Data Quality: Lecture 5

Philipp Drebes

30.03.2023

Task 01

How does the outlier pattern look like? Show the characteristics of the data by using visualization technique.

```
library(outliers)
library(readxl)

hse <- read_excel('data/HSE.xlsx')
head(hse)
```

```
## # A tibble: 6 x 2
##   wtval   sex
##   <dbl> <dbl>
## 1  85.2     1
## 2  72.7     1
## 3  84.8     1
## 4  73.6     1
## 5  61.3     1
## 6  81.9     1
```

Key figures & Graphs

```
summary(hse)
```

```
##      wtval             sex
##  Min.   : 35.60   Min.   :1.000
##  1st Qu.: 64.40   1st Qu.:1.000
##  Median : 74.60   Median :2.000
##  Mean   : 76.28   Mean   :1.556
##  3rd Qu.: 86.00   3rd Qu.:2.000
##  Max.   :184.30   Max.   :2.000
##  NA's   :1389
```

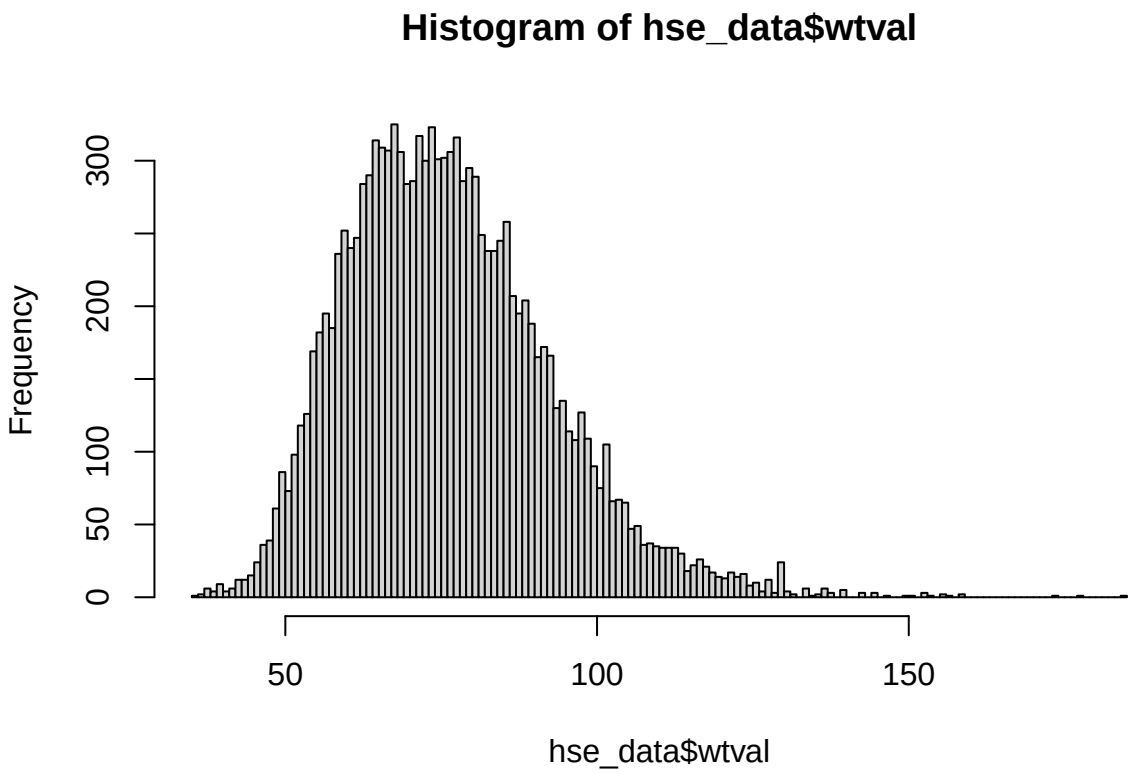
```
min(hse$wtval, na.rm = TRUE)
```

```
## [1] 35.6
```

```
max(hse$wtval, na.rm = TRUE)
```

