Cardiac EDA

Michiel Noback (NOMI)

11/05/2020

Assignment Introduction

The purpose of this assignment is to train you in performing reproducible research. The remainder of this document logs an exploratory data analysis. It is your task to study this EDA and describe weaknesses with respect to reproducibility.

Reproducibility is guaranteed by the following aspects of an analysis:

1. There is a log of all steps of the analysis
2. The log is in English and is easy to read (spelling/grammar/formulations)
3. The data are described in detail. It is known
   - where they came from and how they were collected
   - what the variables are and what abbreviations mean
   - what the data types of the variables are, what values they may have, and in what units they were measured
   - what the dependent / class variable is
4. There are no data processing steps missing in the log (or its rendered result!)
5. It is clear what the sequence of processing or analysis steps is (also chronologically), and why they were undertaken in that particular order
6. Every step has
   - an intro: why is it carried out
   - a result: presented in clear tables or figures or other relevant means
   - a conclusion: was the step succesful, are the results as expected, what action should additionaly be done, what questions do arise as a result etc.

Here follows the actual EDA. It is written entirely using the *tidyverse* packages as well as some others. Study it in pairs or trios and report flaws, errors, weaknesses and possible soluitons/repairs at the end of the session.

Errors

- No numbered steps
- No case introduction
- No background of the data

- No clear steps of what we are doing
- No clear discription of the figures and tables
- No conclusion at every step
- own questions diddn't get answerd

Error fixes

The biggest issue with this journal is the lack of background information and the absent of detail by the few steps. To make it better understandeble, there must be a good and detailed background of the subject and data be added. Also, there must be added more steps with more detail (intro, result and conclusion). The figures and tables seems meaningless now, without a clear description of the findings. there must be a broad and detailed dicription by every figure/table.

with a little re-arrangement of the steps and introducing a few sub steps. The journal is better readeble and reproducible.

-

EDA of cardiac data: dobutamine efficacy

setop and load libraries:

Load the codebook

```
codebook <- read_delim(file = "codebook.txt",
                       delim = ";")

knitr::kable(codebook)
```

column	description
bhr	BASAL HEART RATE
basebp	BASAL BLOOD PRESSURE
basedp	BASAL DOUBLE PRODUCT (= bhr x basebp)
pkhr	PEAK HEART RATE
sbp	SYSTOLIC BLOOD PRESSURE
dp	DOUBLE PRODUCT (= pkhr x sbp)
dose	DOSE OF DOBUTAMINE GIVEN
maxhr	MAXIMUM HEART RATE
%mphr(b	% OF MAXIMUM PREDICTED HEART RATE ACHIEVED BY PATIENT
mbp	MAXIMUM BLOOD PRESSURE
dpmaxdo	DOUBLE PRODUCT ON MAXIMUM DOBUTAMINE DOSE
dobdose	DOBUTAMINE DOSE AT WHICH MAXIMUM DOUBLE PRODUCT OCCURED
age	PATIENT AGE
gender	PATIENT GENDER (male = 0)
baseEF	BASELINE CARDIAC EJECTION FRACTION (a measure of the hearts pumping efficiency)
dobEF	EJECTION FRACTION ON DOBUTAMINE
chestpain	0 MEANS THE PATIENT EXPERIENCED CHEST PAIN
posECG	SIGNS OF HEART ATTACK ON ECG (0 = yes)
equivecg	ECG IS EQUIVOCAL (0 = yes)

column	description
restwma	CARDIOLOGIST SEES WALL MOTION ANAMOLY ON ECHOCARDIOGRAM (0 = yes)
posSE	STRESS ECHOCARDIOGRAM WAS POSITIVE (0 = yes)
newMI	NEW MYOCARDIAL INFARCTION, OR HEART ATTACK (0 = yes)
newPTCA	RECENT ANGIOPLASTY (0 = yes)
newCABG	RECENT BYPASS SURGERY (0 = yes)
death	THE PATIENT DIED (0 = yes)
hxofHT	PATIENT HAS HISTORY OF HYPERTENSION (0 = yes)
hxofdm	PATIENT HAS HISTORY OF DIABETES (0 = yes)
hxofcig	PATIENT HAS HISTORY OF SMOKING (0 = yes)
hxofMI	PATIENT HAS HISTORY OF HEART ATTACK (0 = yes)
hxofPTCA	PATIENT HAS HISTORY OF ANGIOPLASTY (0 = yes)
hxofCABG	PATIENT HAS HISTORY OF BYPASS SURGERY (0 = yes)
any event	THIS IS THE OUTCOME VARIABLE. IT IS DEFINED AS “death OR newMI OR newPTCA OR newCABG”. IF ANY OF THESE VARIABLES IS POSITIVE (= 0) THEN “ANY EVENT” IS ALSO POSTIVE (= 0).

Load the data.

```
cardiac <- read_csv(file = "cardiac.csv")
spec(cardiac)
```

```
## cols(
##   bhr = col_double(),
##   basebp = col_double(),
##   basedp = col_double(),
##   pkhr = col_double(),
##   sbp = col_double(),
##   dp = col_double(),
##   dose = col_double(),
##   maxhr = col_double(),
##   ‘%mphr(b)‘ = col_double(),
##   mbp = col_double(),
##   dpmaxdo = col_double(),
##   dobdose = col_double(),
##   age = col_double(),
##   gender = col_double(),
##   baseEF = col_double(),
##   dobEF = col_double(),
##   chestpain = col_double(),
##   posECG = col_double(),
##   equivecg = col_double(),
##   restwma = col_double(),
##   posSE = col_double(),
##   newMI = col_double(),
##   newPTCA = col_double(),
##   newCABG = col_double(),
##   death = col_double(),
##   hxofHT = col_double(),
##   hxofdm = col_double(),
##   hxofcig = col_double(),
##   hxofMI = col_double(),
##   hxofPTCA = col_double(),
##   hxofCABG = col_double(),
##   ‘any event‘ = col_double(),
##   phat = col_double(),
##   ‘event(#)‘ = col_double(),
##   mics = col_double(),
##   deltaEF = col_double(),
##   newpkmphr = col_double(),
##   gdpkmphr = col_double(),
##   gdmaxmphr = col_double(),
##   gddpeakdp = col_double(),
##   gdmaxdp = col_double(),
##   hardness = col_double()
## )
```

