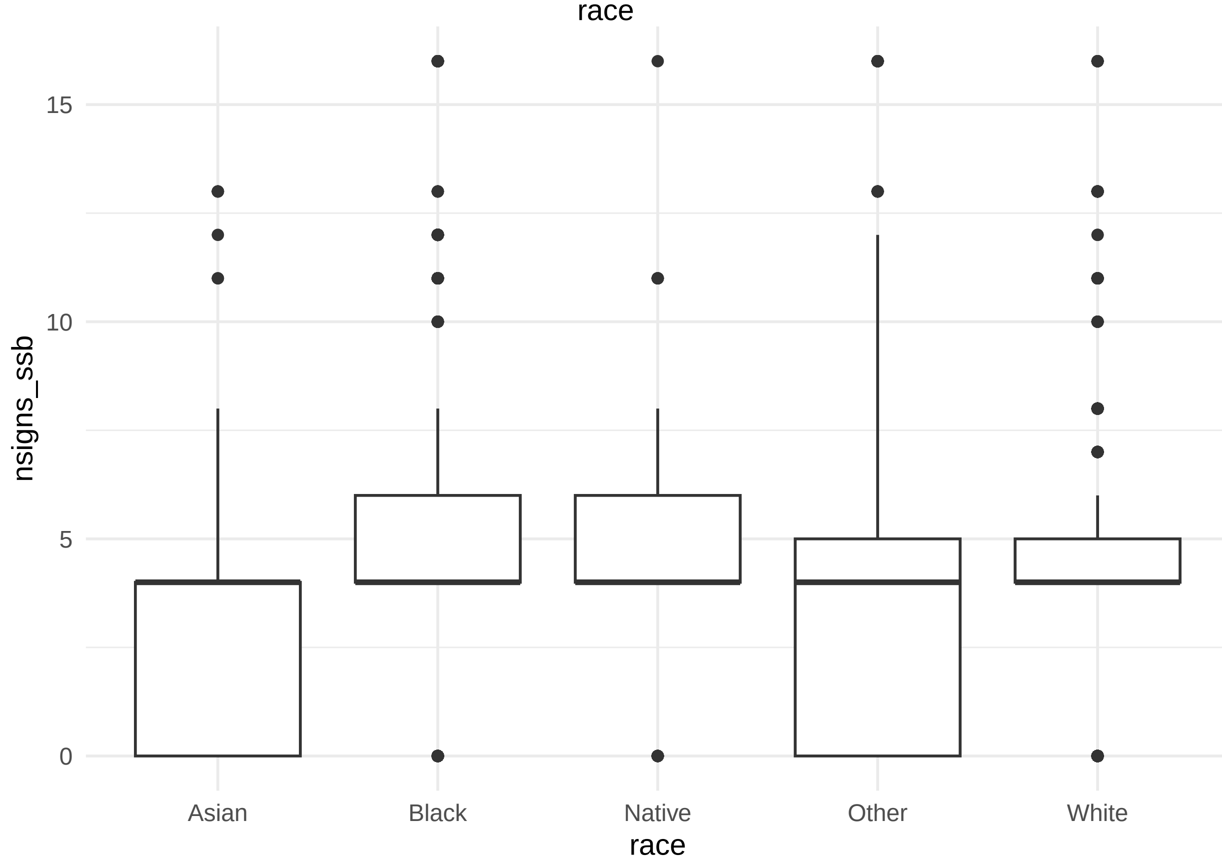


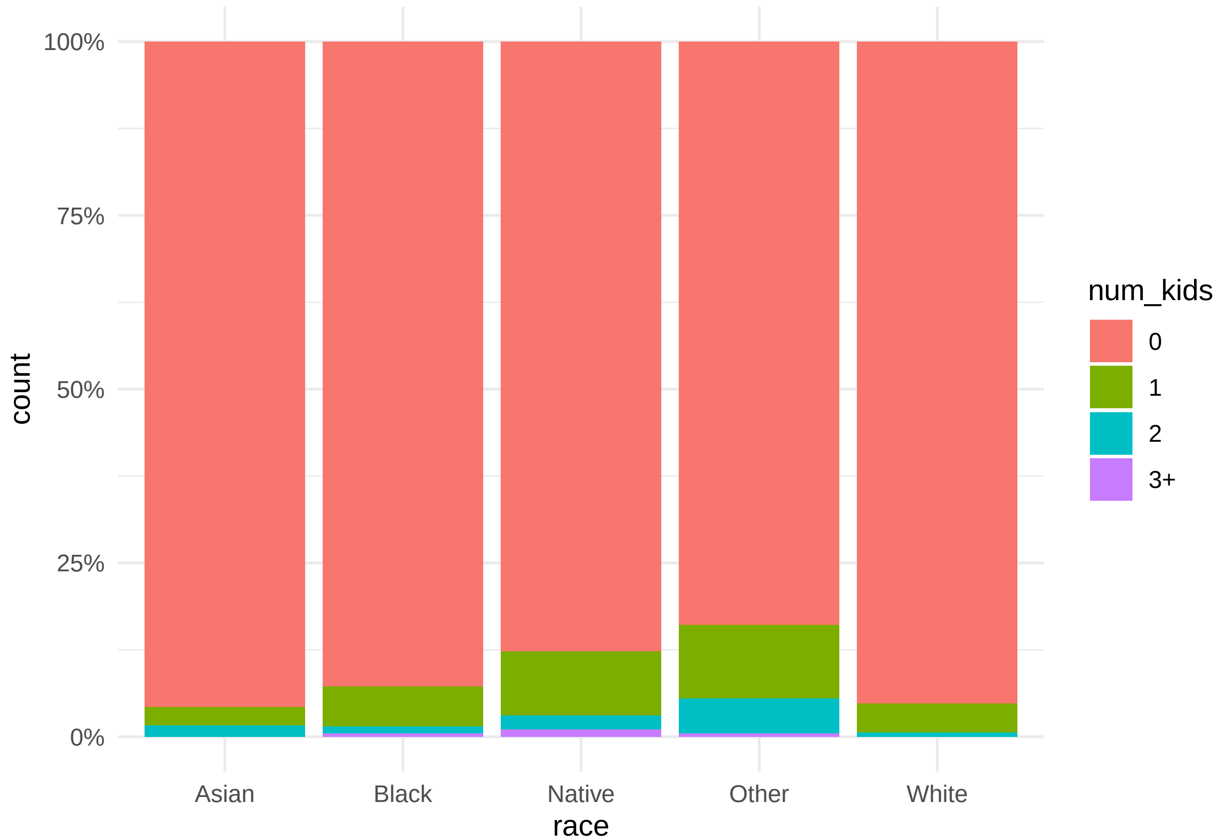
city
New Jersey
New York
count
100%
75%
50%
25%
0%
Asian
Black
Native
Other
White
race

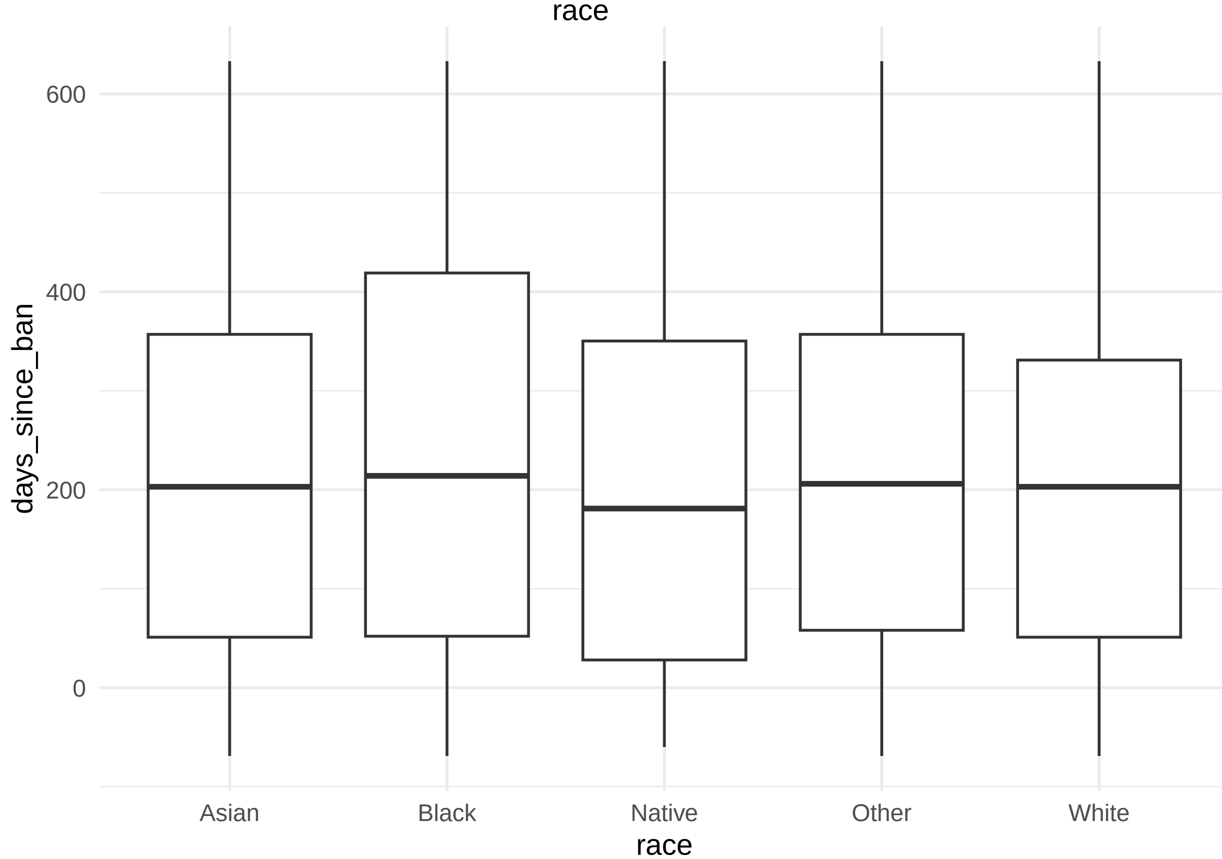


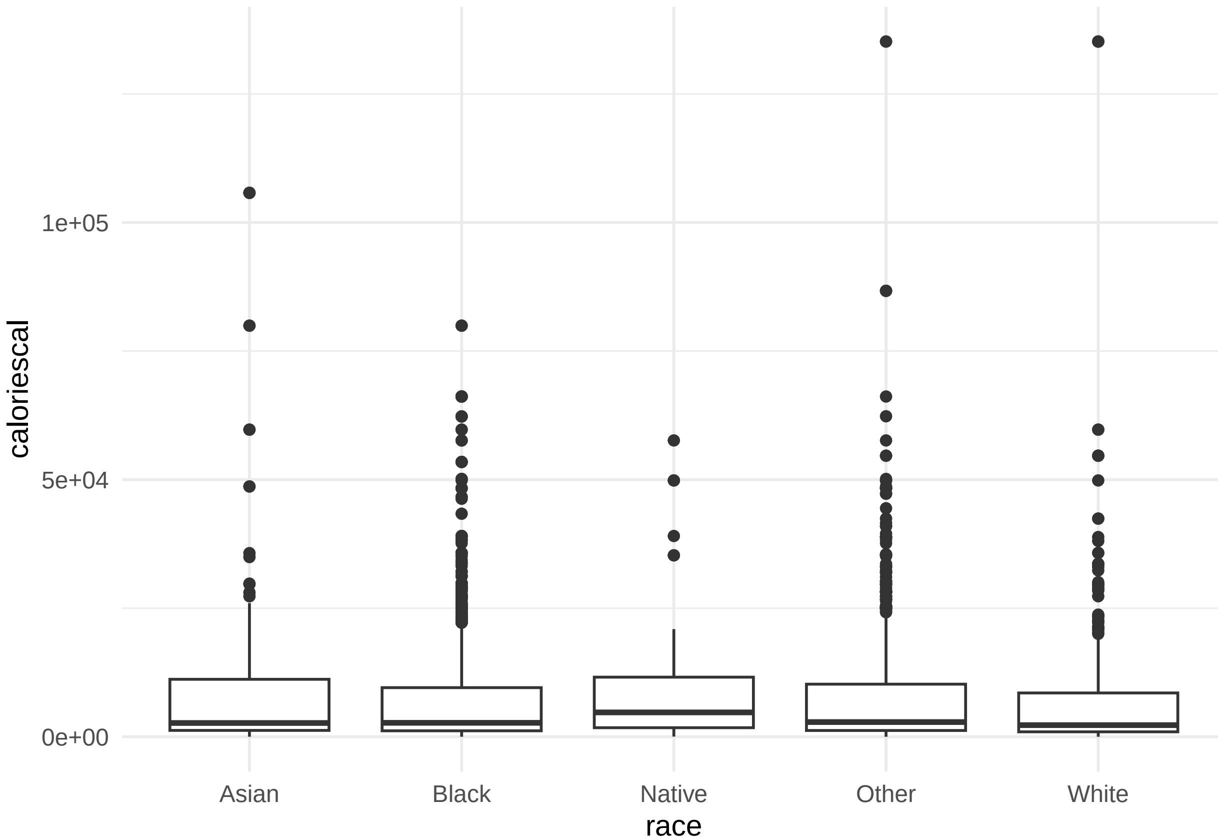
caff
3000
2000
1000
0
Asian
Black
Native
Other
White
race


100%
75%
50%
25%
0%
count
Asian
Black
Native
Other
White
race
round
1
2
3

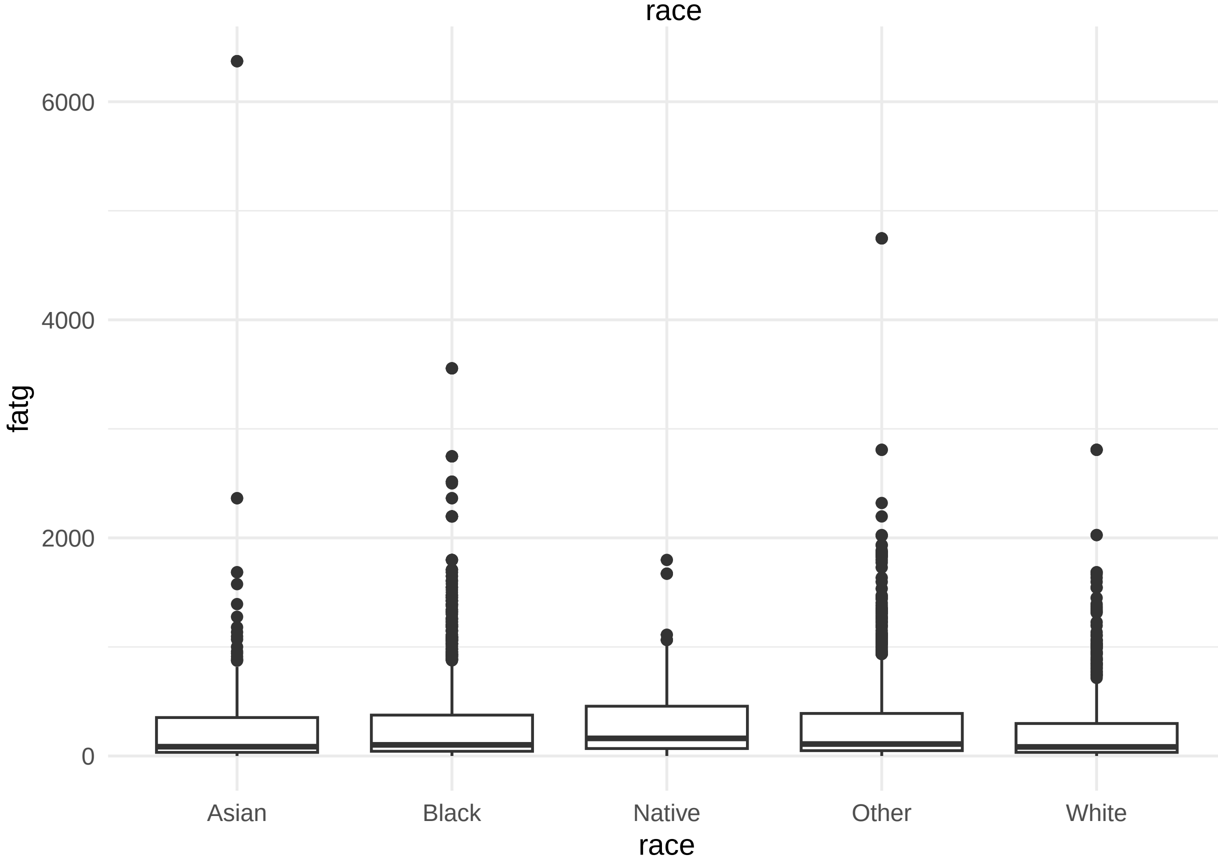

15
10
5
0
nsigns_ssb
Asian
Black
Native
Other
White
race


100%
75%
50%
25%
0%
count
num_kids
0
1
2
3+
Asian
Black
Native
Other
White
race

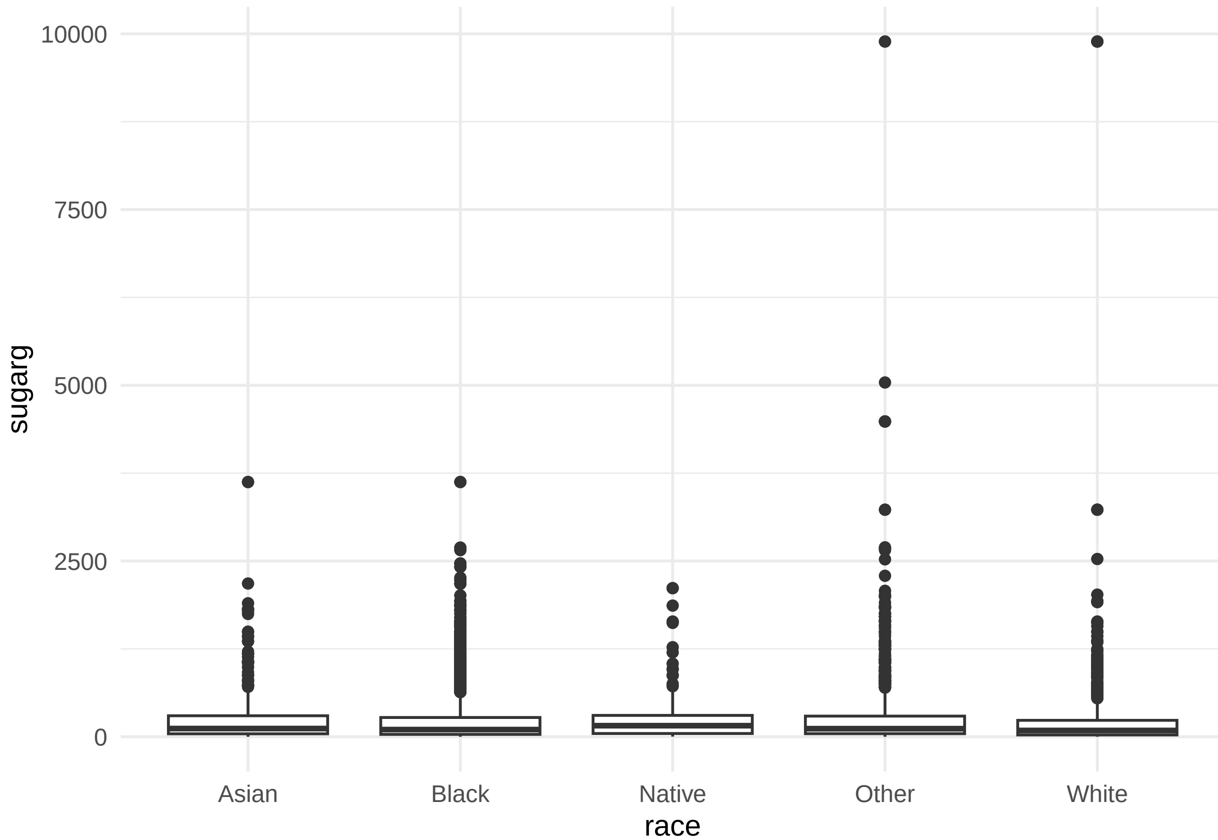

600
400
200
0
days_since_ban
Asian
Black
Native
Other
White
race

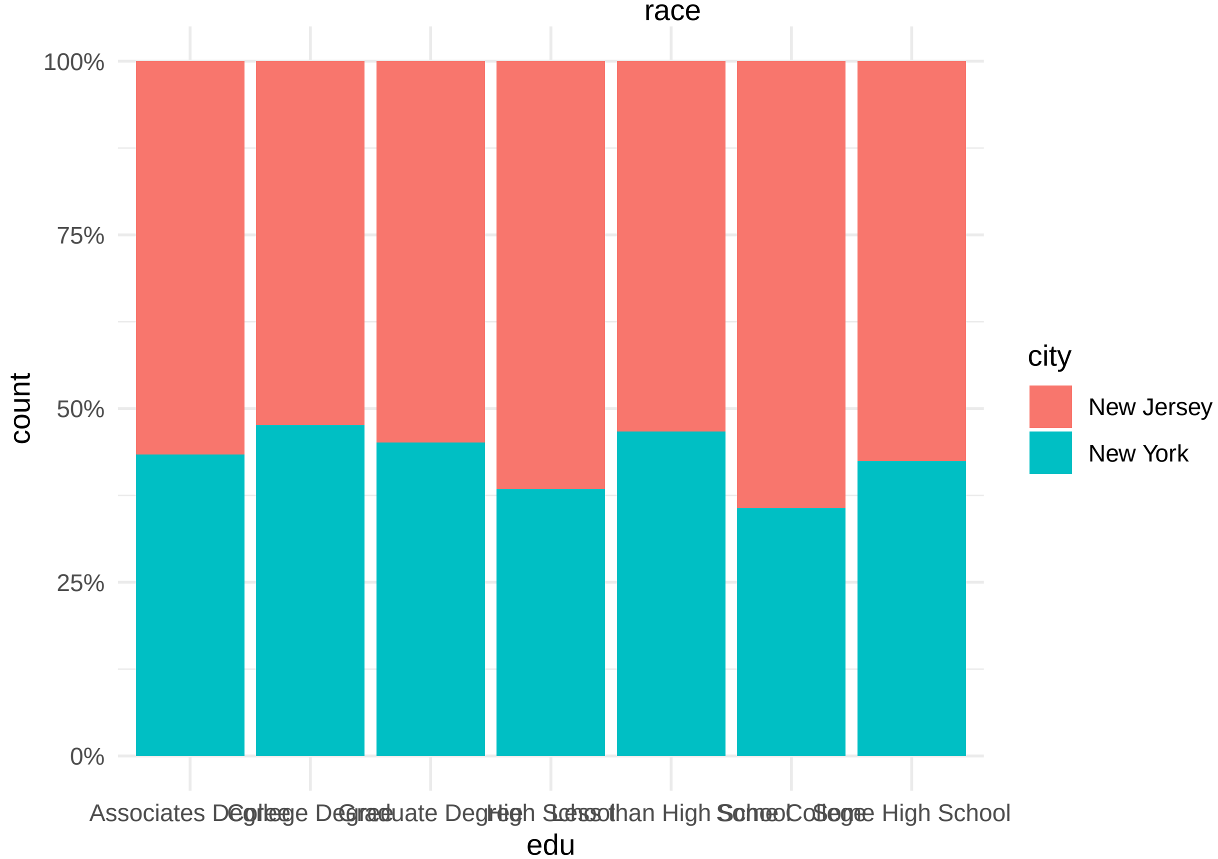


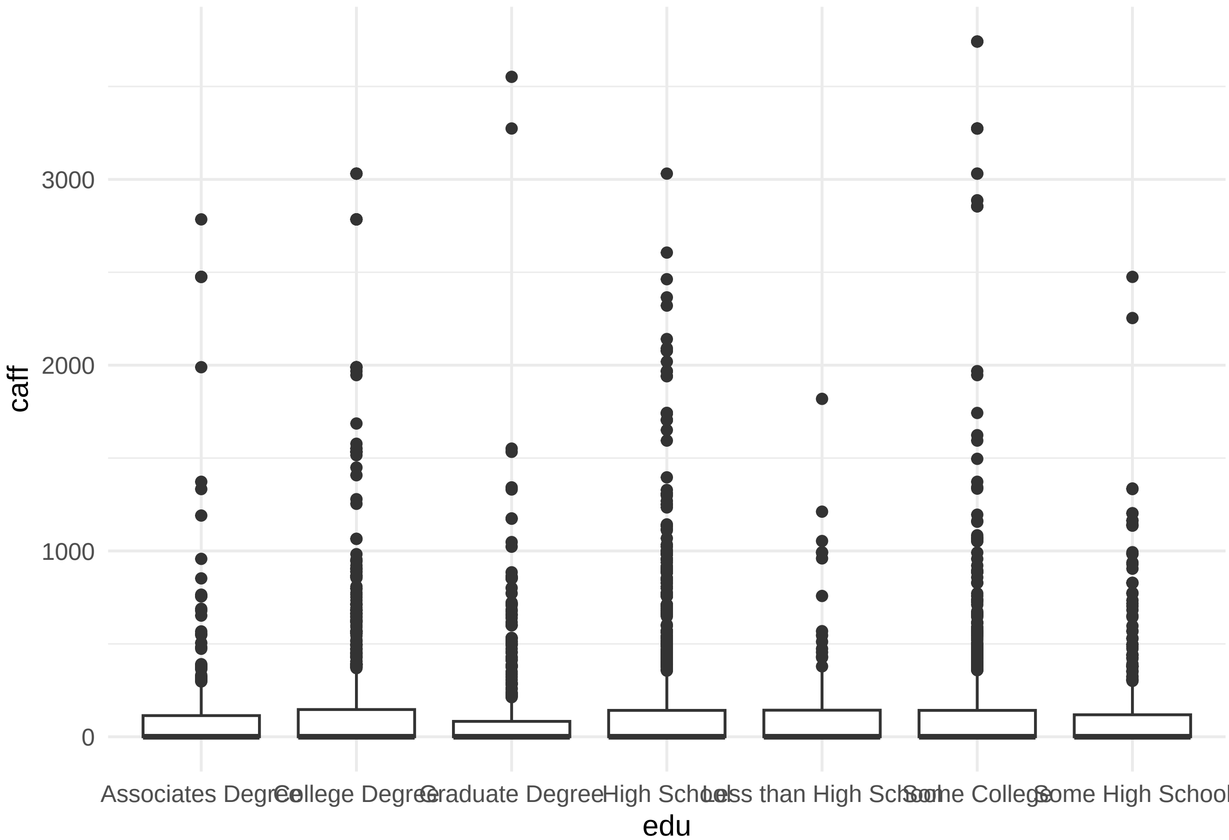
1e+05
5e+04
0e+00
caloriescal
Asian
Black
Native
Other
White
race