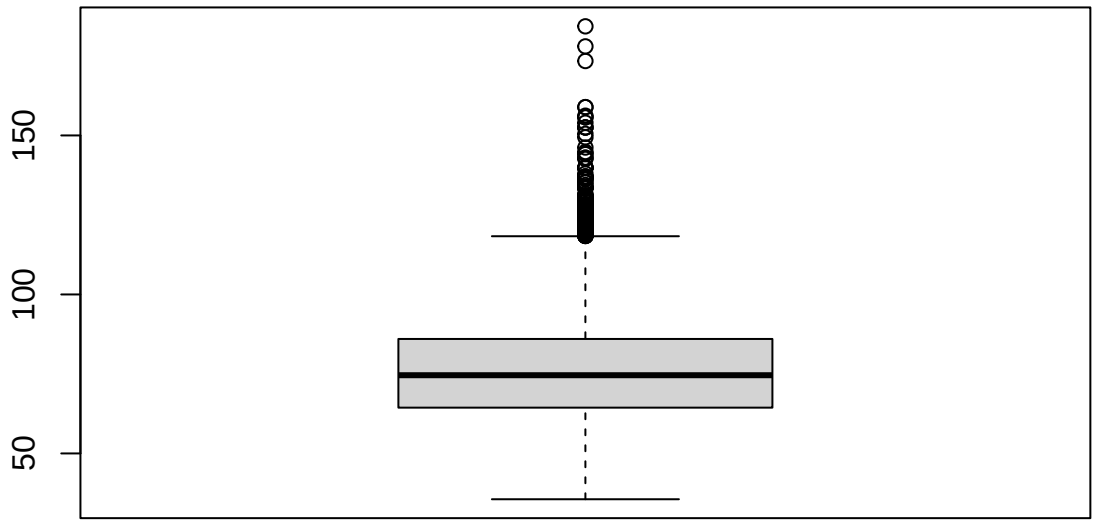
```
## [1] 184.3
```

```
hse_data <- na.omit(hse)
hist(hse_data$wtval, breaks = sqrt(nrow(hse_data)))
```



Box plot

```
boxplot(hse_data$wtval)
```




```
out <- boxplot.stats(hse_data$wtval)$out
length(out)
```

```
## [1] 199
```

We see 199 outliers, of which 3 seem to be more extreme.

```
out_ind <- which(hse_data$wtval %in% c(out))
out_ind
```

