BB503/BB602 - R Training - Week III

Introduction

This week, we will be working with the AIDS data set[1]. Let's start by reading the data. Notice that we had to specify that the separator (via the `sep` argument) is spaces (instead of the default of `read.delim()` which is tabs).

```
aids_df <- read.delim("../data/aids_dataset.txt", sep = " ")
```

For any function in R, we can use `?function_name` to view the help page for that function:

```
?read.delim
```

We can view the first 6 (by default) lines using the function `head()` and, similarly, the last lines using the `tail()` function:

```
head(aids_df)
```

```
##   id treatment   age gender week_1 cd4_1 week_2 cd4_2
## 1  1      trt2 36.43   male      0    23   7.57    21
## 2  2      trt4 47.85   male      0    21   8.00    49
## 3  4      trt3 36.60   male      0    61   7.14    61
## 4  5      trt1 35.95   male      0    36   8.00    31
## 5  6      trt2 38.40   male      0    11   7.29    11
## 6  7      trt2 45.08   male      0    11   9.00    41
```

```
# print first 2 lines
head(aids_df, 2)
```

```
##   id treatment   age gender week_1 cd4_1 week_2 cd4_2
## 1  1      trt2 36.43   male      0    23   7.57    21
## 2  2      trt4 47.85   male      0    21   8.00    49
```

```
tail(aids_df)
```

```
##        id treatment   age gender week_1 cd4_1 week_2 cd4_2
## 1173 1307      trt3 14.90   male      0    11   4.14    15
## 1174 1308      trt1 30.75   male      0     4   8.71     6
## 1175 1309      trt3 39.46   male      0     9   8.86    21
## 1176 1311      trt4 53.65   male      0     9   8.14     8
## 1177 1312      trt1 42.24   male      0    27   7.71    15
## 1178 1313      trt1 15.84 female      0   146   7.29    64
```

Using `str()`, we can display the structure of our data frame `aids_df`. For later use, we turn the variables `treatment` and `gender` into factors.

```
str(aids_df)
```

```
## 'data.frame':    1178 obs. of  8 variables:
##  $ id       : int  1 2 4 5 6 7 8 11 12 13 ...
```

[1]Source: Applied longitudinal analysis [Internet]. [cited 2021 Sep 27]. Available from: https://content.sph.harvard.edu/fitzmaur/ala2e/

```
##  $ treatment: chr  "trt2" "trt4" "trt3" "trt1" ...
##  $ age      : num  36.4 47.9 36.6 36 38.4 ...
##  $ gender   : chr  "male" "male" "male" "male" ...
##  $ week_1   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ cd4_1    : int  23 21 61 36 11 11 16 16 46 1 ...
##  $ week_2   : num  7.57 8 7.14 8 7.29 ...
##  $ cd4_2    : int  21 49 61 31 11 41 11 21 51 1 ...
```

```
aids_df$treatment <- as.factor(aids_df$treatment)
aids_df$gender <- as.factor(aids_df$gender)
str(aids_df)
```

```
## 'data.frame':    1178 obs. of  8 variables:
##  $ id       : int  1 2 4 5 6 7 8 11 12 13 ...
##  $ treatment: Factor w/ 4 levels "trt1","trt2",..: 2 4 3 1 2 2 3 2 4 4 ...
##  $ age      : num  36.4 47.9 36.6 36 38.4 ...
##  $ gender   : Factor w/ 2 levels "female","male": 2 2 2 2 2 2 2 2 2 2 ...
##  $ week_1   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ cd4_1    : int  23 21 61 36 11 11 16 16 46 1 ...
##  $ week_2   : num  7.57 8 7.14 8 7.29 ...
##  $ cd4_2    : int  21 49 61 31 11 41 11 21 51 1 ...
```

