Aufgabenzettel 03

Gruppe 01

09.04.2020

#bibliography: bibliography.bib

Aufgabe 7 Deskriptive Statistik

Laden Sie den Workspace yingtan_20_ueb3.Rdata und arbeiten Sie im Folgenden mit dem Objekt ljz weiter. Einheiten mit Angeben!

```
load("yingtan_20_ueb3.RData")
print(ljz)
```

```
## # A tibble: 335 x 9
##    OBJECTID SAMPLING   EAST   NORTH     C Ca_exch Mg_exch K_exch Na_exch
##  *    <int> <fct>     <dbl>   <dbl> <dbl>   <dbl>   <dbl>  <dbl> <chr>
##  1        1 regular  490591 3123240 0.599   12.4     3.20  3.51  0.452
##  2        2 regular  490591 3123390 0.647   13.1     4.77  0.992 0.592
##  3        3 regular  490591 3123540 0.527    3.77    1.45  4.05  0.122
##  4        4 regular  490741 3123390 0.812   31.7     7.72  2.66  0.687
##  5        5 regular  490741 3123540 0.756   24.0     8.33  4.02  0.244
##  6        6 regular  491191 3122040 1.24    15.8     2.71  0.634 0.383
##  7        7 regular  491191 3123690 0.918   11.2     2.53  1.73  0.244
##  8        8 regular  491341 3121740 0.414    7.54    2.81  2.26  0.001
##  9        9 regular  491341 3121890 0.523   13.0     5.03  3.79  0.001
## 10       10 regular  491341 3122040 0.975   20.4     3.88  2.18  0.435
## # ... with 325 more rows
```

7 a)

Methode Summary auf austauschbaren Ca-Ionen anwenden und aus Funktion gewonnene Parameter erläutern.

```
print(summary(ljz$Ca_exch))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.772  13.633  19.491  21.926  28.718  94.311
```

Min. 1st Qu. Median Mean 3rd Qu. Max.

7 b)

Varianz, Sandardabweichung, sowie Schiefe und Kurtosis (bezogen auf NV mit kurtosis = 0) füraustauschbaren Ca-Ionen ermitteln.

```
var(ljz$Ca_exch)
```

```
## [1] 123.8529
```

```
sd(ljz$Ca_exch)
```

```
## [1] 11.12892
```

```
#skewness als Funktion

skewness <- function(x) {
  a <- (1/length(x))*sum((((x)-mean(x))/sd(x))^3)
  return(a)
}

skewness(ljz$Ca_exch)
```

```
## [1] 1.50357
```

```
#Skewness als Magrittr Pipe

b <- (ljz$Ca_exch-mean(ljz$Ca_exch)) %>%
     divide_by(sd(ljz$Ca_exch)) %>%
        raise_to_power(3) %>%
          sum() %>%
            multiply_by(1/length(ljz$Ca_exch))

b;
```

```
## [1] 1.50357
```

```
#Kurtosis als Funktion
kurtosis <- function(x) {
a <- ((1/length(x))*sum((((x)-mean(x))/sd(x))^4))-3
return(a)
}

kurtosis(ljz$Ca_exch)
```

```
## [1] 5.554256
```

```
#Kurtosis als Magrittr Pipe
c <- (ljz$Ca_exch-mean(ljz$Ca_exch)) %>%
     divide_by(sd(ljz$Ca_exch)) %>%
        raise_to_power(4) %>%
          sum() %>%
            multiply_by(1/length(ljz$Ca_exch)) %>%
              -3

c;
```