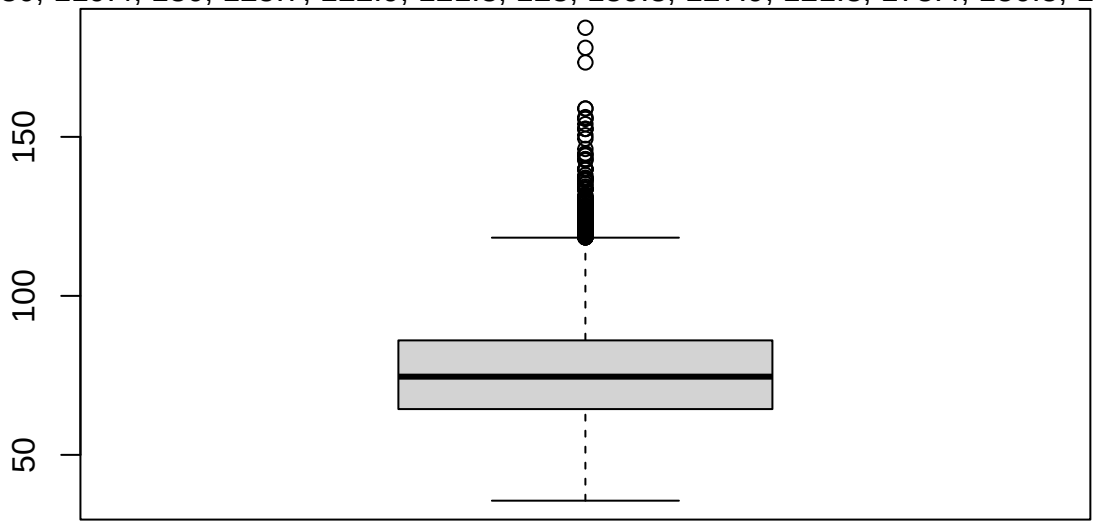
```
##   [1]     8    56    90    95   113   130   136   204   218   253   271   355
##  [13]   412   424   461   566   644   682   685   757  1011  1101  1141  1155
##  [25]  1165  1202  1322  1323  1399  1416  1460  1474  1529  1601  1717  1718
##  [37]  1776  1796  1814  1819  1871  1878  1902  1953  1984  2099  2113  2117
##  [49]  2142  2210  2265  2266  2409  2460  2540  2549  2629  2651  2703  2733
##  [61]  2778  2793  2798  2866  2919  2941  2982  3044  3051  3056  3128  3142
##  [73]  3214  3243  3258  3336  3349  3399  3494  3496  3527  3582  3592  3611
##  [85]  3664  3679  3737  3745  3768  3810  3982  4056  4063  4071  4073  4076
##  [97]  4138  4174  4180  4200  4215  4227  4268  4324  4381  4573  4648  4649
## [109]  4666  4788  4791  4840  4867  4888  5031  5050  5090  5120  5153  5186
## [121]  5264  5281  5298  5299  5388  5392  5654  5681  5817  5821  5838  5887
## [133]  6168  6179  6442  6560  6977  7047  7078  7089  7131  7372  7537  7666
## [145]  7715  7722  7853  7862  7932  7987  8059  8120  8197  8244  8446  8647
## [157]  8701  9139  9160  9186  9347  9396  9453  9463  9477  9509  9707  9995
## [169] 10078 10172 10329 10332 10346 10427 10605 10622 10668 10772 10783 10823
## [181] 10850 10862 10934 11137 11202 11203 11269 11555 11572 11780 11800 11839
## [193] 11860 11937 11977 12010 12079 12183 12320
```

```
hse_data[out_ind, ]
```

```
## # A tibble: 199 x 2
##    wtval   sex
##    <dbl> <dbl>
##  1  123.     1
##  2  126.     1
##  3  130      1
##  4  124.     1
##  5  121      1
##  6  124.     1
##  7  121.     1
##  8  122.     1
##  9  131      1
## 10  123.     1
## # ... with 189 more rows
```

```
boxplot(hse_data$wtval)
mtext(paste("Outliers: ", paste(out, collapse = ", ")))
```

6.7, 130, 119.4, 130, 125.7, 122.9, 121.3, 128, 139.8, 127.9, 121.3, 173.4, 150.5, 134.4,



