Exploratory Data Analysis

Ana Morales

12 September 2019

Practical 1: Explory data analysis of ‘tidy data’

In this practical you will be analysing a sample of the teaching dataset of the Health Survey for England available here (https://beta.ukdataservice.ac.uk/datacatalogue/studies/study?id=6765) from the UK Data Service (https://www.ukdataservice.ac.uk/). But you can access it from the folder **NCRM_UKDS_workshop**.

As a first step in a research project we need to start with a purpose:

Ask an interesting question

Here our purpose, defined through questions is to understand whether there are differences in BMI measures by different groups. For a definition of BMI follow this link (https://www.nhs.uk/common-health-questions/lifestyle/what-is-the-body-mass-index-bmi/)

1. Getting started

Define a working directory

You can use the directory as defined below on University-owned machines, as it provides a stable drive to work on. If you’re using your own machine, you can specify any folder you find convenient.

Remember to Copy the folder **NCRM_UKDS_workshop** into the folder you will define as working directory, as this makes matters easier, you can use “C:Work”.

Also note that you need to take your data and results with you and delete what you store on the C-drive after this session (if you are not using your own laptop):

Open a new Script

First, open a new script in R studio and save it in your working directory, so you will be able to access this script at a later time if you want to revise or modify a code. In R Studio:

Go to File… New File…. R script

Load the packages

Remember to load the packages. We will be using `tidyverse` and `haven`

```
library(haven)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------- tidyverse 1.2.1 --
```

```
## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   0.8.3     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

```
## -- Conflicts ------------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

2. Import the data into R

Import the dataset called hse_data.dta into the console using your desired method (code or point and click). This dataset can only be imported if you have loaded the package `haven` .

```
hse_data<-read_dta("hse_data.dta")
```

3. Data exploration

3.1 Dataset exploration

Task 1: Inspect the dataset, use the function View()

Once we have the data, we always need an initial inspection of the variables.

```
View(hse_data)
```

Task 1.1 How many observations does the data have?

Task 1.2 How many variables?

Another function used to get a glimpse at your data is using the function `glimpse()` .

```
glimpse(hse_data)
```

```
## Observations: 36,389
## Variables: 14
## $ sha      <chr> "Q27", "Q27", "Q27", "Q27", "Q27", "Q27", "Q27", "Q27...
## $ persno   <dbl+lbl> 1, 2, 1, 4, 1, 3, 2, 1, 1, 3, 2, 2, 1, 1, 1, 4, 1...
## $ sex      <dbl+lbl> 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0...
## $ age      <dbl> 55, 57, 21, 6, 24, 10, 28, 59, 41, 8, 50, 16, 24, 58,...
## $ topqual3 <dbl+lbl> 7, 7, 3, NA, 4, NA, 7, 7, 4, NA, 2, 7, 3, 3, 3, N...
## $ nssec5   <dbl+lbl> 5, 4, 1, NA, 4, NA, 4, 5, 5, NA, 2, 99, 1, 1, 1, ...
## $ ethnic   <dbl+lbl> 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ marital  <dbl+lbl> 2, 2, 1, NA, 1, NA, 1, 2, 3, NA, 2, 1, 1, 4, 2, N...
## $ limitill <dbl+lbl> 1, 1, 3, 3, 1, 2, 3, 1, 2, 3, 3, 3, 3, 3, 2, 3, 2...
## $ height   <dbl> 163.1, 174.9, 167.1, 121.2, 180.3, 157.7, 177.4, 145....
## $ weight   <dbl> 85.6, 119.5, 66.9, 26.8, 118.3, 52.9, NA, 59.5, 77.0,...
## $ bmival   <dbl> 32.17850, 39.06504, 23.95925, 18.24440, 36.39094, 21....
## $ hhdtypb  <dbl+lbl> 2, 2, 1, 3, 3, 3, 3, 6, 3, 3, 5, 3, 1, 2, 3, 3, 6...
## $ car      <dbl+lbl> 1, 1, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 2, 1, 1, 1, 1...
```

Note the data type of almost all variables:

These are labelled variables in the haven package. It's the way to identify variables with labels from Stata.

3.2 Variable exploration

Let's use another function to have a closer look at the variable sex: `attributes`

```
attributes(hse_data$sex)
```

```
## $label
## [1] "sex"
##
## $format.stata
## [1] "%8.0g"
##
## $class
## [1] "haven_labelled"
##
## $labels
##   Men Women
##     0     1
```

We have the `$class` attribute which says that `sex` is a haven_labelled type. We also have the `$labels` attribute which are the Value Labels of the variable `sex` : 0 Man, 1 Woman

We could have also used `class(hse$sex)` to check the type of the variable.

Task 2 Inspect other variables with attributes() and class()

3.3 Understanding our data

Now that we have had a look at what variables we have, we need to start looking for other aspects of the exploratory data analysis (EDA) process. These are some of the things to look out for:

- Checking for outliers (unusual values)
- Looking at the distribution of the variables
- Exploring some relationships and patterns
- Checking for missing cases

Here we start the data exploration of the categorical variables (also called factors).

Let's see what happens when we ask for a frequency table

```
table(hse_data$sex)
```

```
##
##     0     1
## 16566 19823
```

Right now, the variable does not contain any visible label that help us to identify each category of response, although we know the underlying values of each label thanks to the `attribute` function.

This is one of the inconveniences of working with data with labels in R, but it is good for you to know what to do in this case.

Although we can use the data as it is, I prefer to see the values of the variables, so we are going to change the class of some varibles to make them easier to analyse.

We can use the function `as_factor()` from the package **haven** to convert the variable into a factor variable (equivalent to categorical variable). Copy the following code in your console.

```
hse_data$sexf<-as_factor(hse_data$sex)
table(hse_data$sexf)
```

```
##
##   Men Women
## 16566 19823
```

Here we created a new variable with the suffix “f” (for factor).

Task 3: Convert the following variables into factor using the example code before.

- topqual3,
- nssec5,
- ethnic,
- marital,
- limitill,
- car,
- hhdtypb

hint: You can create a new variable (recode as a new variable) as I did with ‘sexf’ or you can rewrite the old variable: `hse$sex<-as_factor(hse$sex)` , remember that if you overwrite a variable you can not get it back, though.

```
# solution
hse_data$topqual3f <- as_factor(hse_data$topqual3)
hse_data$nssec5f <- as_factor(hse_data$nssec5)
hse_data$ethnicf <- as_factor(hse_data$ethnic)
hse_data$maritalf <- as_factor(hse_data$marital)
hse_data$limitillf <- as_factor(hse_data$limitill)
hse_data$carf <- as_factor(hse_data$car)
hse_data$hhdtypb <- as_factor(hse_data$hhdtypb)
```

Univariate analysis

We can use a series of descriptive statistics and graphs to help us to undertand and make sense of the data.