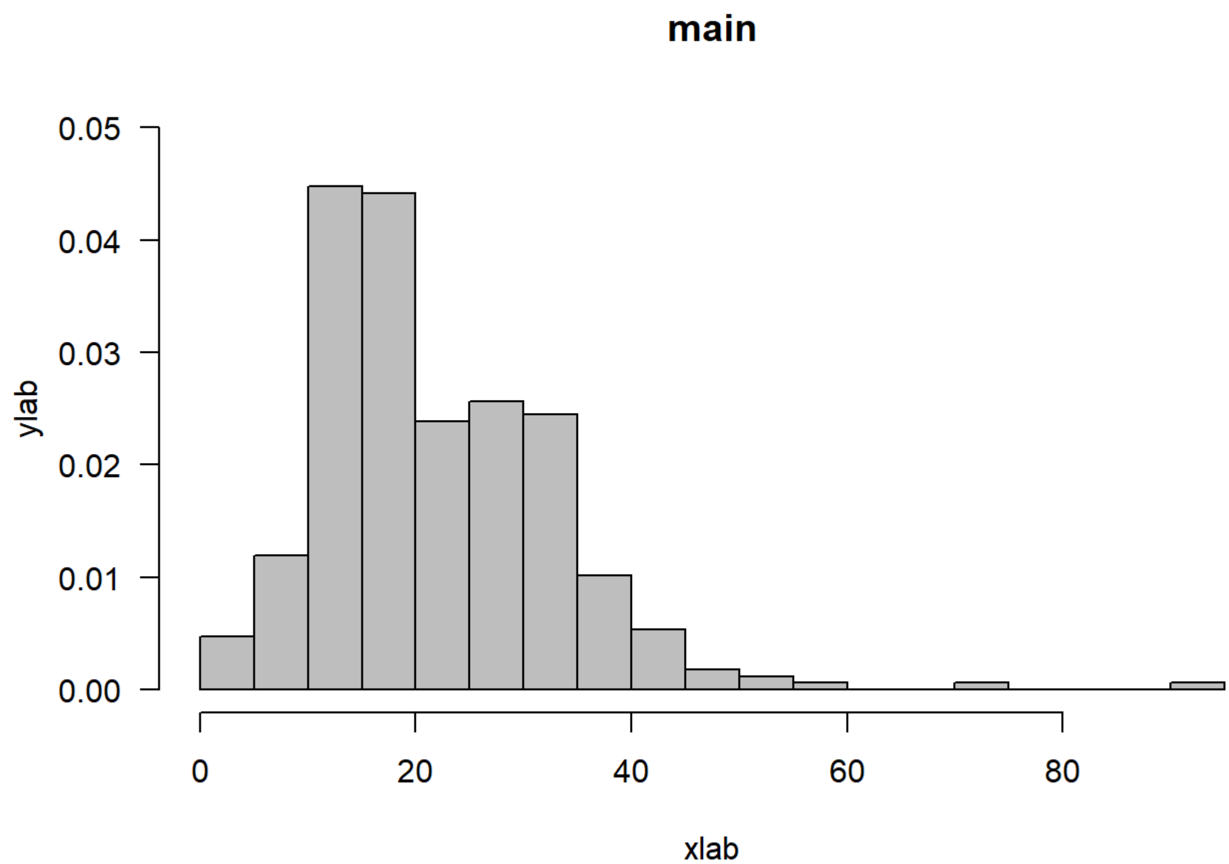
```
## [1] 5.554256
```

Aufgabe 8 Dichte-Histogramme und Box-Whisker-Plots in R

8 a)

Dichte-Histogramm für austauschbare Ca-Ionen erstellen. Aussagekräftige Klassenweiten. Titel und Achsenbeschriftungen sinnvoll ändern und Balken grau einfärben.

```
# The Freedman-Diaconis rule is very robust and works well in practice. The bin-width is set to
 h=2×IQR×n−1/3. So the number of bins is (max−min)/h, where n is the number of observations, max
is the maximum value and min is the minimum value.
#
# In base R, you can use:
#
# hist(x, breaks="FD")
# For other plotting libraries without this option (e.g., ggplot2), you can calculate binwidth a
s:
#
# bw <- 2 * IQR(x) / length(x)^(1/3)
#
# ### for example #####
# ggplot() + geom_histogram(aes(x), binwidth = bw)

# alternative Formel nacj Hedderich für


#?hist()
hist(ljz$Ca_exch,
     breaks="FD", #Bin-Weite nach Freedman-Diaconis
     freq=FALSE, #Wahrscheinlichkeitsdichte statt Häufigkeit, alternative prob=TRUE
     col="grey",
     main="main",
     xlab="xlab",
     ylab="ylab",
     ylim=c(0, 0.05),
     las=1
     )
```