Percentiles method

```
lower_bound <- quantile(hse_data$wtval, 0.025)
lower_bound
```

```
## 2.5%
##   50
```

```
upper_bound <- quantile(hse_data$wtval, 0.975)
upper_bound
```

```
## 97.5%
## 113.2
```

```
outlier_ind <- which(hse_data$wtval < lower_bound | hse_data$wtval > upper_bound)
outlier_ind
```

```
##   [1]     8    39    42    56    90    95   113   130   136   204   207   218
##  [13]   253   271   355   373   378   412   424   432   441   461   480   500
##  [25]   566   573   609   617   644   682   685   713   738   757   768   772
##  [37]   803   835   868   950  1011  1101  1115  1141  1155  1165  1175  1202
##  [49]  1243  1305  1322  1323  1396  1399  1416  1460  1474  1498  1529  1589
##  [61]  1601  1678  1717  1718  1720  1737  1776  1789  1796  1814  1819  1862
##  [73]  1871  1878  1893  1902  1937  1953  1984  2029  2099  2113  2115  2117
##  [85]  2142  2198  2210  2265  2266  2277  2291  2328  2346  2393  2409  2460
##  [97]  2477  2485  2540  2549  2629  2641  2651  2703  2733  2778  2782  2793
## [109]  2798  2853  2855  2866  2893  2919  2941  2982  3000  3044  3051  3056
## [121]  3115  3116  3128  3142  3214  3217  3229  3235  3243  3258  3336  3338
## [133]  3349  3363  3399  3440  3487  3494  3496  3509  3527  3582  3592  3611
## [145]  3664  3672  3679  3709  3732  3737  3745  3758  3768  3790  3791  3810
## [157]  3862  3886  3888  3909  3969  3982  4023  4046  4056  4063  4068  4071
## [169]  4073  4076  4138  4174  4180  4200  4202  4215  4227  4268  4273  4280
## [181]  4324  4341  4381  4389  4394  4398  4562  4573  4602  4648  4649  4666
## [193]  4788  4791  4821  4830  4840  4855  4867  4872  4888  4922  4925  4955
## [205]  5031  5050  5056  5063  5090  5120  5153  5170  5185  5186  5203  5248
## [217]  5264  5281  5298  5299  5336  5371  5376  5388  5392  5639  5654  5681
## [229]  5687  5718  5721  5725  5755  5813  5815  5817  5820  5821  5831  5836
## [241]  5838  5842  5851  5853  5864  5887  5890  5899  5906  5912  5917  5918
## [253]  5957  5961  5981  6035  6053  6122  6157  6168  6179  6194  6212  6214
## [265]  6233  6308  6320  6324  6346  6394  6413  6416  6438  6442  6477  6485
## [277]  6507  6509  6540  6546  6560  6583  6595  6596  6597  6665  6674  6679
## [289]  6692  6730  6737  6747  6788  6829  6830  6832  6861  6869  6882  6888
## [301]  6889  6919  6939  6961  6969  6977  7027  7035  7047  7071  7078  7083
## [313]  7089  7091  7092  7124  7131  7237  7267  7272  7274  7322  7360  7362
## [325]  7364  7372  7418  7447  7477  7521  7537  7574  7580  7587  7625  7635
## [337]  7666  7686  7694  7695  7697  7711  7713  7715  7722  7724  7769  7776
## [349]  7796  7837  7853  7862  7868  7880  7920  7932  7937  7987  8025  8038
## [361]  8059  8120  8125  8128  8147  8154  8184  8197  8231  8237  8238  8244
## [373]  8287  8294  8299  8341  8360  8399  8412  8415  8417  8443  8446  8450
## [385]  8474  8522  8528  8535  8554  8588  8590  8617  8640  8647  8667  8676
## [397]  8679  8701  8734  8739  8768  8784  8825  8863  8877  8891  8901  8912
## [409]  8927  9012  9038  9045  9049  9067  9094  9099  9108  9119  9126  9139
## [421]  9160  9178  9184  9186  9189  9194  9269  9272  9278  9294  9317  9347
## [433]  9359  9391  9396  9413  9419  9440  9453  9463  9477  9509  9521  9531
## [445]  9538  9548  9594  9634  9644  9650  9691  9707  9760  9767  9786  9860
## [457]  9875  9879  9934  9944  9976  9982  9995 10018 10037 10051 10054 10072
## [469] 10078 10136 10141 10149 10150 10172 10179 10202 10203 10218 10220 10225
## [481] 10235 10287 10326 10329 10332 10346 10366 10399 10414 10424 10427 10447
## [493] 10457 10459 10475 10481 10507 10527 10528 10568 10605 10622 10643 10668
## [505] 10670 10678 10686 10693 10711 10749 10765 10772 10783 10799 10811 10820
```

```
## [517] 10823 10827 10850 10862 10869 10906 10921 10934 10945 10961 10994 11054
## [529] 11057 11083 11107 11109 11110 11113 11116 11137 11144 11155 11182 11185
## [541] 11202 11203 11226 11269 11276 11277 11281 11301 11361 11370 11379 11452
## [553] 11482 11521 11526 11539 11545 11548 11555 11572 11589 11591 11592 11594
## [565] 11620 11628 11642 11667 11735 11746 11748 11754 11772 11780 11800 11811
## [577] 11838 11839 11852 11856 11860 11894 11932 11937 11939 11950 11956 11960
## [589] 11977 12003 12010 12032 12045 12072 12079 12086 12098 12103 12107 12109
## [601] 12111 12129 12134 12163 12183 12196 12198 12239 12246 12250 12292 12296
## [613] 12320 12339 12365 12380 12393 12416 12441 12451
```

Task 02

Perform an outlier test according to Grubbs. Also reflect on the hypothesis structure that goes with this test.

