TRAINING Computer exam BFVH15DAVUR

Data Analysis and Visualization using R

YOUR NAME (YOUR STUDENT NUMBER)

June 2016

Test header

- **Teacher** Michiel Noback (NOMI), to be reached at +31 50 595 4691
- **Test size** 4 pages; 7 questions
- **Aiding materials** Computer on the BIN network
- **Data files**
  - `food_constituents.txt`
- **Supplementary materials**
  - `TRAINING_EXAM.pdf` This test as pdf
  - `TRAINING_EXAM.Rmd` This test as R markdown
  - `R_cheatsheet.pdf` Lists all R functions that may be used
  - `rmarkdown-reference.pdf` R markdown reference document

Instructions

In the real test, you should be logged in as guest (username = "gast", password = "gast"). On your desktop you will find all supplied data and supplements, as well as the submit script `submit_your_work`. For this training test, simply quit your browser and time your work; in the real exam, you will have two hours to solve a set of similar questions. Use the supplied R markdown file `TRAINING_EXAM.Rmd` to solve and answer the questions of this test. Fill in your name and student number in the header of this document. **Note: never use `echo = False` in your code chunk headers.**
All questions have the possible number of points to be scored indicated. your grade will be calculated as $Grade = 1 + (\frac{PointsScored}{MaximumScore} * 9)$

After finishing, `knit` the result into a pdf document and rename it to `TRAINING_EXAM_YOUR_NAME.pdf`.

Data description

This test explores a dataset containing measurements of several food constituents in a variety of foods, categorized over several groups.

Code "Book"

These are the columns, and their descriptions, included in the data file `food_constituents.txt`:
id.nr Type kcal protein carb.total carb.sugar carb.other fat.total fat.sat fat.unsat fiber Na 2 chocolate 442 5.00 67.40 64.60 2.80 15.50 9.00 6.50 6.60 0.100

1. **id.nr** simple measurement counter
2. **Type** food group
3. **kcal** energy contents in kcal/100g product
4. **protein** protein content in g/100g product
5. **carb.total** total carbohydrate content in g/100g product
6. **carb.sugar** sugar carbohydrates in g/100g product
7. **carb.other** other carbohydrates in g/100g product
8. **fat.total** total fat content in g/100g product

9. **fat.sat** saturated fats in g/100g product
10. **fat.unsat** unsaturated fats in g/100g product
11. **fiber** fiber contents in g/100g product
12. **Na** Sodium content in g/100g product

Here starts the actual test

Part 1: Data loading and cleaning

Question 1 (10 points)

Load the data from file `food_constituents.txt` and assign it to a variable called `foods`. Take special care with missing/invalid fields, and also make sure the columns are loaded in the right data type.

```
data.file <- "./food_constituents.txt"
foods <- read.table(
    file = data.file,
    head = TRUE,
    sep = "\t",
    na.strings = c("*"),
    row.names = 1,
    comment.char = "@"
)
```

If you fail to load the data as instructed above, you may load the pre-processed file using the following code chunk (uncomment the R code). Make sure your working directory is set appropriately! You will not get any points for this question, however.

```
## Uncomment this line to load pre-processed data
load("./foods_raw.Rdata")
```

Question 2 (5 points)

There are several rows with missing data. Report these and also remove these from the `foods` dastaset. Hint: use the function `complete.cases()` to achieve this.

```
#report incomplete cases
foods[!complete.cases(foods), ]
```

```
##     Type kcal protein carb.total carb.sugar carb.other fat.total fat.sat
## 70 pizza  215     8.6       25.7         NA         21       8.6      NA
## 72 pizza  266     9.9       29.2         NA         27      12.2     3.7
##    fat.unsat fiber   Na
## 70        NA   1.7 0.49
## 72       8.5   2.0   NA
```

```
foods <- foods[complete.cases(foods), ]
```

Part 2: Data exploration

Question 3 (6 points)

Question 3 a (2 points) What is the average caloric value of this food listing?

```
mean(foods$kcal)
```

```
## [1] 292.5276
```

