Wrap Up 2015-06-08

Data

Load Data

- first we have to load the data (do not forget to set the working directory first)

```
load("201404zeiten.rdata")
```

- the file *201404zeiten.rdata* contains seven data frame, each one with the same structure, I choose **t179**
- we can see all on them using the **ls()** function (ls stand for *list*)

```
ls()
```

```
## [1] "t098" "t151" "t152" "t158" "t159" "t179" "t320"
```

- then we use the **head()** function to have a look on the first six lines of one of the

```
head(t179)
```

```
##    SIC              BEGINN                ENDE visnr
## 1 1175 2012-11-06 12:35:03 2012-11-06 12:36:24     1
## 2  219 2012-11-07 09:16:50 2012-11-07 09:19:22     1
## 3  190 2012-11-07 09:57:57 2012-11-07 10:01:03     1
## 4  256 2011-11-19 13:10:27 2011-11-19 13:11:43     1
## 5 1206 2012-11-07 10:40:23 2012-11-07 10:42:22     1
## 6 1183 2012-11-07 10:55:59 2012-11-07 10:58:47     1
```

- we see four columns named SIC, BEGINN, ENDE and visnr; the first numbers (we see 1-6) are only the row names
- to inspect what data types (numeric or character or logical values) each of the columns has we type **str()** (str stands for structure)

```
str(t179)
```

```
## Classes 'grouped_df', 'tbl_df', 'tbl' and 'data.frame':  3774 obs. of  4 variables:
##  $ SIC   : num  1175 219 190 256 1206 ...
##  $ BEGINN: POSIXct, format: "2012-11-06 12:35:03" "2012-11-07 09:16:50" ...
##  $ ENDE  : POSIXct, format: "2012-11-06 12:36:24" "2012-11-07 09:19:22" ...
##  $ visnr : int  1 1 1 1 1 1 1 1 2 ...
##  - attr(*, "vars")=List of 1
##   ..$ : symbol SIC
##  - attr(*, "labels")='data.frame':   2104 obs. of  1 variable:
##   ..$ SIC: num  1 2 3 4 5 6 7 8 9 10 ...
##   ..- attr(*, "vars")=List of 1
##   .. ..$ : symbol SIC
##  - attr(*, "indices")=List of 2104
```

```
##   ..$ : int  861 1634 1992
##   ..$ : int  162 2110
##   ..$ : int  71 2268
##   ..$ : int 213
##   ..$ : int  1681 3673
##   ..$ : int 72
##   ..$ : int  2592 3766
##   ..$ : int  517 1307 2348
##   ..$ : int 1251
##   ..$ : int  822 1440 2975
##   ..$ : int  452 1254
##   ..$ : int  932 1700 3234
##   ..$ : int  859 1639
##   ..$ : int 1444
##   ..$ : int  632 1379 2670
##   ..$ : int 2513
##   ..$ : int 2524
##   ..$ : int  1051 1793 3309
##   ..$ : int  756 2568 2931
##   ..$ : int 2546
##   ..$ : int  1054 1798 3302
##   ..$ : int 2535
##   ..$ : int 1487
##   ..$ : int  694 1225
##   ..$ : int  125 2136 2556
##   ..$ : int 2833
##   ..$ : int  950 1735 2557
##   ..$ : int  1558 2233 2572 3492
##   ..$ : int 2514
##   ..$ : int  784 1591 2893
##   ..$ : int  2574 3059 3443 3611
##   ..$ : int  501 1305 2491
##   ..$ : int 1690
##   ..$ : int  783 1592 2883
##   ..$ : int 2573
##   ..$ : int 1421
##   ..$ : int  672 1304 2494
##   ..$ : int  752 2515 2916
##   ..$ : int 2540
##   ..$ : int  674 1422 2516 3087
##   ..$ : int 753
##   ..$ : int 2517
##   ..$ : int  1011 1664 2518 3258
##   ..$ : int 2520
##   ..$ : int 2519
##   ..$ : int  141 2188 2521
##   ..$ : int  1858 2256 2523 3359
##   ..$ : int  2522 2602 3118 3721
##   ..$ : int  995 1768
##   ..$ : int  2528 3073 3255
##   ..$ : int 1767
##   ..$ : int 3554
##   ..$ : int  1784 2527 3206
##   ..$ : int 2525
```

```
##   ..$ : int  22 2526 3125 3710
##   ..$ : int 3444
##   ..$ : int  2530 2696 3002 3613
##   ..$ : int 2529
##   ..$ : int  2533 2584 2726
##   ..$ : int 2532
##   ..$ : int 2531
##   ..$ : int  472 1252
##   ..$ : int  1712 2211
##   ..$ : int 1633
##   ..$ : int  2536 2826 3046
##   ..$ : int  2593 3121
##   ..$ : int 2534
##   ..$ : int 2537
##   ..$ : int  1638 2641
##   ..$ : int  1667 2145 3406
##   ..$ : int 1691
##   ..$ : int  910 1705 2538 3278
##   ..$ : int  2541 2827 3045
##   ..$ : int  1827 3410
##   ..$ : int  911 1703 2539 3279
##   ..$ : int  1039 1782 3378
##   ..$ : int  1650 1914 2544
##   ..$ : int 2545
##   ..$ : int  2542 2582 3119 3674
##   ..$ : int  827 1749 1933 2543
##   ..$ : int  2548 3042 3543
##   ..$ : int  2547 3044 3606
##   ..$ : int  2333 3060 3515
##   ..$ : int 2561
##   ..$ : int 2550
##   ..$ : int  1884 2555 2580
##   ..$ : int  127 2062 2554
##   ..$ : int  2356 2549 3062 3512
##   ..$ : int 2551
##   ..$ : int  1866 2552 2851 3294
##   ..$ : int  68 899 1464 1952
##   ..$ : int  1865 2553 2862 3298
##   ..$ : int  1787 3151 3343
##   ..$ : int  897 1747 1950 2562
##   ..$ : int 2558
##   ..$ : int  1847 2560
##   ..$ : int  881 1551 1942
##   ..$ : int  809 1791
##   ..$ : int  1872 2438 2559
##   .. [list output truncated]
```