Examining Distributions

Mean/Median/Mode

We can inspect the mean of a vector using the function `mean()`:

```
# mean of age
mean(aids_df$age)
```

```
## [1] 37.683
```

```
# mean of CD4 count at week 1
mean(aids_df$cd4_1)
```

```
## [1] 26.511
```

Similarly, we can calculate the median of a vector using the function `median()` :

```
# median of CD4 count at week 2
median(aids_df$cd4_2)
```

```
## [1] 21
```

Mode can be calculated using the function `table()` (calculates the frequency of each value):

```
tbl <- table(aids_df$cd4_1)
sort(tbl, decreasing = TRUE)
```

```
##
##  11  21  13   9  16  31   5   7  17  41   6  23  26  25  33   8  29  51   3  15
##  83  64  46  40  40  39  37  37  34  29  28  25  25  24  23  22  22  22  21  21
##   1  27  36  19  39  20  28  10  18  37   4  35  12  14  22  46  30  45  49  32
##  20  20  20  19  18  17  17  16  16  16  15  15  14  14  14  13  12  12  12  11
##  40  43  61   2  47  24  56  34  38  53  59  73  48  52  54  68  69  70  81  42
##  11  11  11  10  10   8   8   6   5   5   5   5   4   4   4   4   4   4   4   3
##  44  50  55  63  65  66  71  85  60  64  67  79  87 109  76  80  82  83  90  91
##   3   3   3   3   3   3   3   3   2   2   2   2   2   2   1   1   1   1   1   1
```

```
##  93  94  96  97  98 106 107 111 116 117 138 139 146 157 166 175 181
##   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
```

Quantiles

For calculating quantiles given a variable (a vector), we use the `quantile()` function:

```
# 10th, 25th, 33rd and 78th percentiles of age
quantile(aids_df$age, probs = c(0.1, 0.25, 0.33, 0.78))
```

```
##    10%    25%    33%    78%
## 28.411 31.770 33.444 43.391
```

By default, R uses "type 7" out of the 9 available algorithms for calculating quantiles. We can change this by altering the `type` argument:

```
# 10th, 25th, 33rd and 78th percentiles of age - type 2
quantile(aids_df$age, probs = c(0.1, 0.25, 0.33, 0.78), type = 2)
```

```
##   10%   25%   33%   78%
## 28.39 31.76 33.44 43.39
```

Spread

Range

```
# range of CD4 counts at week 1
max(aids_df$cd4_1) - min(aids_df$cd4_1)
```

```
## [1] 180
```

```
diff(range(aids_df$cd4_1))
```

```
## [1] 180
```

Inter-quantile range

```
### IQR of CD4 counts at week 1
# calculate Q1 and Q3
quantile(aids_df$cd4_1, probs = c(0.25, 0.75))
```

```
## 25% 75%
##  11  36
```

```
# IQR = Q3 - Q1
36 - 11
```

```
## [1] 25
```

alternatively:

```
IQR(aids_df$cd4_1)
```

```
## [1] 25
```

```
IQR(aids_df$cd4_1, type = 2)
```

```
## [1] 25
```

Variance and standard deviation

```
# Variance of CD4 counts at week 1
var(aids_df$cd4_1)
```

```
## [1] 476.82
```

```
# Std. deviation of CD4 counts at week 1
sqrt(var(aids_df$cd4_1))
```

```
## [1] 21.836
```

```
sd(aids_df$cd4_1)
```

```
## [1] 21.836
```

Outliers

We can manually determine outliers in a variable using the definition provided in the slides.

```
### Outliers for CD4 counts at weeek 1
Q1 <- quantile(aids_df$cd4_1, 0.25)
Q3 <- quantile(aids_df$cd4_1, 0.75)

IQR_val <- IQR(aids_df$cd4_1)

UL <- Q3 + 1.5 * IQR_val
LL <- Q1 - 1.5 * IQR_val

cond <- aids_df$cd4_1 > UL | aids_df$cd4_1 < LL
table(cond)
```

```
## cond
## FALSE  TRUE
##  1142    36
```

```
which(cond)
```