Question 3 b (2 points) Tabulate the frequencies of the different food categories (e.g. Type)

```
table(foods$Type)
```

```
## 
##  beverage     bread      cake    cheese     chips chocolate   cookies 
##        16        16         8        18        10        31        24 
##       jam      meat      milk      nuts     pasta     pizza    potato 
##         7        27        12         6        13        21         9 
##      rice vegetable 
##         9        27
```

Question 3 c (2 points) Show the "6-number summary" for -only- the fat measurements.

```
summary(foods[, 7:9])
```

```
##    fat.total        fat.sat          fat.unsat    
##  Min.   : 0.00   Min.   : 0.000   Min.   : 0.00  
##  1st Qu.: 1.00   1st Qu.: 0.200   1st Qu.: 0.60  
##  Median : 9.15   Median : 3.450   Median : 5.60  
##  Mean   :14.18   Mean   : 6.756   Mean   : 7.42  
##  3rd Qu.:26.30   3rd Qu.:11.075   3rd Qu.:11.15  
##  Max.   :51.00   Max.   :33.100   Max.   :41.00
```

Question 4 (12 points)

Question 4 a (4 points) Create a new column called `fat.cat` that divides the foods into 3 food categories based on total fat content: `high.fat`, `medium.fat` and `low.fat`. Take into account that this is an ordinal scale!.

```
foods$fat.cat <- cut(foods$fat.total, breaks = 3, labels = c("low.fat", "medium.fat", "high.fat"), o
```

If you are not able to create this factor, load it from file and attach it to your foods dataframe. You will not get points for this question of course.

```
##uncomment this if you could not create the factor yourself
#load("foods_fat_cat.RData")
```

Question 4 b (4 points) Calculate mean energy content for each fat.cat category.

```
tapply(X=foods$kcal, INDEX=foods$fat.cat, FUN=mean)
```

```
##    low.fat medium.fat   high.fat 
##   198.5472   424.1549   525.7500
```

```
#OR
aggregate(formula = kcal ~ fat.cat, data = foods, FUN = mean)
```

```
##      fat.cat     kcal
## 1    low.fat 198.5472
## 2 medium.fat 424.1549
## 3   high.fat 525.7500
```

Question 4 c (8 points) -Challenge question- Report which foods from each fat.cat group have the largest fraction of saturated fat relative to total fat.

```
#create fraction
foods$sat.fat.fraction <- (foods$fat.sat / foods$fat.total)
#split on fat.cat
split.foods <- split(foods, foods$fat.cat)
#create max reporting function
max.reporting <- function(x) {
    fr.order <- order(x$sat.fat.fraction, na.last = TRUE, decreasing = T)
    #report food
    print(x[fr.order[1], c(1, 2, 7, 8, 9, 12, 13)])
}
lapply(split.foods, max.reporting)
```

```
##   Type kcal fat.total fat.sat fat.unsat fat.cat sat.fat.fraction
## 6  jam  244       0.1     0.1         0 low.fat                1
##       Type kcal fat.total fat.sat fat.unsat    fat.cat sat.fat.fraction
## 176 cheese  265        21      15         6 medium.fat        0.7142857
##      Type kcal fat.total fat.sat fat.unsat  fat.cat sat.fat.fraction
## 405 chips  553      36.8    33.1       3.7 high.fat        0.8994565
```

```
## $low.fat
##   Type kcal fat.total fat.sat fat.unsat fat.cat sat.fat.fraction
## 6  jam  244       0.1     0.1         0 low.fat                1
##
## $medium.fat
##       Type kcal fat.total fat.sat fat.unsat    fat.cat sat.fat.fraction
## 176 cheese  265        21      15         6 medium.fat        0.7142857
##
## $high.fat
##      Type kcal fat.total fat.sat fat.unsat  fat.cat sat.fat.fraction
## 405 chips  553      36.8    33.1       3.7 high.fat        0.8994565
```

Is there anything funny in these results? Discuss/explain these!