Let’s drop columns 33-42

```
cardiac <- cardiac %>% select(1:32)
```

The dimensions are supposed to be 32 columns and 558 rows. Check:

```
cardiac <- as_tibble(cardiac)
cardiac
```

```
## # A tibble: 558 x 32
##      bhr basebp basedp  pkhr   sbp    dp  dose maxhr ‘%mphr(b)‘   mbp dpmaxdo
##    <dbl>  <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>      <dbl> <dbl>   <dbl>
##  1    92    103   9476   114    86  9804    40   100         74   121   12100
##  2    62    139   8618   120   158 18960    40   120         82   158   18960
##  3    62    139   8618   120   157 18840    40   120         82   157   18840
##  4    93    118  10974   118   105 12390    30   118         72   105   12390
##  5    89    103   9167   129   173 22317    40   129         69   176   22704
##  6    58    100   5800   123   140 17220    40   123         83   140   17220
##  7    63    120   7560    98   130 12740    40    98         71   130   12740
##  8    86    161  13846   144   157 22608    40   144        111   157   22608
##  9    69    143   9867   115   118 13570    40   113         81   151   17063
## 10    76    105   7980   126   125 15750    40   126         94   125   15750
## # ... with 548 more rows, and 21 more variables: dobdose <dbl>, age <dbl>,
## #   gender <dbl>, baseEF <dbl>, dobEF <dbl>, chestpain <dbl>, posECG <dbl>,
## #   equivecg <dbl>, restwma <dbl>, posSE <dbl>, newMI <dbl>, newPTCA <dbl>,
## #   newCABG <dbl>, death <dbl>, hxofHT <dbl>, hxofdm <dbl>, hxofcig <dbl>,
## #   hxofMI <dbl>, hxofPTCA <dbl>, hxofCABG <dbl>, any event <dbl>
```

Have a look at the structure of the df:

```
glimpse(cardiac)
```

```
## Rows: 558
## Columns: 32
## $ bhr         <dbl> 92, 62, 62, 93, 89, 58, 63, 86, 69, 76, 105, 72, 90, 81, 8~
## $ basebp      <dbl> 103, 139, 139, 118, 103, 100, 120, 161, 143, 105, 134, 112~
## $ basedp      <dbl> 9476, 8618, 8618, 10974, 9167, 5800, 7560, 13846, 9867, 79~
## $ pkhr        <dbl> 114, 120, 120, 118, 129, 123, 98, 144, 115, 126, 171, 127,~
## $ sbp         <dbl> 86, 158, 157, 105, 173, 140, 130, 157, 118, 125, 182, 95, ~
## $ dp          <dbl> 9804, 18960, 18840, 12390, 22317, 17220, 12740, 22608, 135~
## $ dose        <dbl> 40, 40, 40, 30, 40, 40, 40, 40, 40, 40, 40, 30, 40, 40, 40~
## $ maxhr       <dbl> 100, 120, 120, 118, 129, 123, 98, 144, 113, 126, 171, 125,~
## $ ‘%mphr(b)‘  <dbl> 74, 82, 82, 72, 69, 83, 71, 111, 81, 94, 108, 80, 126, 58,~
## $ mbp         <dbl> 121, 158, 157, 105, 176, 140, 130, 157, 151, 125, 182, 101~
## $ dpmaxdo     <dbl> 12100, 18960, 18840, 12390, 22704, 17220, 12740, 22608, 17~
## $ dobdose     <dbl> 40, 40, 40, 30, 40, 40, 40, 40, 40, 40, 40, 20, 40, 40, 40~
## $ age         <dbl> 85, 73, 73, 57, 34, 71, 81, 90, 81, 86, 61, 63, 86, 29, 71~
## $ gender      <dbl> 0, 0, 0, 1, 0, 0, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1~
## $ baseEF      <dbl> 27, 39, 39, 42, 45, 46, 48, 50, 52, 52, 52, 53, 54, 55, 55~
## $ dobEF       <dbl> 32, 40, 40, 57, 57, 57, 54, 57, 62, 62, 65, 65, 70, 65, 65~
## $ chestpain   <dbl> 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1~
## $ posECG      <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1~
## $ equivecg    <dbl> 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 1, 0, 0~
## $ restwma     <dbl> 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 1~
## $ posSE       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ newMI       <dbl> 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ newPTCA     <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ newCABG     <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ death       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ hxofHT      <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ hxofdm      <dbl> 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ hxofcig     <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ hxofMI      <dbl> 0, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1~
## $ hxofPTCA    <dbl> 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ hxofCABG    <dbl> 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ ‘any event‘ <dbl> 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
```