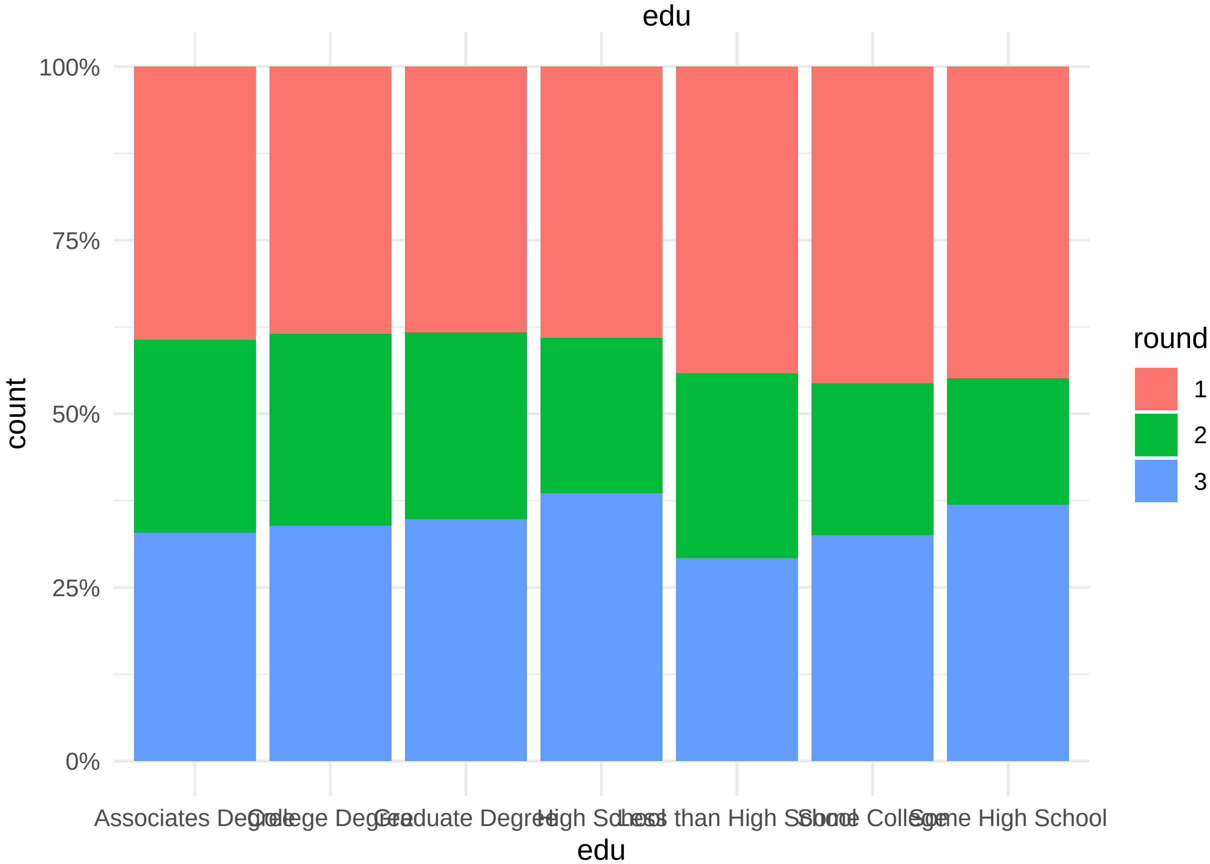


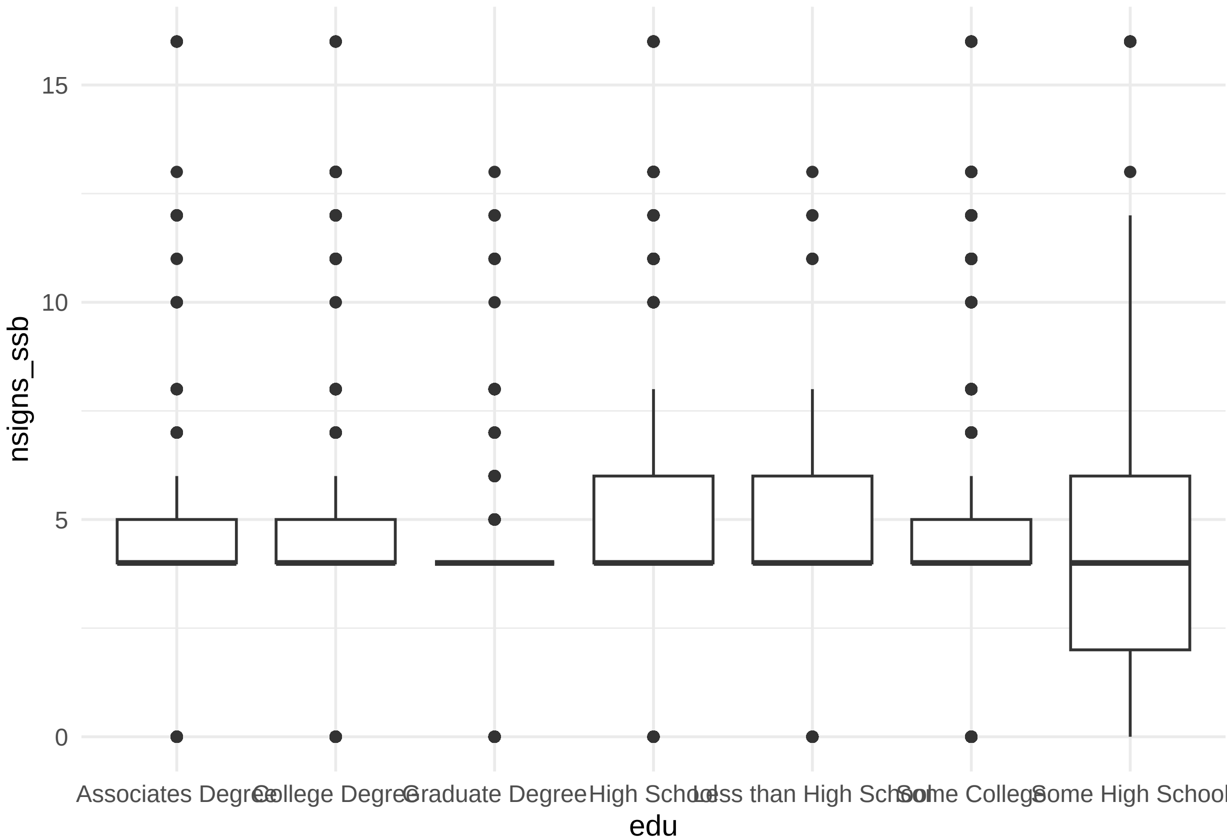
6000
4000
2000
0
fatg
Asian
Black
Native
Other
White
race

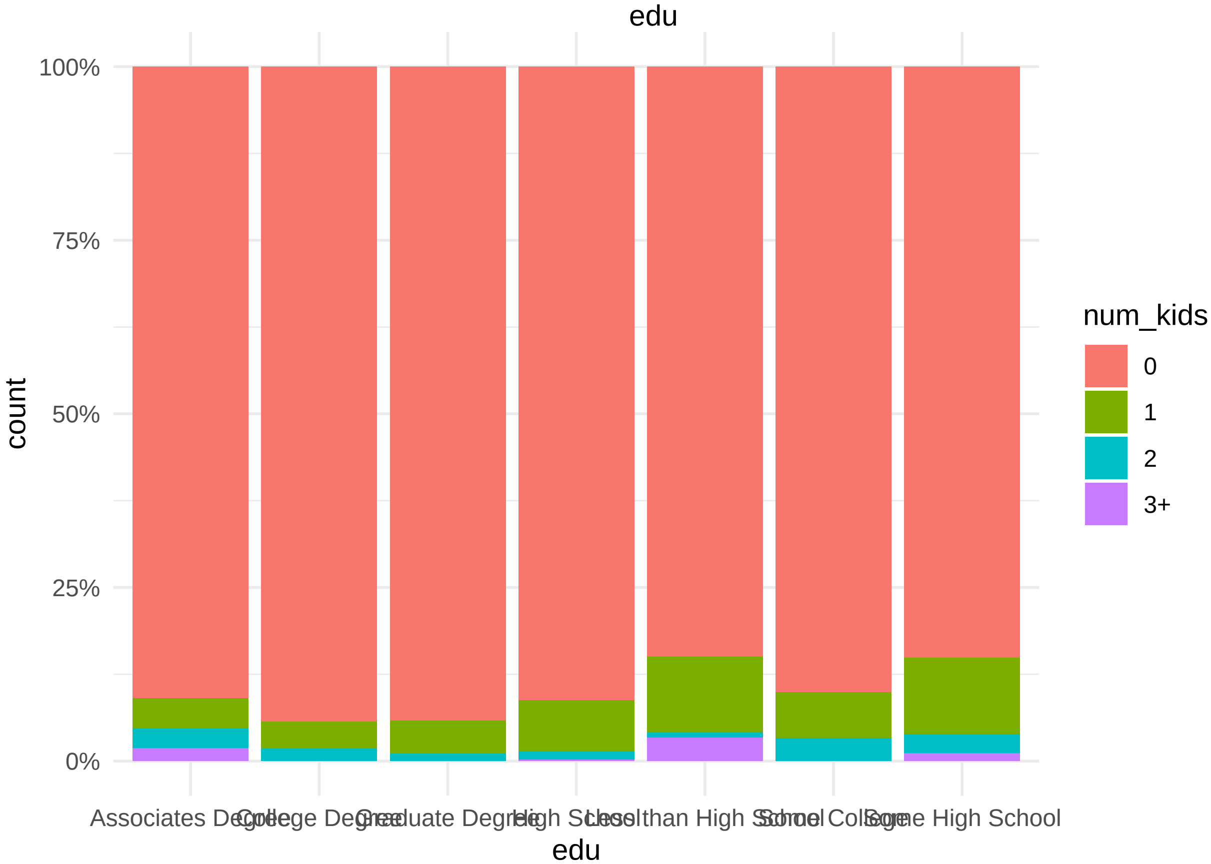

10000
7500
5000
2500
0
sugarg
Asian
Black
Native
Other
White
race

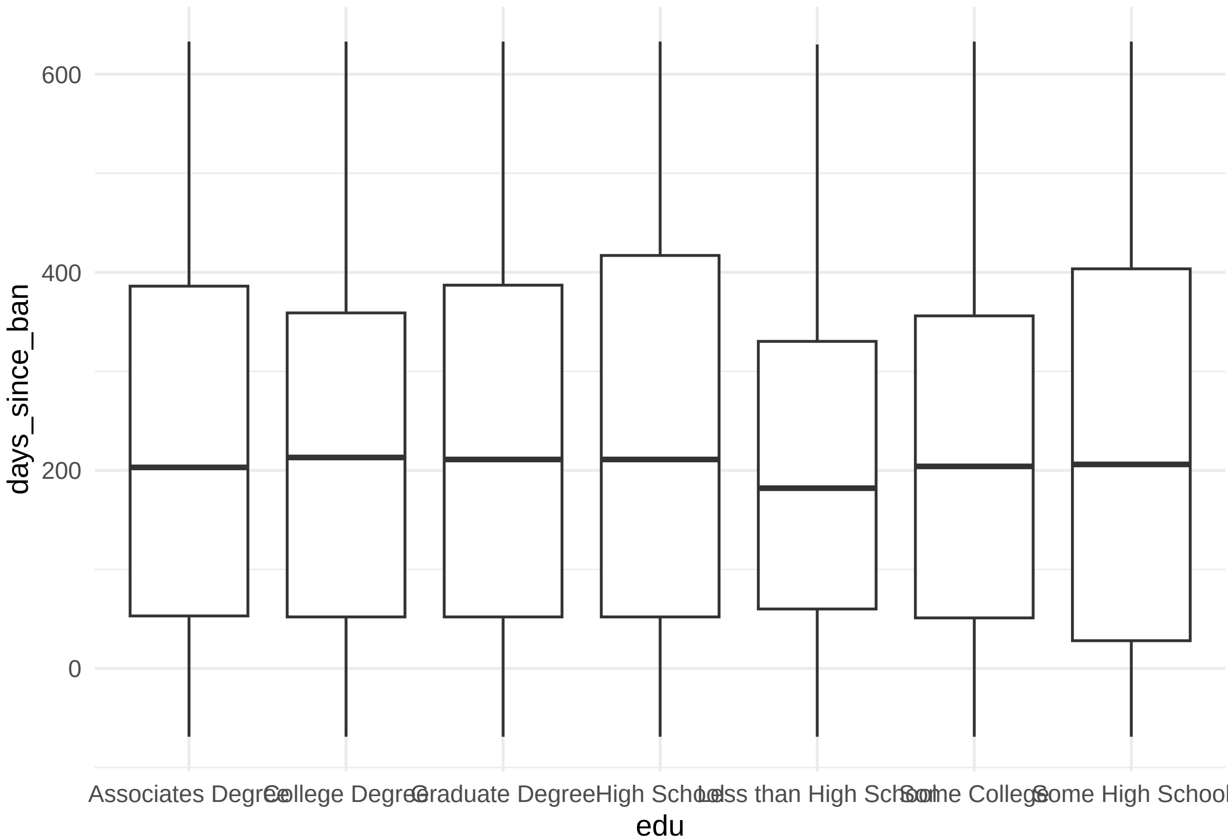

100%
75%
50%
25%
0%
count
Some High School
edu
city
New Jersey
New York


caff
3000
2000
1000
0
Associates Degree
College Degree
Graduate Degree
High School
Less than High School
Some College
Some High School
edu

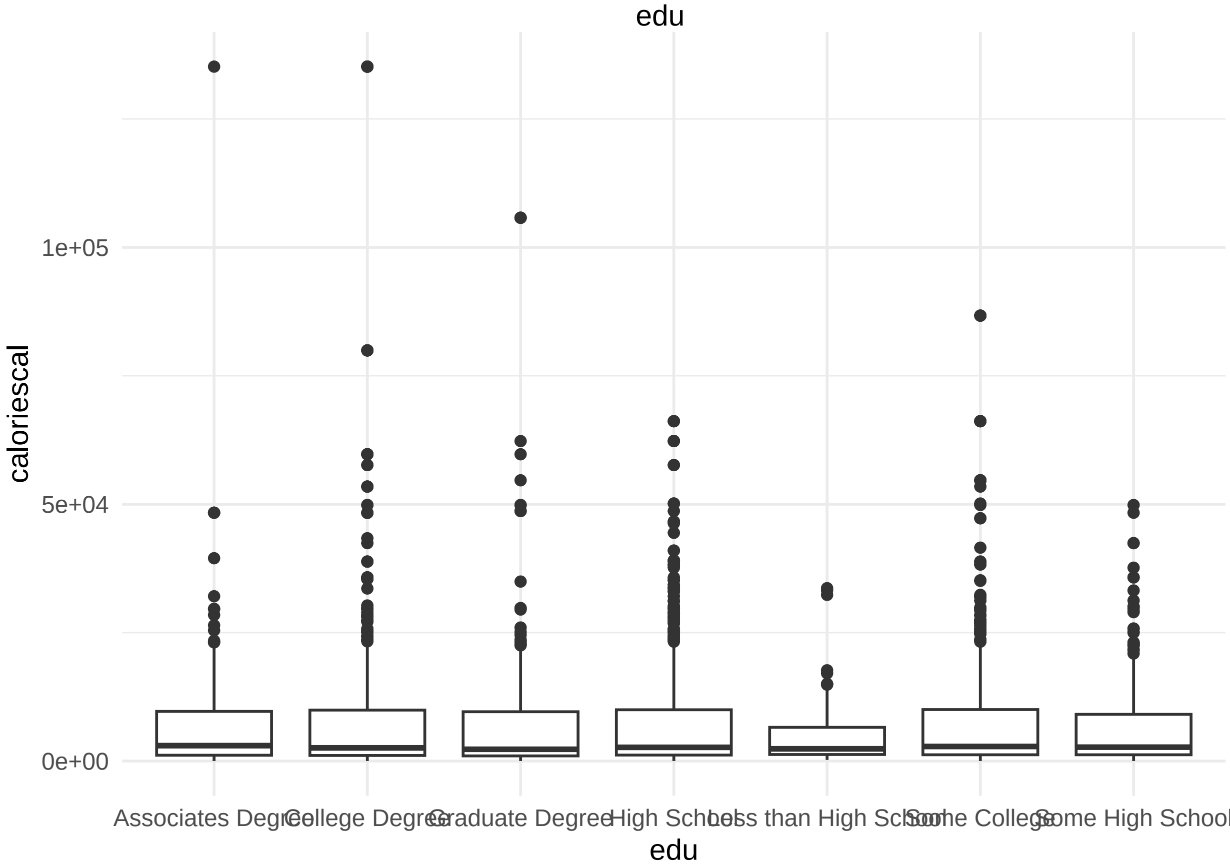

count
100%
75%
50%
25%
0%
Associates Degree
College Degree
Graduate Degree
High School
Less than High School
Some College
Some High School
edu
round
1
2
3


nsigns_ssb
0
5
10
15
Associates Degree
College Degree
Graduate Degree
High School
Less than High School
Some College
Some High School
edu

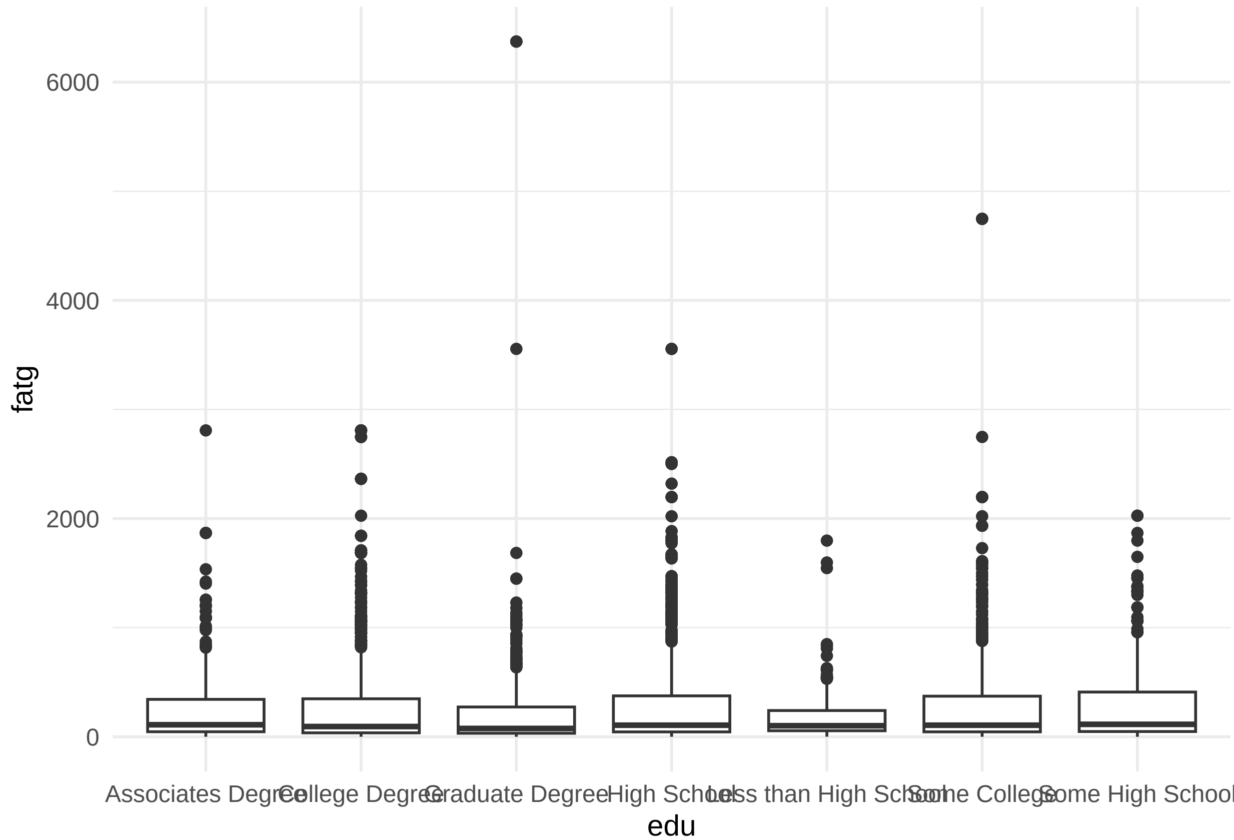

count
0%
25%
50%
75%
100%
edu
num_kids
0
1
2
3+

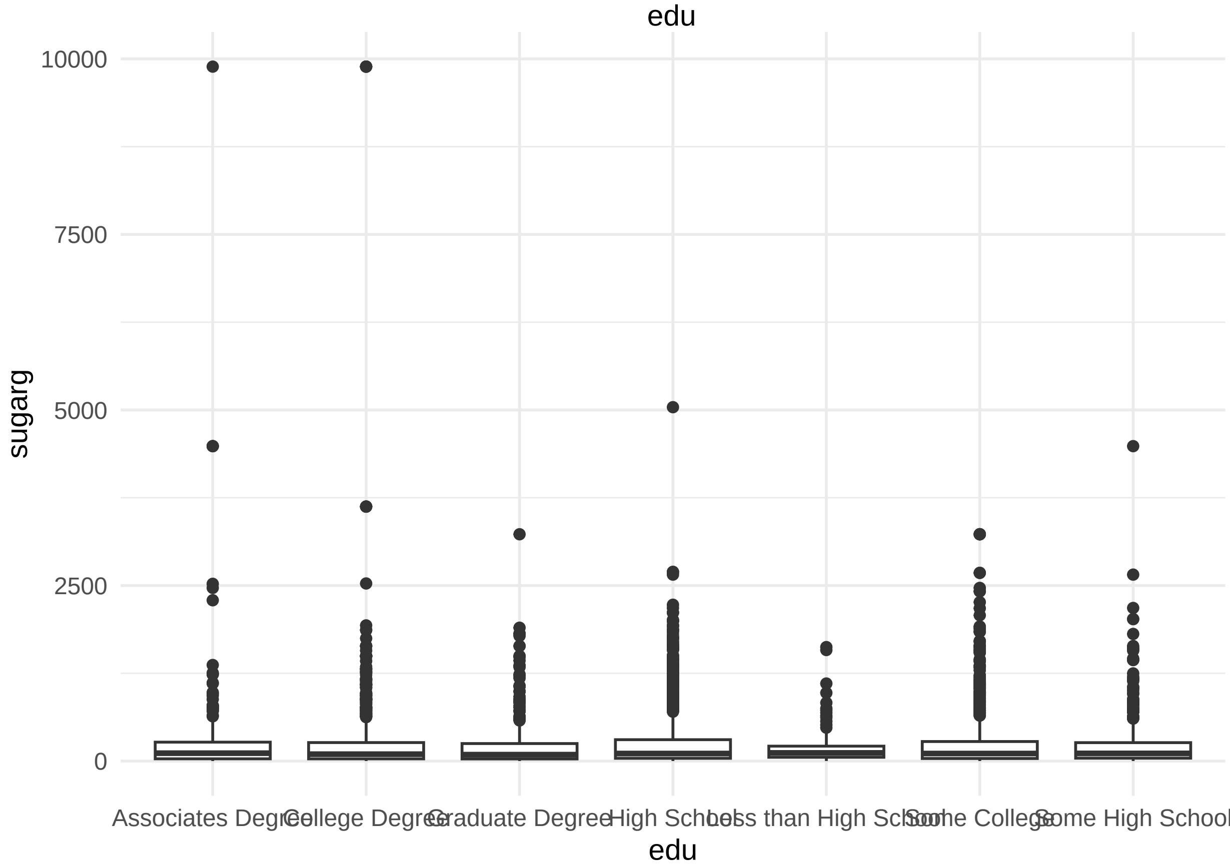


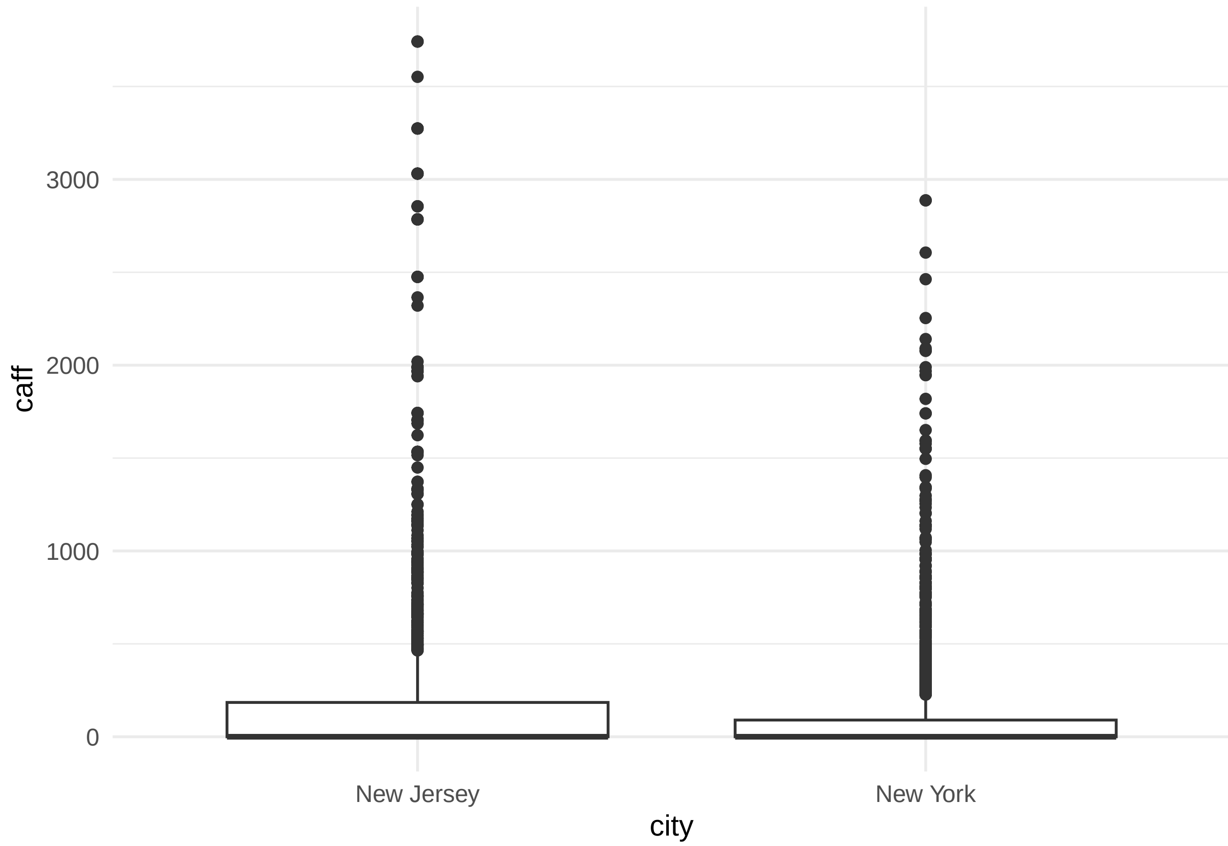
600
400
200
0
days_since_ban
Associates Degree
College Degree
Graduate Degree
High School
Less than High School
Some College
Some High School
edu