Question 5 (8 points)

Sort (and list) the Pasta foods by energy content, from high to low.

```
pastas <- foods[foods$Type == "pasta", ]
pastas[order(pastas$kcal, decreasing = T), ]
```

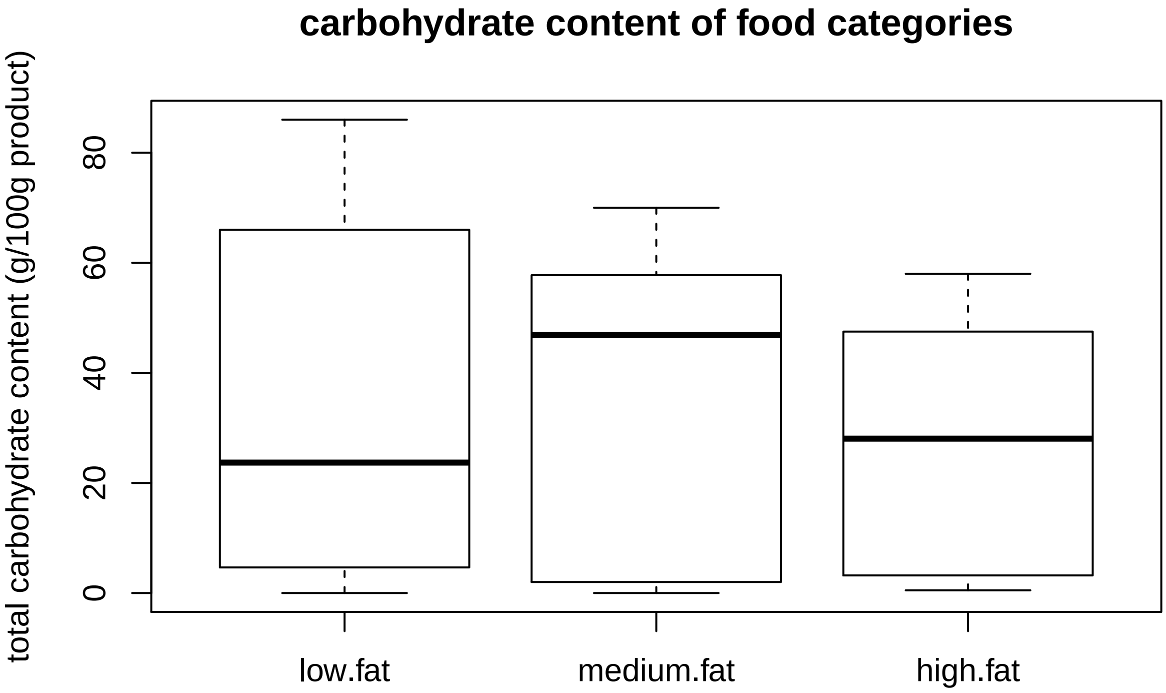
```
##      Type kcal protein carb.total carb.sugar carb.other fat.total fat.sat
## 356 pasta  372    15.0         68        3.0       65.0       3.7     1.2
## 33  pasta  355    12.0         72        2.0       70.0       1.5     0.1
## 40  pasta  355    12.5         73        2.4       70.6       1.4     0.3
## 46  pasta  355    10.7         75        4.7       70.3       1.5     0.5
## 251 pasta  355    12.5         73        2.4       70.6       1.4     0.3
## 279 pasta  355    12.0         72        2.0       70.0       1.5     0.1
## 343 pasta  351    11.0         72        2.0       70.0       1.5     0.1
## 94  pasta  350    11.0         74        2.5       71.5       1.0      0.0
## 361 pasta  350    11.0         74        2.5       71.5       1.0      0.0
## 372 pasta  349    10.5         72        3.5       68.5       1.5      0.1
## 303 pasta  345    12.0         71        2.0       69.0       1.0      0.2
## 348 pasta  340    11.0         69        2.0       67.0       2.0      0.1
## 402 pasta  190     4.5         30        0.1       29.9       5.5      2.5
##     fat.unsat fiber    Na fat.cat sat.fat.fraction
## 356       2.5   3.1 0.030 low.fat       0.32432432
## 33        1.4   2.5 0.010 low.fat       0.06666667
## 40        1.1   2.6 0.000 low.fat       0.21428571
## 46        1.0   1.8 0.050 low.fat       0.33333333
## 251       1.1   2.6 0.000 low.fat       0.21428571
## 279       1.4   2.5 0.010 low.fat       0.06666667
## 343       1.4   2.5 0.010 low.fat       0.06666667
## 94        1.0   2.5 0.000 low.fat       0.00000000
## 361       1.0   2.5 0.000 low.fat       0.00000000
## 372       1.4   2.5 0.385 low.fat       0.06666667
## 303       0.8   3.0 0.010 low.fat       0.20000000
## 348       1.9   3.5 0.010 low.fat       0.05000000
## 402       3.0   0.7 0.160 low.fat       0.45454545
```

Part 3: Visualization

Question 6 (8 points)

Create a -well annotated- box plot showing distributions of total total carbohydrate content for the three fat categories (low.fat, medium.fat and high.fat).