8 b)

Um welche Verteilung handelt es sich bei der Ca-Ionenkonzentration dem visuellen Eindruck nach? Setzen Sie ihre Vermutung in Bezug zu den in Aufgabe 7b) errechneten Formparametern.

8 c)

Für die austauschbaren Ca-Ionen…

Box-Whisker-Plot erstellen. Randverhalten der Verteilung untersuchen.

Mögliche Ausreißer angeben (range = 1,5)

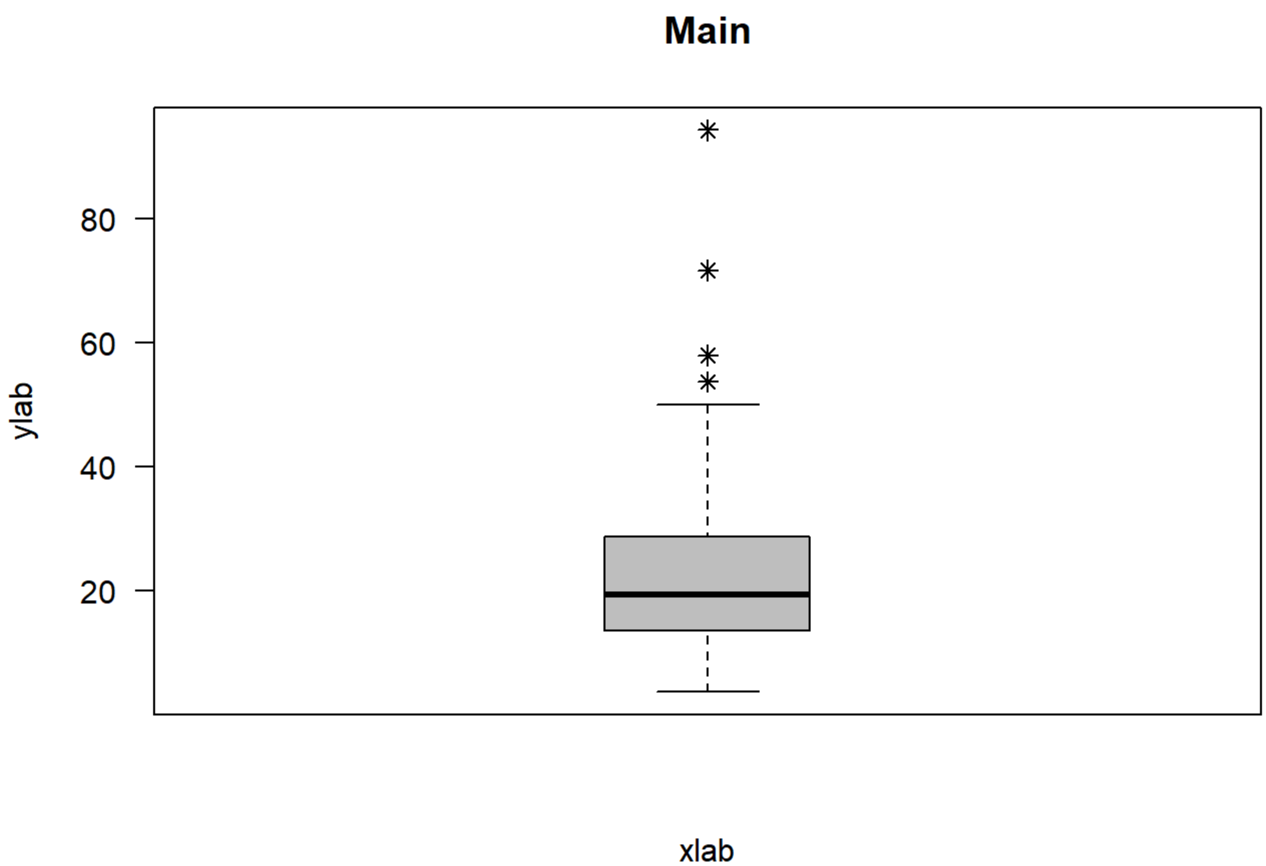
Sinvoller Titel und passende Achsenbeschriftungen.