Univariate exploration is when we look at the characteristics of each variable of interest independently

Univariate Descriptive statistic for continuous (numeric) data

```
#This uses basic functions from R
summary(hse_data$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   21.00   40.00   40.34   58.00   99.00
```

```
sd(hse_data$age)
```

```
## [1] 23.27309
```

```
quantile(hse_data$age)
```

```
##   0%  25%  50%  75% 100%
##    0   21   40   58   99
```

```
# Subsetting observations -Getting statistics for a sample of the data
sum(hse_data$age[hse_data$age<5])
```

```
## [1] 4354
```

```
#using function from the package dplyr (tidyverse)
hse_data%>%
  summarise(mean= mean(age),
            sd= sd(age),
            median = median(age))
```

mean	sd	median
<dbl>	<dbl>	<dbl>
40.33615	23.27309	40

1 row

```
# Subsetting observations -Getting statistics for a sample of the data
hse_data %>%
  filter(age <5) %>%
  summarise(younger_5 = sum(age))
```

younger_5
<dbl>
4354

1 row

The distribution of the variable age seems reasonable, the mean and the median are very similar.

The standard deviation shows how spread the data are. A standard deviation of 23 is quite high, which might indicate a lot of variation in the distribution of age.

The maximum value for age is 99, which might be considered an unusual value (outlier). We need more visual exploration to check this.

There are a lot of small children children in this dataset (4354 younger than 5), we saw this using the `table(hse_data$age[hse_data$age<5])` which gives the total number of children younger than 5.

Task 4: Get summary statistics for the variables height, BMI and weight. Use the subsetting operation for exploring certain groups.

Task 4.1 Write down a short description of the main statistics you've obtained

Task 4.2 Is there any unusual value in the variables you have analysed?

Univariate Descriptive statistic for categorica data

We can look at each variable independently. Make sure to see if there is any unusual value

```
table(hse_data$sexf)
```

```
##
##   Men Women
## 16566 19823
```

```
table(hse_data$maritalf)
```

```
##
##       single, that is never married    married and living with husbwife
##                                7581                               15823
## married and separated from husbwife                             divorced
##                                 762                                2473
##                             widowed
##                                2521
```

**Missing values*

In R missing values are identified as NA.

Now inspect whether there are missing values in our variables

```
#is.na(hse_data$sexf)

sum(is.na(hse_data$sexf))
```

```
## [1] 0
```

```
sum(is.na(hse_data$maritalf))
```

```
## [1] 7229
```

The function `is.na` is a logical function which looks at each observation and evaluates whether it is a valid case or a missing case. It will show a series of values `TRUE` for those observation that are missing and `FALSE` for valid cases (is.na = FALSE means that the cases are valid).

It is a good function to explore missingness, but it is not very practical on its own (as you could see in the example) that is why it is often used along other statistics.

I used the statistc `sum` which will add all the `TRUE` values of `is.na` so the result will be the number of NA (missing cases) for a particular variable.

You can see that there are no missing cases in the variable sex, but some cases are missing for the variable marital status.

Task 5: Now inspect the variables ethnicf and topqual3f and report you exploration. Look at unusual values and missing observations

Task 5.1 Describe the variables. Did you fing any unusual pattern?

Task 5.2 Are there missing values in any of the variables? How many?

Looking at relationship

More insights can be gained when looking at relationships between variables.

For categorical data we can use contingency tables (also called crosstabs)

```
table(hse_data$sexf, hse_data$maritalf)
```

```
##
##         single, that is never married married and living with husbwife
##   Men                            3705                             7501
##   Women                          3876                             8322
##
##         married and separated from husbwife divorced widowed
##   Men                                  269      919     539
##   Women                                493     1554    1982
```

```
# we can get a proportion table (contigency table)

prop.table(table(hse_data$sexf, hse_data$maritalf), 1)# % of rows
```

```
##
##         single, that is never married married and living with husbwife
##   Men                      0.28647646                       0.57998917
##   Women                    0.23886116                       0.51284896
##
##         married and separated from husbwife   divorced    widowed
##   Men                            0.02079951 0.07105853 0.04167633
##   Women                          0.03038146 0.09576632 0.12214211
```

```
prop.table(table(hse_data$sexf, hse_data$maritalf), 2)# % of columns
```

```
##
##         single, that is never married married and living with husbwife
##   Men                       0.4887218                        0.4740568
##   Women                     0.5112782                        0.5259432
##
##         married and separated from husbwife  divorced   widowed
##   Men                             0.3530184 0.3716134 0.2138040
##   Women                           0.6469816 0.6283866 0.7861960
```

```
# For percentage, we multipliy by 100
prop.table(table(hse_data$sexf, hse_data$maritalf), 1)*100# % of rows
```

```
##
##         single, that is never married married and living with husbwife
##   Men                       28.647646                        57.998917
##   Women                     23.886116                        51.284896
##
##         married and separated from husbwife  divorced   widowed
##   Men                              2.079951  7.105853  4.167633
##   Women                            3.038146  9.576632 12.214211
```

```
prop.table(table(hse_data$sexf, hse_data$maritalf), 2)*100# % of columns
```