- we got a lot of information here and we look only at some of them:
- at the end of line one the number of observations (=rows) and variables (=columns) are given, which is in my case 3774 and 4 resp.
- the next four lines describe the columns, first the name followed by the type and the first elements:
  - the column SIC is numeric (**num**) and the first elements are 1175 219 190 256 1206
  - the next two columns have the type **POSIXct** - this is a date time format (ct stands for calender

time), when we look at the first element we see it is containing a date as well as time
  - visnr is the visit number and has the type int, int stands for integer and means it is numeric without any decimal place
- everything else we ignore (for now)

calculate the duration

- now we want to calculate the duration (which is ENDE - BEGINN)
- first remember how to access a column of a table:
- the table name is **t179** and we have to use the **$**-sign
- so accessing the BEGINN column of t179 can be down by **t179$BEGINN** (I use head() in addition because the vector is so long. . .)

```
head(t179$BEGINN)
```

```
## [1] "2012-11-06 12:35:03 CET" "2012-11-07 09:16:50 CET"
## [3] "2012-11-07 09:57:57 CET" "2011-11-19 13:10:27 CET"
## [5] "2012-11-07 10:40:23 CET" "2012-11-07 10:55:59 CET"
```

- you have to type both, the name of the table (data frame) and the name of the column, only typing BEGINN will give an error
- Also you have really type `t179$BEGINN`, when you type `T179$beginn` or `t179$Beginn` you will also get an error
- now we know how two access a column, therefore we can subtract one from the other, There is no fancy command to do this we use the **minus sign (-)**

```
t179$ENDE-t179$BEGINN
```

- doing this we print out the resulting vector (all differences one after another, you can do that but for this document it would be to much)
- we need the the results as column in our table, so we have to create a new column and again we have to use the name of the table **t179** and the **$** sign to do this
- so we choose a name, say **duration** and put it in the *table-column* syntax: **t179$duration**
- and we have to put data in this new vector using the arrow **<-**
- and of course we need our difference vector as content, so let's put the things together

```
t179$duration <- t179$ENDE-t179$BEGINN
```

- and now we have a look on the result

```
head(t179)
```

```
##    SIC              BEGINN               ENDE visnr duration
## 1 1175 2012-11-06 12:35:03 2012-11-06 12:36:24     1  81 secs
## 2  219 2012-11-07 09:16:50 2012-11-07 09:19:22     1 152 secs
## 3  190 2012-11-07 09:57:57 2012-11-07 10:01:03     1 186 secs
## 4  256 2011-11-19 13:10:27 2011-11-19 13:11:43     1  76 secs
## 5 1206 2012-11-07 10:40:23 2012-11-07 10:42:22     1 119 secs
## 6 1183 2012-11-07 10:55:59 2012-11-07 10:58:47     1 168 secs
```

- and there it is: our new column **duration**
- we see that it consists of a number and *secs*
- if we type **summary(t179$duration)** (which gives a summary of the vector)

```
summary(t179$duration)
```

```
##   Length    Class     Mode
##     3774 difftime  numeric
```

- we see that it is of Class difftime (we also see the length and that it is numeric)
- that means that we can not really calculate with it so we have to convert it into *real* numbers
- in the same step we may convert it into minutes because time durations using seconds may be a bit unhandy (dividing by 60)
- we can put this in the same column or we can add another one (I add another one so we can see the differences)

```
t179$duration2 <- as.numeric(t179$ENDE-t179$BEGINN)/60
head(t179)
```

```
##    SIC              BEGINN                ENDE visnr duration duration2
## 1 1175 2012-11-06 12:35:03 2012-11-06 12:36:24     1  81 secs  1.350000
## 2  219 2012-11-07 09:16:50 2012-11-07 09:19:22     1 152 secs  2.533333
## 3  190 2012-11-07 09:57:57 2012-11-07 10:01:03     1 186 secs  3.100000
## 4  256 2011-11-19 13:10:27 2011-11-19 13:11:43     1  76 secs  1.266667
## 5 1206 2012-11-07 10:40:23 2012-11-07 10:42:22     1 119 secs  1.983333
## 6 1183 2012-11-07 10:55:59 2012-11-07 10:58:47     1 168 secs  2.800000
```

