Descriptive Statistics

Jan Rovny

Basic Data Descriptions

To find out about a dataset in R we can run a names command on the particular dataframe (dataset). Let's return to our example ESS dataset, first loading it into R:

```
library(rio)
D<-import("https://jan-rovny.squarespace.com/s/ESS_FR.dta")
```

Let's explore our data:

```
names(D)
```

```
##  [1] "essround" "lrscale"  "stflife"  "gincdif"  "euftf"    "imwbcnt"
##  [7] "prtvtcfr" "happy"    "rlgatnd"  "gndr"     "yrbrn"    "eduyrs"
## [13] "hinctnta"
```

This produces a list of all the variable names in the dataset. Alternatively, we can ask to receive both variable names and some basic summary statistics:

```
summary(D)
```

```
##     essround    lrscale          stflife          gincdif
##  Min.   :7   Min.   : 0.000   Min.   : 0.000   Min.   :1.000
##  1st Qu.:7   1st Qu.: 3.000   1st Qu.: 5.000   1st Qu.:1.000
##  Median :7   Median : 5.000   Median : 7.000   Median :2.000
##  Mean   :7   Mean   : 5.065   Mean   : 6.395   Mean   :2.183
##  3rd Qu.:7   3rd Qu.: 7.000   3rd Qu.: 8.000   3rd Qu.:3.000
##  Max.   :7   Max.   :10.000   Max.   :10.000   Max.   :5.000
##              NA's   :114      NA's   :6        NA's   :8
##      euftf           imwbcnt          prtvtcfr          happy
##  Min.   : 0.000   Min.   : 0.000   Min.   : 1.000   Min.   : 0.000
##  1st Qu.: 3.000   1st Qu.: 4.000   1st Qu.: 9.000   1st Qu.: 6.000
##  Median : 5.000   Median : 5.000   Median : 9.000   Median : 8.000
##  Mean   : 4.946   Mean   : 4.848   Mean   : 8.775   Mean   : 7.192
##  3rd Qu.: 7.000   3rd Qu.: 6.000   3rd Qu.:10.000   3rd Qu.: 8.000
##  Max.   :10.000   Max.   :10.000   Max.   :16.000   Max.   :10.000
##  NA's   :74       NA's   :25       NA's   :857      NA's   :3
##     rlgatnd           gndr           yrbrn          eduyrs
##  Min.   :1.000   Min.   :1.000   Min.   :1916   Min.   : 0.00
##  1st Qu.:5.000   1st Qu.:1.000   1st Qu.:1950   1st Qu.:10.00
##  Median :7.000   Median :2.000   Median :1964   Median :13.00
##  Mean   :5.978   Mean   :1.524   Mean   :1964   Mean   :12.82
##  3rd Qu.:7.000   3rd Qu.:2.000   3rd Qu.:1979   3rd Qu.:16.00
##  Max.   :7.000   Max.   :2.000   Max.   :2000   Max.   :50.00
##  NA's   :3                       NA's   :3      NA's   :13
##     hinctnta
##  Min.   : 1.000
##  1st Qu.: 3.000
##  Median : 5.000
##  Mean   : 5.102
##  3rd Qu.: 7.000
```

```
##  Max.   :10.000
##  NA's   :118
```

We can also ask about the nature of each variable by typing:

```
is.character(D$party)
```

```
## [1] FALSE
```

```
is.numeric(D$yrbrn)
```

```
## [1] TRUE
```

```
is.factor(D$hinctnta)
```

```
## [1] FALSE
```

```
is.integer(D$gndr)
```

```
## [1] FALSE
```

```
is.vector(D$happy)
```

```
## [1] FALSE
```

R answers TRUE or FALSE.

Descriptive Statistics

Next, we should run some descriptive statistics on our data. Descriptive statistics do not test any hypotheses and do not try to infer any general rules from the data. They simply describe the data we have in front of us. The most basic descriptive statistics are measures of central tendency, such as mean, mode and median, and measures of dispersion, such as variance and standard deviation.