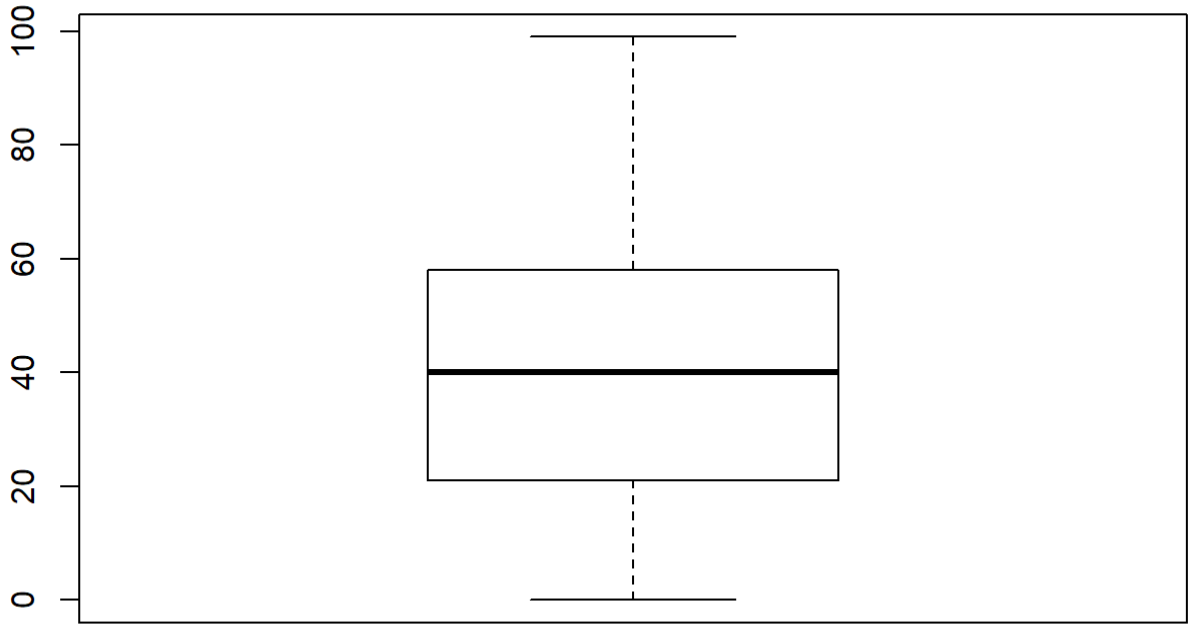
```
##
##         single, that is never married married and living with husbwife
##   Men                        48.87218                         47.40568
##   Women                      51.12782                         52.59432
##
##         married and separated from husbwife divorced  widowed
##   Men                              35.30184 37.16134 21.38040
##   Women                            64.69816 62.83866 78.61960
```

Task 6: Explore the relationship between sex and ethnic group, marital status and ethnic group

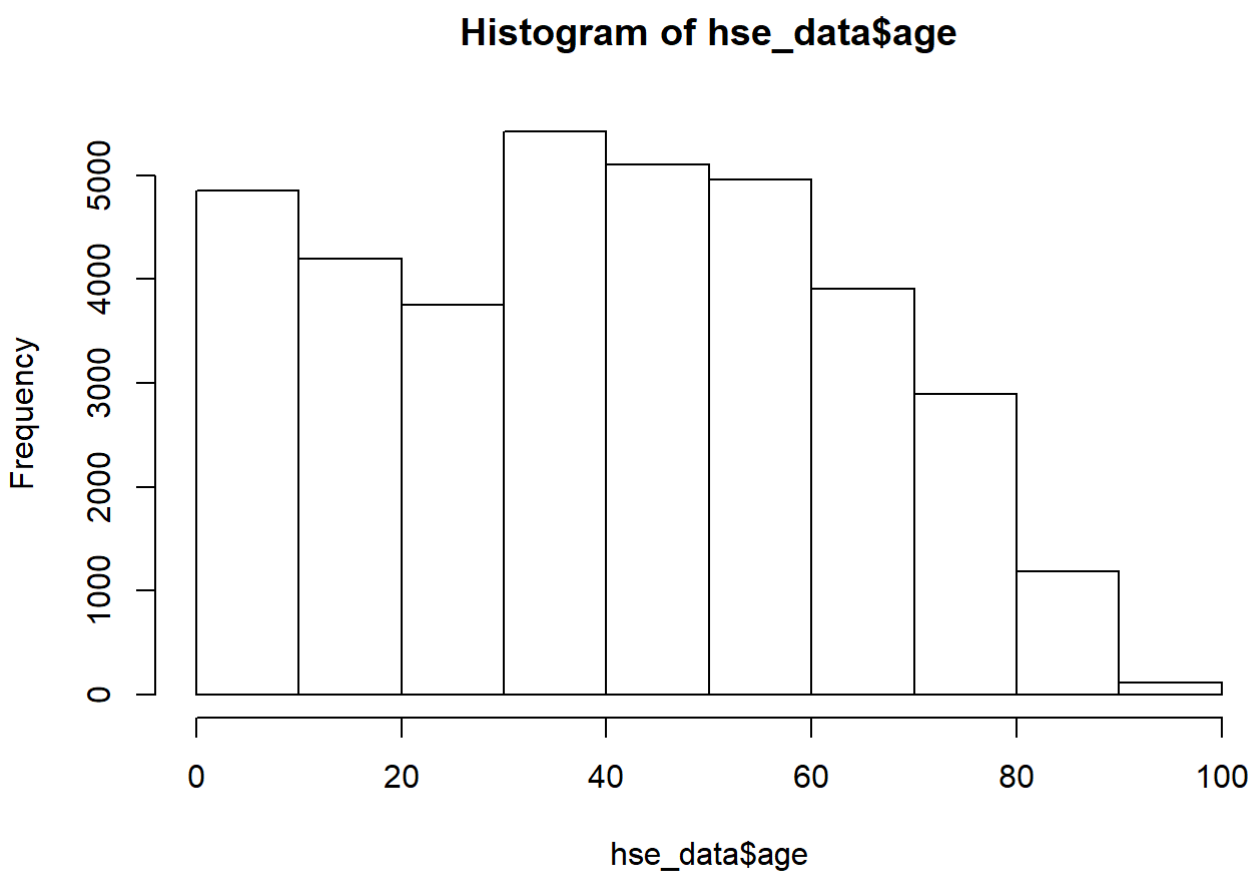
Visualisations

We can use a combination of histograms, boxplots and scatterplots as the basis for our data exploration

```
boxplot(hse_data$age)
```




```
hist(hse_data$age, breaks = 10) # use the option breaks to explore the distribution, by incre
asing the number of bars. Use this to get more details of the distribution
```



Boxplot and histograms are really useful to detect outliers and unusual values in continuous variables.