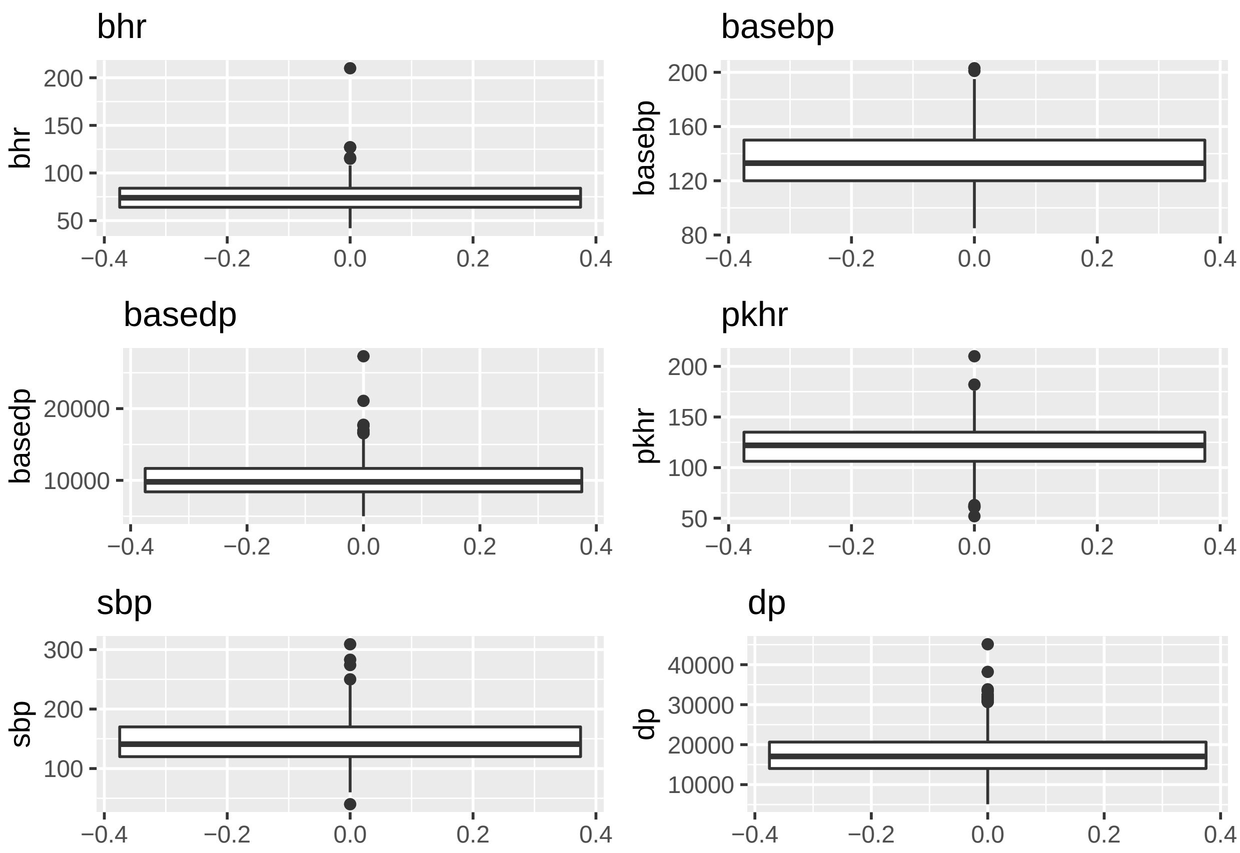
All colums are numeric. Some however should be factors. That will be dealt with later.

```
summary(cardiac)
```

```
##       bhr            basebp        basedp           pkhr          sbp     
##  Min.   : 42.0   Min.   : 85   Min.   : 5000   Min.   : 52   Min.   : 40  
##  1st Qu.: 64.0   1st Qu.:120   1st Qu.: 8400   1st Qu.:106   1st Qu.:120  
##  Median : 74.0   Median :133   Median : 9792   Median :122   Median :141  
##  Mean   : 75.3   Mean   :135   Mean   :10181   Mean   :121   Mean   :147  
##  3rd Qu.: 84.0   3rd Qu.:150   3rd Qu.:11663   3rd Qu.:135   3rd Qu.:170  
##  Max.   :210.0   Max.   :203   Max.   :27300   Max.   :210   Max.   :309  
##        dp             dose          maxhr        %mphr(b)          mbp     
##  Min.   : 5100   Min.   :10.0   Min.   : 58   Min.   : 38.0   Min.   : 84  
##  1st Qu.:14033   1st Qu.:30.0   1st Qu.:104   1st Qu.: 69.0   1st Qu.:133  
##  Median :17060   Median :40.0   Median :120   Median : 78.0   Median :150  
##  Mean   :17634   Mean   :33.8   Mean   :119   Mean   : 78.6   Mean   :156  
##  3rd Qu.:20644   3rd Qu.:40.0   3rd Qu.:133   3rd Qu.: 88.0   3rd Qu.:176  
##  Max.   :45114   Max.   :40.0   Max.   :200   Max.   :133.0   Max.   :309  
##     dpmaxdo         dobdose          age           gender          baseEF    
##  Min.   : 7130   Min.   : 5.0   Min.   :26.0   Min.   :0.000   Min.   :20.0  
##  1st Qu.:15260   1st Qu.:20.0   1st Qu.:60.0   1st Qu.:0.000   1st Qu.:52.0  
##  Median :18118   Median :30.0   Median :69.0   Median :1.000   Median :57.0  
##  Mean   :18550   Mean   :30.2   Mean   :67.3   Mean   :0.606   Mean   :55.6  
##  3rd Qu.:21239   3rd Qu.:40.0   3rd Qu.:75.0   3rd Qu.:1.000   3rd Qu.:62.0
##  Max.   :45114   Max.   :40.0   Max.   :93.0   Max.   :1.000   Max.   :83.0
##      dobEF        chestpain         posECG         equivecg        restwma
##  Min.   :23.0   Min.   :0.000   Min.   :0.000   Min.   :0.000   Min.   :0.000
##  1st Qu.:62.0   1st Qu.:0.000   1st Qu.:1.000   1st Qu.:0.000   1st Qu.:0.000
##  Median :67.0   Median :1.000   Median :1.000   Median :1.000   Median :0.000
##  Mean   :65.2   Mean   :0.692   Mean   :0.873   Mean   :0.685   Mean   :0.461
##  3rd Qu.:73.0   3rd Qu.:1.000   3rd Qu.:1.000   3rd Qu.:1.000   3rd Qu.:1.000
##  Max.   :94.0   Max.   :1.000   Max.   :1.000   Max.   :1.000   Max.   :1.000
##      posSE           newMI          newPTCA         newCABG          death
##  Min.   :0.000   Min.   :0.00   Min.   :0.000   Min.   :0.000   Min.   :0.000
##  1st Qu.:1.000   1st Qu.:1.00   1st Qu.:1.000   1st Qu.:1.000   1st Qu.:1.000
##  Median :1.000   Median :1.00   Median :1.000   Median :1.000   Median :1.000
##  Mean   :0.756   Mean   :0.95   Mean   :0.952   Mean   :0.941   Mean   :0.957
##  3rd Qu.:1.000   3rd Qu.:1.00   3rd Qu.:1.000   3rd Qu.:1.000   3rd Qu.:1.000
##  Max.   :1.000   Max.   :1.00   Max.   :1.000   Max.   :1.000   Max.   :1.000
##      hxofHT          hxofdm          hxofcig           hxofMI
##  Min.   :0.000   Min.   :0.000   Min.   :0.000   Min.   :0.000
##  1st Qu.:0.000   1st Qu.:0.000   1st Qu.:0.500   1st Qu.:0.000
##  Median :0.000   Median :1.000   Median :1.000   Median :1.000
##  Mean   :0.296   Mean   :0.631   Mean   :0.658   Mean   :0.724
##  3rd Qu.:1.000   3rd Qu.:1.000   3rd Qu.:1.000   3rd Qu.:1.000
##  Max.   :1.000   Max.   :1.000   Max.   :1.000   Max.   :1.000
##     hxofPTCA        hxofCABG       any event
##  Min.   :0.000   Min.   :0.000   Min.   :0.000
##  1st Qu.:1.000   1st Qu.:1.000   1st Qu.:1.000
##  Median :1.000   Median :1.000   Median :1.000
##  Mean   :0.927   Mean   :0.842   Mean   :0.841
##  3rd Qu.:1.000   3rd Qu.:1.000   3rd Qu.:1.000
##  Max.   :1.000   Max.   :1.000   Max.   :1.000
```