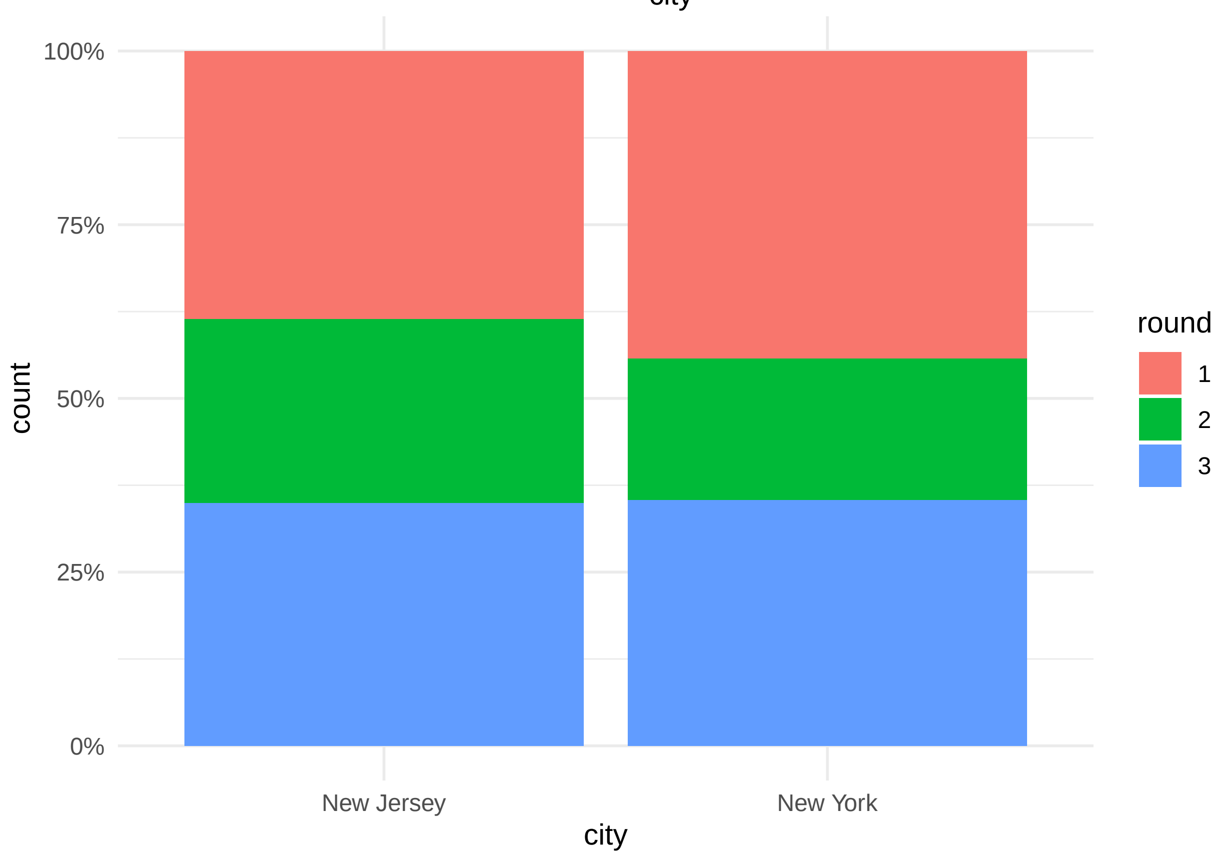


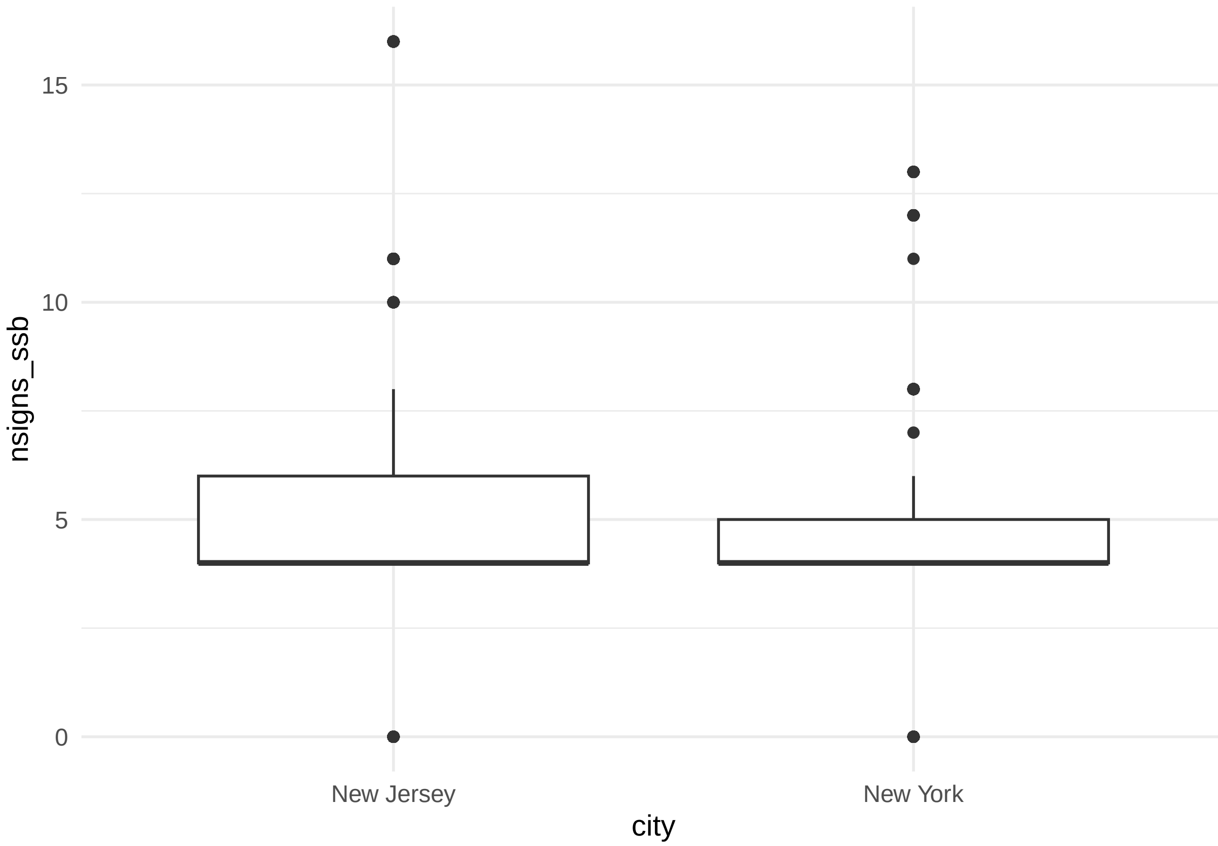
1e+05
5e+04
0e+00
caloriescal
Associates Degree
College Degree
Graduate Degree
High School
Less than High School
Some College
Some High School
edu

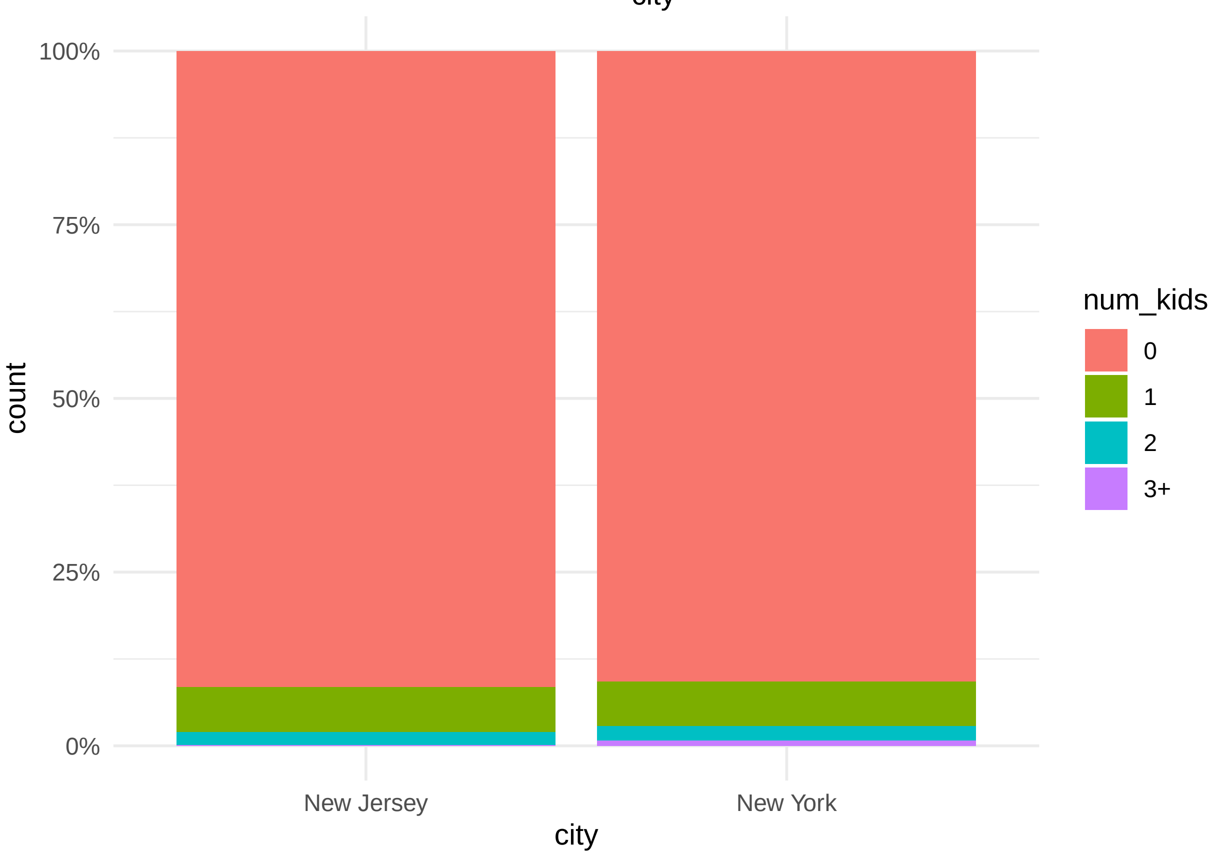

6000
4000
2000
0
fatg
Associates Degree
College Degree
Graduate Degree
High School
Less than High School
Some College
Some High School
edu


10000
7500
5000
2500
0
sugarg
Associates Degree
College Degree
Graduate Degree
High School
Less than High School
Some College
Some High School
edu