Now, we don't see any unusual value in the variable age

Task 7: Visually explore the distribution of BMIVAI, Height and Weight

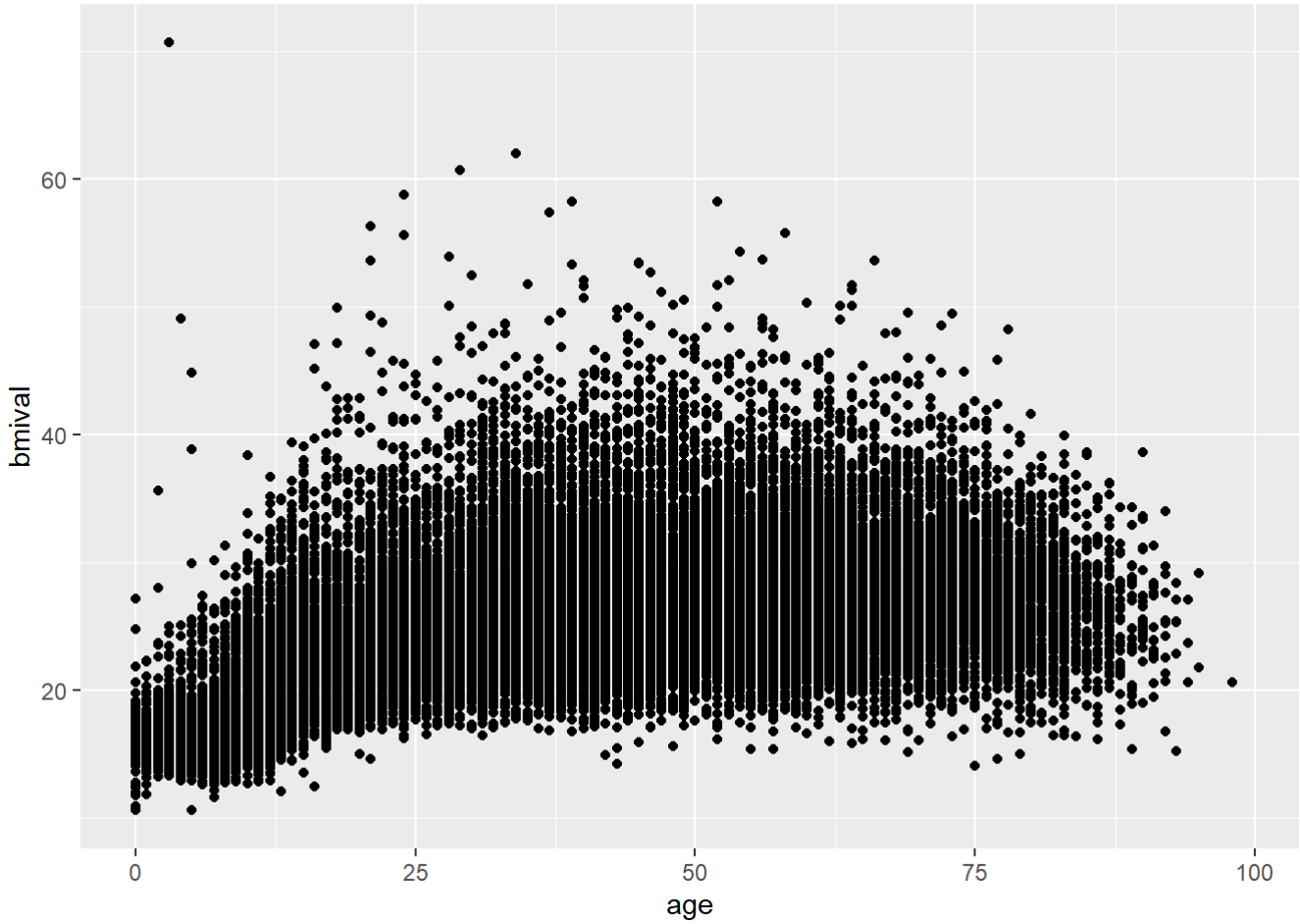
Now we need to focus on our variable of interest (outcome usually represented as the 'y' variable) and its association with other explanatory variables (x variables).

Scatter plots: For looking at the relationship between two continous variables

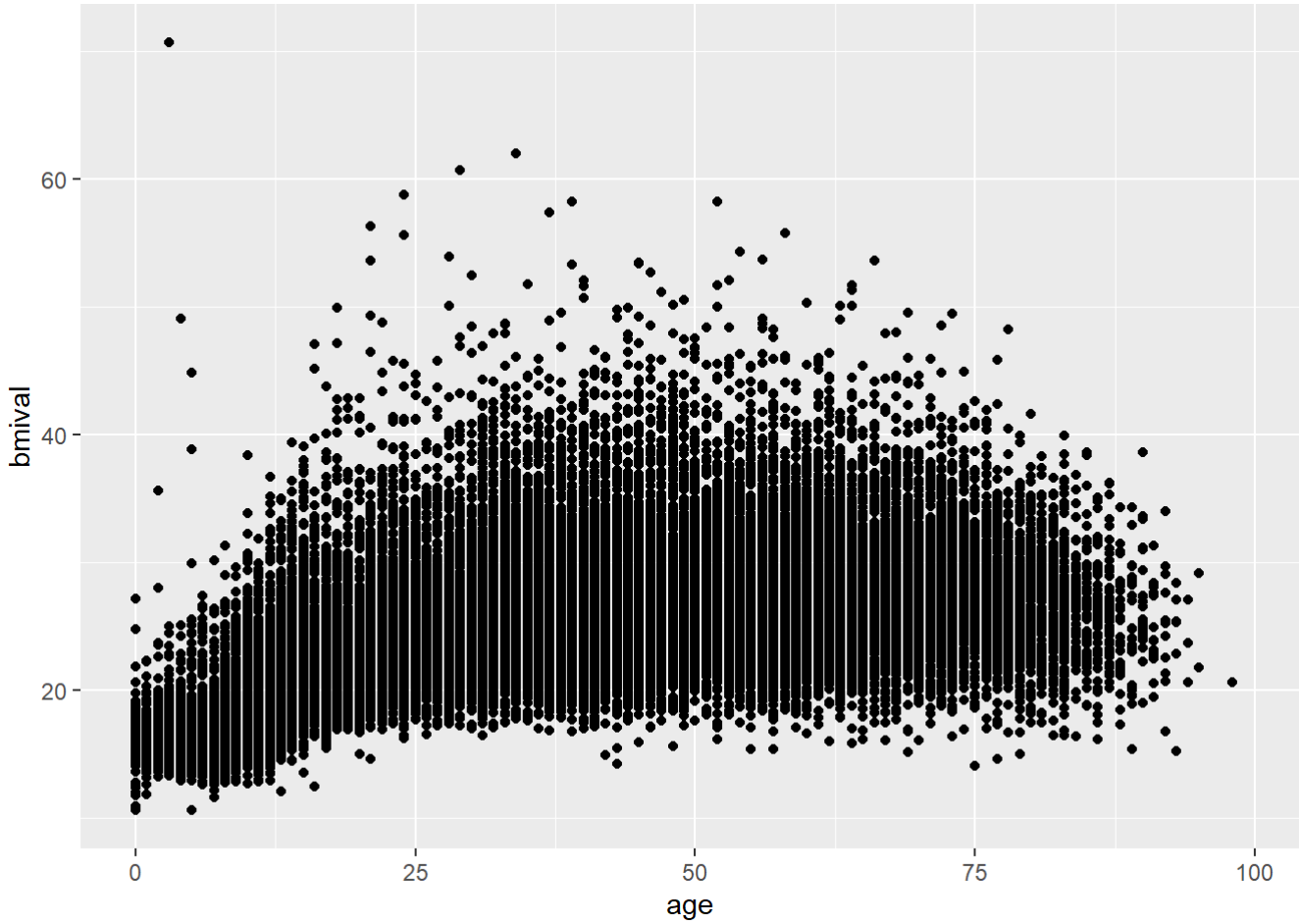
This is usuful to assess co-variation

```
#Relationship of two continous variables

ggplot(data = hse_data, mapping = aes(x = age, y = bmival)) +
  geom_point()
```