Balkenbreite verkleinern um eine ansprechende Grafik zu erzeugen.

```
?boxplot()
```

```
## starting httpd help server ... done
```

```
?plot()
#legend()
#box()
boxplot(ljz$Ca_exch
        ,range = 1.5
        #,width = 1
        ,boxwex= 0.2
        ,varwidth=TRUE
        ,col="grey"
        ,main= "Main"
        ,xlab="xlab"
        ,ylab="ylab"
        ,pch = 8
        ,las=1 #Orientierung der Zahlen an X-Achse
        #, aspect= 0.2,
        #box.ratio= 1
        )
```




```
x <- boxplot.stats(ljz$Ca_exch)$out #Vektor der Outlier
filter(ljz, ljz$Ca_exch >= min(x))
```

OBJECTID <int>	SAMPLING <fctr>	EAST <dbl>	NORTH <dbl>	C <dbl>	Ca_exch <dbl>	Mg_exch <dbl>	K_exch <dbl>	Na_exch <chr>
86	regular	492991	3121890	0.909	71.707	10.304	2.020	0.052

OBJECTID <int>	SAMPLING <fctr>	EAST <dbl>	NORTH <dbl>	C <dbl>	Ca_exch <dbl>	Mg_exch <dbl>	K_exch <dbl>	Na_exch <chr>
91	regular	492991	3123990	1.231	58.034	7.901	0.419	0.887
99	regular	493291	3123090	0.432	94.311	7.921	4.179	0.035
197	regular	492541	3123390	1.459	53.743	6.944	1.176	0.887

4 rows

Aufgabe 9 Plotten in R

'base graphics', 'lattice' und 'ggplot2'…

jeweils einen ansehnlichen Box-Whisker-Plot: 'base graphics' (boxplot), lattice (bwplot) und ggplot2 (geom_boxplot).

In jeder dieser Abbildungen stellen Sie im Prinzip das gleiche dar, nämlich die austauschbaren Ca-Ionen für drei unterschiedliche Datensätze: regular, catA und catQ. Das heißt, Sie müssen den ursprünglichen Datensatz vorab entsprechend reduzieren.

```
levels(ljz$SAMPLING)
```

```
##  [1] "catA"     "catF"     "catK"     "catQ"     "catS"     "profile1"
##  [7] "profile2" "profile3" "profile4" "regular"
```

```
f <- c("catA" , "catQ" , "regular")
ljz2 <- filter(ljz, f == ljz$SAMPLING)
```

```
## Warning in `==.default`(f, ljz$SAMPLING): longer object length is not a multiple
## of shorter object length
```

```
## Warning in is.na(e1) | is.na(e2): longer object length is not a multiple of
## shorter object length
```

```
ljz2$SAMPLING <- factor(ljz2$SAMPLING) #alternativ droplevels(). Entferne übrige Faktoren
levels(ljz2$SAMPLING)
```

```
## [1] "catA"    "catQ"    "regular"
```

```
print(ljz2)
```

```
## # A tibble: 97 x 9
##    OBJECTID SAMPLING   EAST   NORTH     C Ca_exch Mg_exch K_exch Na_exch
##       <int> <fct>     <dbl>   <dbl> <dbl>   <dbl>   <dbl>  <dbl> <chr>
##  1        3 regular  490591 3123540 0.527    3.77    1.45  4.05  0.122
##  2        6 regular  491191 3122040 1.24    15.8     2.71  0.634 0.383
##  3        9 regular  491341 3121890 0.523   13.0     5.03  3.79  0.001
##  4       12 regular  491341 3122340 0.545    4.20    2.73  3.04  0.001
##  5       15 regular  491341 3123540 0.896   15.7     4.89  1.20  0.592
##  6       18 regular  491491 3121440 1.19     4.92    2.82  3.05  0.174
##  7       21 regular  491491 3122190 1.87    18.9     6.74  3.58  0.452
##  8       24 regular  491491 3122640 1.20    14.7     7.19  1.38  0.505
##  9       27 regular  491491 3123240 1.69    31.3     6.29  1.25  0.8
## 10       30 regular  491641 3121290 0.738   32.2     8.02  5.12  0.13
## # ... with 87 more rows
```