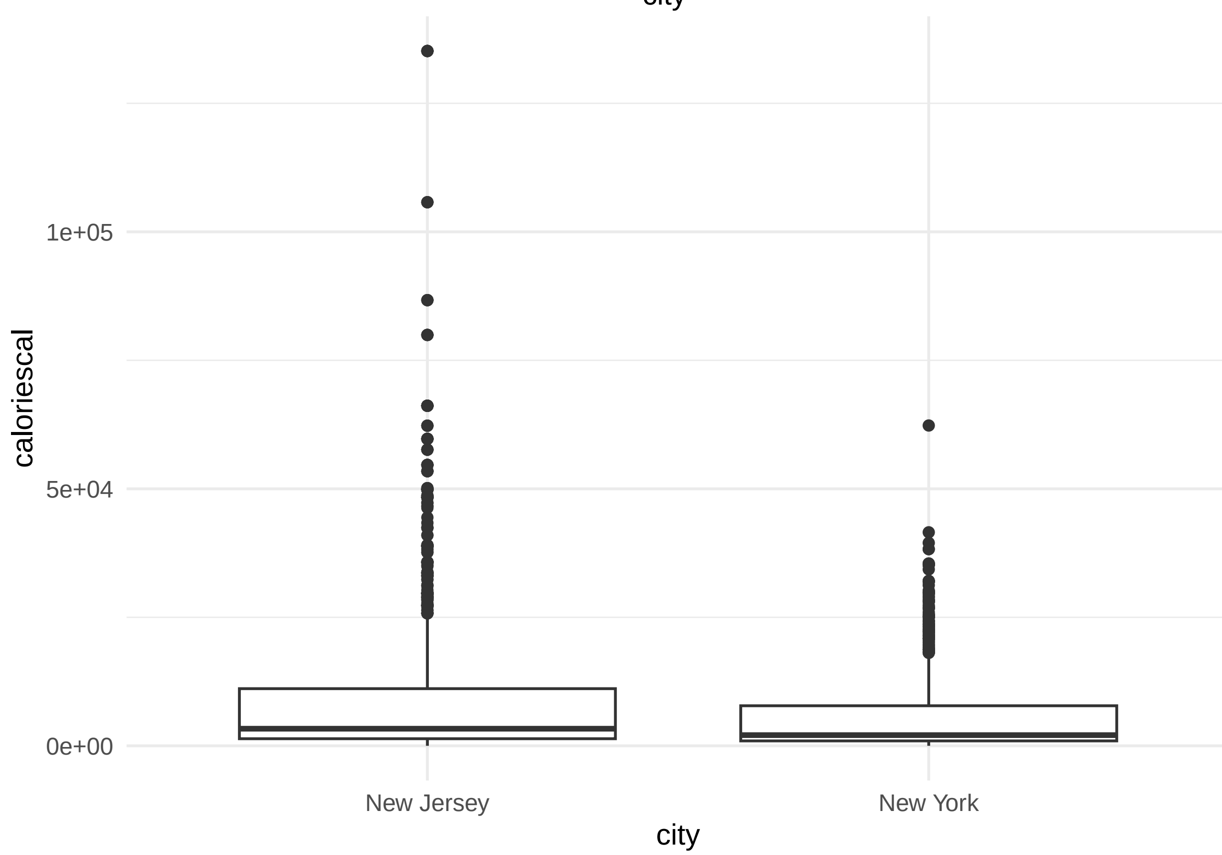

caff
3000
2000
1000
0
New Jersey
New York
city

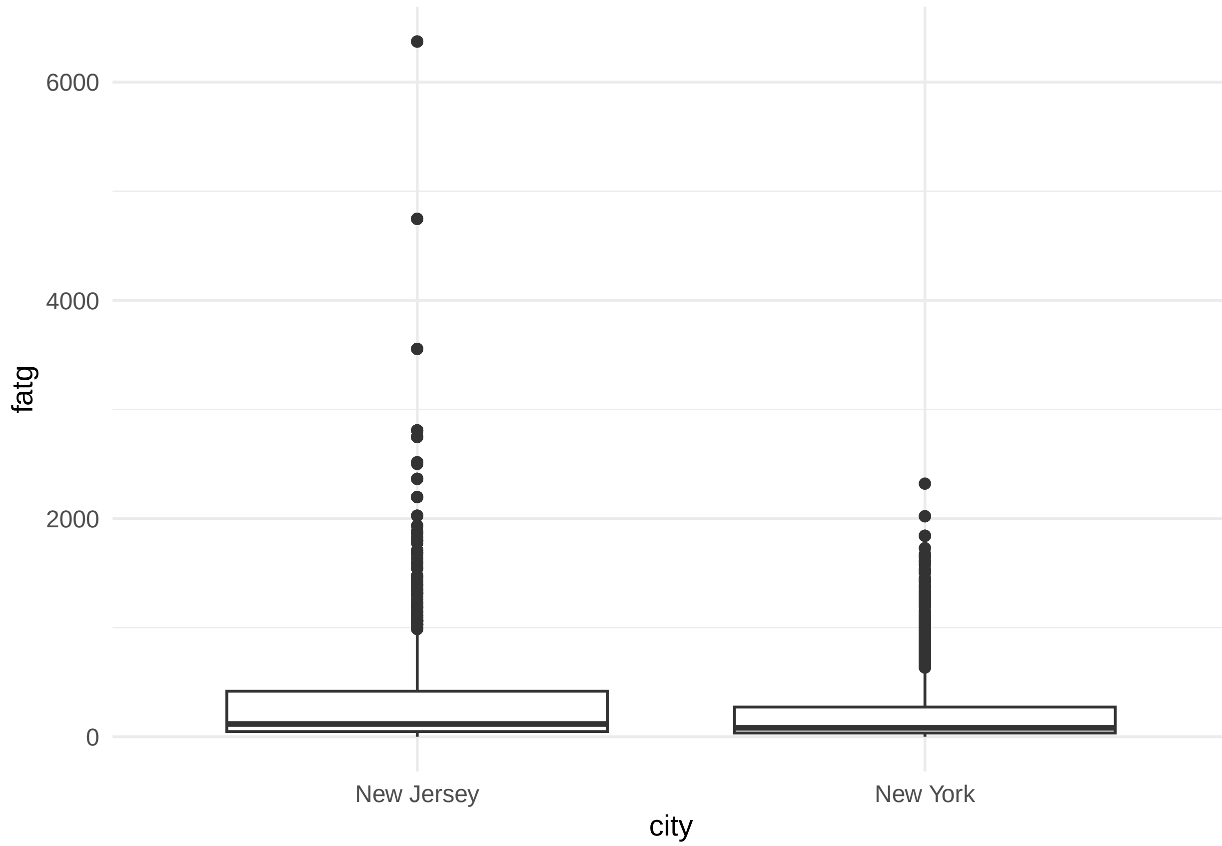

count
100%
75%
50%
25%
0%
New Jersey
New York
city
round
1
2
3

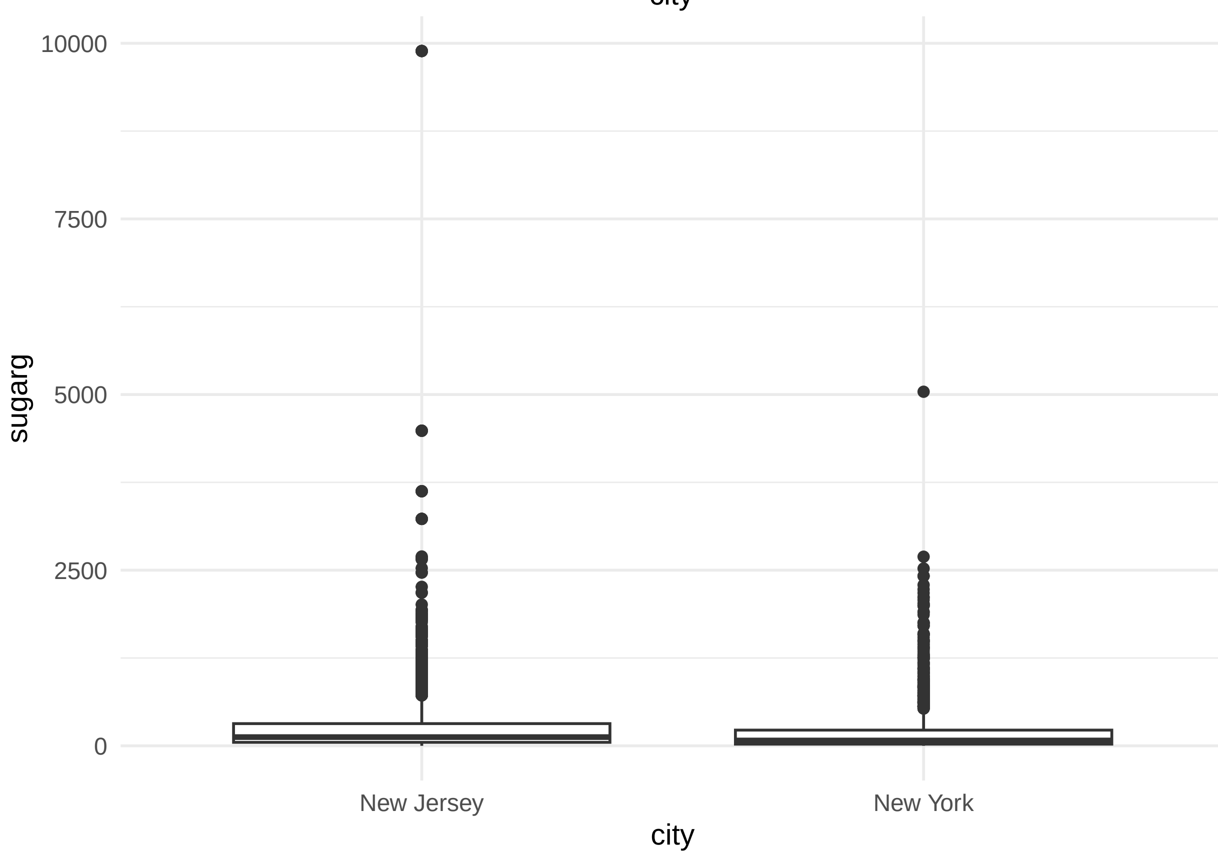

15
10
5
0
nsigns_ssb
New Jersey
New York
city

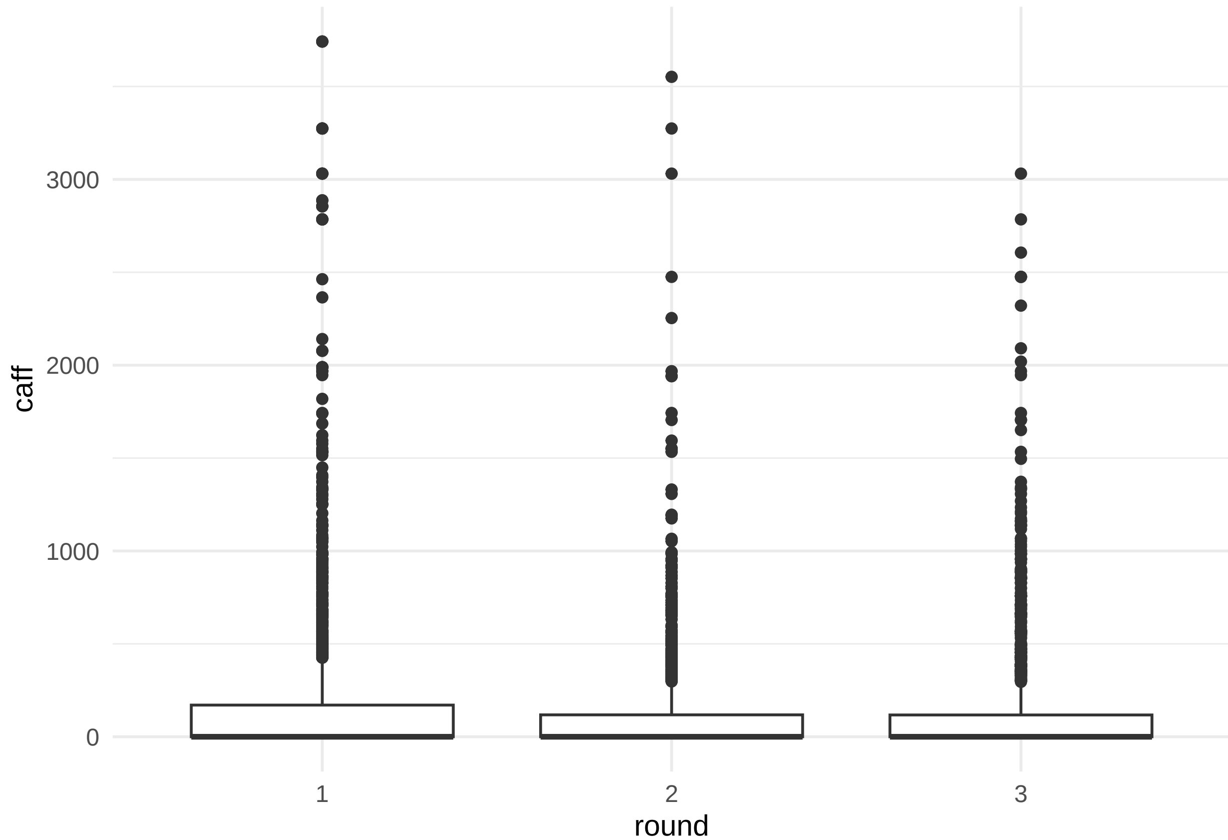

100%
75%
50%
25%
0%
count
New Jersey
New York
city
num_kids
0
1
2
3+

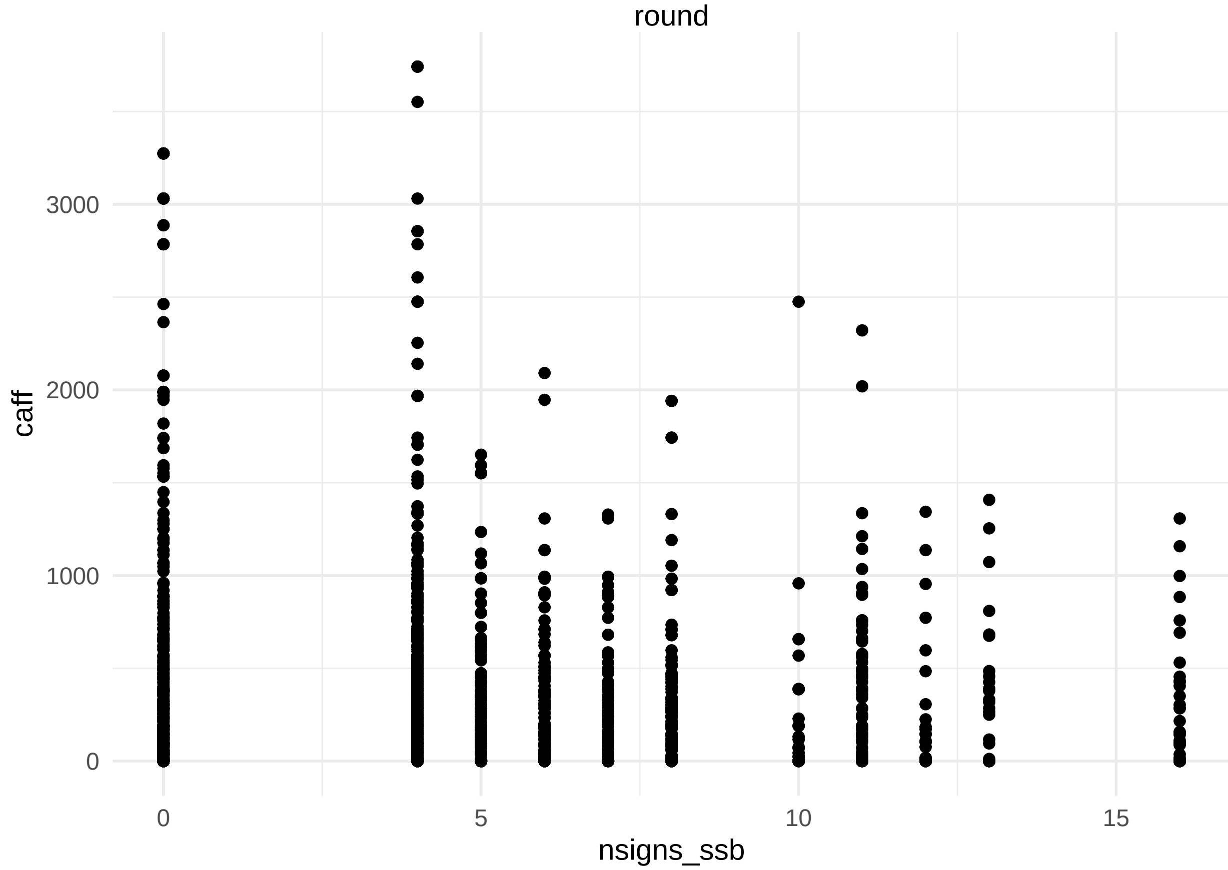

600
400
200
0
days_since_ban
New Jersey
New York
city

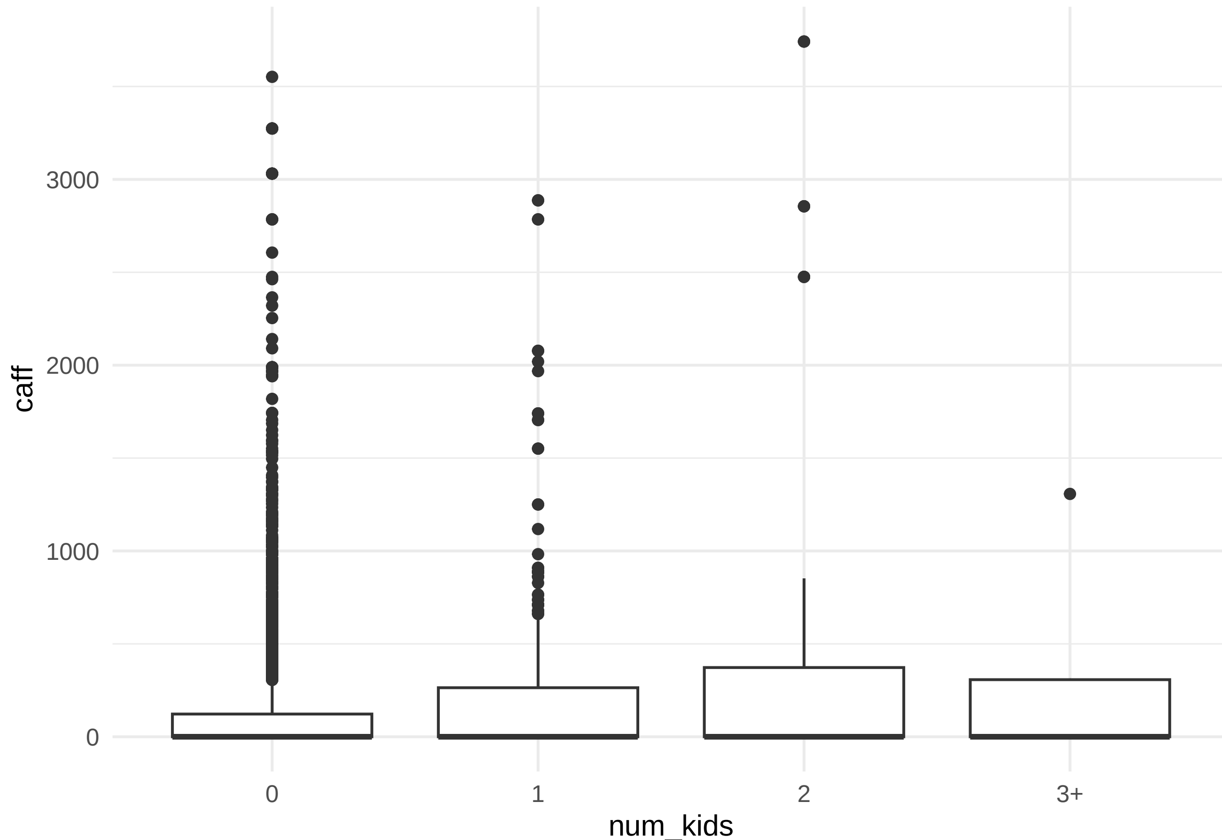

1e+05
5e+04
0e+00
caloriescal
New Jersey
New York
city

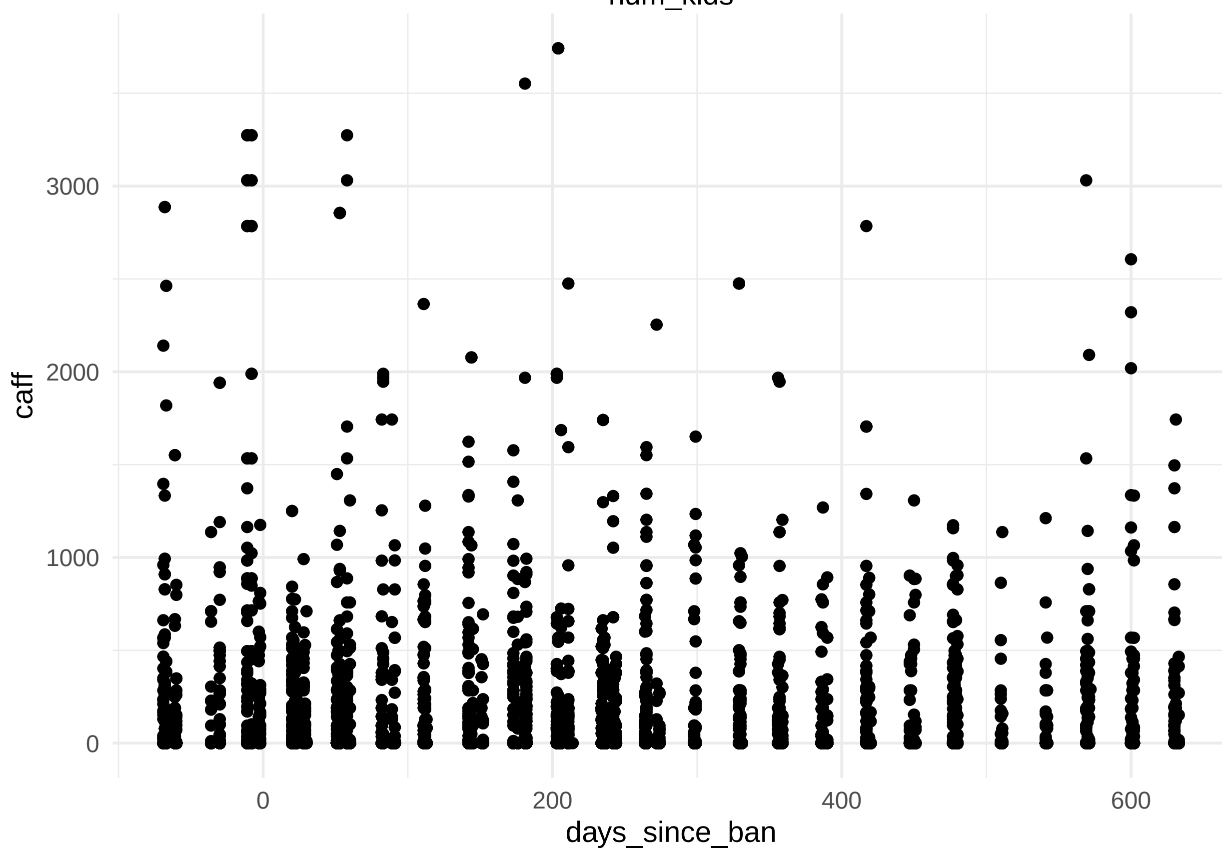