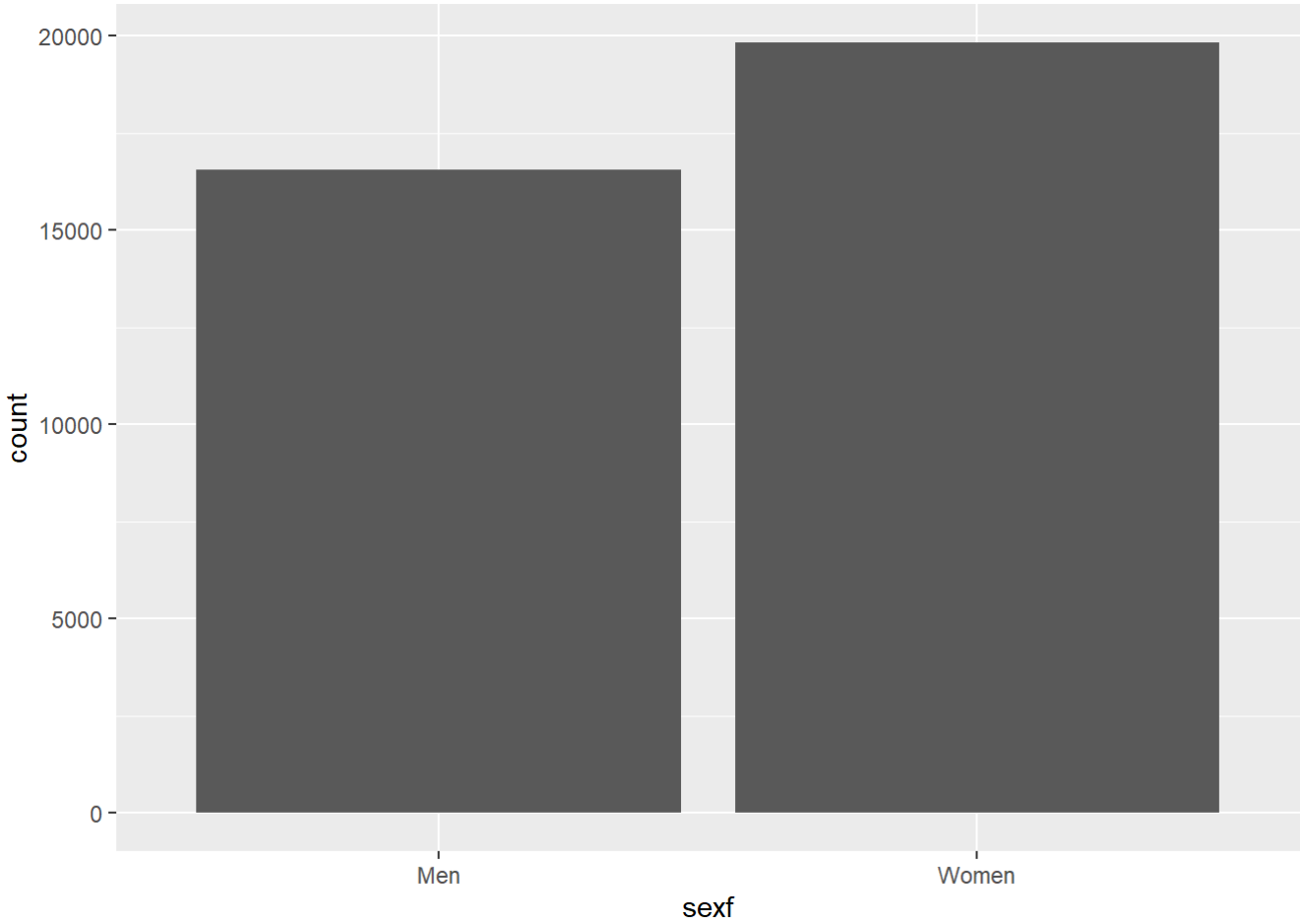
```
## Warning: Removed 5765 rows containing missing values (geom_point).
```




```
#
# To suppress that warning, set na.rm = TRUE: (remove NAs =TRUE)
ggplot(data = hse_data, mapping = aes(x = age, y = bmival)) +
  geom_point(na.rm = TRUE)
```