sinnvolle Titel sowie passende Achsenbeschriftungen

verkleinern Sie die Balkenbreite, sodass ansprechend formatierte Graphiken entstehen.

Färben Sie außerdem die Box-Elemente grau ein

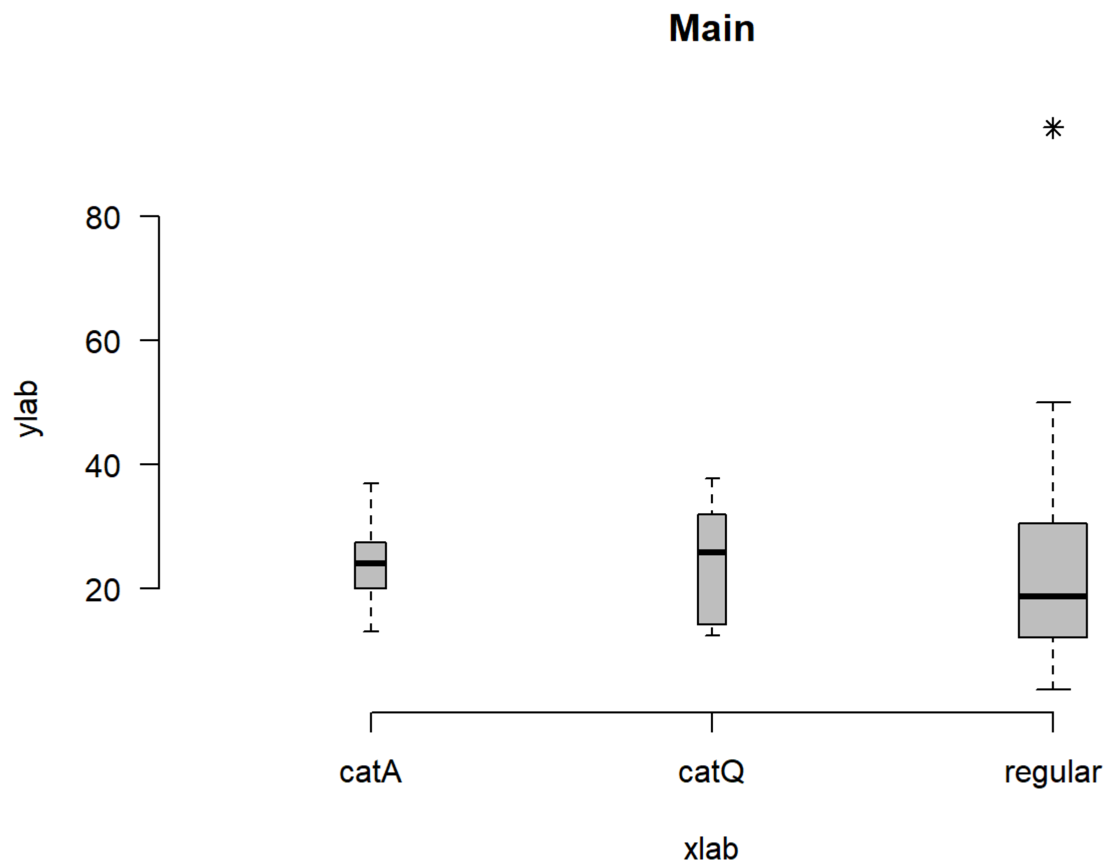
und ändern Sie die Symbolik möglicher Ausreißer zu schwarzen Sternchen (Asterisk).