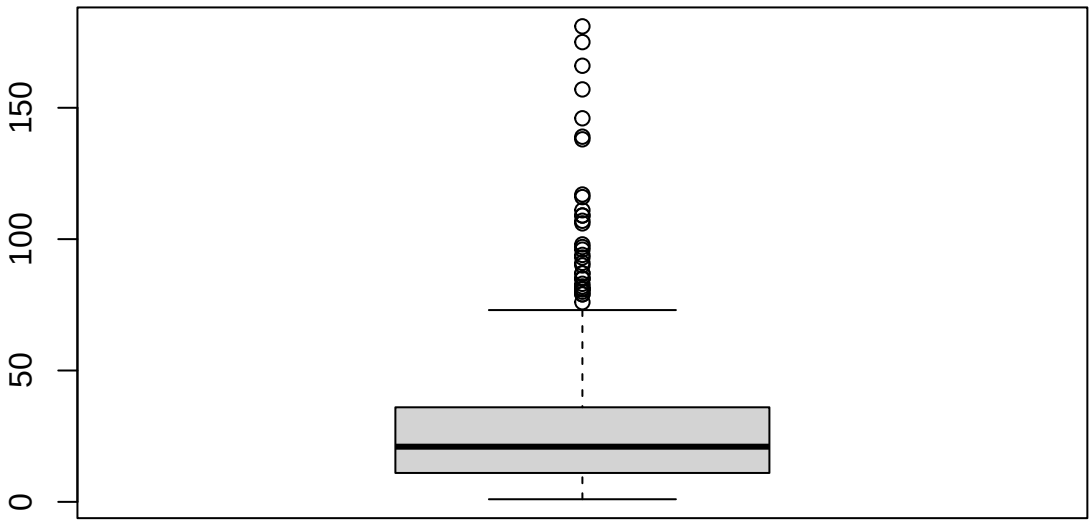
```
##  [1]   19  141  143  166  170  363  400  497  522  527  535  539  540  555  560
## [16]  575  587  640  678  692  727  800  853  858  877  899  900  942  958  970
## [31] 1050 1065 1073 1095 1120 1178
```

```
aids_df$cd4_1[cond]
```

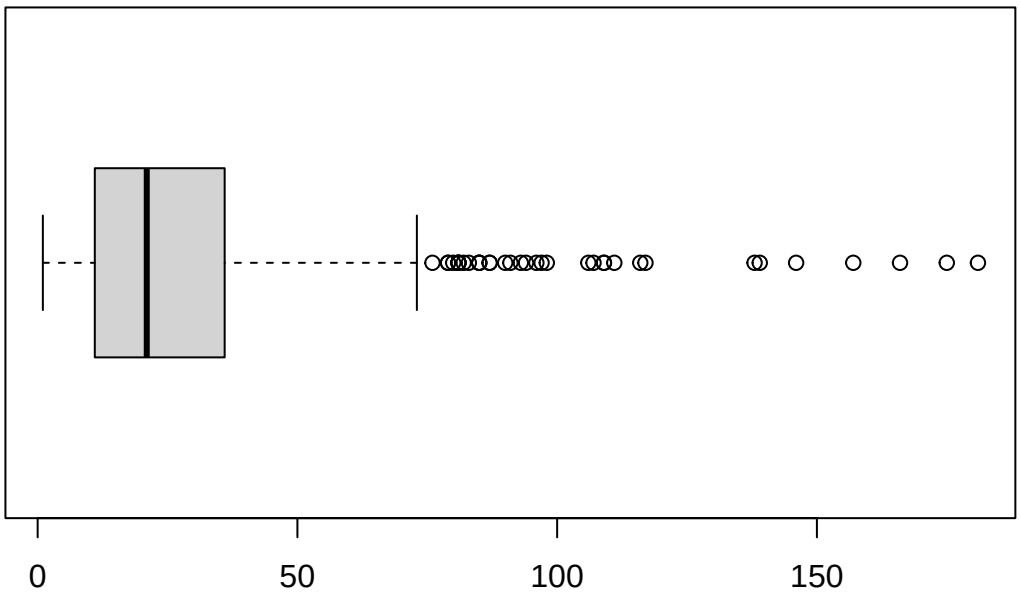
```
##  [1] 116  79 157  85 181 107 109  93  82 138 111  76  81 166  96 117  91  85 175
## [20]  80  90  81  83  79  87 109  98  85  94  97  87 139 106  81  81 146
```