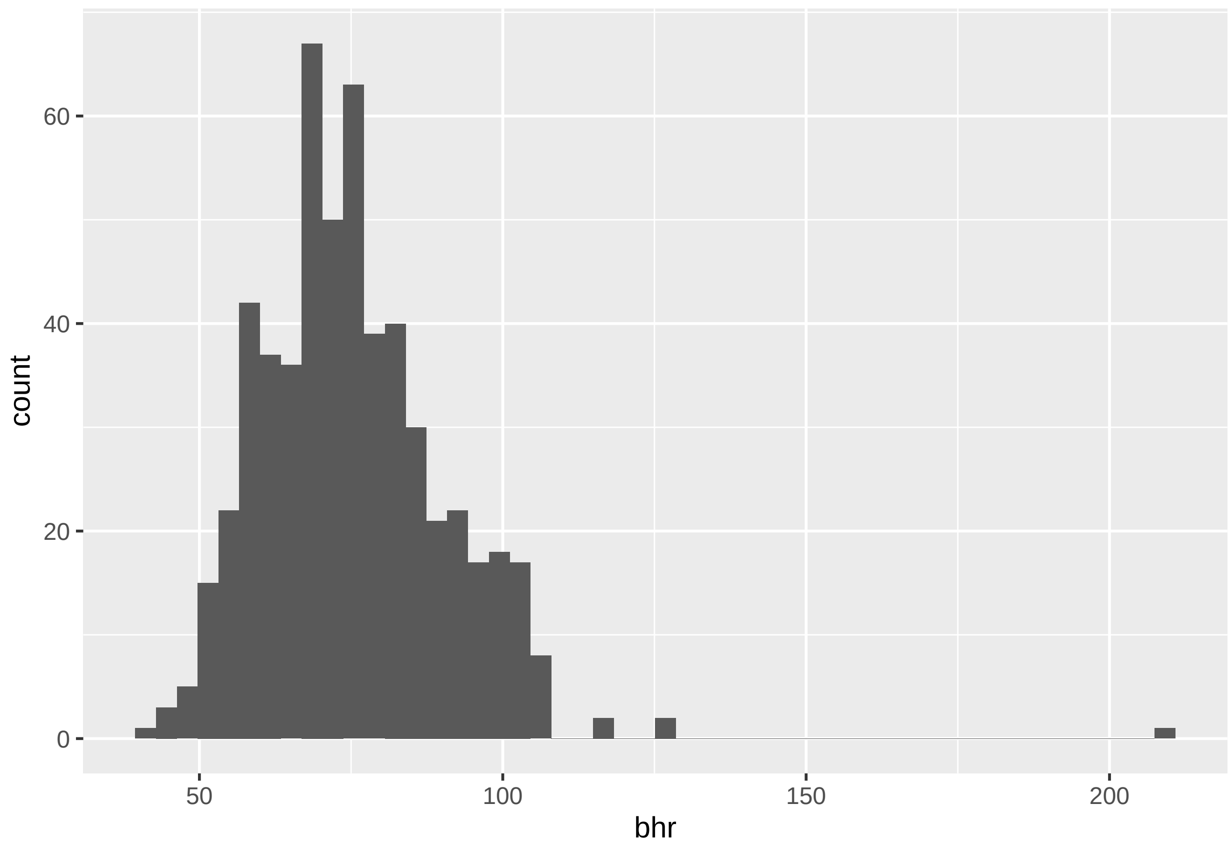
Some variables were selected for boxplot:

```
my_plots <- list()
#use indices is important!
for (i in 1:6) {
    n <- names(cardiac)[i]
    #use aes_string() !!!
    g <- ggplot(data = cardiac, mapping = aes_string(y = n)) +
        geom_boxplot() +
        ylab(n) +
        ggtitle(n)
    my_plots[[i]] <- g ##has to be integer, not name!
}
#use do.call() to process the list in grid.arrange
do.call(grid.arrange, c(my_plots, nrow = 3))
```



The `bhr` variable has a maximum of 210 which is highly unlikely. Let's investigate the distribution of it:

```r
ggplot(data = cardiac, mapping = aes(x = bhr)) +
    geom_histogram(bins = 50)
```



The 210 value is absurd and wrong for sure. I am going to take it out.

```
cardiac <- cardiac %>%
    filter(bhr != max(bhr))
```

Data transformations

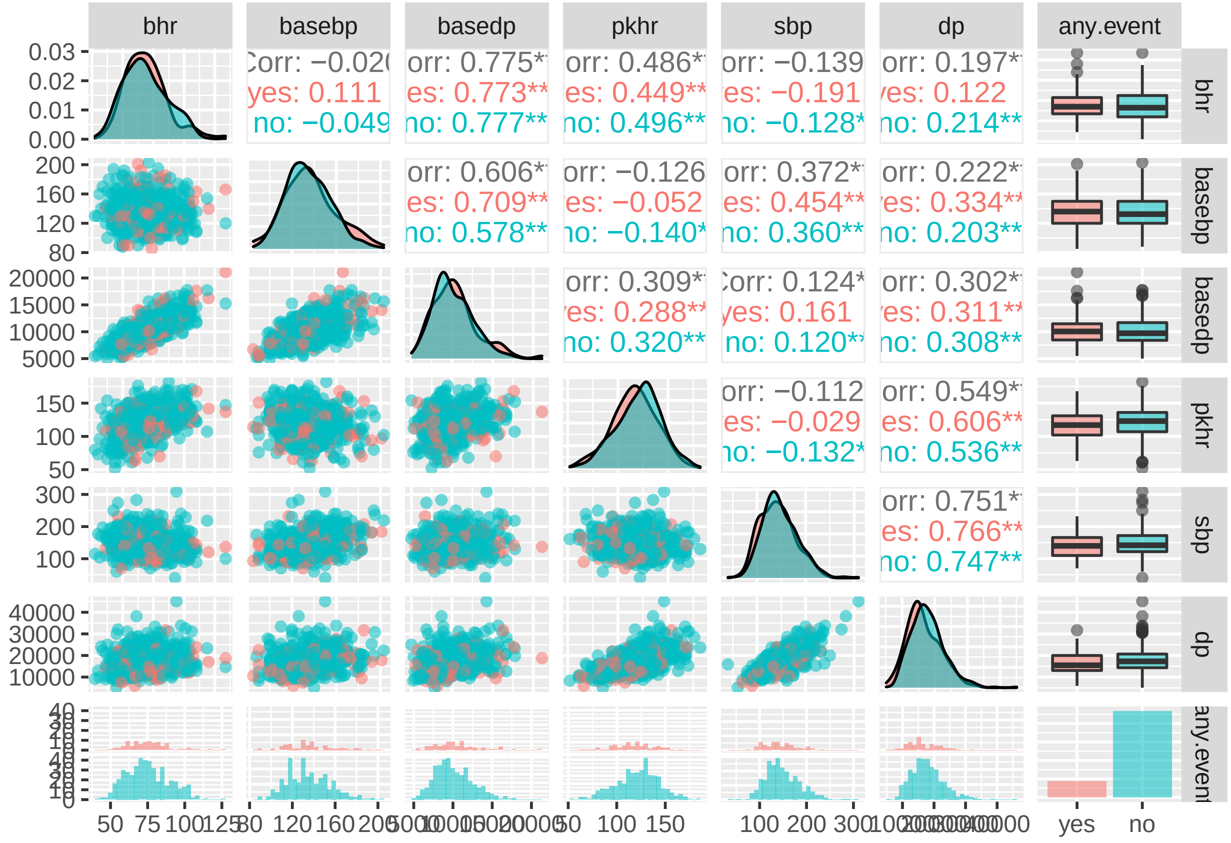
The gender attribute will be converted into a factor for easy and readable analysis.

```
cardiac <- cardiac %>%
    mutate(gender = factor(gender, labels = c("m", "f"), levels = c(0, 1)),
           chestpain = recode(chestpain, "0" = "yes", "1" = "no"),
           any.event = factor(`any event`, labels = c("yes", "no"), levels = c(0, 1)))

# 0 MEANS THE PATIENT EXPERIENCED CHEST PAIN
head(cardiac)
```

```
## # A tibble: 6 x 33
##      bhr basebp basedp  pkhr   sbp    dp  dose maxhr ‘%mphr(b)‘   mbp dpmaxdo
##    <dbl>  <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>      <dbl> <dbl>   <dbl>
## 1     92    103   9476   114    86  9804    40   100         74   121   12100
## 2     62    139   8618   120   158 18960    40   120         82   158   18960
## 3     62    139   8618   120   157 18840    40   120         82   157   18840
## 4     93    118  10974   118   105 12390    30   118         72   105   12390
## 5     89    103   9167   129   173 22317    40   129         69   176   22704
## 6     58     100    5800    123    140  17220     40    123          83    140    17220
## # ... with 22 more variables: dobdose <dbl>, age <dbl>, gender <fct>,
## #   baseEF <dbl>, dobEF <dbl>, chestpain <chr>, posECG <dbl>, equivecg <dbl>,
## #   restwma <dbl>, posSE <dbl>, newMI <dbl>, newPTCA <dbl>, newCABG <dbl>,
## #   death <dbl>, hxofHT <dbl>, hxofdm <dbl>, hxofcig <dbl>, hxofMI <dbl>,
## #   hxofPTCA <dbl>, hxofCABG <dbl>, any event <dbl>, any.event <fct>
```