Wie heißen die Argumente, die die Auswahl auffällig hoher (oder tiefer) Messwerte in den unterschiedlichen Boxplot-Umgebungen definieren?

Was sind die Default-Einstellungen dieses kritischen Parameters?

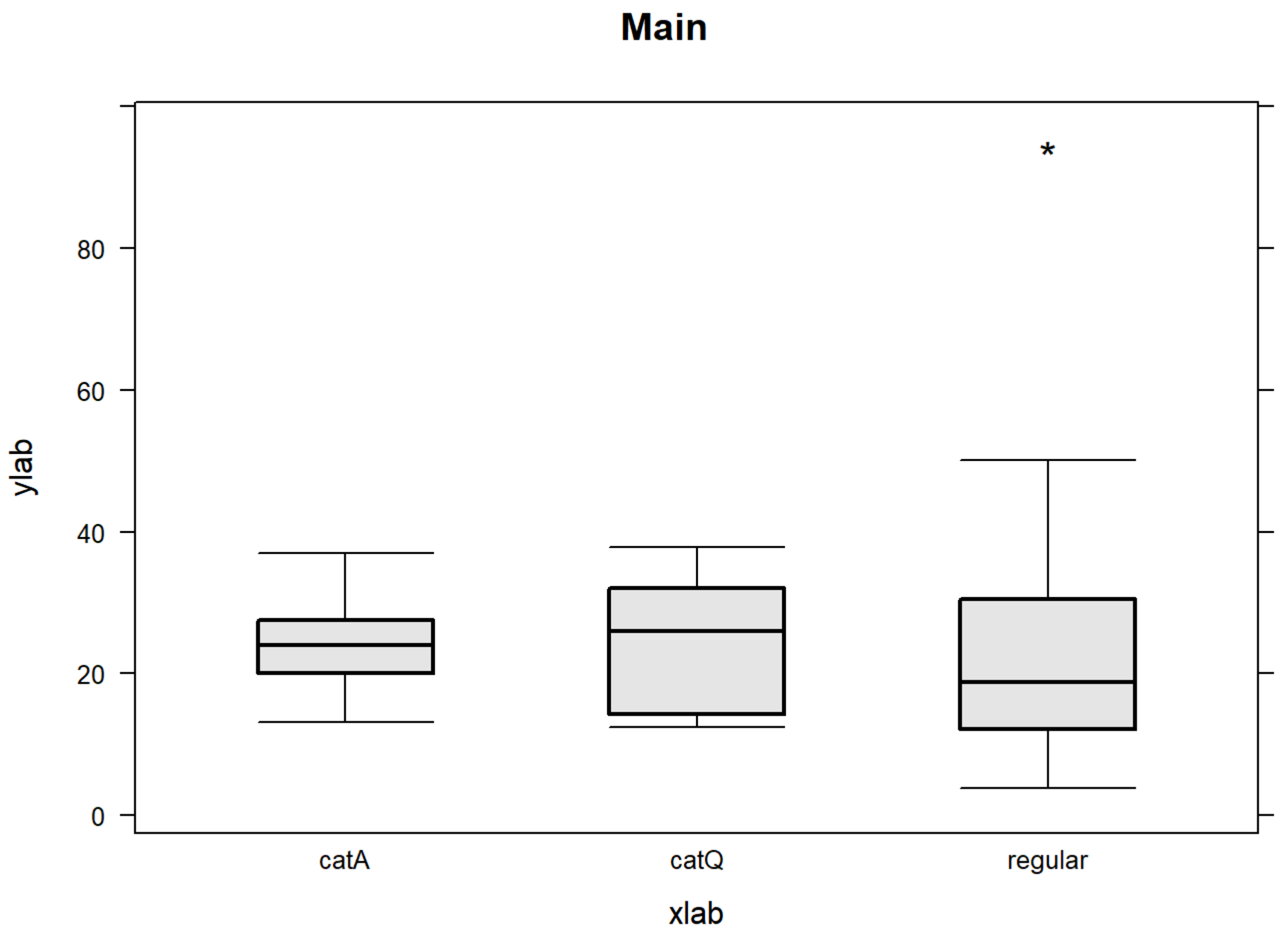
9 base graphic

```
#base grahpics: boxplot()

par(bty="n")  #kein Rahmen um Inhalt
boxplot(ljz2$Ca_exch ~ ljz2$SAMPLING,
        range = 1.5,
        #width = 1,
        boxwex= 0.2,
        varwidth=TRUE,
        col="grey",
        main= "Main",
        xlab="xlab",
        ylab="ylab",
        #asp=1, #plot.window
        pch = 8, #Outlier/Punkte als Sternchen
        las=1,
        pars=list(outcol="black") #?pars
          )
```