We can visualize the boxplot of this variable to see the outlier values:

```
boxplot(aids_df$cd4_1)
```




```
boxplot(aids_df$cd4_1, horizontal = TRUE)
```



Using the function `boxplot.stats()`, to gather and display the statistics necessary for producing box plots, including outliers.

```
boxplot.stats(aids_df$cd4_1)
```

```
## $stats
## [1]  1 11 21 36 73
##
## $n
## [1] 1178
##
## $conf
## [1] 19.849 22.151
##
## $out
##  [1] 116  79 157  85 181 107 109  93  82 138 111  76  81 166  96 117  91  85 175
## [20]  80  90  81  83  79  87 109  98  85  94  97  87 139 106  81  81 146
```

The `summary()` function

```
summary(aids_df$cd4_1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     1.0    11.0    21.0    26.5    36.0   181.0
```

```
summary(aids_df)
```

```
##        id       treatment       age          gender         week_1
```

```
##  Min.   :   1   trt1:289   Min.   :14.9   female: 142   Min.   :0
##  1st Qu.: 331   trt2:288   1st Qu.:31.8   male  :1036   1st Qu.:0
##  Median : 650   trt3:293   Median :36.8                 Median :0
##  Mean   : 659   trt4:308   Mean   :37.7                 Mean   :0
##  3rd Qu.: 993              3rd Qu.:42.5                 3rd Qu.:0
##  Max.   :1313              Max.   :74.2                 Max.   :0
##      cd4_1           week_2          cd4_2
##  Min.   :  1.0   Min.   : 2.14   Min.   :  1.0
##  1st Qu.: 11.0   1st Qu.: 7.86   1st Qu.: 11.0
##  Median : 21.0   Median : 8.14   Median : 21.0
##  Mean   : 26.5   Mean   :10.12   Mean   : 36.7
##  3rd Qu.: 36.0   3rd Qu.:10.54   3rd Qu.: 43.0
##  Max.   :181.0   Max.   :38.00   Max.   :543.0
```

The "improved" summary function

```
new_summary <- function(x){
    out <- list(
        min = min(x),
        max = max(x),
        quants = quantile(x, prob = c(0.1, 0.25, 0.5, 0.75, 0.9),
                          type = 2),
        mean    = mean(x),
        var     = var(x),
        std_dev = sd(x),
        length  = length(x)
    )
  class(out) <- "new_summary"
  return(out)
}

print.new_summary <- function(object, ...){
    cat("Min:", object$min, "\n")
    cat("10th percentile:", object$quants[1], "\n")
    cat("25th percentile:", object$quants[2], "\n")
    cat("50th percentile (median):", object$quants[3], "\n")
    cat("Mean:", object$mean, "\n")
    cat("75th percentile:", object$quants[4], "\n")
    cat("90th percentile:", object$quants[5], "\n")
    cat("Max:", object$max, "\n")
    cat("Var:", object$var, "\n")
    cat("Sd:", object$std_dev, "\n")
    cat("Length:", object$length, "\n")
}
```