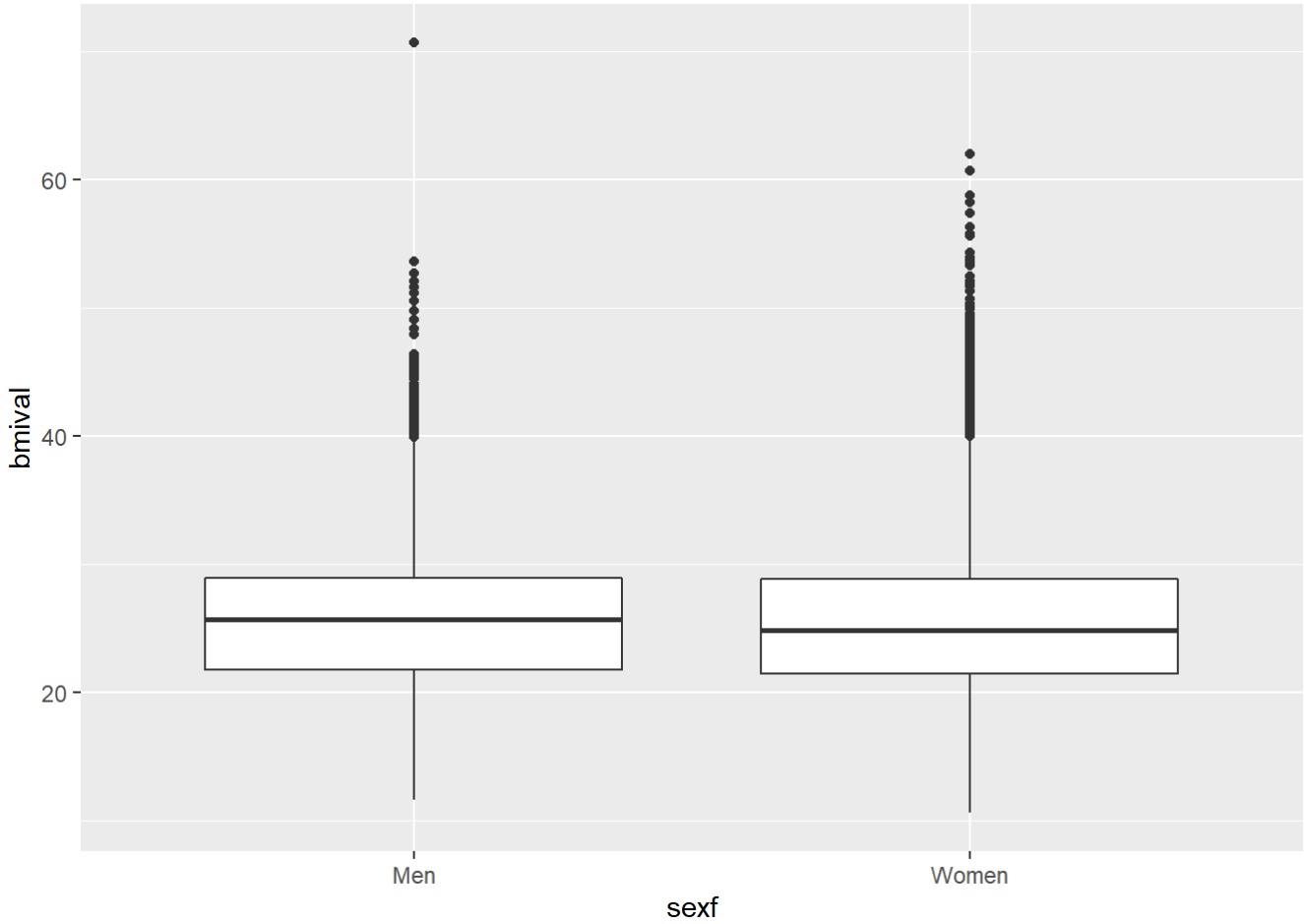

```
# bar plot of categorical data
ggplot(data = hse_data) +
  geom_bar(mapping = aes(x = sexf))
```



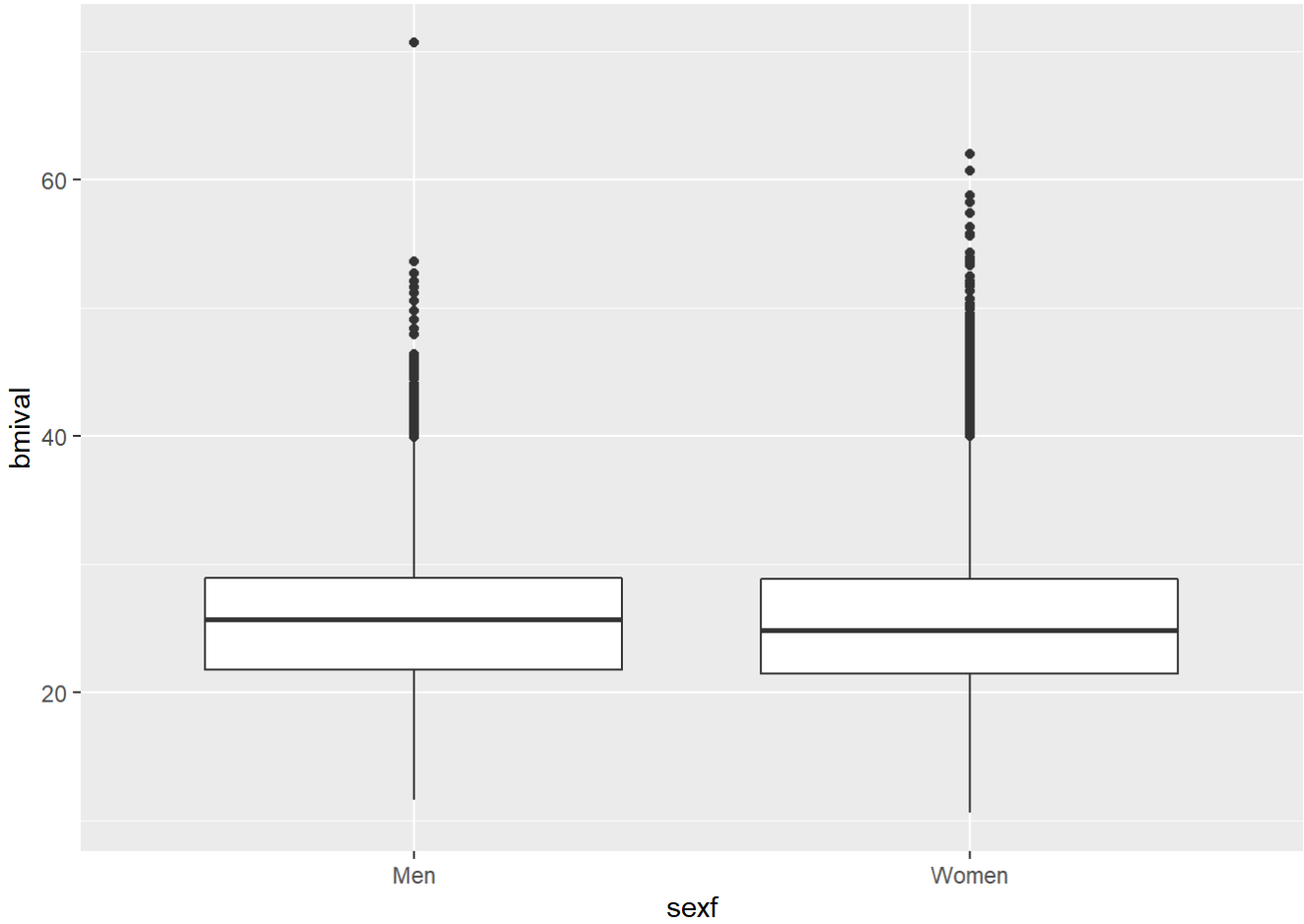
Boxplot for detecting outliers. Here we obtain boxplots by groups

```
# boxplot of categorical and continous data
ggplot(data = hse_data, mapping = aes(x = sexf, y = bmival)) +
  geom_boxplot()
```

```
## Warning: Removed 5765 rows containing non-finite values (stat_boxplot).
```




```
# remove NAs

ggplot(data = hse_data, mapping = aes(x = sexf, y = bmival)) +
  geom_boxplot(na.rm = TRUE)
```