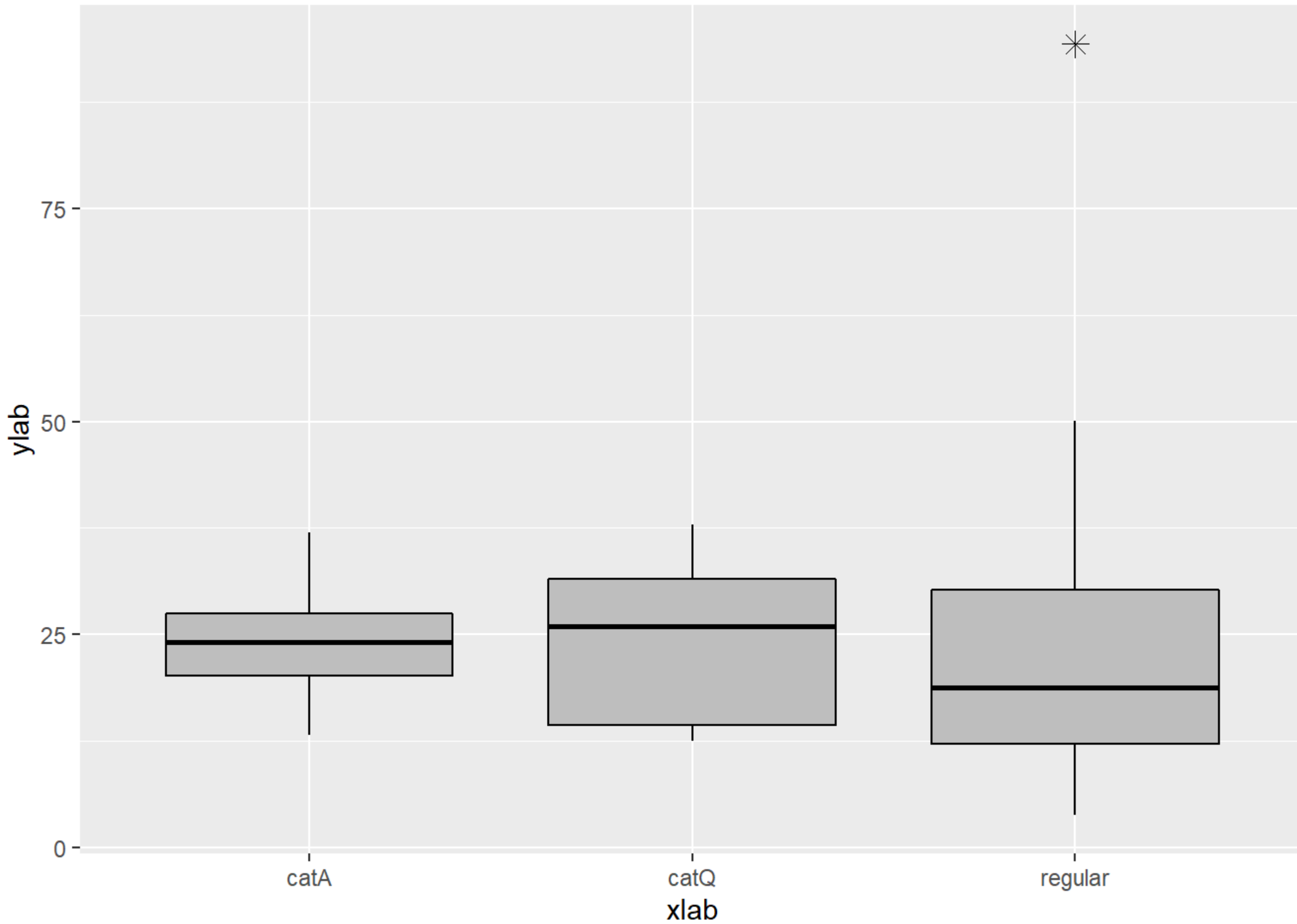
9 Lattice

```
#lattice, bwplot()
#install.packages("lattice")
library(lattice)

#show.settings(bwplot) #gibt die optionen dot rectangle, umbrella
#trellis.par.get()
#str(trellis.par.get())
#str(trellis.par.get(), max.level = 1)

# https://www.rdocumentation.org/packages/lattice/versions/0.10-10/topics/xyplot
# https://moc.online.uni-marburg.de/gitbooks/publicationQualityGraphics/_book/chapters/02_data_v
isualisation/box_lattice.html
# https://www.stat.ubc.ca/~jenny/STAT545A/block16_colorsLattice.html


bw <- bwplot(ljz2$Ca_exch ~ ljz2$SAMPLING,
        main= "Main",
         xlab="xlab",
         ylab="ylab",
         range=1.5
        # box.ratio= 0.7,
        # aspect= 1,
        # pch=8,
        # col="grey"
        #horizontal=FALSE
       )

bw_theme <- trellis.par.get()
bw_theme$box.dot$pch <- "|"
bw_theme$box.rectangle$col <- "black"
bw_theme$box.rectangle$lwd <- 2
bw_theme$box.rectangle$fill <- "grey90"
bw_theme$box.umbrella$lty <- 1
bw_theme$box.umbrella$col <- "black"
bw_theme$plot.symbol$col <- "black"
bw_theme$plot.symbol$pch <- "*"