6000
4000
fatg
2000
0
New Jersey
New York
city

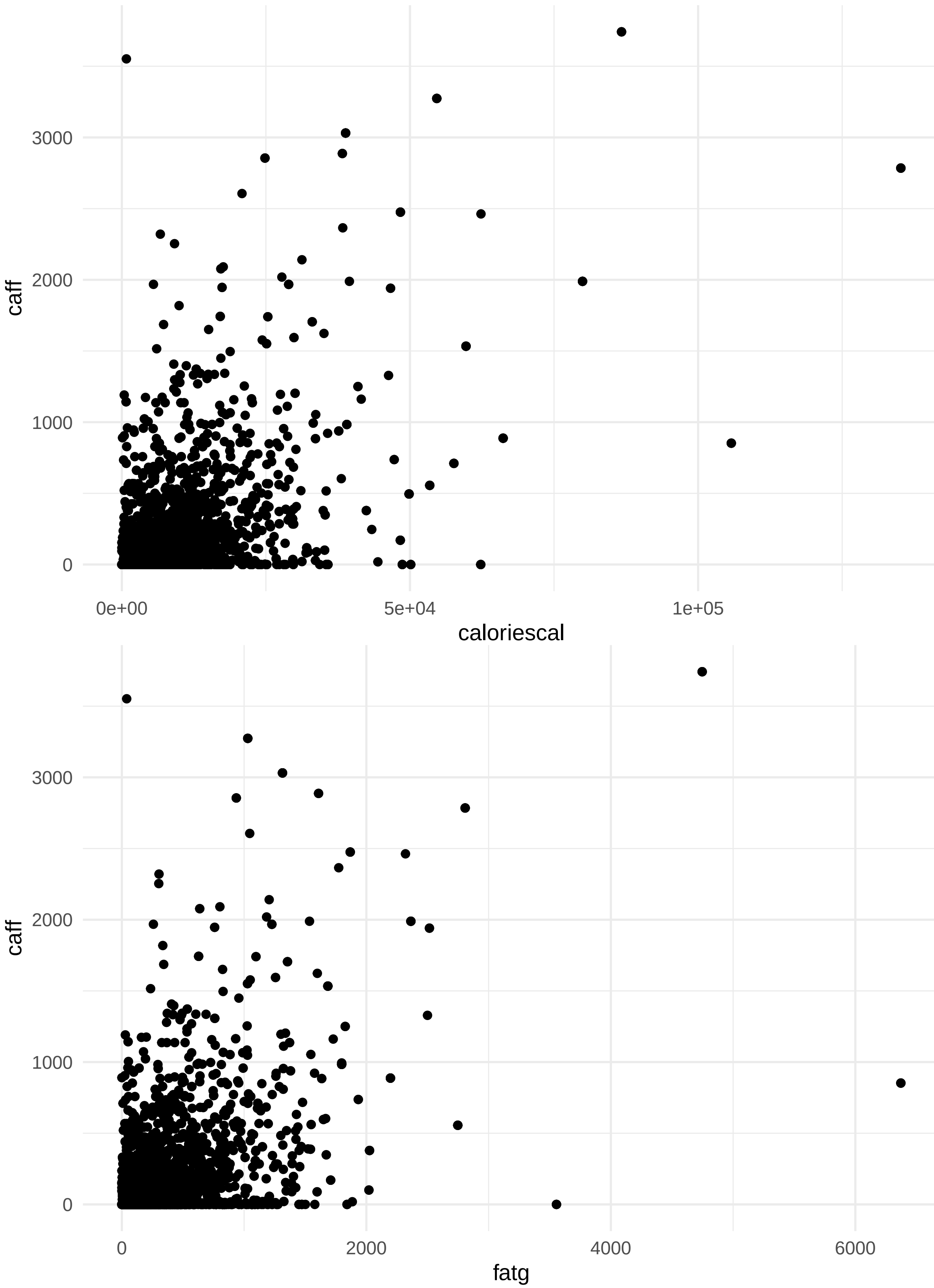

10000
7500
sugarg
5000
2500
0
New Jersey
New York
city

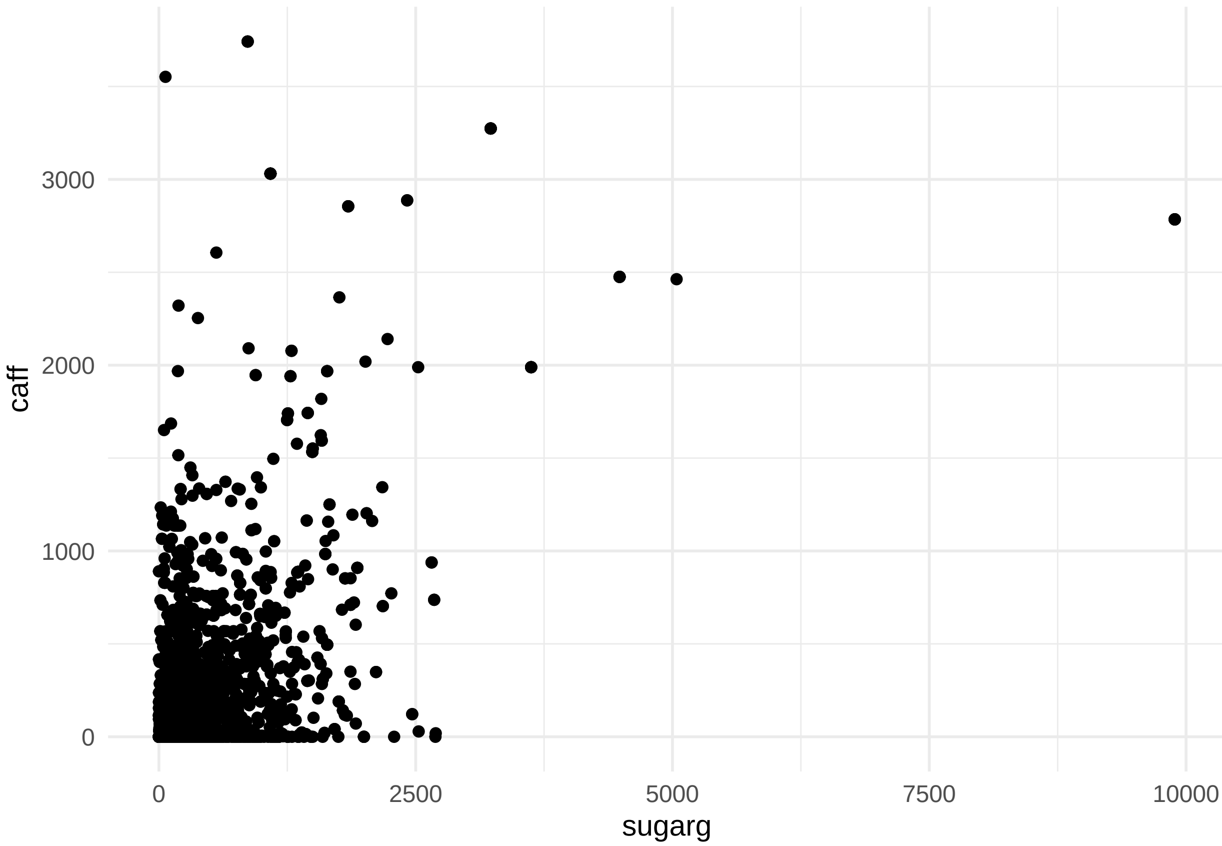

caff
3000
2000
1000
0
1
2
3
round

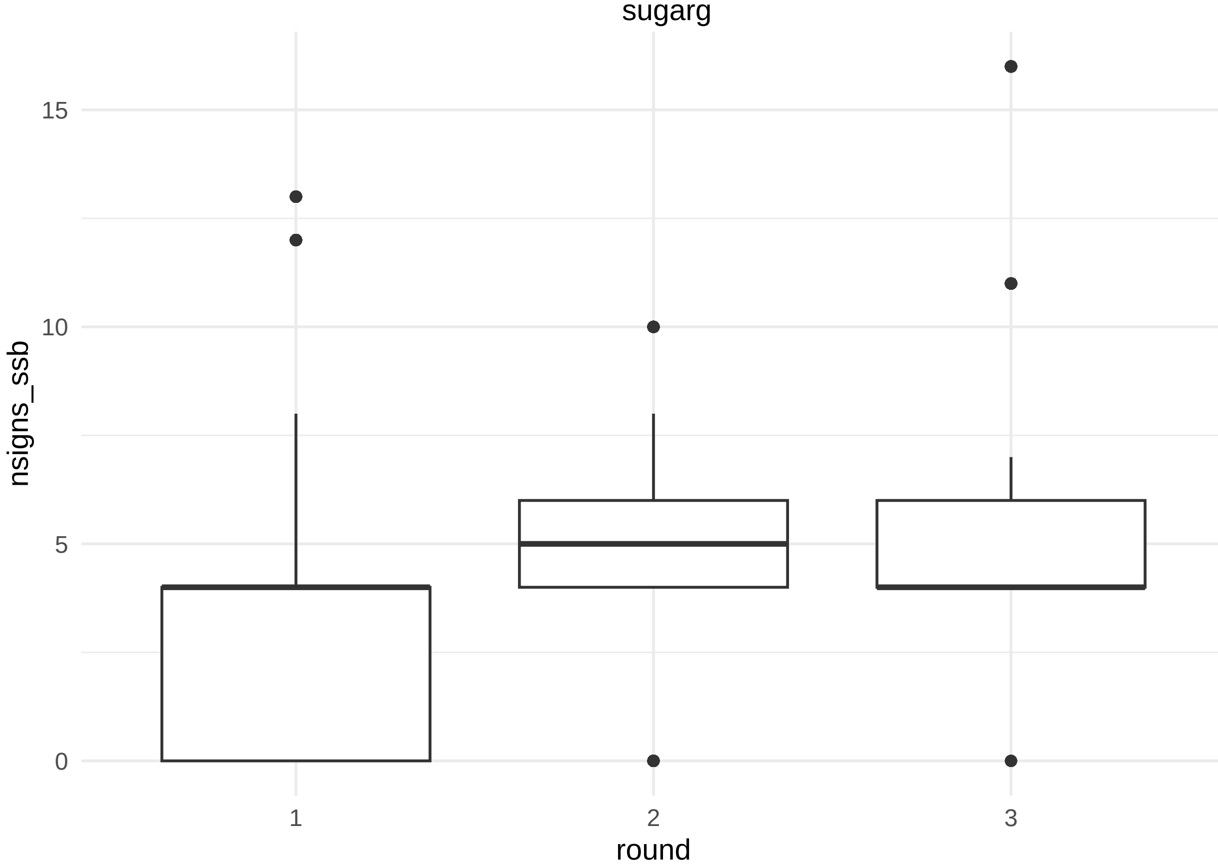

caff
3000
2000
1000
0
0
5
10
15
nsigns_ssb

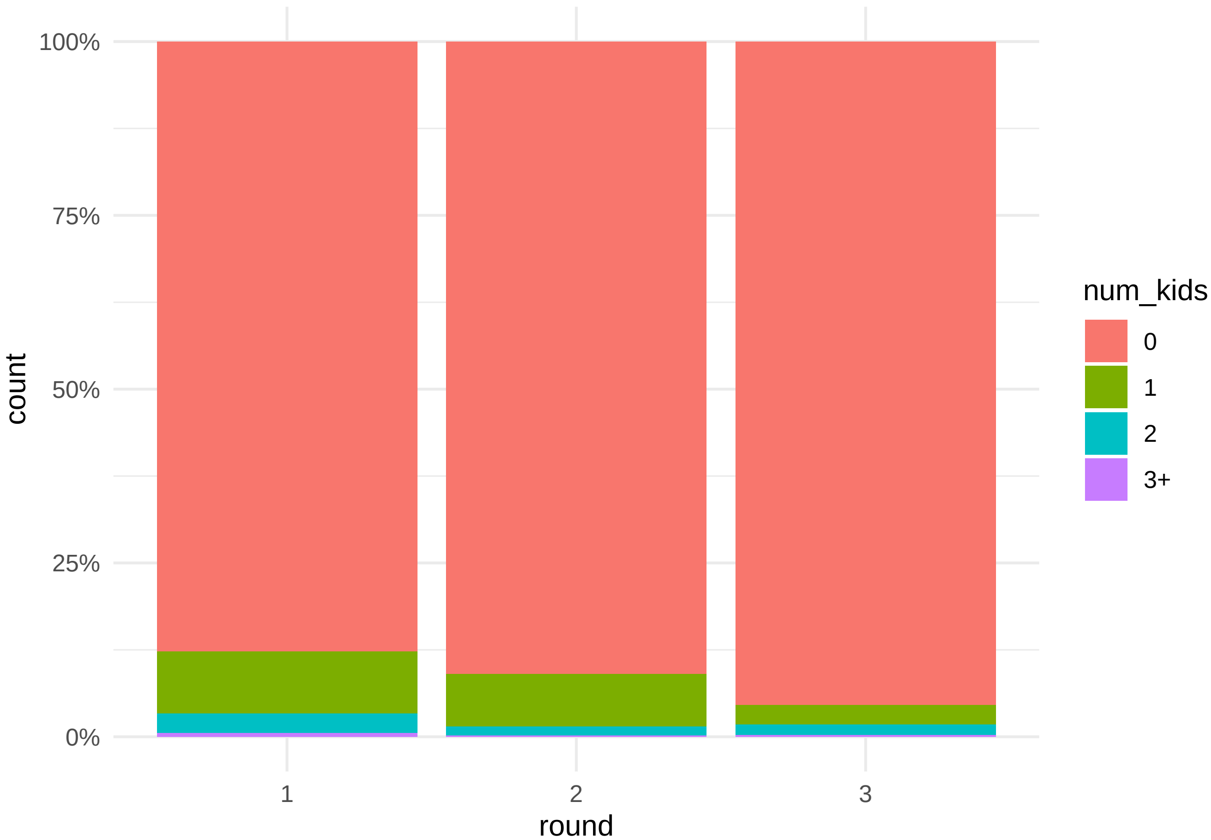

caff
3000
2000
1000
0
0
1
2
3+
num_kids

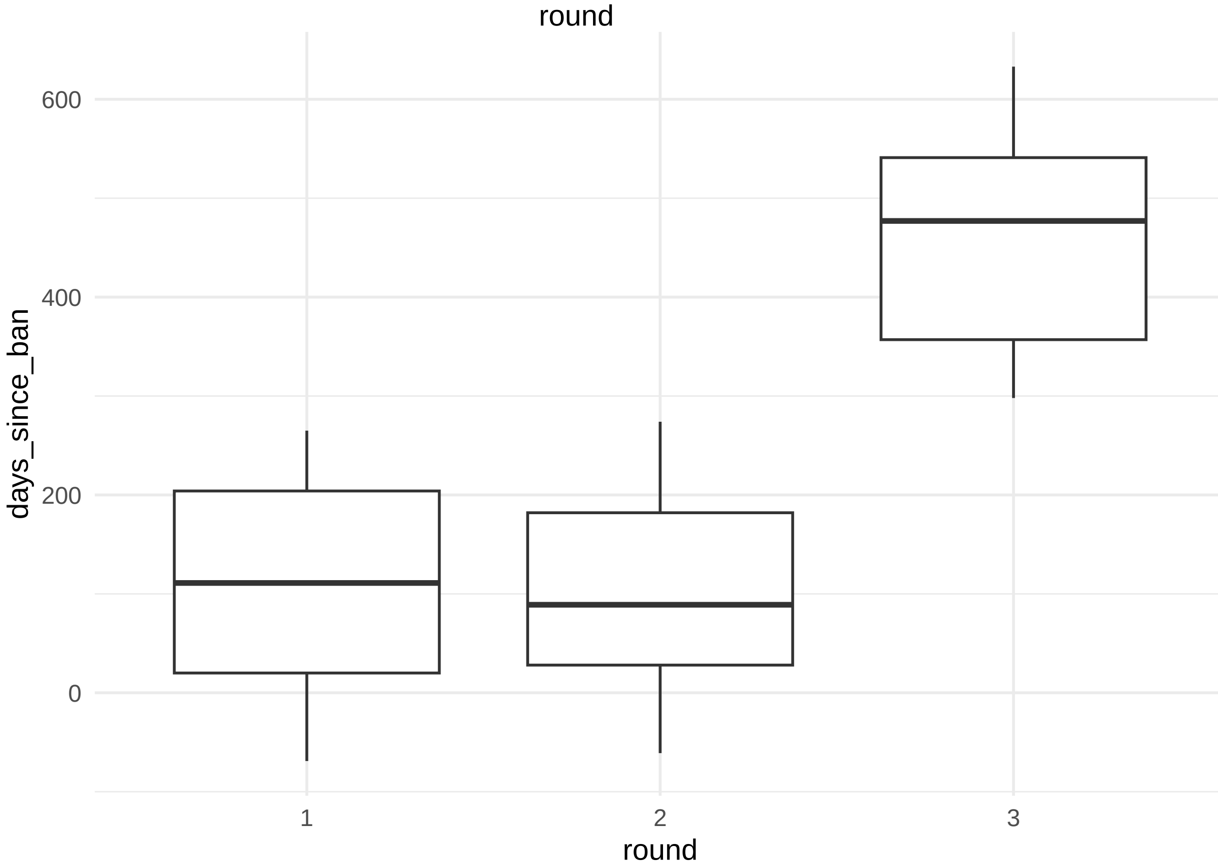

caff
3000
2000
1000
0
0
200
400
600
days_since_ban

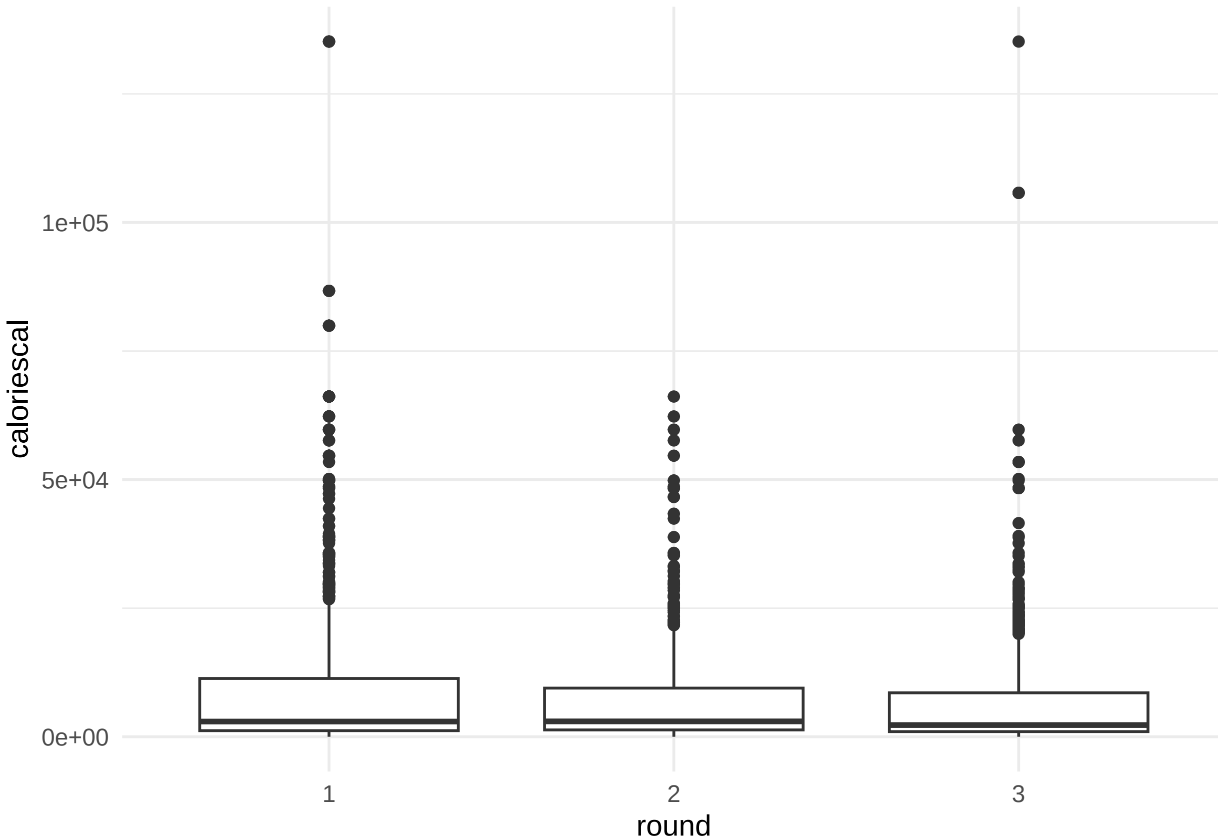

caff
3000
2000
1000
0
0e+00
5e+04
1e+05
caloriescal
caff
3000
2000
1000
0
0
2000
4000
6000
fatg