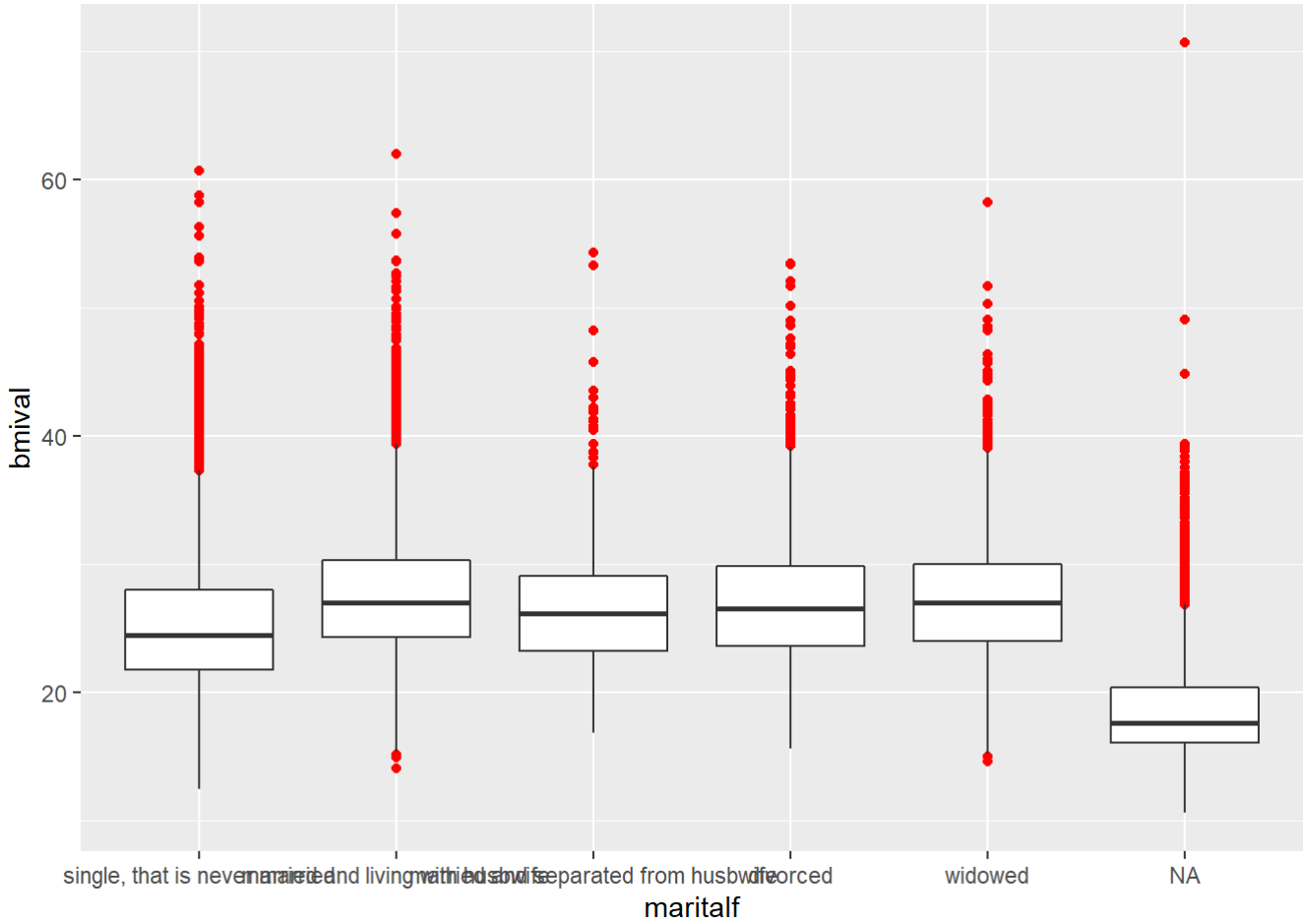


Now we can see a lot of unusual values in the variable bmival when analysing by different sexes.

Task 8 Explore the association between bmival and marital status and ethnic group

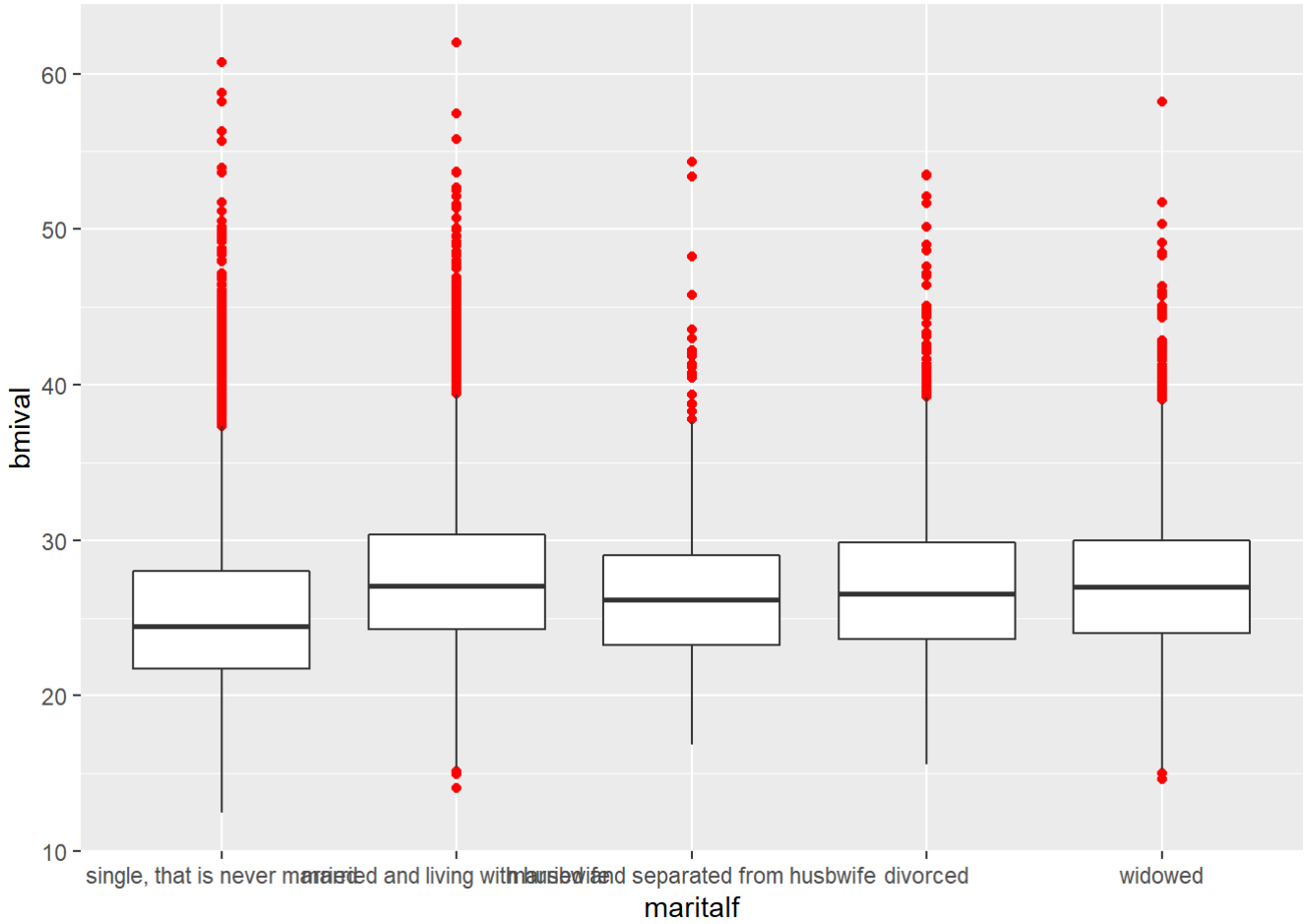
```
#solutions
# base package

boxplot(hse_data$bmival ~ hse_data$maritalf)
```

```
#marital status

#this code still plots the missing cases
ggplot(data = hse_data, mapping = aes(x = maritalf, y = bmival)) +
  geom_boxplot(outlier.colour = "red", na.rm = TRUE)
```