Examples:

```
new_summary(aids_df$cd4_1)
```

```
## Min: 1
## 10th percentile: 6
## 25th percentile: 11
## 50th percentile (median): 21
## Mean: 26.511
## 75th percentile: 36
```

```
## 90th percentile: 51
## Max: 181
## Var: 476.82
## Sd: 21.836
## Length: 1178
```

```
new_summary(aids_df[aids_df$treatment == "trt1", "cd4_1"])
```

```
## Min: 1
## 10th percentile: 7
## 25th percentile: 12
## 50th percentile (median): 21
## Mean: 25.619
## 75th percentile: 35
## 90th percentile: 51
## Max: 146
## Var: 356.85
## Sd: 18.891
## Length: 289
```

```
new_summary(aids_df[aids_df$treatment == "trt2", "cd4_1"])
```

```
## Min: 1
## 10th percentile: 6
## 25th percentile: 11
## 50th percentile (median): 22
## Mean: 27.26
## 75th percentile: 37
## 90th percentile: 51
## Max: 175
## Var: 496.19
## Sd: 22.275
## Length: 288
```

```
new_summary(aids_df[aids_df$treatment == "trt3", "cd4_1"])
```

```
## Min: 1
## 10th percentile: 6
## 25th percentile: 11
## 50th percentile (median): 21
## Mean: 27.416
## 75th percentile: 37
## 90th percentile: 53
## Max: 181
## Var: 559.54
## Sd: 23.655
## Length: 293
```

```
new_summary(aids_df[aids_df$treatment == "trt4", "cd4_1"])
```

```
## Min: 1
## 10th percentile: 5
## 25th percentile: 11
## 50th percentile (median): 21
## Mean: 25.786
## 75th percentile: 34.5
## 90th percentile: 51
```

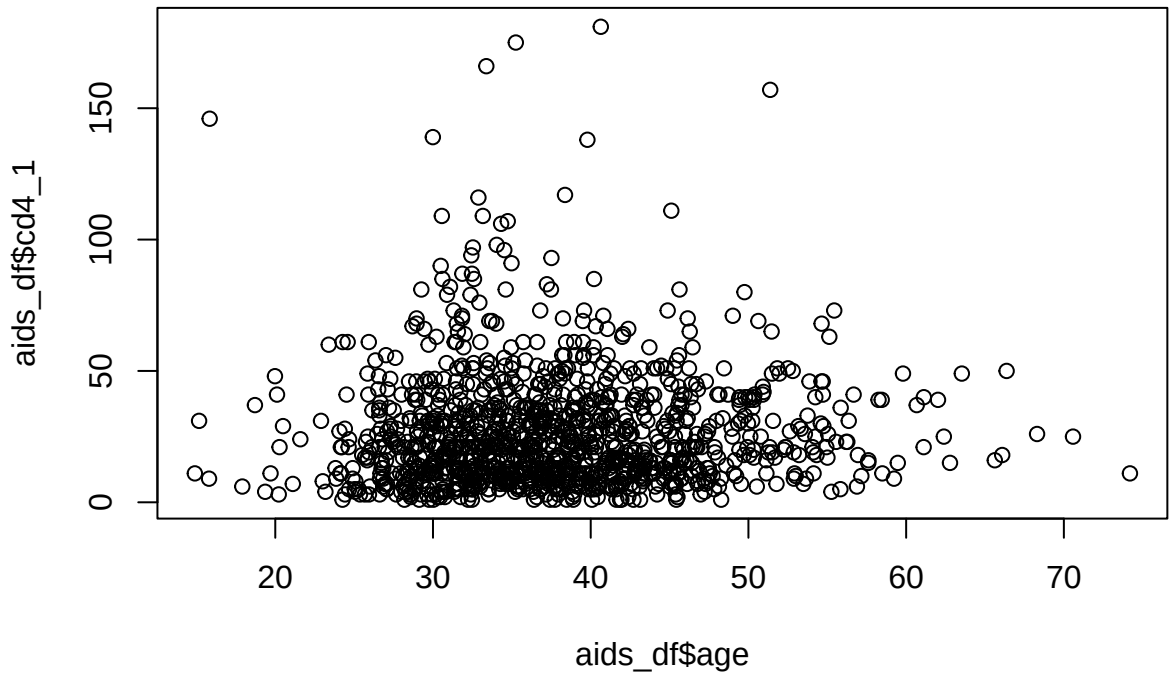
```
## Max: 157
## Var: 494.64
## Sd: 22.241
## Length: 308
```