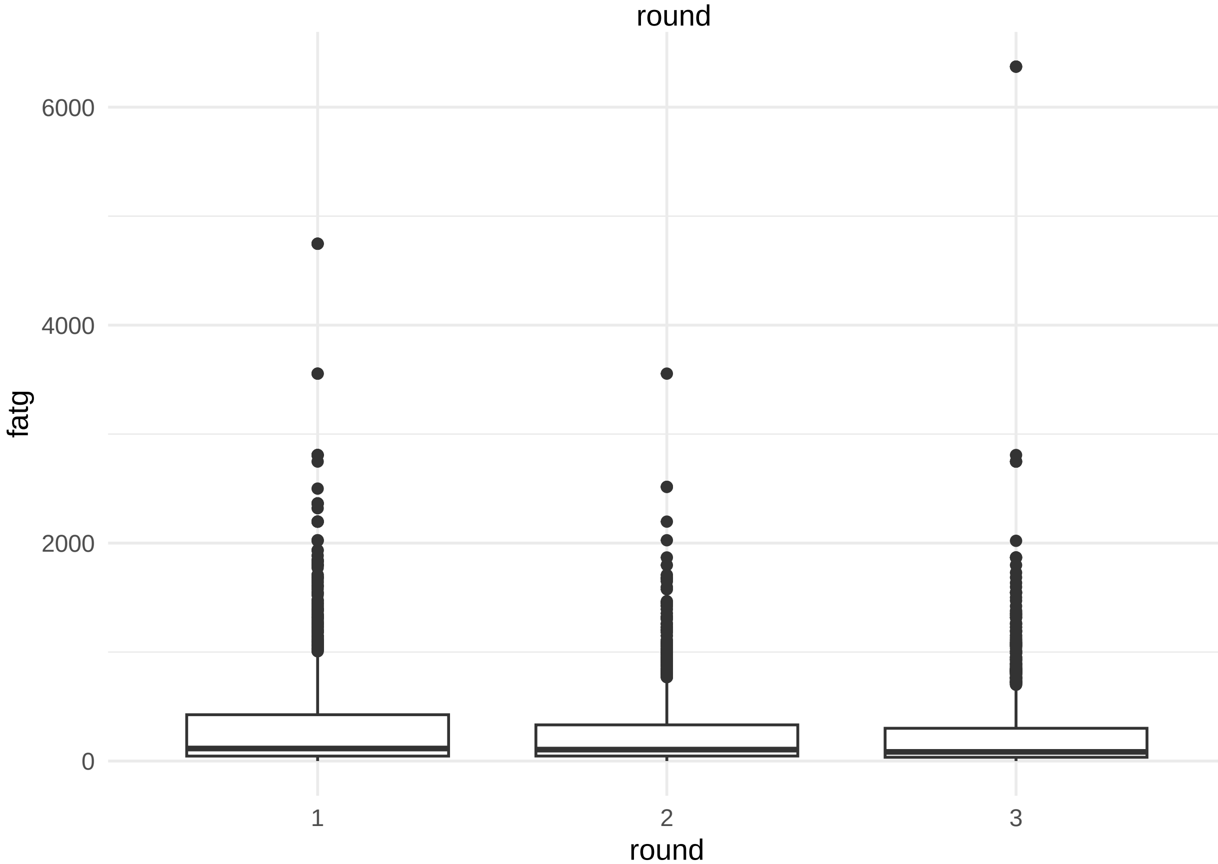

caff
3000
2000
1000
0
0
2500
5000
7500
10000
sugarg

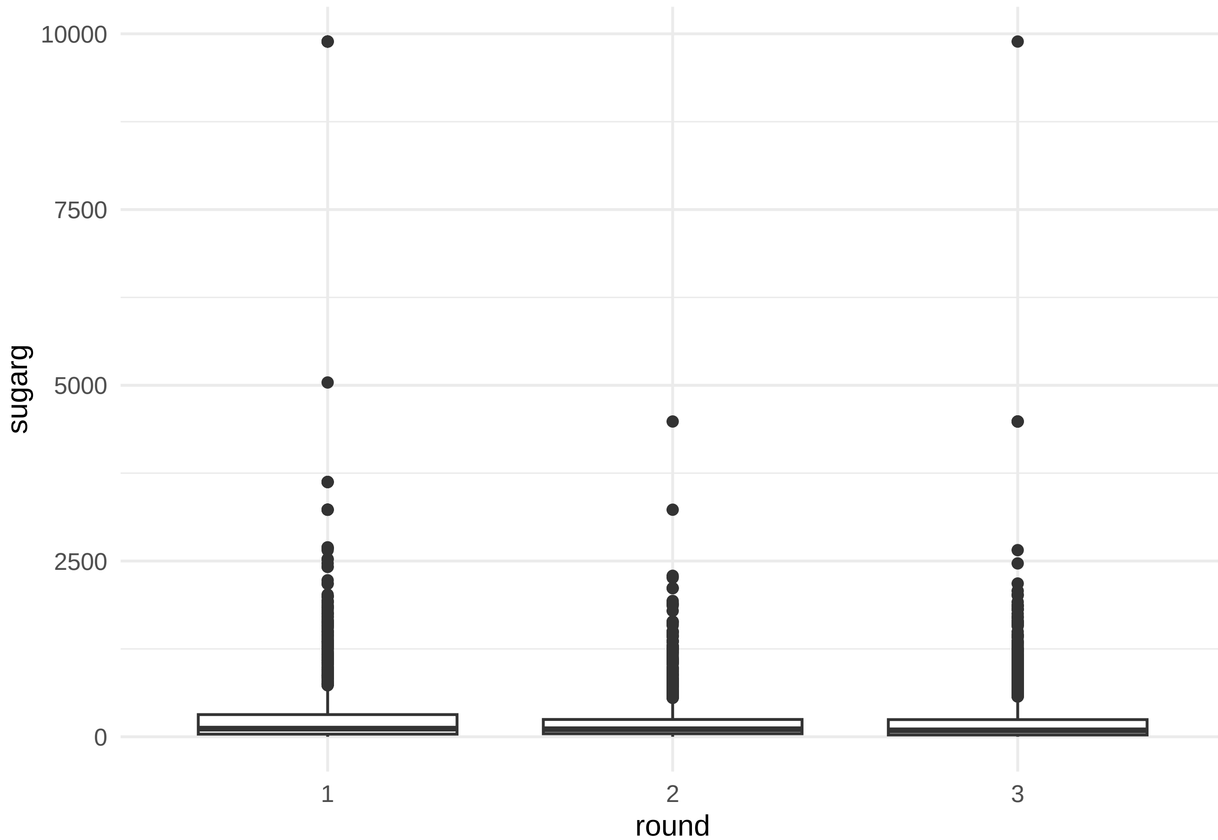

nsigns_ssb
15
10
5
0
1
2
3
round

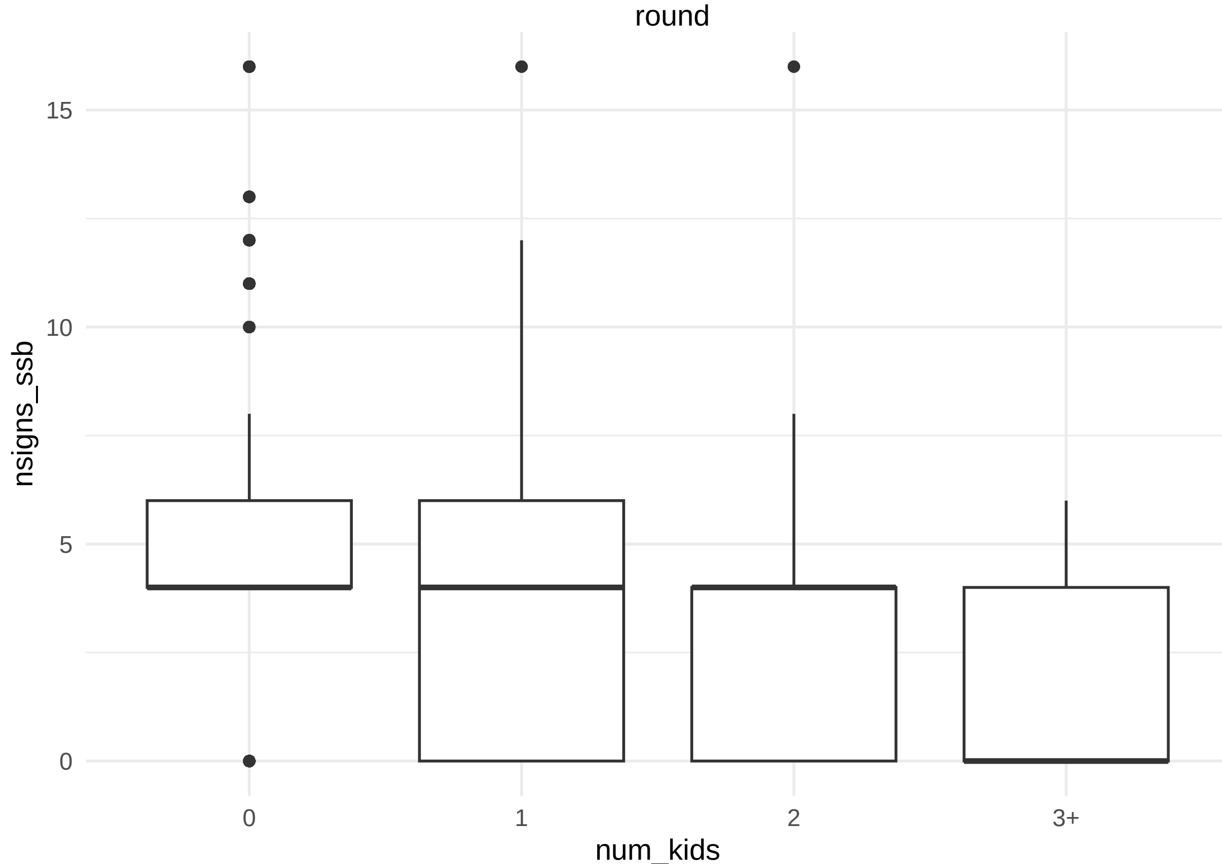

100%
75%
50%
25%
0%
count
1
2
3
round
num_kids
0
1
2
3+

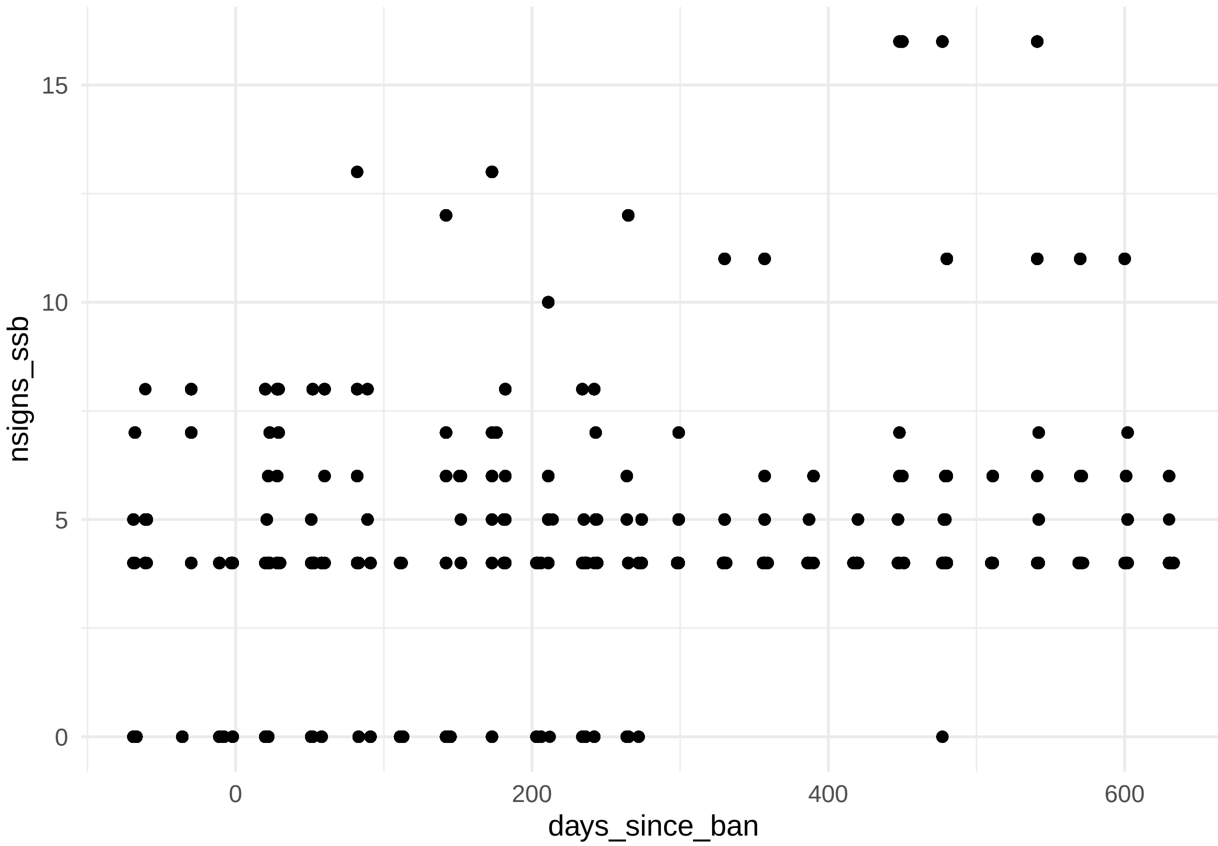

600
400
200
0
days_since_ban
1
2
3
round

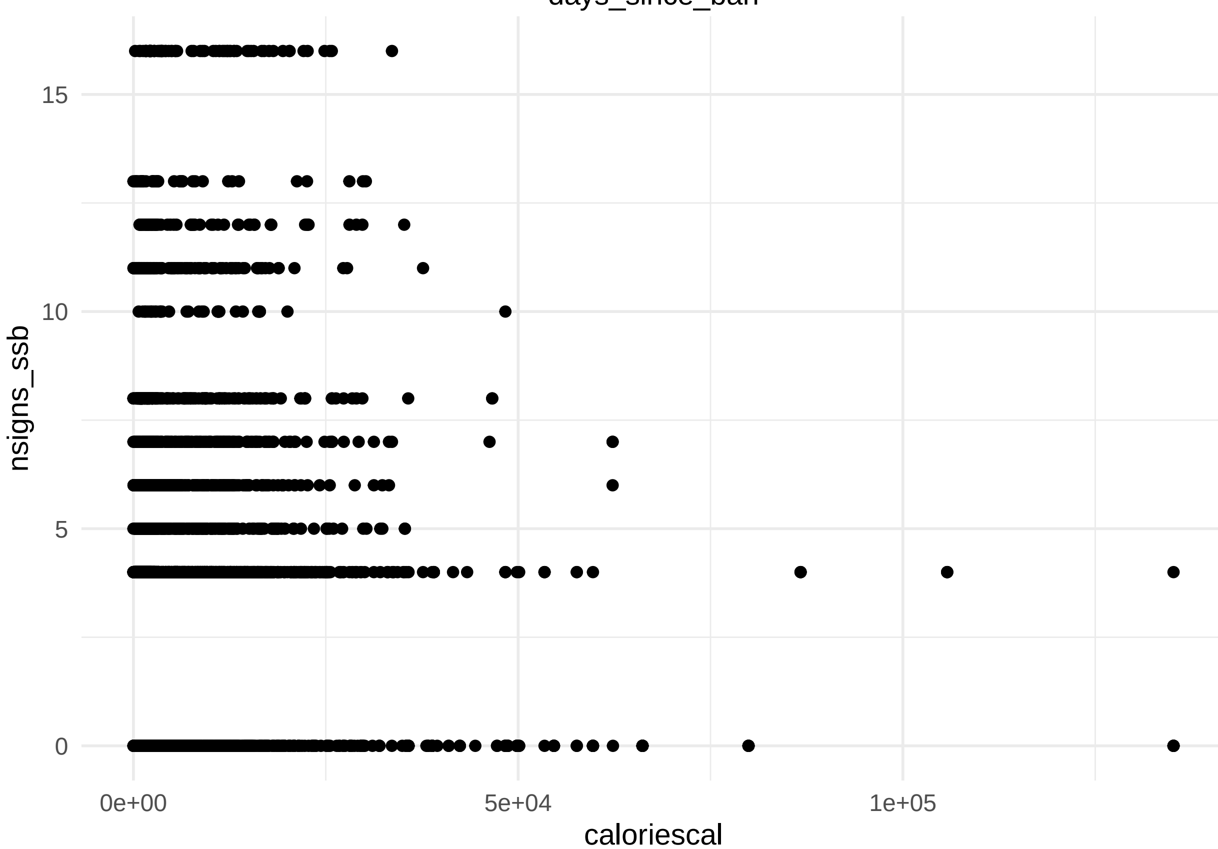

1e+05
5e+04
0e+00
caloriescal
1
2
3
round

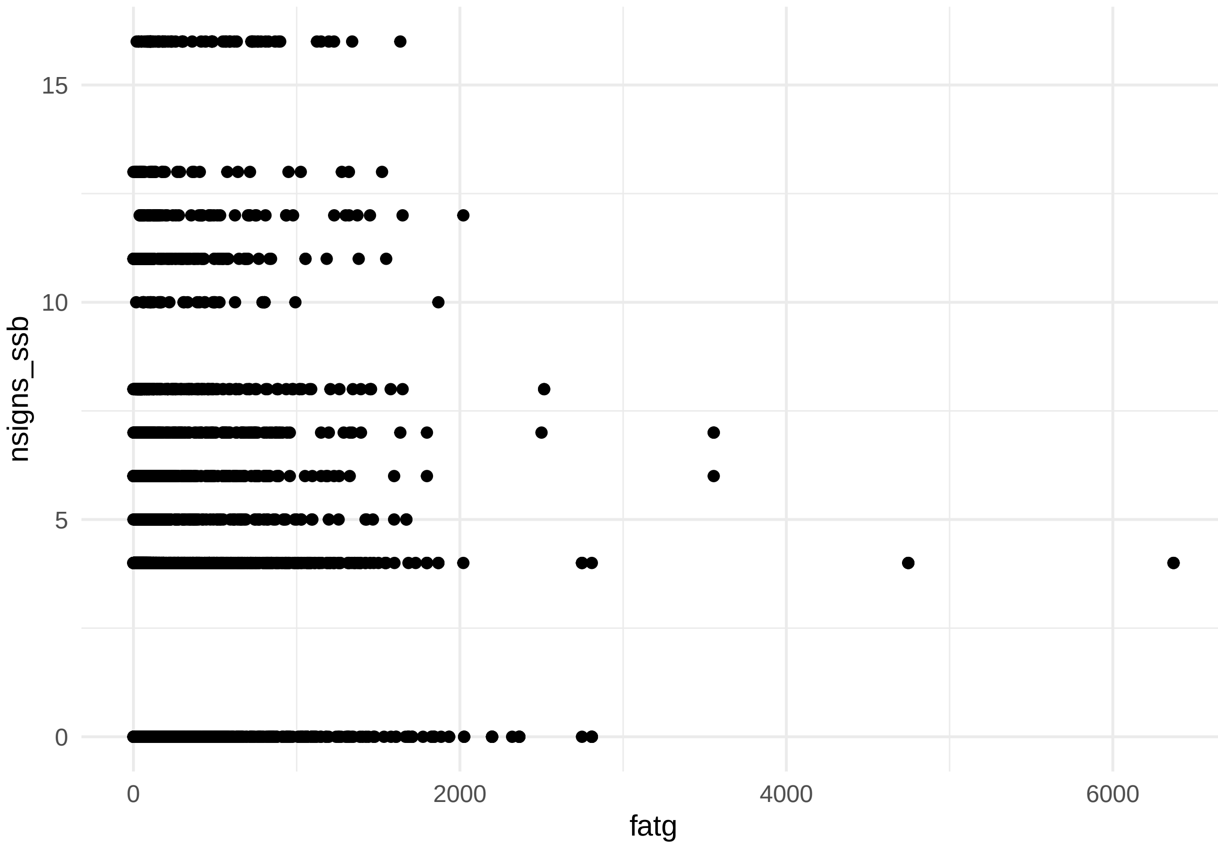

6000
4000
2000
0
fatg
1
2
3
round


10000
7500
5000
2500
0
sugarg
1
2
3
round

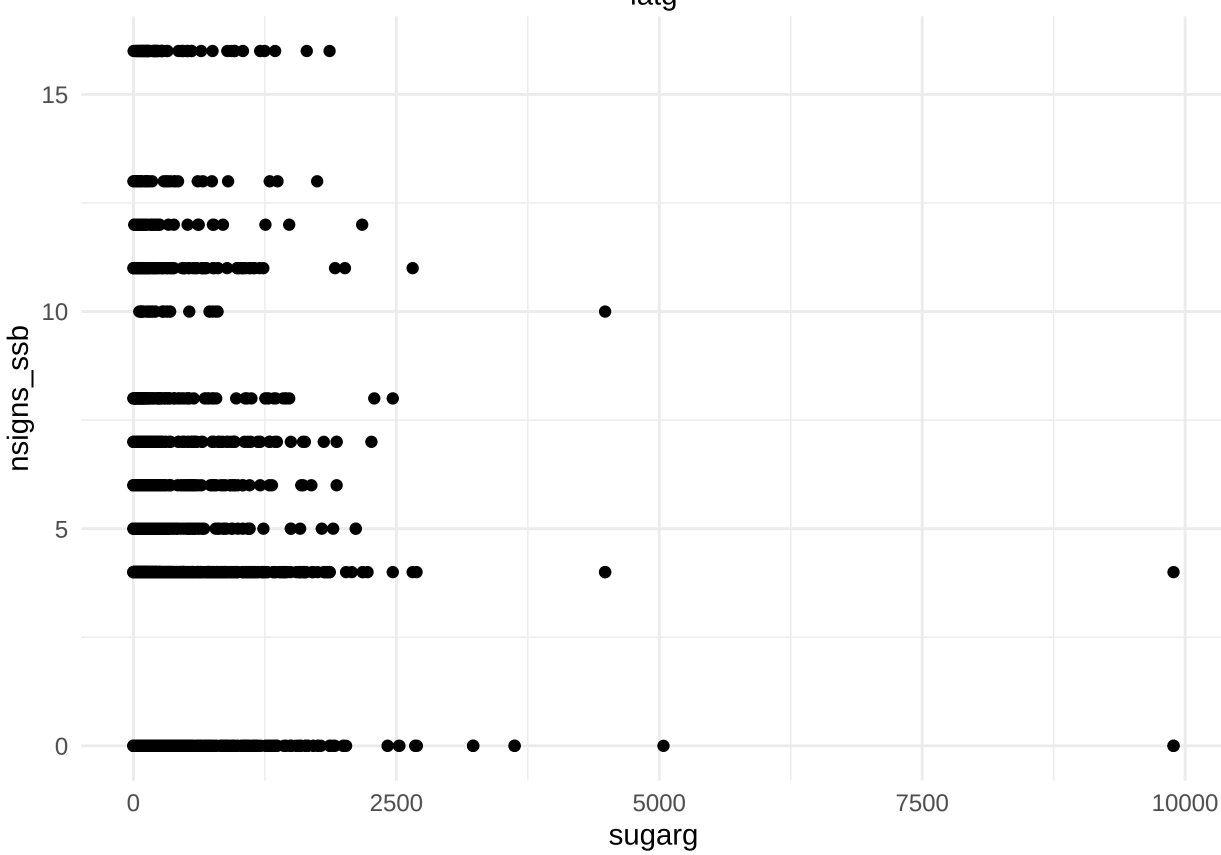

15
10
5
0
nsigns_ssb
0
1
2
3+
num_kids


nsigns_ssb
15
10
5
0
0
200
400
600
days_since_ban

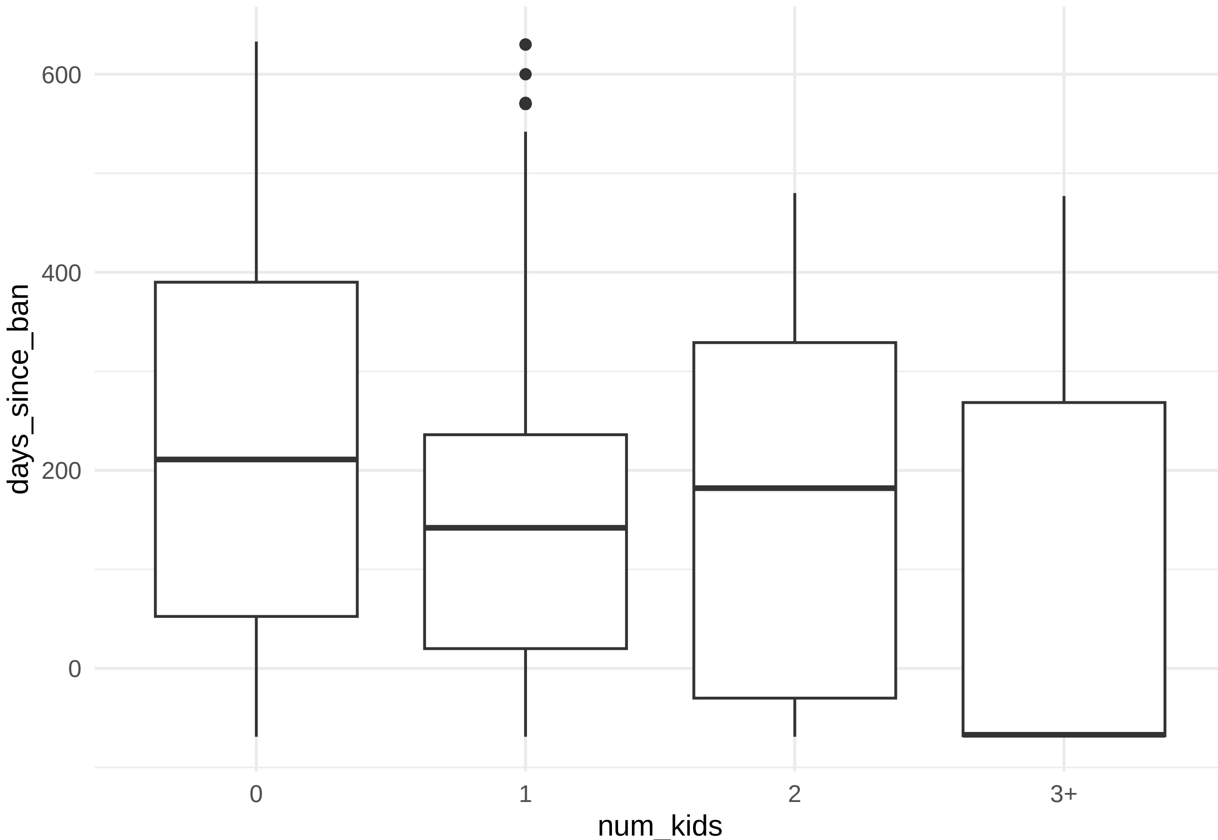

nsigns_ssb
15
10
5
0
0e+00
5e+04
1e+05
caloriescal

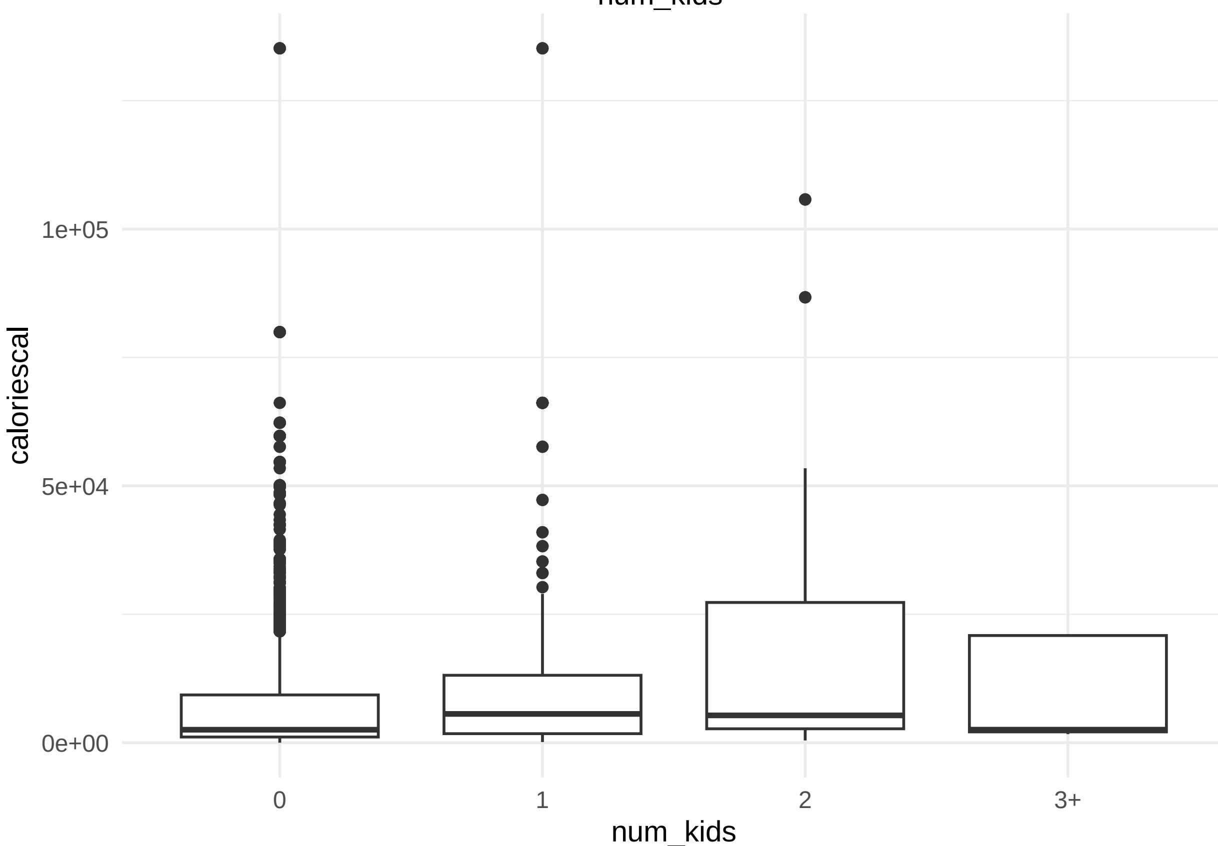


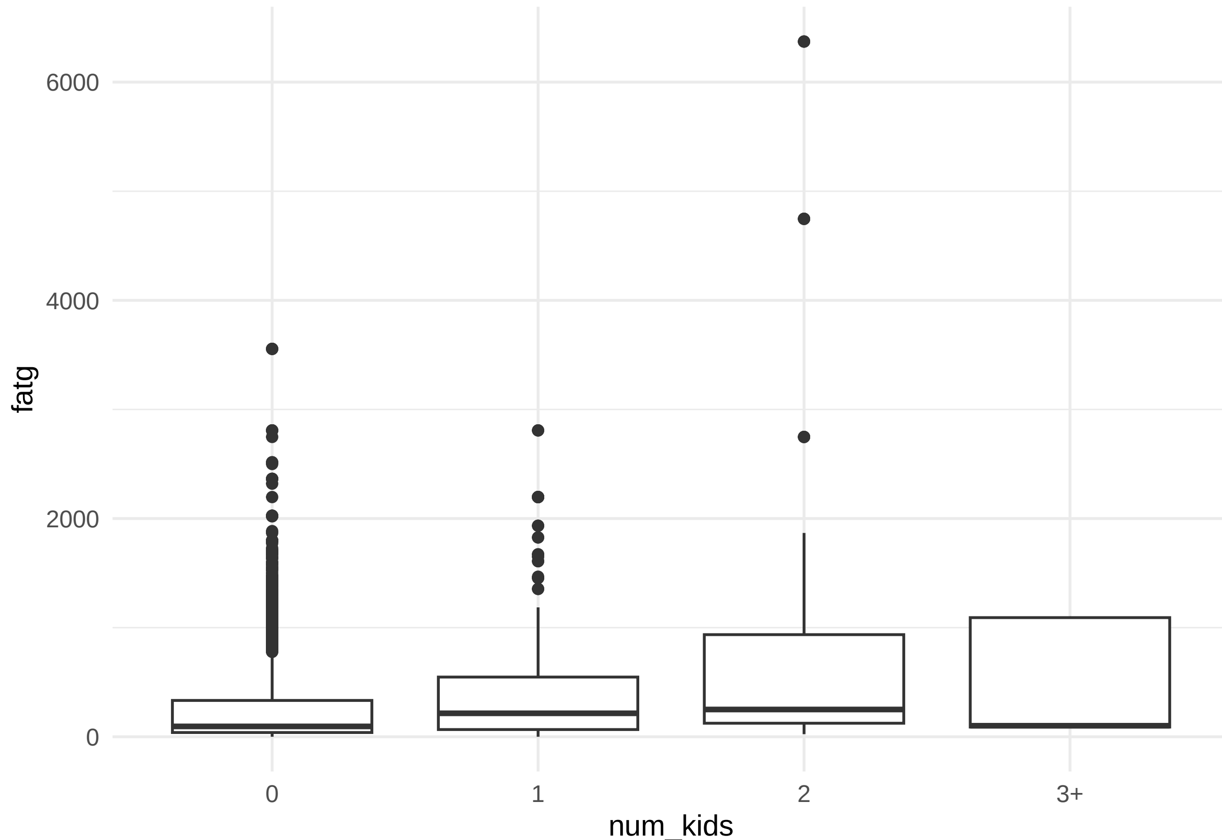
nsigns_ssb
15
10
5
0
0
2000
4000
6000
fatg

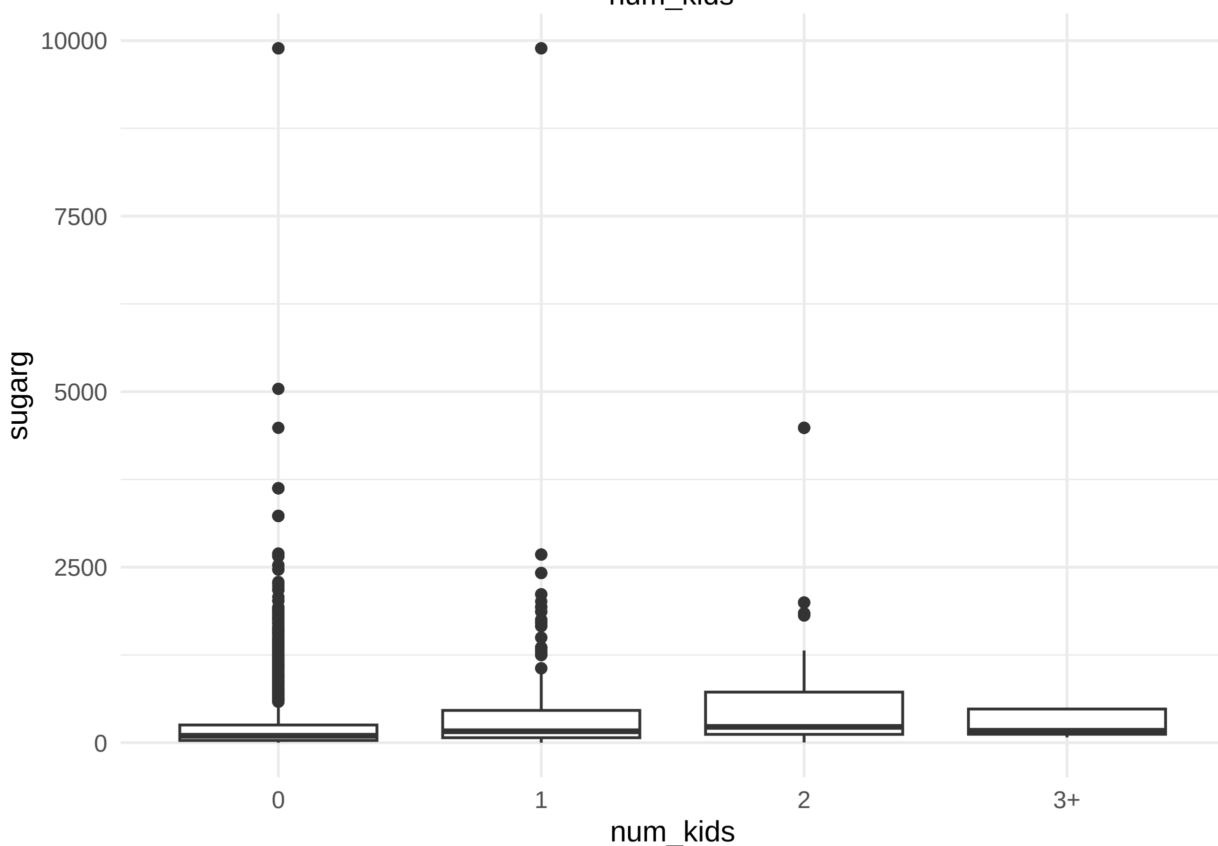


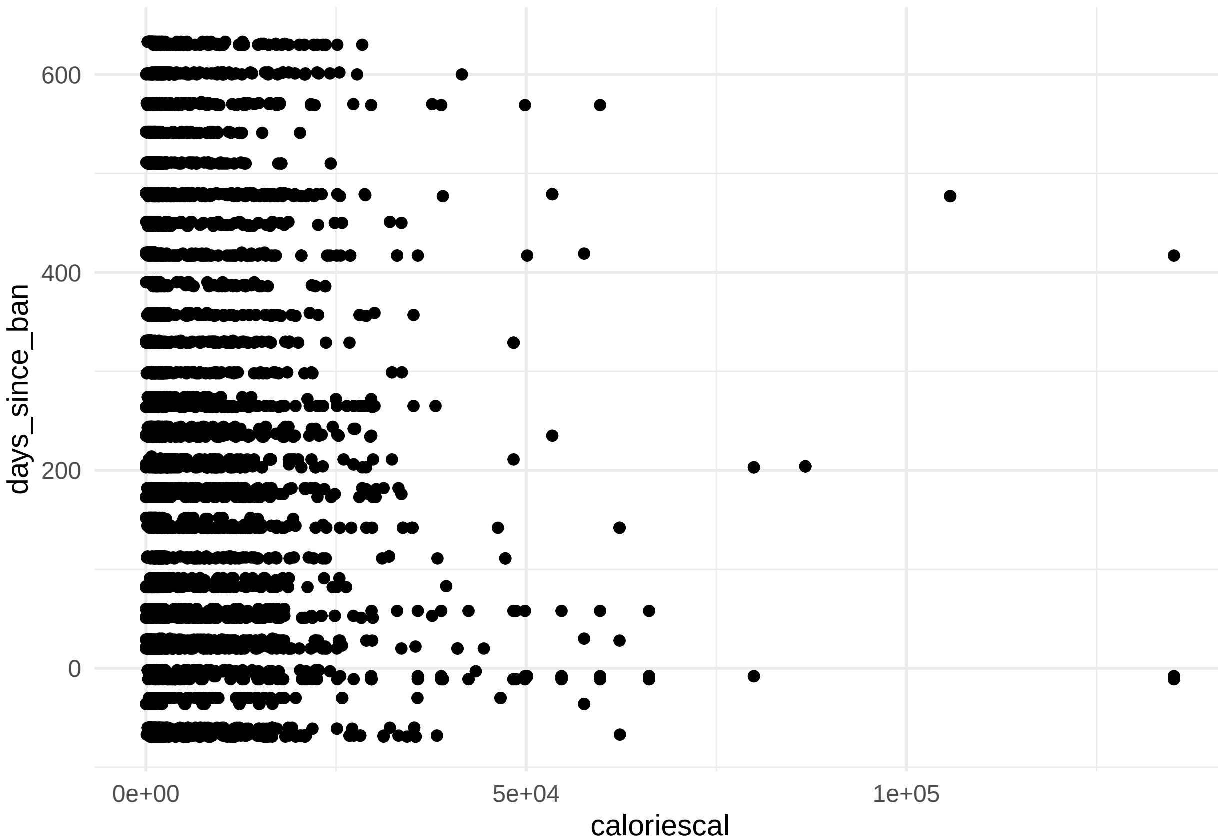
nsigns_ssb
15
10
5
0
0
2500
5000
7500
10000
sugarg