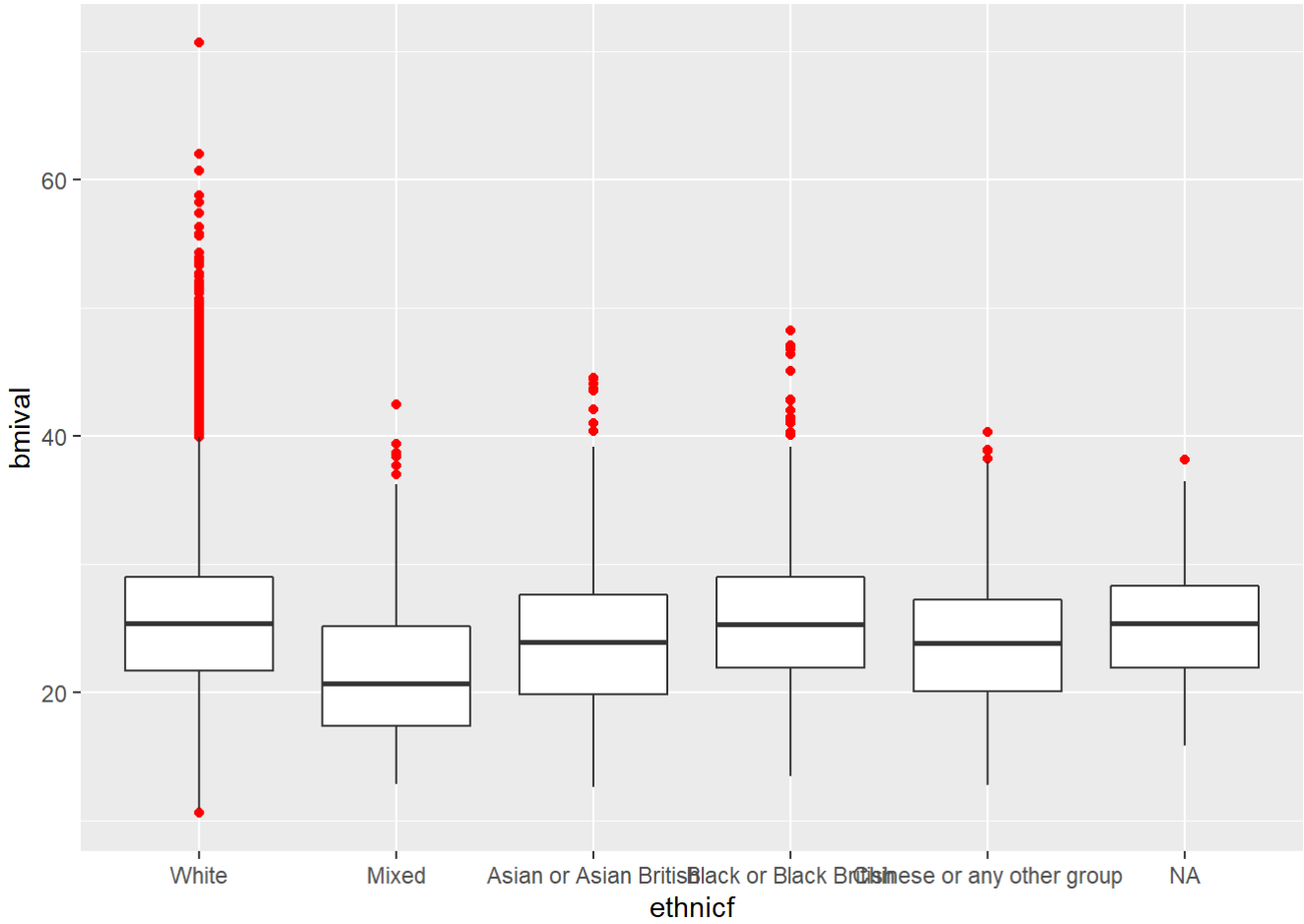



```
# solution: plot only valid cases for maritalf, using filter to include only those observatio
n that are not missing (the symbol'!' means 'is not')

 hse_data%>%
   filter(!is.na(maritalf)) %>% # pick maritalf when not missing
ggplot(mapping = aes(x = maritalf, y = bmival)) +
  geom_boxplot(outlier.colour = "red")
```

```
## Warning: Removed 4188 rows containing non-finite values (stat_boxplot).
```




```
# ethnic group

ggplot(data = hse_data, mapping = aes(x = ethnicf, y = bmival)) +
  geom_boxplot(outlier.colour = "red", na.rm = TRUE)
```


60
40
20
bmival
White
Mixed
Asian or Asian Britis
Black or Black Brit
Chinese or any other group
NA
ethnicf