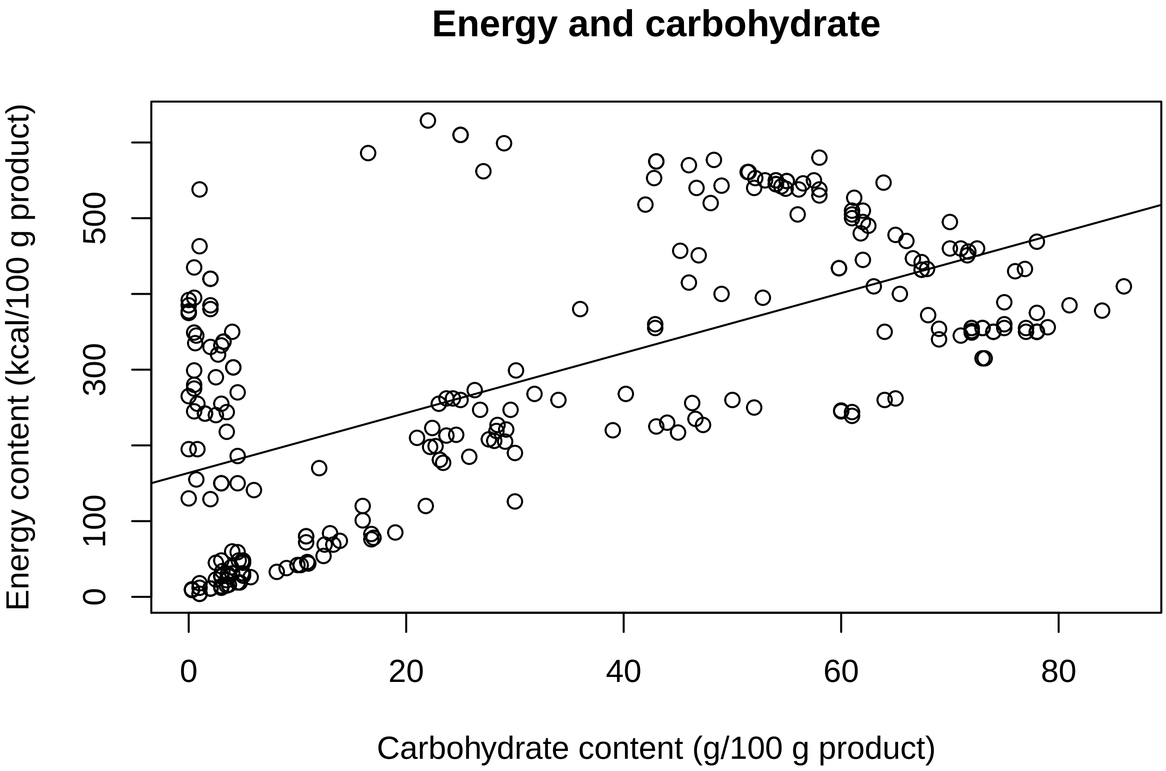
```
boxplot(foods$carb.total ~ foods$fat.cat,
        ylab = "total carbohydrate content (g/100g product)",
        main = "carbohydrate content of food categories")
```



Question 7 (15 points)

Create a -well annotated- scatter plot exploring the total carbohydrate content relative to energy content. You should add a linear regression line to emphasise the relationship.

```
plot(foods$carb.total, foods$kcal,
     xlab = "Carbohydrate content (g/100 g product)",
     ylab = "Energy content (kcal/100 g product)",
     main = "Energy and carbohydrate")
rl <- lm(foods$kcal ~ foods$carb.total)
abline(rl)
```



Is there a clear relationship as you would expect? If not, can you explain?