- now we summarise this new column

```
summary(t179$duration2)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##        1        2        2    60890        3 46360000
```

- because the new columsn **t179$duration2** is numeric we got the basic information minimum, first quartile, median, mean third quartile and the maximum
- we see the maximum is 46.360.000 minutes which is `46360000/60/24` (32194.44) which is still a ridiculous high number
- so this have to be a error in the data, we do not know what kind of error but we can be sure that is an error
- so we have to think what durations are not plausible
- we see 3/4 of the values are 3 and below (because the 3rd quartile is 3), so the most people will finished the questionnaire within 3 minutes, some may need more time, but normally it should not be take more than then minutes
- so let's set us all values above 10 to missing
- remember: missings are coded as `NA` in are (which mean not available)

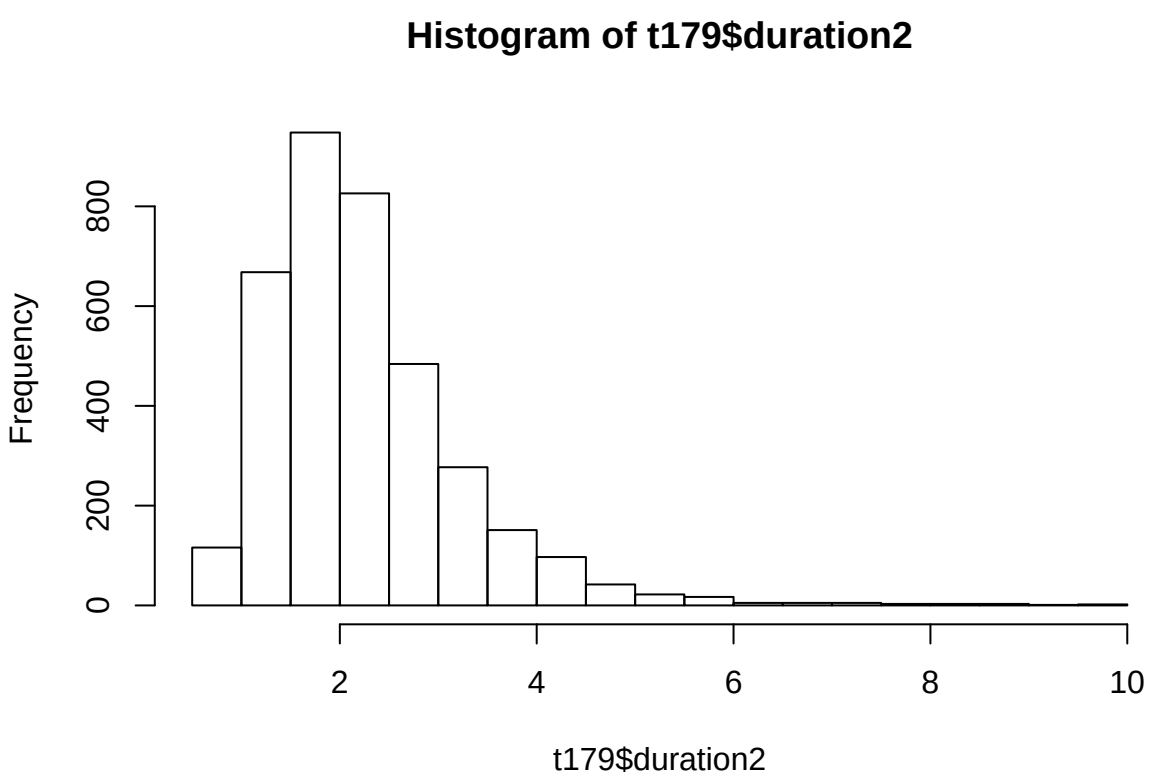
- we have to use indexing ([], because we want to idendify all values greater than 10) **t179$duration2 > 10**
- we have to do this indexing on hour duration2 column (because we want to change the values there) (the vector's name is **t179$duration2**)
- we have to assign `NA` to these indexed values **<- NA**
- so let's put it together and summarise the vector again

```
t179$duration2[t179$duration2 > 10] <- NA
summary(t179$duration2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  0.5833  1.5830  2.0670  2.2550  2.6830  9.6330      99
```

- now we see again min, max, mean, median and the quartiles and in addition the number of NAs
- and now the data shut be ready for plotting a histogram

```
hist(t179$duration2)
```



- now we change some elements (to get a more nice looking graph)
- `main` can be used to change the title
- `xlab` for the title of the x-axis
- `ylab` for the title of the y-axis
- `xlim` for the limits of the x-axis
- `col` for the colour of the bins
- `border` for the colour of the borders of the bins

- `breaks` for the number of breaks along the x-axis

```
hist(t179$duration2,
     main="Ausfuelldauer fuer den Fragebogen T00179",
     xlab="Dauer",
     ylab="Anzahl",
     xlim=c(0,10),
     col="midnightblue",
     border="mediumvioletred",
     breaks=20)
```