Pairwise plot

```
cardiac %>%
  select(c(1:6, 33)) %>%
  ggpairs(mapping = aes(color = any.event, alpha = 0.3))
```



Dose dependency of variables

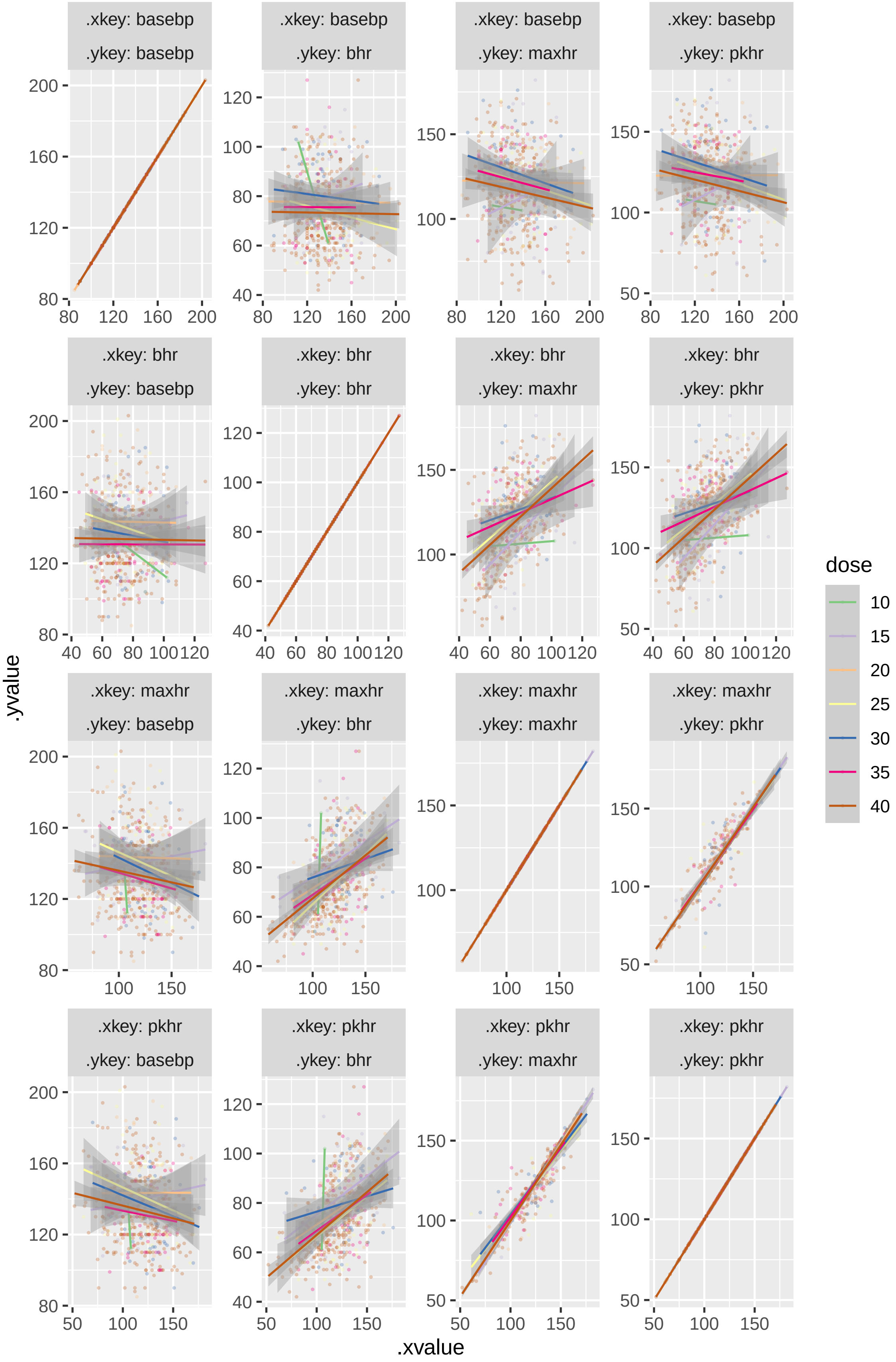
A custom function ripped from the internet (https://stackoverflow.com/questions/3735286/create-a-matrix-of-scatterplots-pairs-equivalent-in-ggplot2)

```
gatherpairs <- function(data,
                        ...,
                        xkey = '.xkey',
                        xvalue = '.xvalue',
                        ykey = '.ykey',
                        yvalue = '.yvalue',
                        na.rm = FALSE,
                        convert = FALSE,
                        factor_key = FALSE) {
  vars <- quos(...)
  xkey <- enquo(xkey)
  xvalue <- enquo(xvalue)
  ykey <- enquo(ykey)
  yvalue <- enquo(yvalue)

  data %>% {
    cbind(
      gather(
        .,
        key = !!xkey,
        value = !!xvalue,
        !!!vars,
        na.rm = na.rm,
        convert = convert,
        factor_key = factor_key
      ),
      select(.,!!!vars)
    )
  } %>% gather(
    .,
    key = !!ykey,
    value = !!yvalue,
    !!!vars,
    na.rm = na.rm,
    convert = convert,
    factor_key = factor_key
  )
}
```