Examining Relationships

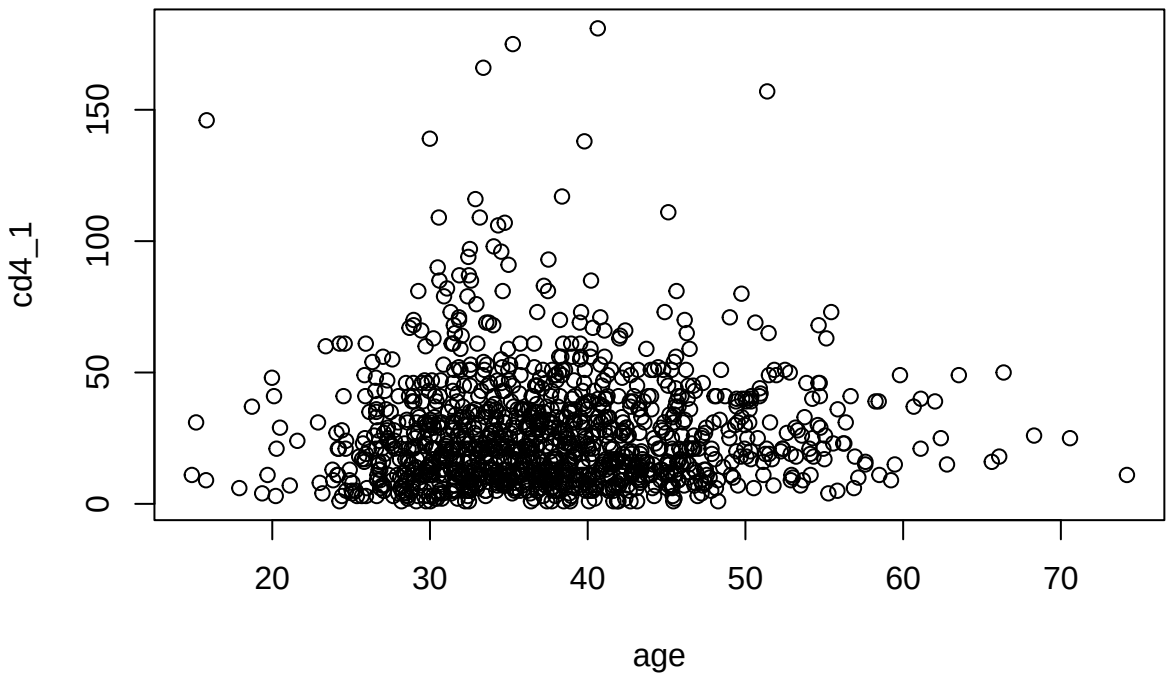
Scatter plots

Age vs. baseline (week 1) CD4 cell counts:

```
plot(aids_df$age, aids_df$cd4_1)
```




```
plot(cd4_1 ~ age, aids_df)
```



Correlation

Pearson Correlation

```
?cor
cor(aids_df$age, aids_df$cd4_1)
```

```
## [1] 0.024636
```

Spearman's Rank Correlation

```
cor(aids_df$age, aids_df$cd4_1, method = "spearman")
```

```
## [1] 0.063961
```

Correlation Test

```
cor.test(aids_df$age, aids_df$cd4_1, method = "spearman")
```

```
## Warning in cor.test.default(aids_df$age, aids_df$cd4_1, method = "spearman"):
## Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  aids_df$age and aids_df$cd4_1
## S = 2.55e+08, p-value = 0.028
```

```
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.063961
```