bw_theme$plot.symbol$cex <- 2
bw_theme$strip.background$col <- "grey80"


update(bw, par.settings = bw_theme)
```



9 ggplot

```
#ggplot(),geom_boxplot

#https://moc.online.uni-marburg.de/gitbooks/publicationQualityGraphics/_book/chapters/02_data_vi
sualisation/box_ggplot2.html
box_plot <- ggplot(ljz2, aes(x=SAMPLING, y=Ca_exch))

box_plot +
  geom_boxplot(
  outlier.colour = "black"
        ,outlier.shape = 8
        ,outlier.size = 3
        ,color = "black"
        , fill = "grey"
       ,coef = 1.5 # whisker
        #, bg = "White"
        #, alpha = 0.2
        ) +
  xlab("xlab") +
  ylab("ylab")
```




```
#https://moc.online.uni-marburg.de/gitbooks/publicationQualityGraphics/_book/chapters/02_data_vi
sualisation/box_ggplot2.html
```

Beenden und Speichern

```
save(list = ls(all.names = TRUE), file = "z01_Geostatistik_s20_Gr01.RData",
     envir = .GlobalEnv) #gleichbedeutend wie save.image()
#savehistory("z01_Geostatistik_s20_Gr01.Rhistory")
# alternativ quit()  schließt das R-Skript und RStudio wobei auch save.image()
# und savehistory mit ausgefuehrt werden
```

Literatur