Usage:

```
cardiac %>%
  mutate(dose = factor(dose)) %>%
  drop_na(c(bhr, basebp, pkhr, maxhr)) %>%
  gatherpairs(bhr, basebp, pkhr, maxhr) %>% {
    ggplot(., aes(x = .xvalue, y = .yvalue, color = dose)) +
      geom_point(size = 0.2, alpha = 0.3) +
      geom_smooth(method = 'lm', size = 0.5) +
      facet_wrap(
        .xkey ~ .ykey,
        ncol = length(unique(.$.ykey)),
        scales = 'free',
        labeller = label_both
      ) +
      scale_color_brewer(type = 'qual')
  }
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```


.xkey: basebp
.ykey: basebp
.xkey: basebp
.ykey: bhr
.xkey: basebp
.ykey: maxhr
.xkey: basebp
.ykey: pkhr
.xkey: bhr
.ykey: basebp
.xkey: bhr
.ykey: bhr
.xkey: bhr
.ykey: maxhr
.xkey: bhr
.ykey: pkhr
.xkey: maxhr
.ykey: basebp
.xkey: maxhr
.ykey: bhr
.xkey: maxhr
.ykey: maxhr
.xkey: maxhr
.ykey: pkhr
.xkey: pkhr
.ykey: basebp
.xkey: pkhr
.ykey: bhr
.xkey: pkhr
.ykey: maxhr
.xkey: pkhr
.ykey: pkhr
.yvalue
.xvalue
200
160
120
80
40
60
100
150
50
dose
10
15
20
25
30
35
40

Relationships with outcome variable

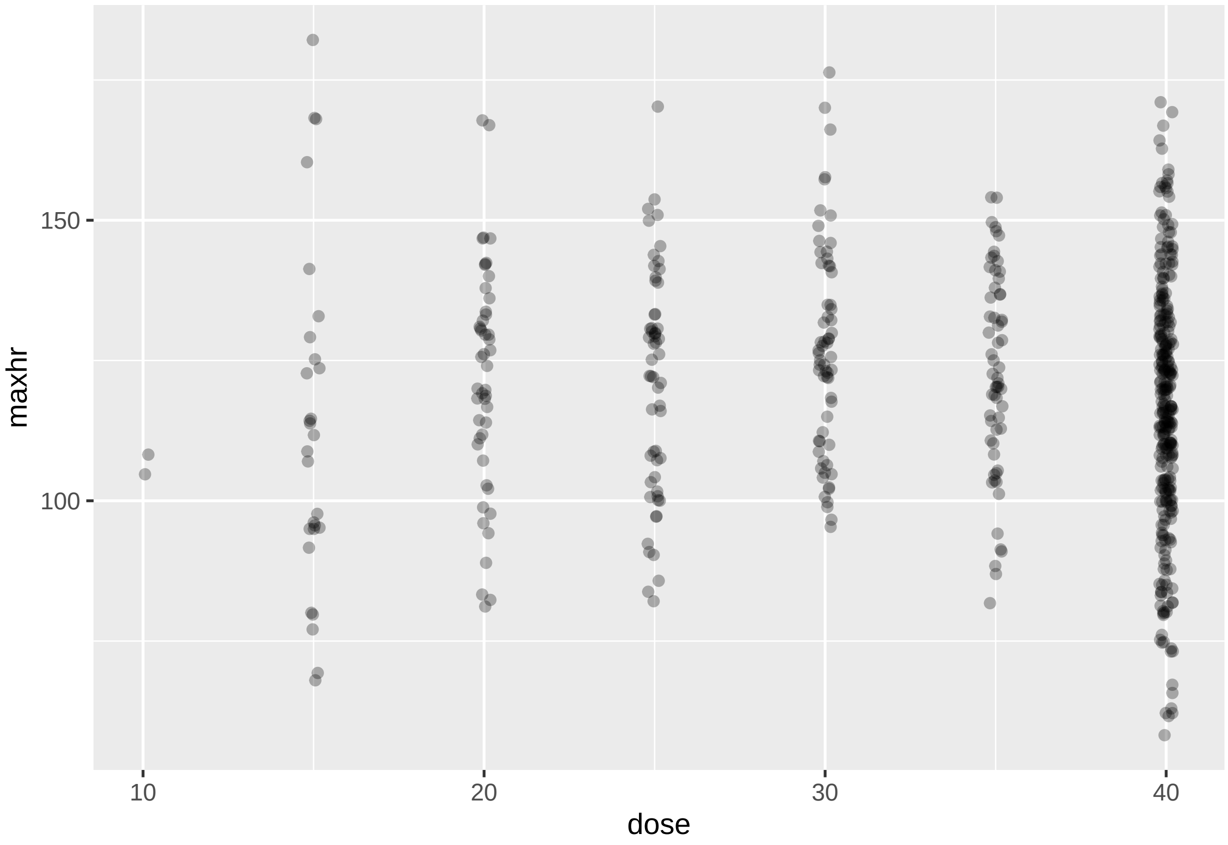
Different outcome variables

Dose : DOSE OF DOBUTAMINE GIVEN

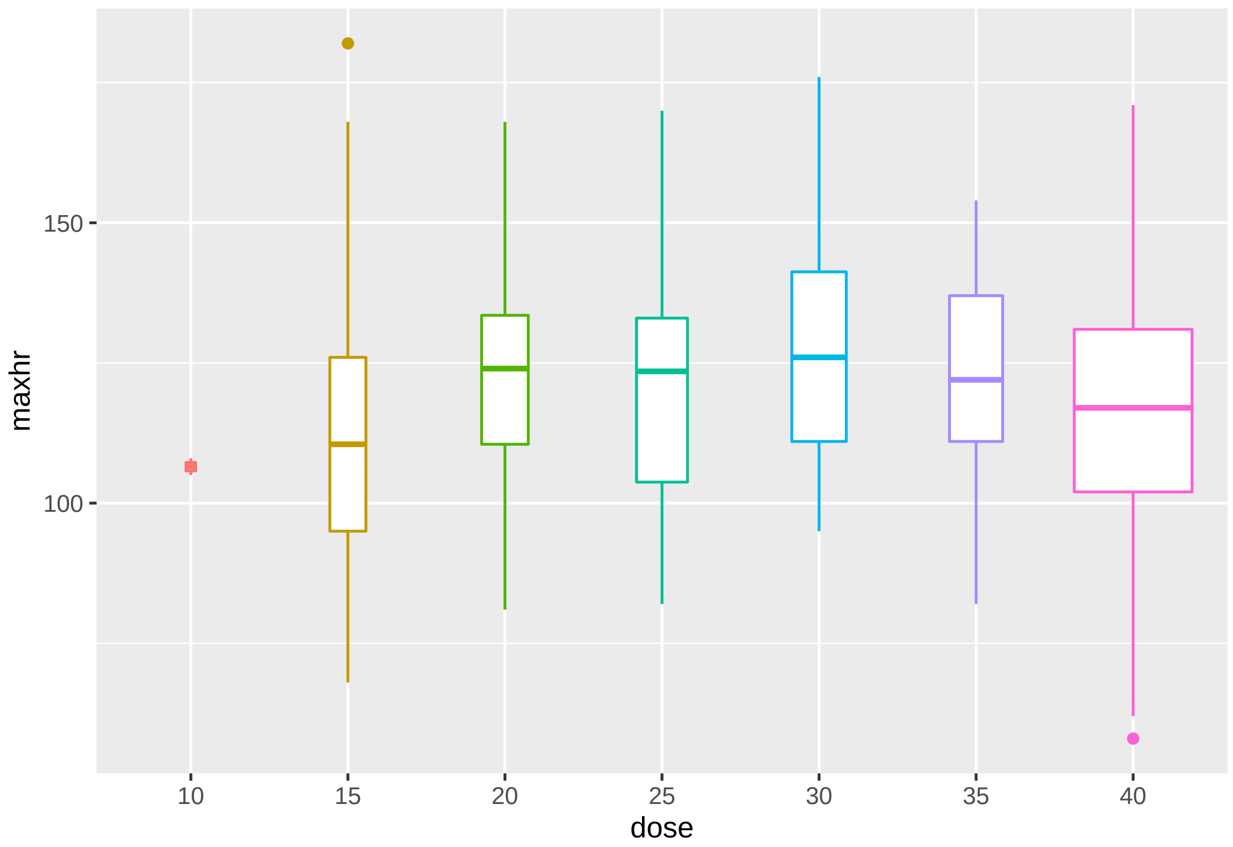
Which variable is interesting?

Maybe maxhr?

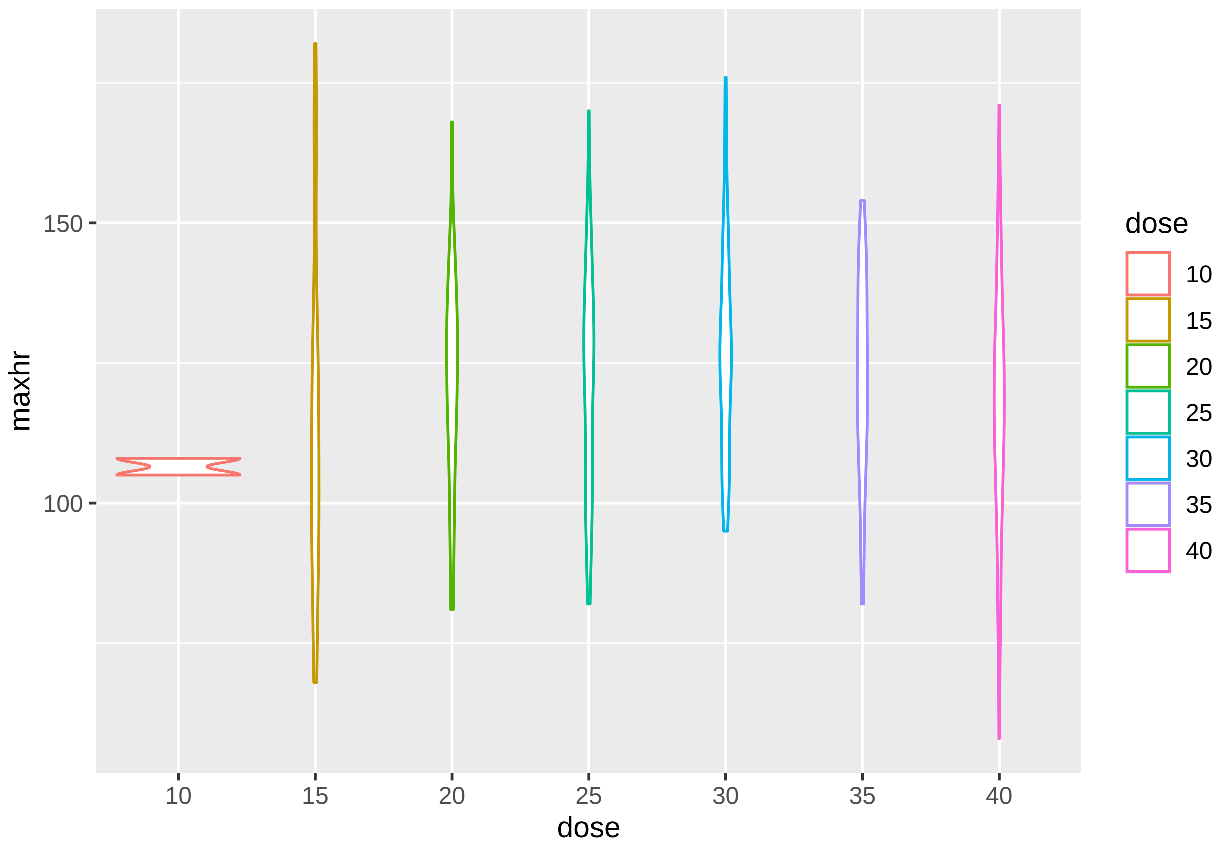
```
cardiac %>%
  ggplot(mapping = aes(x = dose, y = maxhr)) +
    geom_jitter(width = 0.2, alpha = 0.3)
```