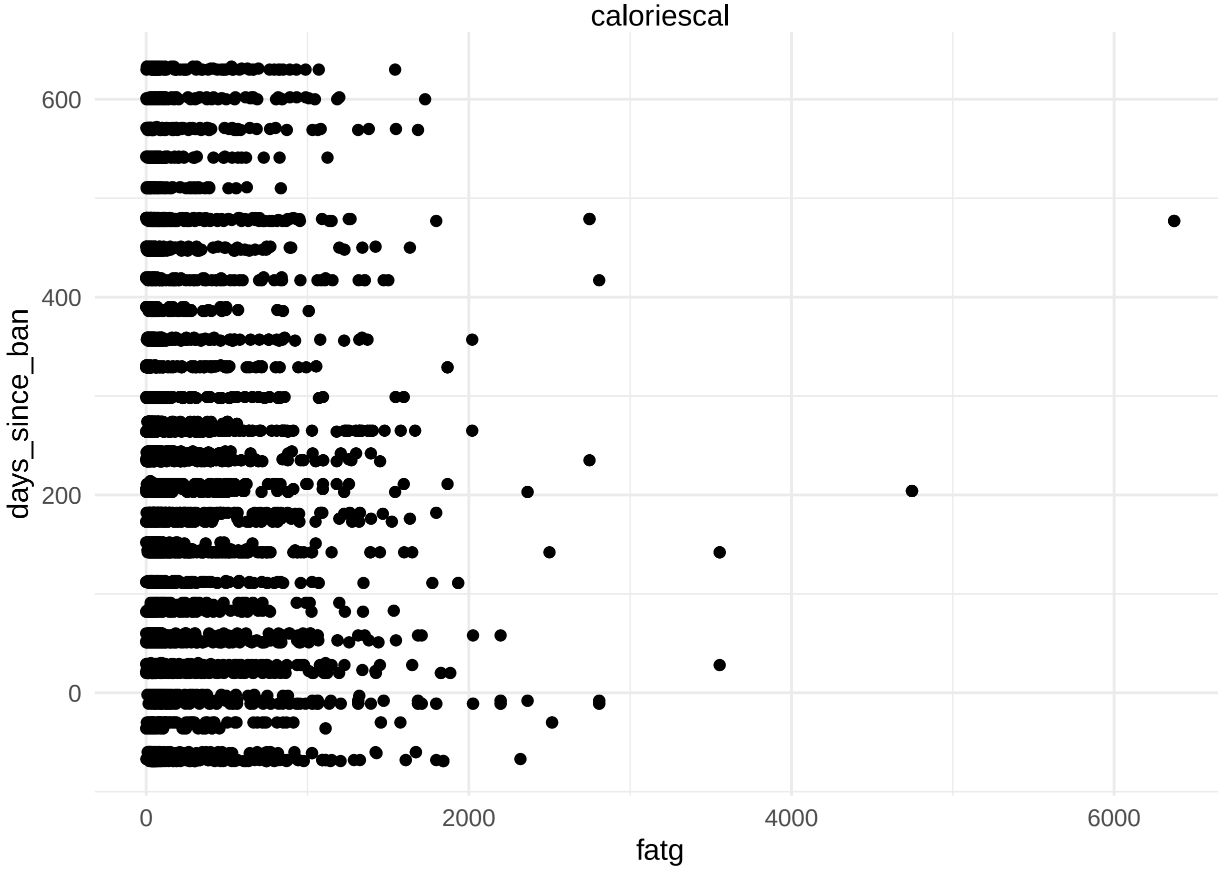

days_since_ban
600
400
200
0
0
1
2
3+
num_kids

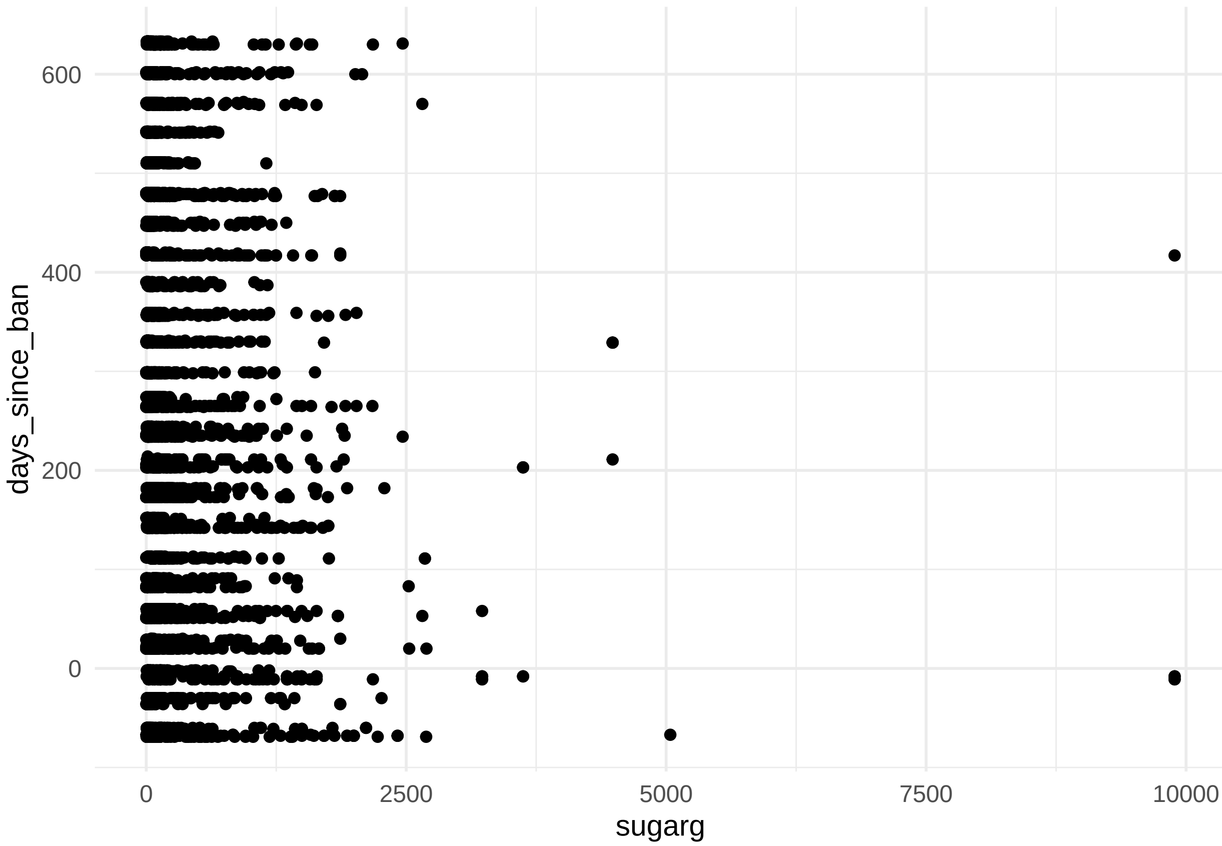

caloriescal
1e+05
5e+04
0e+00
0
1
2
3+
num_kids

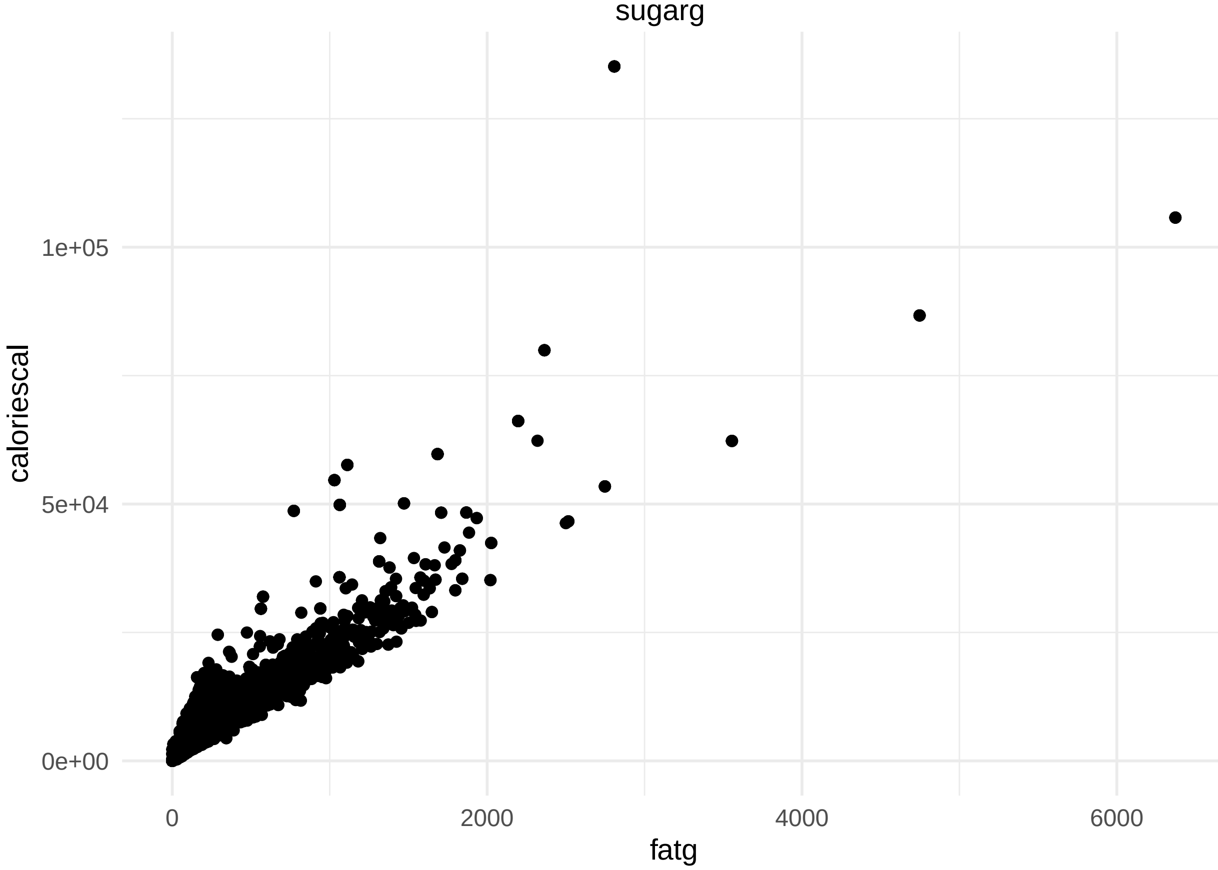

6000
4000
2000
0
fatg
0
1
2
3+
num_kids

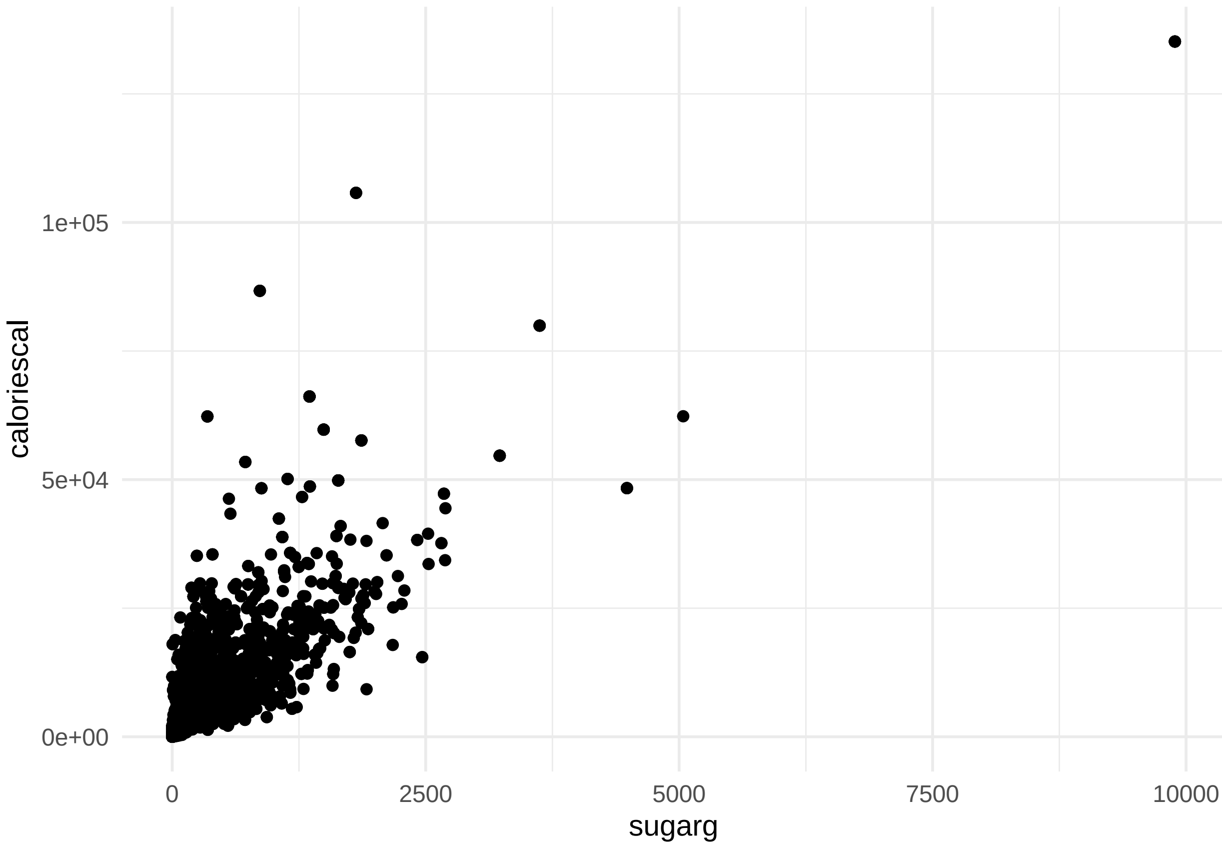

10000
7500
5000
2500
0
sugarg
0
1
2
3+
num_kids

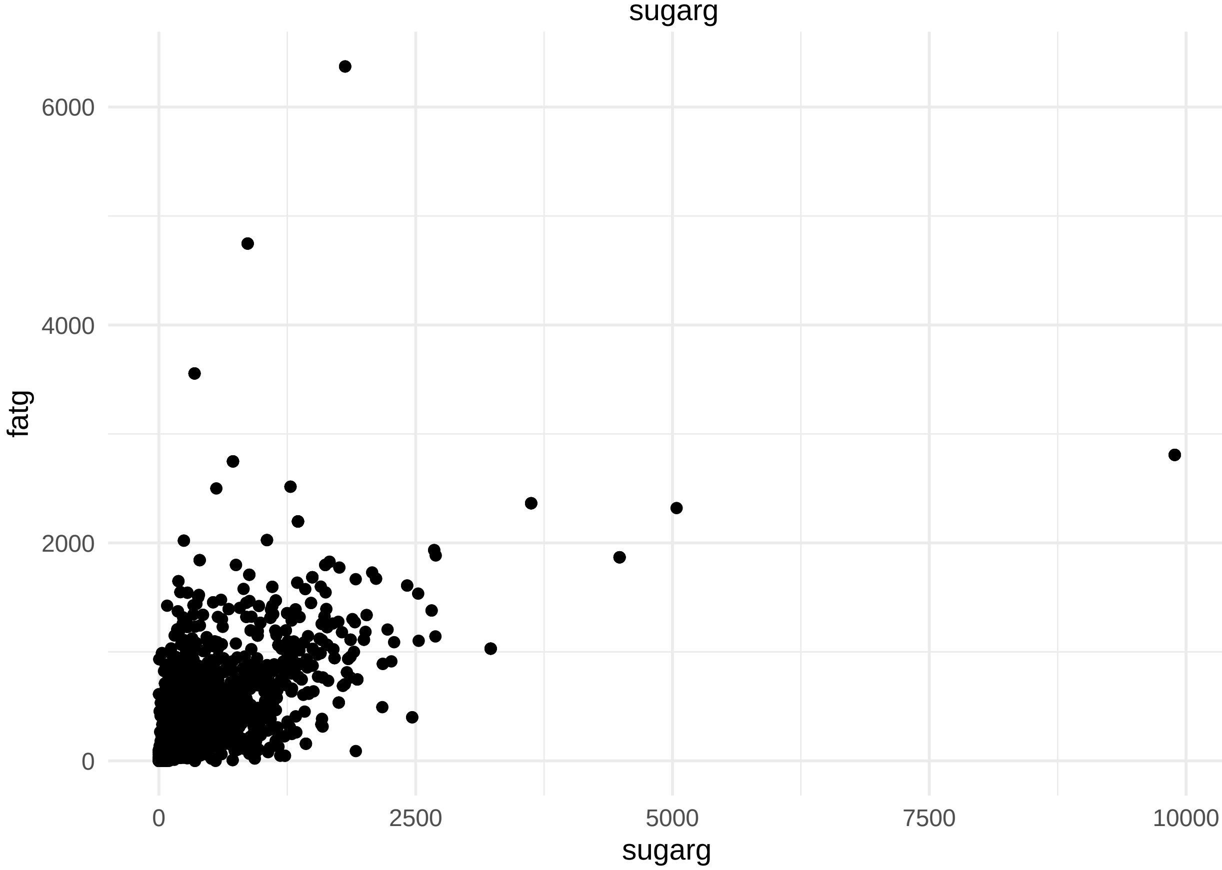

600
400
200
0
days_since_ban
0e+00
5e+04
1e+05
caloriescal


600
400
200
0
days_since_ban
0
2000
4000
6000
fatg


600
400
200
0
days_since_ban
0
2500
5000
7500
10000
sugarg


1e+05
5e+04
0e+00
caloriescal
0
2000
4000
6000
fatg


caloriescal
1e+05
5e+04
0e+00
0
2500
5000
7500
10000
sugarg


fatg
6000
4000
2000
0
0
2500
5000
7500
10000
sugarg

Modeling Process

Testing Different Optimization Methods

For models with no random effects, best to use Newton's approximation. For models with random effects, best to use `nlminb`, which is the default.

```
# No random effects
control_clm_full <- clm(limit ~ 1 + age + gender + race + edu + caff +
                          nsigns_ssb + num_kids + days_since_ban,
                   data = reduced_data, control = list(
  maxIter = 10000,
  maxLineIter = 2000,
  maxModIter = 2000,
  method = "Newton",
  trace = 1))
control_clm <- clm(limit ~ 1 + age + gender + race + edu + caff +
                    nsigns_ssb + num_kids + days_since_ban, data = reduced_data, control = list(
  method = "ucminf",
  stepmax = 1,
  grad = "central",
  maxeval = 500000,
  gradstep = c(1e-10, 1e-12),
  trace = 1))
control_clm <- clm(limit ~ 1 + age + gender + race + edu + caff +
                    nsigns_ssb + num_kids + days_since_ban, data = reduced_data, control = list(
  method = "nlminb",
  eval.max = 2000,
  iter.max = 1500,
  abs.tol = 1e-20,
  trace = 1))
control_clm <- clm(limit ~ 1 + age + gender + race + edu + caff +
                    nsigns_ssb + num_kids + days_since_ban, data = reduced_data, control = list(
  method = "optim",
  tmax = 100,
  maxit = 100000,
  type = 1,
  ndeps = 1e-10,
  REPORT = 1,
  trace = 1))

## Check with alternative packages. Produced the same intercepts
control_vglm <- vglm(limit ~ 1 + age + gender + race + edu + caff +
                      nsigns_ssb + num_kids + days_since_ban,
                    data = reduced_data, family = cumulative(parallel = TRUE))

## Random effects. Omit the rest for brevity
control_clmm_full <- clmm(limit ~ 1 + age + gender + race + edu + city + caff +
                            nsigns_ssb + num_kids + days_since_ban +
                            (1 | location) + (1 | round),
                          control = list(method = "nlminb",
                                         useMatrix = T,
                                         maxIter = 200,
                                         gradTol = 1e-4,
                                         maxLineIter = 200,
```

```
                                          trace = 1),
                      data = reduced_data, link = "logit")

# Same intercepts
summary(control_clm)
summary(control_vglm)
coef(control_vglm, matrix = T)

summary(control_clmm_full)
coef(control_clmm_full, matrix = T)
```

Full Model

Note that we also tested the non-standardized model. They both produced the similar conclusions. However, the non-standardized model couldn't fit properly because of the `kcal` variable. We proceeded with the standardized model for predictions.

```
control_clmm_full_std <- clmm(limit ~ 1 + age_std + gender + race + edu + city + caff_std +
                                nsigns_ssb_std + num_kids + days_since_ban_std +
                                caloriescal_std + fatg_std + sugarg_std +
                                (1 | location) + (1 | round),
                          control = list(method = "nlminb",
                                         useMatrix = T,
                                         maxIter = 200,
                                         gradTol = 1e-4,
                                         maxLineIter = 200
                                         # , trace = 1
                                         ),
                      data = reduced_data, link = "logit")

summary(control_clmm_full_std)
```

```
## Cumulative Link Mixed Model fitted with the Laplace approximation
##
## formula: limit ~ 1 + age_std + gender + race + edu + city + caff_std +
##     nsigns_ssb_std + num_kids + days_since_ban_std + caloriescal_std +
##     fatg_std + sugarg_std + (1 | location) + (1 | round)
## data:    reduced_data
##
##  link  threshold nobs logLik   AIC      niter       max.grad cond.H
##  logit flexible  4296 -6498.59 13053.18 6481(13170) 3.83e-03 1.6e+03
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  location (Intercept) 0.02328  0.1526
##  round    (Intercept) 0.00000  0.0000
## Number of groups:  location 57,  round 3
##
## Coefficients:
##                           Estimate Std. Error z value Pr(>|z|)
## age_std                  0.2259749  0.0296596   7.619 2.56e-14 ***
## genderM                 -0.2933068  0.0566808  -5.175 2.28e-07 ***
## raceBlack               -0.1940720  0.1185131  -1.638  0.10151
## raceNative              -0.2588840  0.2185724  -1.184  0.23624
```

```
## raceOther                   0.0470979  0.1266716   0.372  0.71003
## raceWhite                  -0.0452073  0.1228145  -0.368  0.71280
## eduCollege Degree           0.2579721  0.1262403   2.044  0.04100 *
## eduGraduate Degree          0.2480524  0.1462447   1.696  0.08986 .
## eduHigh School             -0.3044348  0.1217535  -2.500  0.01240 *
## eduLess than High School -0.1760818  0.2027481  -0.868  0.38513
## eduSome College            -0.0670136  0.1251944  -0.535  0.59246
## eduSome High School        -0.4089046  0.1461276  -2.798  0.00514 **
## cityNew York                0.1652177  0.0735426   2.247  0.02467 *
## caff_std                   -0.0220947  0.0368587  -0.599  0.54888
## nsigns_ssb_std              0.0679932  0.0383321   1.774  0.07610 .
## num_kids1                   0.0732200  0.1159336   0.632  0.52767
## num_kids2                  -0.2779861  0.2079258  -1.337  0.18124
## num_kids3+                 -1.4221950  0.5391322  -2.638  0.00834 **
## days_since_ban_std         -0.0003282  0.0304117  -0.011  0.99139
## caloriescal_std             0.2954583  0.1209173   2.443  0.01455 *
## fatg_std                   -0.1628537  0.0902732  -1.804  0.07123 .
## sugarg_std                 -0.1083325  0.0567918  -1.908  0.05645 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Threshold coefficients:
##                   Estimate Std. Error z value
## Never|Seldom       -1.1865     0.1641  -7.230
## Seldom|Sometimes  -0.7759     0.1635  -4.745
## Sometimes|Often    0.2743     0.1630   1.683
## Often|Always       1.1073     0.1640   6.751
```

```r
## Non-standardized model
# control_clmm_full_non <- clmm(limit ~ 1 + age + gender + race + edu + city + caff +
# nsigns_ssb + num_kids + days_since_ban + kcal + fv +
# (1 | location) + (1 | round),
#                            control = list(method = "nlminb",
#                                           useMatrix = T,
#                                           maxIter = 200,
#                                           gradTol = 1e-4,
#                                           maxLineIter = 200,
#                                           trace = 1),
#                       data = reduced_data, link = "logit")
```

Fixed Effects

```r
control_clmm_red <- clmm(limit ~ 1 + age_std + gender + race + edu + city +
                           caff_std + num_kids + caloriescal_std +
                                (1 | location) + (1 | round),
                         data = reduced_data, link = "logit")
anova(control_clmm_red, control_clmm_full_std)
```

```
## Likelihood ratio tests of cumulative link models:
##
##                         formula:
## control_clmm_red      limit ~ 1 + age_std + gender + race + edu + city + caff_std + num_kids + calor
## control_clmm_full_std limit ~ 1 + age_std + gender + race + edu + city + caff_std + nsigns_ssb_std +
##                         link: threshold:
```

```
## control_clmm_red      logit flexible
## control_clmm_full_std logit flexible
##
##                       no.par   AIC  logLik LR.stat df Pr(>Chisq)
## control_clmm_red          24 13053 -6502.7
## control_clmm_full_std     28 13053 -6498.6  8.1352  4    0.08675 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(control_clmm_red)
```

```
## Cumulative Link Mixed Model fitted with the Laplace approximation
##
## formula: limit ~ 1 + age_std + gender + race + edu + city + caff_std +
##     num_kids + caloriescal_std + (1 | location) + (1 | round)
## data:    reduced_data
##
##  link  threshold nobs logLik   AIC      niter       max.grad cond.H
##  logit flexible  4296 -6502.66 13053.31 5174(10480) 1.56e-02 1.6e+03
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  location (Intercept) 2.112e-02 1.453e-01
##  round    (Intercept) 4.405e-12 2.099e-06
## Number of groups:  location 57,  round 3
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## age_std                   0.22612    0.02963   7.632 2.32e-14 ***
## genderM                  -0.28990    0.05663  -5.119 3.06e-07 ***
## raceBlack                -0.18376    0.11848  -1.551  0.12091
## raceNative               -0.24394    0.21813  -1.118  0.26344
## raceOther                 0.04708    0.12665   0.372  0.71007
## raceWhite                -0.03634    0.12274  -0.296  0.76718
## eduCollege Degree         0.27474    0.12604   2.180  0.02928 *
## eduGraduate Degree        0.25441    0.14602   1.742  0.08145 .
## eduHigh School           -0.30339    0.12164  -2.494  0.01262 *
## eduLess than High School -0.17518    0.20273  -0.864  0.38753
## eduSome College          -0.06414    0.12512  -0.513  0.60819
## eduSome High School      -0.41190    0.14599  -2.822  0.00478 **
## cityNew York              0.14847    0.07197   2.063  0.03912 *
## caff_std                 -0.03447    0.03624  -0.951  0.34153
## num_kids1                 0.05873    0.11583   0.507  0.61211
## num_kids2                -0.33591    0.20648  -1.627  0.10377
## num_kids3+               -1.45623    0.53976  -2.698  0.00698 **
## caloriescal_std           0.05965    0.03619   1.648  0.09936 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Threshold coefficients:
##                  Estimate Std. Error z value
## Never|Seldom      -1.1754     0.1638  -7.177
## Seldom|Sometimes  -0.7652     0.1632  -4.689
## Sometimes|Often    0.2825     0.1627   1.737
## Often|Always       1.1135     0.1638   6.799
```

Random Effects

Note that we couldn't perform bootstrap because the `simulate` command is not implemented in `ordinal`, but the effects are fairly marginal and not significant.

```
control_clmm_loc <- clmm(limit ~ 1 + age_std + gender + race + edu + city +
                           caff_std + num_kids + caloriescal_std +
                           (1 | location),
                         data = reduced_data, link = "logit")

lrt_obs_round <- as.numeric(2*(logLik(control_clmm_red) -
                                 logLik(control_clmm_loc)))
.5*(1 - pchisq(lrt_obs_round, 0)) + .5*(1 - pchisq(lrt_obs_round, 1))
```

Level 3 Random Intercept

```
## [1] 0.4973872
```

```
control_clm <- clm(limit ~ 1 + age_std + gender + race + edu + city +
                           caff_std + num_kids + caloriescal_std,
                   data = reduced_data, link = "logit")
lrt_obs_loc <- as.numeric(2*(logLik(control_clmm_loc) - logLik(control_clm)))
.5*(1 - pchisq(lrt_obs_loc, 0)) + .5*(1 - pchisq(lrt_obs_loc, 1))
```

Level 2 Random Intercept

```
## [1] 0.002519061
```

```
summary(control_clmm_loc)
```

```
## Cumulative Link Mixed Model fitted with the Laplace approximation
##
## formula: limit ~ 1 + age_std + gender + race + edu + city + caff_std +
##     num_kids + caloriescal_std + (1 | location)
## data:    reduced_data
##
##  link  threshold nobs logLik   AIC      niter      max.grad cond.H
##  logit flexible  4296 -6502.66 13051.31 4822(9818) 1.05e-02 1.6e+03
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  location (Intercept) 0.02112  0.1453
## Number of groups:  location 57
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## age_std                   0.22611    0.02963   7.632 2.32e-14 ***
## genderM                  -0.28989    0.05663  -5.119 3.07e-07 ***
## raceBlack                -0.18376    0.11847  -1.551  0.12088
## raceNative               -0.24392    0.21812  -1.118  0.26344
## raceOther                 0.04710    0.12663   0.372  0.70996
## raceWhite                -0.03632    0.12273  -0.296  0.76725
## eduCollege Degree         0.27476    0.12602   2.180  0.02924 *
## eduGraduate Degree        0.25443    0.14600   1.743  0.08138 .
```

```
## eduHigh School           -0.30337    0.12162  -2.494  0.01262 *
## eduLess than High School -0.17513    0.20274  -0.864  0.38767
## eduSome College          -0.06412    0.12510  -0.513  0.60827
## eduSome High School      -0.41188    0.14598  -2.822  0.00478 **
## cityNew York              0.14847    0.07197   2.063  0.03911 *
## caff_std                 -0.03447    0.03624  -0.951  0.34151
## num_kids1                 0.05873    0.11583   0.507  0.61215
## num_kids2                -0.33592    0.20648  -1.627  0.10376
## num_kids3+               -1.45623    0.53974  -2.698  0.00698 **
## caloriescal_std           0.05965    0.03619   1.648  0.09936 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Threshold coefficients:
##                   Estimate Std. Error z value
## Never|Seldom       -1.1753     0.1637  -7.179
## Seldom|Sometimes  -0.7651     0.1631  -4.690
## Sometimes|Often    0.2826     0.1626   1.737
## Often|Always       1.1135     0.1637   6.801
```

Separate slopes for each level

Ordinal provides two built-in commands for testing whether we need separate slopes for predictors of each level and whether we need to scale our response by each predictors. None of them showed significance.

```
nominal_test(control_clm)
```

```
## Tests of nominal effects
##
## formula: limit ~ 1 + age_std + gender + race + edu + city + caff_std + num_kids + caloriescal_std
##                  Df  logLik   AIC     LRT Pr(>Chi)
## <none>              -6506.6 13057
## age_std           3 -6502.0 13054  9.1813 0.026975 *
## gender            3 -6499.4 13049 14.2836 0.002543 **
## race             12 -6499.0 13066 15.2702 0.226990
## edu              18 -6496.0 13072 21.1543 0.271698
## city              3 -6498.8 13048 15.5804 0.001382 **
## caff_std          3 -6499.6 13049 13.9194 0.003017 **
## num_kids
## caloriescal_std  3 -6502.5 13055  8.1489 0.043033 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
scale_test(control_clm)
```

```
## Tests of scale effects
##
## formula: limit ~ 1 + age_std + gender + race + edu + city + caff_std + num_kids + caloriescal_std
##                  Df  logLik   AIC     LRT  Pr(>Chi)
## <none>              -6506.6 13057
## age_std           1 -6503.3 13053  6.5478 0.0105013 *
## gender            1 -6504.5 13055  4.2001 0.0404219 *
## race              4 -6500.4 13053 12.2898 0.0153214 *
## edu               6 -6503.4 13063  6.4602 0.3736573
## city              1 -6501.1 13048 10.9697 0.0009261 ***
## caff_std          1 -6504.8 13056  3.5489 0.0595853 .
```

```
## num_kids         3 -6505.4 13061  2.3111 0.5103986
## caloriescal_std  1 -6505.6 13057  2.0258 0.1546482
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
control_clm_nom <- clm(limit ~ 1 + race + edu + num_kids + caloriescal_std,
                       nominal = ~ age_std + gender + city + caff_std,
                   data = reduced_data, link = "logit")
anova(control_clm_nom, control_clm)
```

```
## Likelihood ratio tests of cumulative link models:
##
##                 formula:
## control_clm     limit ~ 1 + age_std + gender + race + edu + city + caff_std + num_kids + caloriescal_
## control_clm_nom limit ~ 1 + race + edu + num_kids + caloriescal_std
##                 nominal:                                link: threshold:
## control_clm     ~1                                      logit flexible
## control_clm_nom ~age_std + gender + city + caff_std logit flexible
##
##                 no.par   AIC  logLik LR.stat df Pr(>Chisq)
## control_clm         22 13057 -6506.6
## control_clm_nom     34 13029 -6480.5  52.194 12  5.728e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Overall fit Compared to the only intercept model.