Perform an outlier test according to Rosner.

```
library(readxl)
library(outliers)
library(EnvStats, quietly = T)

hse <- read_excel('data/HSE.xlsx')
hse_data <- na.omit(hse)
head(hse_data, 10)
```

```
## # A tibble: 10 x 2
##    wtval   sex
##    <dbl> <dbl>
##  1  85.2     1
##  2  72.7     1
##  3  84.8     1
##  4  73.6     1
##  5  61.3     1
##  6  81.9     1
##  7  60       1
##  8 123.      1
##  9  98.4     1
## 10  73.3     1
```

Grubbs

```
grubbs.test(hse_data$wtval)
```

```
##
##  Grubbs test for one outlier
##
## data:  hse_data$wtval
## G = 6.61164, U = 0.99651, p-value = 2.29e-07
## alternative hypothesis: highest value 184.3 is an outlier
```

```
grubbs.test(hse_data$wtval, opposite = TRUE)
```

```
##
##  Grubbs test for one outlier
##
```

```
## data:  hse_data$wtval
## G = 2.4903, U = 0.9995, p-value = 1
## alternative hypothesis: lowest value 35.6 is an outlier
```

We assume the data is normal distributed. Then the Grubbs test for one outlier indicates that there is strong evidence to suggest that the highest value of 184.3 is an outlier. There is no significant evidence to suggest that the lowest value of 35.6 is an outlier.

Rosner

```
test.rosner <- rosnerTest(hse_data$wtval, k = 25)
```

```
## Warning in rosnerTest(hse_data$wtval, k = 25): The true Type I error may be larger than assumed.
## Although the help file for 'rosnerTest' has a table with information
## on the estimated Type I error level,
## simulations were not run for k > 10 or k > floor(n/2).
```

```
test.rosner$all.stats
```

```
##     i   Mean.i     SD.i Value Obs.Num    R.i+1 lambda.i+1 Outlier
## 1   0 76.28451 16.33717 184.3   11203 6.611640   4.609832    TRUE
## 2   1 76.27588 16.30927 178.0    5120 6.237196   4.609815    TRUE
## 3   2 76.26776 16.28455 173.4    4268 5.964685   4.609798    TRUE
## 4   3 76.26000 16.26204 158.9    3582 5.081773   4.609782    TRUE
## 5   4 76.25339 16.24590 158.9    5838 5.087228   4.609765    TRUE
## 6   5 76.24679 16.22974 156.2   11780 4.926339   4.609748    TRUE
## 7   6 76.24040 16.21464 155.7    4791 4.900485   4.609731    TRUE
## 8   7 76.23405 16.19972 155.7    7537 4.905391   4.609715    TRUE
## 9   8 76.22770 16.18478 154.0    1796 4.805274   4.609698    TRUE
## 10  9 76.22148 16.17048 152.5    2982 4.717146   4.609681    TRUE
## 11 10 76.21538 16.15674 152.5    5388 4.721536   4.609664    TRUE
## 12 11 76.20928 16.14298 152.5   10329 4.725939   4.609648    TRUE
## 13 12 76.20318 16.12920 150.5    4324 4.606355   4.609631   FALSE
## 14 13 76.19724 16.11615 149.4    2703 4.542198   4.609614   FALSE
## 15 14 76.19139 16.10350 146.2    1717 4.347417   4.609597   FALSE
## 16 15 76.18579 16.09196 144.8    3496 4.263880   4.609581   FALSE
## 17 16 76.18030 16.08090 144.4    2117 4.242281   4.609564   FALSE
## 18 17 76.17484 16.06997 144.1    2265 4.226839   4.609547   FALSE
## 19 18 76.16941 16.05912 143.0    7862 4.161535   4.609530   FALSE
## 20 19 76.16406 16.04863 143.0    9463 4.164588   4.609514   FALSE
## 21 20 76.15872 16.03813 142.6    2629 4.142707   4.609497   FALSE
## 22 21 76.15340 16.02776 140.0     355 3.983501   4.609480   FALSE
## 23 22 76.14829 16.01822 139.8    3679 3.973707   4.609463   FALSE
## 24 23 76.14320 16.00874 139.8    4200 3.976379   4.609447   FALSE
## 25 24 76.13810 15.99924 139.8    4867 3.979057   4.609430   FALSE
```

We assume the data is normal distributed. Then for the RosnerTest the first 12 observations (which are the furthest away from the median) are indeed outliers. All values above 152.5 are outliers. Values below are marked as NON-outliers by this test.