```
cardiac %>%
  mutate(dose = factor(dose)) %>%
  ggplot(mapping = aes(x = dose, y = maxhr, color = dose)) +
    geom_boxplot(varwidth = TRUE) + theme(legend.position = "none")
```



```
cardiac %>%
  mutate(dose = factor(dose)) %>%
  ggplot(mapping = aes(x = dose, y = maxhr)) +
    geom_violin(aes(color = dose))
```



PCA

Does Principal Components Analysis tell me anything about patterns of variation?

```
library(devtools)
```

```
## Warning: package 'devtools' was built under R version 4.0.5
```

```
## Loading required package: usethis
```

```
## Warning: package 'usethis' was built under R version 4.0.5
```

```
install_github("vqv/ggbiplot")
```

```
## WARNING: Rtools is required to build R packages, but is not currently installed.
##
## Please download and install Rtools 4.0 from https://cran.r-project.org/bin/windows/Rtools/.
```

```
## Skipping install of 'ggbiplot' from a github remote, the SHA1 (7325e880) has not changed since last
##   Use `force = TRUE` to force installation
```

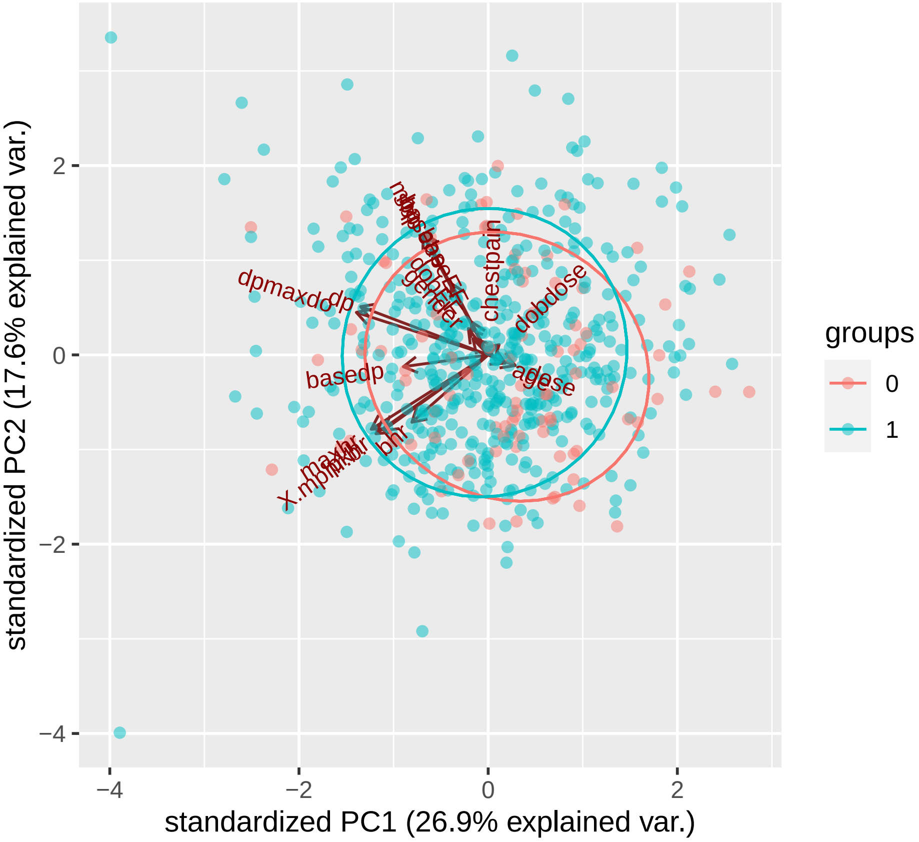
```r
library(ggbiplot)
```