```
control_null <- clm(limit ~ 1, data = reduced_data, link = "logit")
```

```
## Warning: (-1) Model failed to converge with max|grad| = 5.30767e-06 (tol = 1e-06)
## In addition: iteration limit reached
```

```
# Overall fit
anova(control_null, control_clm_nom)
```

```
## Likelihood ratio tests of cumulative link models:
##
##                 formula:
## control_null    limit ~ 1
## control_clm_nom limit ~ 1 + race + edu + num_kids + caloriescal_std
##                 nominal:                                link: threshold:
## control_null    ~1                                      logit flexible
## control_clm_nom ~age_std + gender + city + caff_std logit flexible
##
##                 no.par   AIC  logLik LR.stat df Pr(>Chisq)
## control_null         4 13243 -6617.6
## control_clm_nom     34 13029 -6480.5  274.31 30  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
control_null_re <- clmm(limit ~ 1 + (1 | location), data = reduced_data, link = "logit")
```

```
## Warning: (-1) Model failed to converge with max|grad| = 5.30767e-06 (tol = 1e-06)
## In addition: iteration limit reached
```

```
anova(control_null, control_clmm_loc)
```

```
## Likelihood ratio tests of cumulative link models:
```

```
##
##                  formula:
## control_null     limit ~ 1
## control_clmm_loc limit ~ 1 + age_std + gender + race + edu + city + caff_std + num_kids + caloriescal
##                  link: threshold:
## control_null     logit flexible
## control_clmm_loc logit flexible
##
##                  no.par   AIC  logLik LR.stat df Pr(>Chisq)
## control_null          4 13243 -6617.6
## control_clmm_loc     23 13051 -6502.7  229.99 19  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model Diagnostics

Accuracy Metrics

Because residual analysis are not well understood in ordinal models, we opted for accuracy metrics. Note that our model doesn't predict well.

```r
library(tidymodels)
```

```
## -- Attaching packages -------------------------------------- tidymodels 1.3.0 --

## v broom        1.0.7     v rsample      1.2.1
## v dials        1.4.0     v tune         1.3.0
## v infer        1.0.7     v workflows    1.2.0
## v modeldata    1.4.0     v workflowsets 1.1.0
## v parsnip      1.3.0     v yardstick    1.3.2
## v recipes      1.1.1

## -- Conflicts ----------------------------------------- tidymodels_conflicts() --
## x rlang::%@%()                 masks purrr::%@%()
## x scales::discard()            masks purrr::discard()
## x Matrix::expand()             masks tidyr::expand()
## x dplyr::filter()              masks stats::filter()
## x recipes::fixed()             masks stringr::fixed()
## x rlang::flatten()             masks purrr::flatten()
## x rlang::flatten_chr()         masks purrr::flatten_chr()
## x rlang::flatten_dbl()         masks purrr::flatten_dbl()
## x rlang::flatten_int()         masks purrr::flatten_int()
## x rlang::flatten_lgl()         masks purrr::flatten_lgl()
## x rlang::flatten_raw()         masks purrr::flatten_raw()
## x rlang::invoke()              masks purrr::invoke()
## x dplyr::lag()                 masks stats::lag()
## x Matrix::pack()               masks tidyr::pack()
## x car::recode()                masks dplyr::recode()
## x ordinal::slice()             masks dplyr::slice()
## x car::some()                  masks purrr::some()
## x yardstick::spec()            masks readr::spec()
## x rlang::splice()              masks purrr::splice()
## x recipes::step()              masks stats::step()
## x Matrix::unpack()             masks tidyr::unpack()
## x recipes::update()            masks stats4::update(), Matrix::update(), stats::update()
## x workflows::update_formula() masks VGAM::update_formula()
```

```r
library(workflows)

model_accuracy <- function(model = control_clm, adj = F) {
    comp_metrics <- function(model = model, predict) {
      control_results <- reduced_data %>%
      bind_cols(fit = predict)


    # Confusion matrix
    # table(control_results$limit, control_results$fit)
    conf_mat(control_results, truth = limit, estimate = fit) -> conf

    # accuracy metrics
    accuracy(control_results, truth = limit, estimate = fit) -> acc
    sensitivity(control_results, truth = limit, estimate = fit) -> sen
    specificity(control_results, truth = limit, estimate = fit) -> spe
    # ppv(control_results, truth = limit, estimate = fit)

    # Goodness of fit
    chisq.test(control_results$limit, control_results$fit) -> gof

    return(list(control_results = control_results, conf = conf, acc = acc, sen = sen, spe = spe, gof = g
    }
  if (adj) {
    # Predict response
    control_vglm_pred <- predict(model, type = "response")
    level_counts <- table(reduced_data$limit)
    total_counts <- sum(level_counts)
    proportions <- as.numeric(level_counts / total_counts)
    names(proportions) <- names(level_counts)

    adjusted_probs <- control_vglm_pred / proportions[colnames(control_vglm_pred)]
    adjusted_probs <- adjusted_probs / rowSums(adjusted_probs)
    fit <- ordered(colnames(adjusted_probs)[max.col(adjusted_probs)],
                     levels = c("Never", "Seldom", "Sometimes",
                                "Often", "Always"))
    comp_metrics(model = model, predict = fit) -> result
  } else {
    # Predict response
    control_pred <- predict(model, type = "class")
    comp_metrics(model = model, control_pred) -> result
  }
  return(result)
}

model_accuracy(control_clm_nom)
```

```
## $control_results
## # A tibble: 4,296 x 27
##    receiptid person_id limit       age age_std gender race  edu    city    caff
##    <fct>     <fct>     <ord>     <dbl>   <dbl> <chr>  <chr> <chr>  <chr>  <dbl>
##  1 B103-098  900563    Sometimes    30 -0.670  F      Black Associ~ New ~     0
##  2 B103-022  900076    Sometimes    22 -1.22   F      Other Some C~ New ~     0
##  3 B103-091  900582    Never        40  0.0172 F      White High S~ New ~ 1137.
```

```
##  4 B103-081  900569    Never        20 -1.36   F      Other Some H~ New ~    0
##  5 B103-080  900568    Sometimes    34 -0.395  M      Other High S~ New ~    0
##  6 B103-090  900578    Seldom       19 -1.43   F      Other High S~ New ~  101.
##  7 B103-024  900078    Sometimes    61  1.46   M      Black Some C~ New ~   75.8
##  8 B103-067  900021    Always       51  0.773  F      Black Colleg~ New ~    0
##  9 B103-086  900574    Never        58  1.25   F      Black Some C~ New ~    0
## 10 B103-023  900077    Never        50  0.704  F      Black Colleg~ New ~    0
## # i 4,286 more rows
## # i 17 more variables: location <fct>, round <fct>, nsigns_ssb <dbl>,
## #   num_kids <chr>, surveydate <date>, days_since_ban <dbl>, caloriescal <dbl>,
## #   fatg <dbl>, sugarg <dbl>, black <chr>, caff_std <dbl>,
## #   nsigns_ssb_std <dbl>, days_since_ban_std <dbl>, caloriescal_std <dbl>,
## #   fatg_std <dbl>, sugarg_std <dbl>, fit <fct>
##
## $conf
##            Truth
## Prediction  Never Seldom Sometimes Often Always
##   Never       893    233       585   356    432
##   Seldom        0      0         0     0      0
##   Sometimes   209     73       260   179    216
##   Often         0      0         0     0      0
##   Always      171     71       209   160    249
##
## $acc
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy multiclass     0.326
##
## $sen
## # A tibble: 1 x 3
##   .metric     .estimator .estimate
##   <chr>       <chr>          <dbl>
## 1 sensitivity macro          0.245
##
## $spe
## # A tibble: 1 x 3
##   .metric     .estimator .estimate
##   <chr>       <chr>          <dbl>
## 1 specificity macro          0.816
##
## $gof
##
##  Pearson's Chi-squared test
##
## data:  control_results$limit and control_results$fit
## X-squared = 141.64, df = 8, p-value < 2.2e-16
```

```r
## Similar results under different model specifications
control_clm_probit <- clm(limit ~ 1 + age_std + gender + race + edu + city +
                           caff_std + num_kids + caloriescal_std,
                   data = reduced_data, link = "probit")
model_accuracy(control_clm_probit)
```

```
## $control_results
```

```
## # A tibble: 4,296 x 27
##    receiptid person_id limit       age age_std gender race  edu     city    caff
##    <fct>     <fct>     <ord>     <dbl>   <dbl> <chr>  <chr> <chr>   <chr>  <dbl>
##  1 B103-098  900563    Sometimes    30 -0.670  F      Black Associ~ New ~    0
##  2 B103-022  900076    Sometimes    22 -1.22   F      Other Some C~ New ~    0
##  3 B103-091  900582    Never        40  0.0172 F      White High S~ New ~ 1137.
##  4 B103-081  900569    Never        20 -1.36   F      Other Some H~ New ~    0
##  5 B103-080  900568    Sometimes    34 -0.395  M      Other High S~ New ~    0
##  6 B103-090  900578    Seldom       19 -1.43   F      Other High S~ New ~  101.
##  7 B103-024  900078    Sometimes    61  1.46   M      Black Some C~ New ~   75.8
##  8 B103-067  900021    Always       51  0.773  F      Black Colleg~ New ~    0
##  9 B103-086  900574    Never        58  1.25   F      Black Some C~ New ~    0
## 10 B103-023  900077    Never        50  0.704  F      Black Colleg~ New ~    0
## # i 4,286 more rows
## # i 17 more variables: location <fct>, round <fct>, nsigns_ssb <dbl>,
## #   num_kids <chr>, surveydate <date>, days_since_ban <dbl>, caloriescal <dbl>,
## #   fatg <dbl>, sugarg <dbl>, black <chr>, caff_std <dbl>,
## #   nsigns_ssb_std <dbl>, days_since_ban_std <dbl>, caloriescal_std <dbl>,
## #   fatg_std <dbl>, sugarg_std <dbl>, fit <fct>
##
## $conf
##            Truth
## Prediction  Never Seldom Sometimes Often Always
##   Never       989    264       692   432    487
##   Seldom        0      0         0     0      0
##   Sometimes    58     21        78    62     76
##   Often         0      0         0     0      0
##   Always      226     92       284   201    334
##
## $acc
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy multiclass     0.326
##
## $sen
## # A tibble: 1 x 3
##   .metric     .estimator .estimate
##   <chr>       <chr>          <dbl>
## 1 sensitivity macro          0.245
##
## $spe
## # A tibble: 1 x 3
##   .metric     .estimator .estimate
##   <chr>       <chr>          <dbl>
## 1 specificity macro          0.815
##
## $gof
##
##  Pearson's Chi-squared test
##
## data:  control_results$limit and control_results$fit
## X-squared = 143.72, df = 8, p-value < 2.2e-16
```

```r
control_clm_sym <- clm(limit ~ 1 + age_std + gender + race + edu + city +
                           caff_std + num_kids + caloriescal_std,
                  data = reduced_data,
                  link = "probit", threshold = "equidistant")
model_accuracy(control_clm_sym)
```

```
## $control_results
## # A tibble: 4,296 x 27
##    receiptid person_id limit       age age_std gender race  edu     city    caff
##    <fct>     <fct>     <ord>     <dbl>   <dbl> <chr>  <chr> <chr>   <chr>  <dbl>
##  1 B103-098  900563    Sometimes    30 -0.670  F      Black Associ~ New ~     0
##  2 B103-022  900076    Sometimes    22 -1.22   F      Other Some C~ New ~     0
##  3 B103-091  900582    Never        40  0.0172 F      White High S~ New ~ 1137.
##  4 B103-081  900569    Never        20 -1.36   F      Other Some H~ New ~     0
##  5 B103-080  900568    Sometimes    34 -0.395  M      Other High S~ New ~     0
##  6 B103-090  900578    Seldom       19 -1.43   F      Other High S~ New ~  101.
##  7 B103-024  900078    Sometimes    61  1.46   M      Black Some C~ New ~   75.8
##  8 B103-067  900021    Always       51  0.773  F      Black Colleg~ New ~     0
##  9 B103-086  900574    Never        58  1.25   F      Black Some C~ New ~     0
## 10 B103-023  900077    Never        50  0.704  F      Black Colleg~ New ~     0
## # i 4,286 more rows
## # i 17 more variables: location <fct>, round <fct>, nsigns_ssb <dbl>,
## #   num_kids <chr>, surveydate <date>, days_since_ban <dbl>, caloriescal <dbl>,
## #   fatg <dbl>, sugarg <dbl>, black <chr>, caff_std <dbl>,
## #   nsigns_ssb_std <dbl>, days_since_ban_std <dbl>, caloriescal_std <dbl>,
## #   fatg_std <dbl>, sugarg_std <dbl>, fit <fct>
##
## $conf
##            Truth
## Prediction  Never Seldom Sometimes Often Always
##   Never       956    252       649   412    455
##   Seldom        0      0         0     0      0
##   Sometimes     0      0         0     0      0
##   Often         0      0         0     0      0
##   Always      317    125       405   283    442
##
## $acc
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy multiclass     0.325
##
## $sen
## # A tibble: 1 x 3
##   .metric     .estimator .estimate
##   <chr>       <chr>          <dbl>
## 1 sensitivity macro          0.249
##
## $spe
## # A tibble: 1 x 3
##   .metric     .estimator .estimate
##   <chr>       <chr>          <dbl>
## 1 specificity macro          0.817
##
```

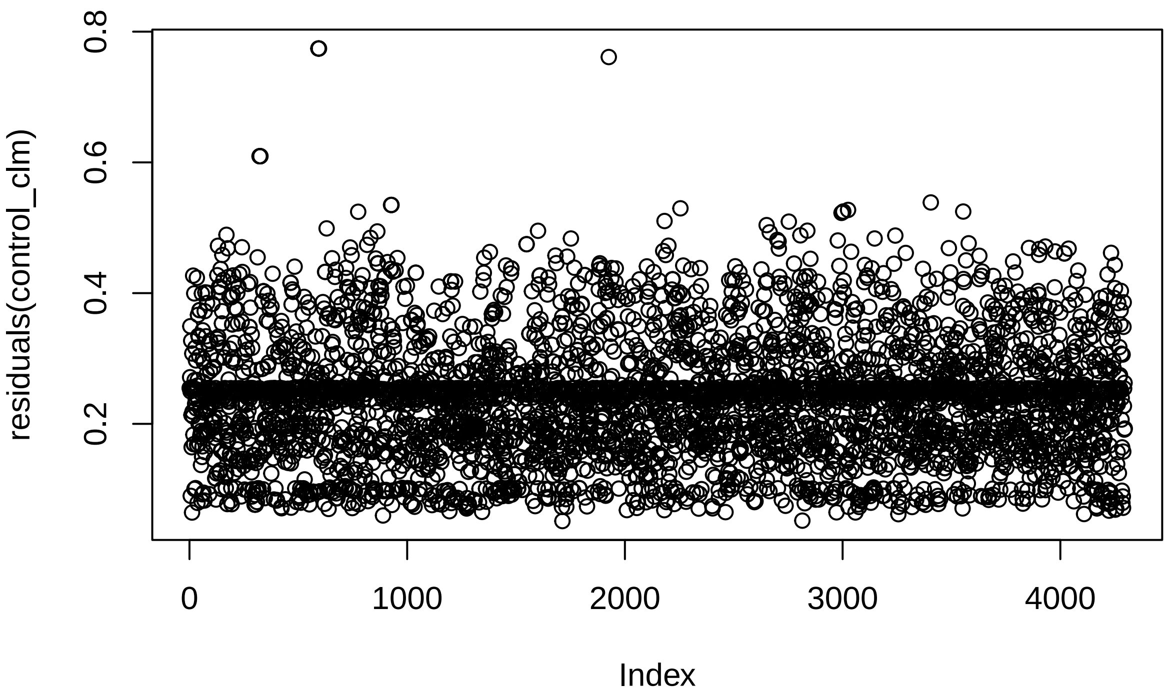
```
## $gof
##
##  Pearson's Chi-squared test
##
## data:  control_results$limit and control_results$fit
## X-squared = 145.72, df = 4, p-value < 2.2e-16
```

```
## Use VGAM to get prob for each level of resp, not implemented in Ordinal
## Similarly inaccurate model
control_vglm_sig <- vglm(limit ~ 1 + age_std + gender + race + edu + city +
                           caff_std + num_kids + caloriescal_std,
                     data = reduced_data,
                     family = cumulative(parallel = TRUE))
model_accuracy(control_vglm_sig, adj = T)
```

```
## $control_results
## # A tibble: 4,296 x 27
##    receiptid person_id limit       age age_std gender race  edu     city    caff
##    <fct>     <fct>     <ord>     <dbl>   <dbl> <chr>  <chr> <chr>   <chr>  <dbl>
##  1 B103-098  900563    Sometimes    30 -0.670  F      Black Associ~ New ~     0
##  2 B103-022  900076    Sometimes    22 -1.22   F      Other Some C~ New ~     0
##  3 B103-091  900582    Never        40  0.0172 F      White High S~ New ~ 1137.
##  4 B103-081  900569    Never        20 -1.36   F      Other Some H~ New ~     0
##  5 B103-080  900568    Sometimes    34 -0.395  M      Other High S~ New ~     0
##  6 B103-090  900578    Seldom       19 -1.43   F      Other High S~ New ~  101.
##  7 B103-024  900078    Sometimes    61  1.46   M      Black Some C~ New ~   75.8
##  8 B103-067  900021    Always       51  0.773  F      Black Colleg~ New ~     0
##  9 B103-086  900574    Never        58  1.25   F      Black Some C~ New ~     0
## 10 B103-023  900077    Never        50  0.704  F      Black Colleg~ New ~     0
## # i 4,286 more rows
## # i 17 more variables: location <fct>, round <fct>, nsigns_ssb <dbl>,
## #   num_kids <chr>, surveydate <date>, days_since_ban <dbl>, caloriescal <dbl>,
## #   fatg <dbl>, sugarg <dbl>, black <chr>, caff_std <dbl>,
## #   nsigns_ssb_std <dbl>, days_since_ban_std <dbl>, caloriescal_std <dbl>,
## #   fatg_std <dbl>, sugarg_std <dbl>, fit <ord>
##
## $conf
##            Truth
## Prediction  Never Seldom Sometimes Often Always
##   Never       587    150       425   277    264
##   Seldom       25      7        27     9     18
##   Sometimes   273     67       217   152    192
##   Often       111     39       131    98    136
##   Always      277    114       254   159    287
##
## $acc
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy multiclass     0.278
##
## $sen
## # A tibble: 1 x 3
##   .metric     .estimator .estimate
##   <chr>       <chr>          <dbl>
```

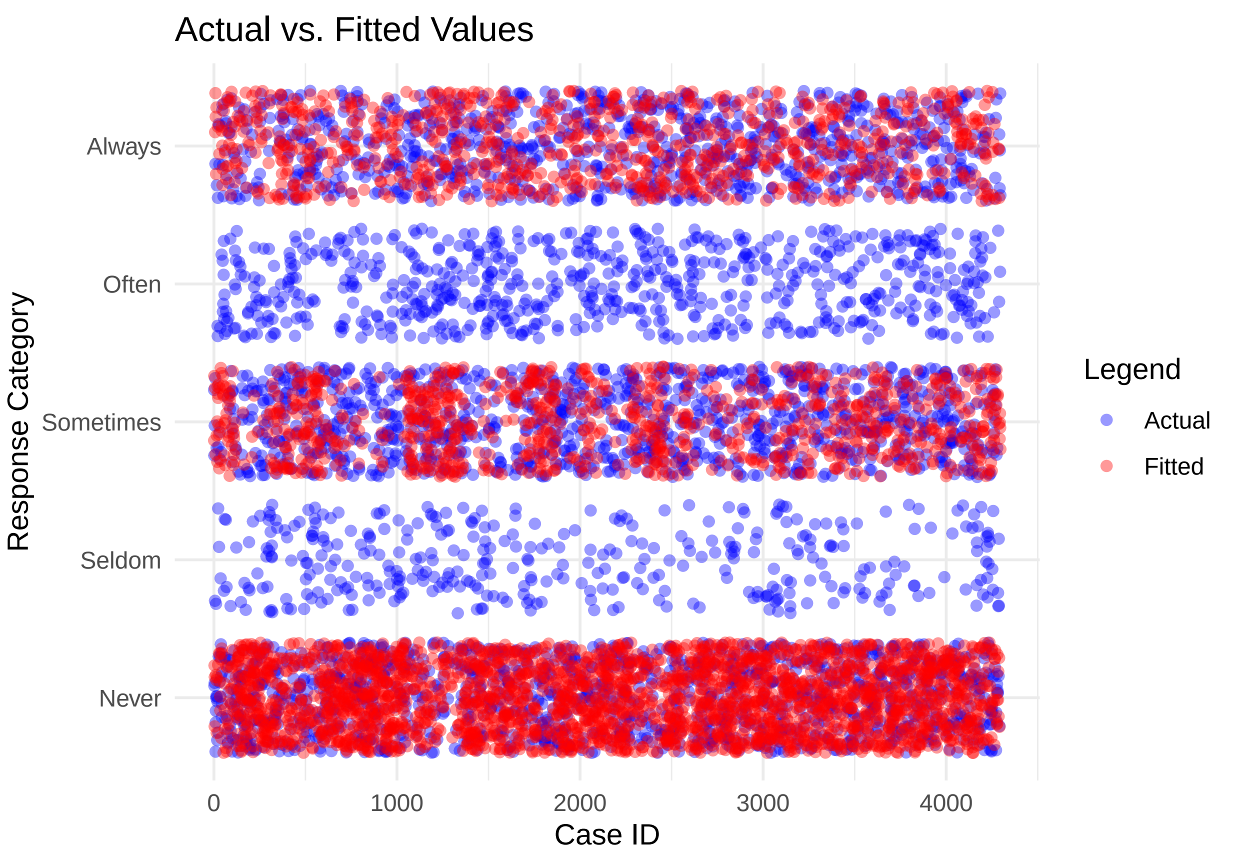
```
## 1 sensitivity macro          0.229
##
## $spe
## # A tibble: 1 x 3
##   .metric     .estimator .estimate
##   <chr>       <chr>          <dbl>
## 1 specificity macro          0.809
##
## $gof
##
##  Pearson's Chi-squared test
##
## data:  control_results$limit and control_results$fit
## X-squared = 93.413, df = 16, p-value = 5.856e-13
```

```
plot(fitted(control_clm), residuals(control_clm))
```

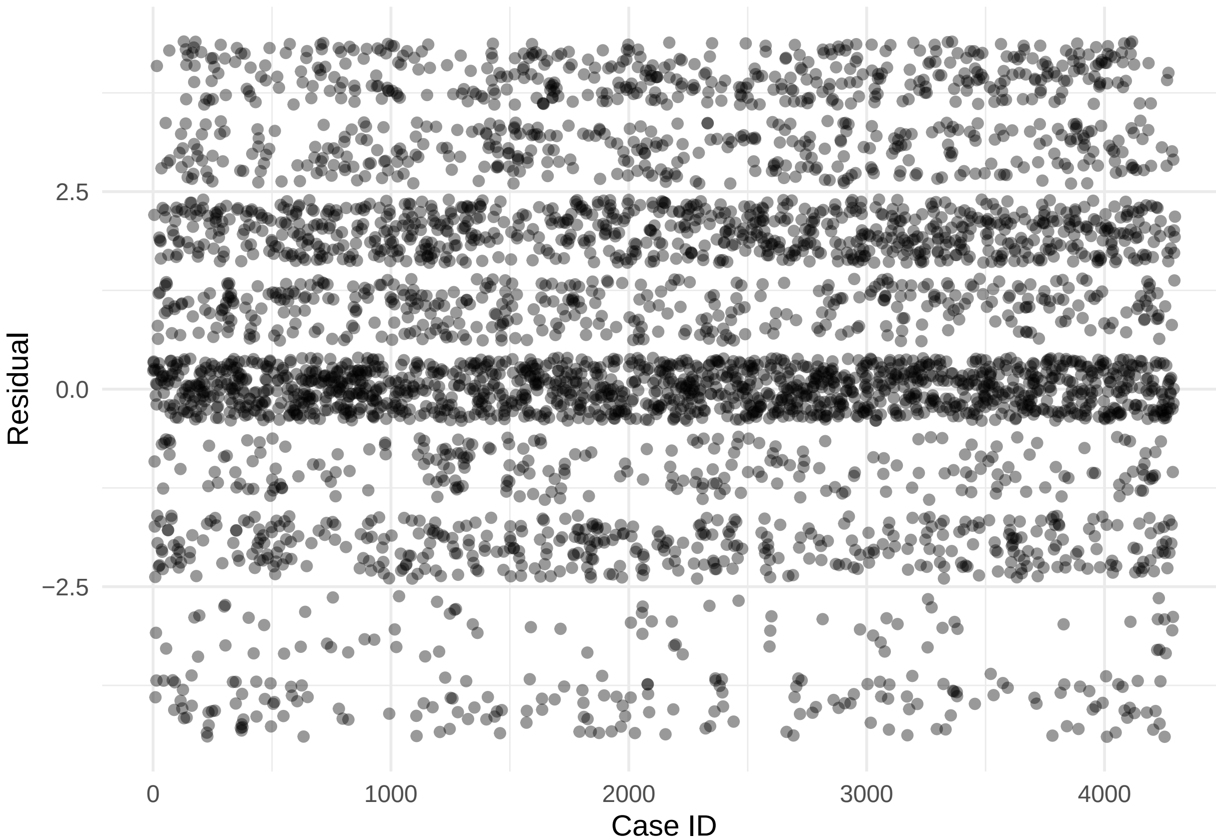



```
control_resid <- model_accuracy(control_clm_nom)$control_results %>%
  mutate(
    case_id = row_number(),
    fit = ordered(fit, levels = c("Never", "Seldom", "Sometimes", "Often", "Always"))
  ) %>%
  select(case_id, limit, fit) %>%
  mutate(
    limit_num = case_when(
      limit == "Never" ~ 0,
      limit == "Seldom" ~ 1,
      limit == "Sometimes" ~ 2,
      limit == "Often" ~ 3,
      limit == "Always" ~ 4),
    fit_num = case_when(
      fit == "Never" ~ 0,
      fit == "Seldom" ~ 1,
```

```r
      fit == "Sometimes" ~ 2,
      fit == "Often" ~ 3,
      fit == "Always" ~ 4),
    ) %>%
  mutate(resid = limit_num - fit_num)

ggplot(control_resid, aes(x = case_id)) +
  geom_jitter(aes(y = limit, color = "Actual"), alpha = 0.4) +
  geom_jitter(aes(y = fit, color = "Fitted"), alpha = 0.4) +
  scale_color_manual(values = c("Actual" = "blue", "Fitted" = "red")) +
  labs(
    x = "Case ID",
    y = "Response Category",
    title = "Actual vs. Fitted Values",
    color = "Legend"
  ) +
  theme_minimal()
```




```r
ggplot(control_resid, aes(x = case_id, y = resid)) +
  geom_jitter(alpha = 0.4) +
  labs(
    x = "Case ID",
    y = "Residual"
  ) +
  theme_minimal()
```



Effects Interpretation

Confidence Intervals

```r
confint(control_clmm_loc) %>% kable(digits = 3)
```

	2.5 %	97.5 %
Never|Seldom	-1.496	-0.854
Seldom|Sometimes	-1.085	-0.445
Sometimes|Often	-0.036	0.601
Often|Always	0.793	1.434
age_std	0.168	0.284
genderM	-0.401	-0.179
raceBlack	-0.416	0.048
raceNative	-0.671	0.184
raceOther	-0.201	0.295
raceWhite	-0.277	0.204
eduCollege Degree	0.028	0.522
eduGraduate Degree	-0.032	0.541
eduHigh School	-0.542	-0.065
eduLess than High School	-0.572	0.222
eduSome College	-0.309	0.181
eduSome High School	-0.698	-0.126
cityNew York	0.007	0.290
caff_std	-0.105	0.037
num_kids1	-0.168	0.286

	2.5 %	97.5 %
num_kids2	-0.741	0.069
num_kids3+	-2.514	-0.398
caloriescal_std	-0.011	0.131

```r
exp(confint(control_clmm_loc)) %>% kable(digits = 3)
```

	2.5 %	97.5 %
Never|Seldom	0.224	0.426
Seldom|Sometimes	0.338	0.641
Sometimes|Often	0.964	1.825
Often|Always	2.209	4.197
age_std	1.183	1.329
genderM	0.670	0.836
raceBlack	0.660	1.050
raceNative	0.511	1.202
raceOther	0.818	1.344
raceWhite	0.758	1.227
eduCollege Degree	1.028	1.685
eduGraduate Degree	0.969	1.717
eduHigh School	0.582	0.937
eduLess than High School	0.564	1.249
eduSome College	0.734	1.198
eduSome High School	0.498	0.882
cityNew York	1.007	1.336
caff_std	0.900	1.037
num_kids1	0.845	1.331
num_kids2	0.477	1.071
num_kids3+	0.081	0.671
caloriescal_std	0.989	1.139

```r
(100*(exp(confint(control_clmm_loc))-1)) %>% kable(digits = 3)
```

	2.5 %	97.5 %
Never|Seldom	-77.603	-57.448
Seldom|Sometimes	-66.206	-35.942
Sometimes|Often	-3.553	82.454
Often|Always	120.922	319.714
age_std	18.299	32.868
genderM	-33.027	-16.382
raceBlack	-34.029	4.963
raceNative	-48.902	20.151
raceOther	-18.217	34.352
raceWhite	-24.185	22.657
eduCollege Degree	2.815	68.498
eduGraduate Degree	-3.122	71.700
eduHigh School	-41.827	-6.293
eduLess than High School	-43.588	24.885
eduSome College	-26.604	19.850
eduSome High School	-50.241	-11.819

	2.5 %	97.5 %
cityNew York	0.744	33.580
caff_std	-10.012	3.723
num_kids1	-15.490	33.077
num_kids2	-52.318	7.119
num_kids3+	-91.906	-32.857
caloriescal_std	-1.123	13.950