```
## Loading required package: plyr

## Warning: package 'plyr' was built under R version 4.0.3

## ------------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)

## ------------------------------------------------------------------------------

##
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize

## Loading required package: scales

## Warning: package 'scales' was built under R version 4.0.3

##
## Attaching package: 'scales'

## The following object is masked from 'package:readr':
##
##     col_factor

## Loading required package: grid
```

```r
#reload data
cardiac <- read.csv(file = "cardiac.csv")
cardiac <- cardiac %>% select(1:32)

cardiac.pca <- prcomp(cardiac[, 1:17], center = TRUE, scale. = TRUE)
summary(cardiac.pca)
```

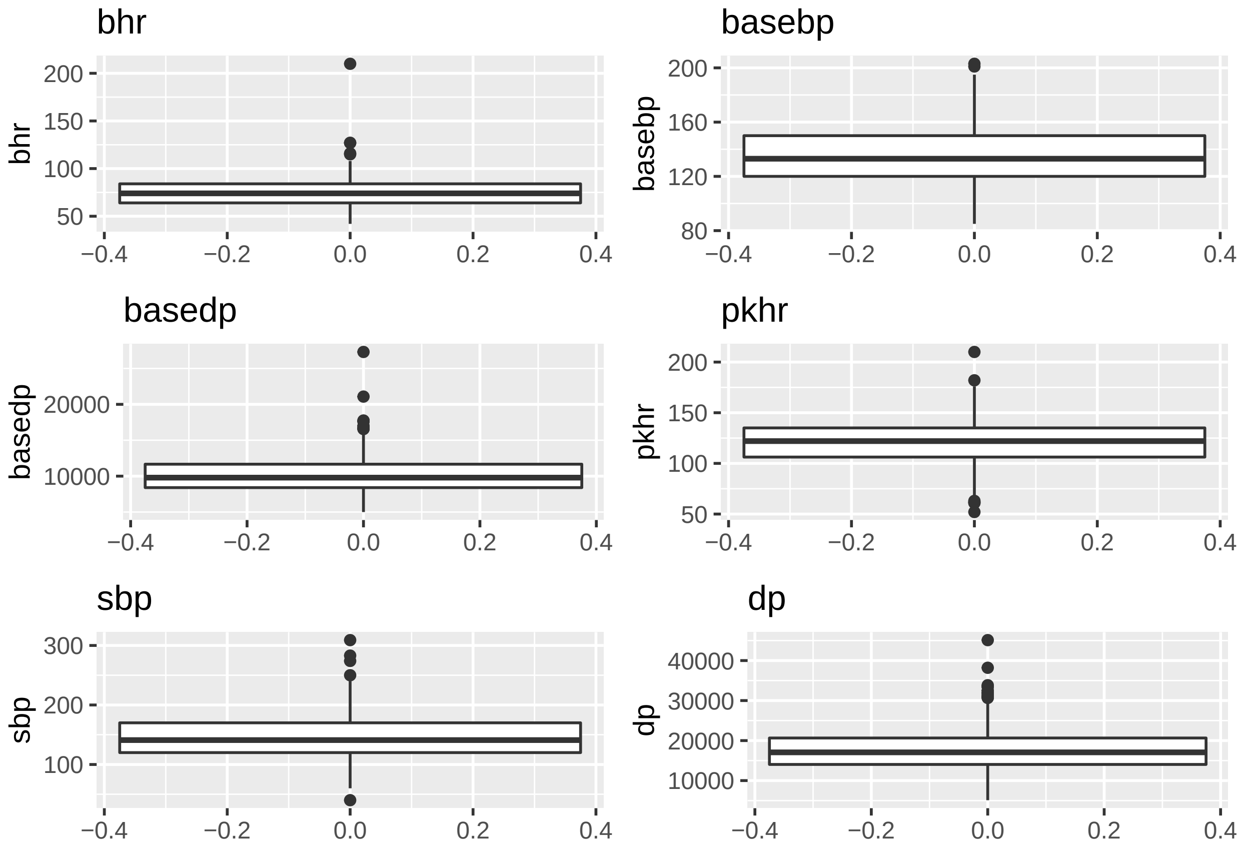
```
## Importance of components:
##                          PC1   PC2   PC3   PC4    PC5    PC6    PC7    PC8
## Standard deviation     2.139 1.730 1.454 1.404 1.1389 1.1081 0.9895 0.9016
## Proportion of Variance 0.269 0.176 0.124 0.116 0.0763 0.0722 0.0576 0.0478
## Cumulative Proportion  0.269 0.445 0.569 0.685 0.7617 0.8339 0.8915 0.9393
##                           PC9   PC10    PC11    PC12   PC13    PC14    PC15
## Standard deviation     0.7169 0.5118 0.31186 0.25819 0.2405 0.14947 0.07488
## Proportion of Variance 0.0302 0.0154 0.00572 0.00392 0.0034 0.00131 0.00033
## Cumulative Proportion  0.9696 0.9850 0.99069 0.99461 0.9980 0.99932 0.99965
##                           PC16    PC17
## Standard deviation     0.06066 0.04694
## Proportion of Variance 0.00022 0.00013
## Cumulative Proportion  0.99987 1.00000
```

```
ggbiplot(cardiac.pca, ellipse=TRUE, groups = factor(cardiac$any.event), alpha = 0.5, size = 0.2)
```



Boxplots

```
my_plots <- list()
#use indices is important!
for (i in 1:6) {
    n <- names(cardiac)[i]
    #use aes_string() !!!
    g <- ggplot(data = cardiac, mapping = aes_string(y = n)) +
        geom_boxplot() +
        ylab(n) +
        ggtitle(n)
    my_plots[[i]] <- g ##has to be integer, not name!
}
#use do.call() to process the list in grid.arrange
do.call(grid.arrange, c(my_plots, nrow = 3))
```


bhr
bhr
200
150
100
50
−0.4
−0.2
0.0
0.2
0.4
basebp
basebp
200
160
120
80
−0.4
−0.2
0.0
0.2
0.4
basedp
basedp
20000
10000
−0.4
−0.2
0.0
0.2
0.4
pkhr
pkhr
200
150
100
50
−0.4
−0.2
0.0
0.2
0.4
sbp
sbp
300
200
100
−0.4
−0.2
0.0
0.2
0.4
dp
dp
40000
30000
20000
10000
−0.4
−0.2
0.0
0.2
0.4