Measure	Calculation	Description
Mean	$\bar{X} = \mu = \frac{\sum x_i}{N}$	the arithmetic mean
Mode		the most frequently occurring value
Median		the central value
Variance	$\sigma^2 = \frac{\sum (x_i - \bar{X})^2}{N-1}$	square deviation from mean
Standard Deviation	$\sigma = \sqrt{\frac{\sum (x_i - \bar{X})^2}{N-1}}$	deviation from mean

In R, we can easily obtain these measures:

```
summary(D$yrbrn)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    1916    1950    1964    1964    1979    2000       3
```

This gives us the minimum, maximum, mean and median values of the year of birth. If we want particular statistics, we can (at any point, even within other commands) ask R to produce them by issuing the following commands:

```
mean(D$yrbrn, na.rm=T)
```

```
## [1] 1964.296
```

```
median(D$yrbrn, na.rm=T)
```

```
## [1] 1964
```

```
var(D$yrbrn, na.rm=T)
```

```
## [1] 351.7442
```

```
sd(D$yrbrn, na.rm=T)
```

```
## [1] 18.75484
```

```
min(D$yrbrn, na.rm=T)
```

```
## [1] 1916
```

```
max(D$yrbrn, na.rm=T)
```

```
## [1] 2000
```

```
range(D$yrbrn, na.rm=T)
```

```
## [1] 1916 2000
```

The mode of a vector is a little harder to obtain. To get the mode of vector x, you can get it like this:

```
names(sort(-table(D$yrbrn)))[1]
```

```
## [1] "1980"
```

This creates a table of the frequencies of each value, multipling by -1 and sorting puts the largest frequency first, and 'names()[1]' extracts the name of the first element, which is the sample mode. You may need to then use 'as.numeric()' on the result if you want a number.

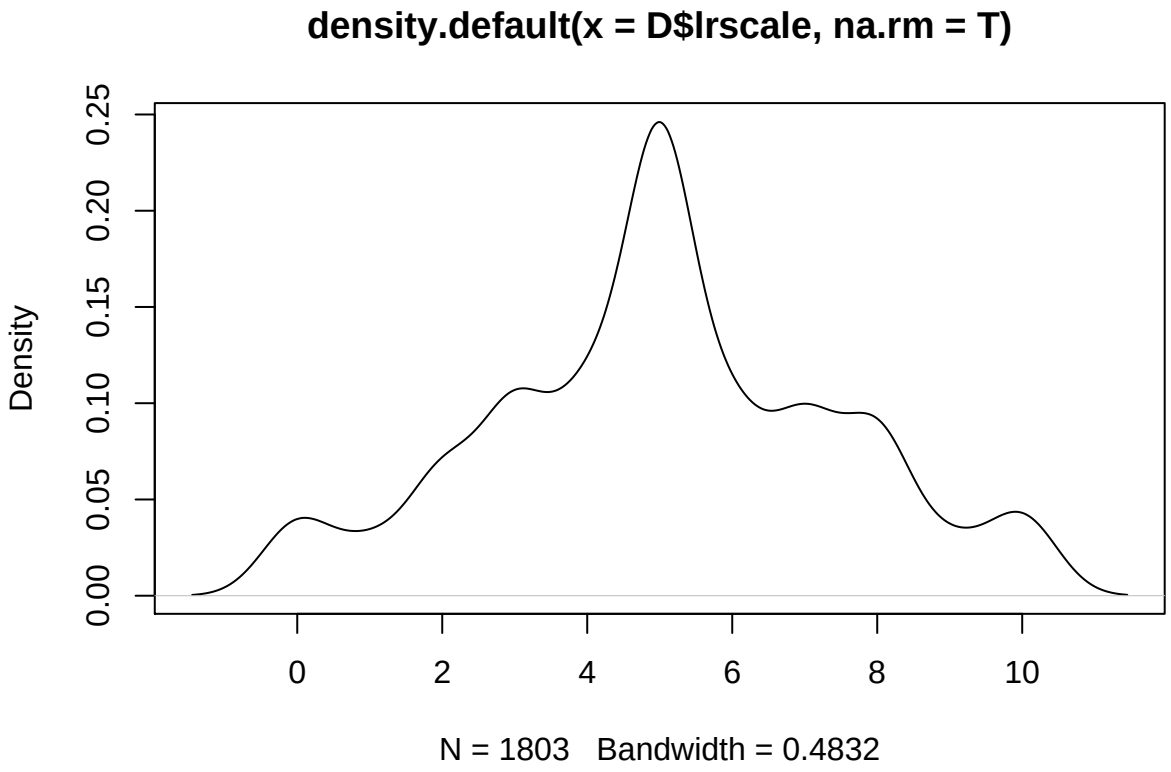
Basic plots

The subsequent step in learning about our data should be the plotting of the data. Mean and variance give us a good idea as to the central tendency and dispersion of a variable, but it is even more interesting to see the frequency distribution across its values. To see a distribution of of a variable we first need to create the distribution density function:

```
den<-density(D$lrscale, na.rm=T)
```

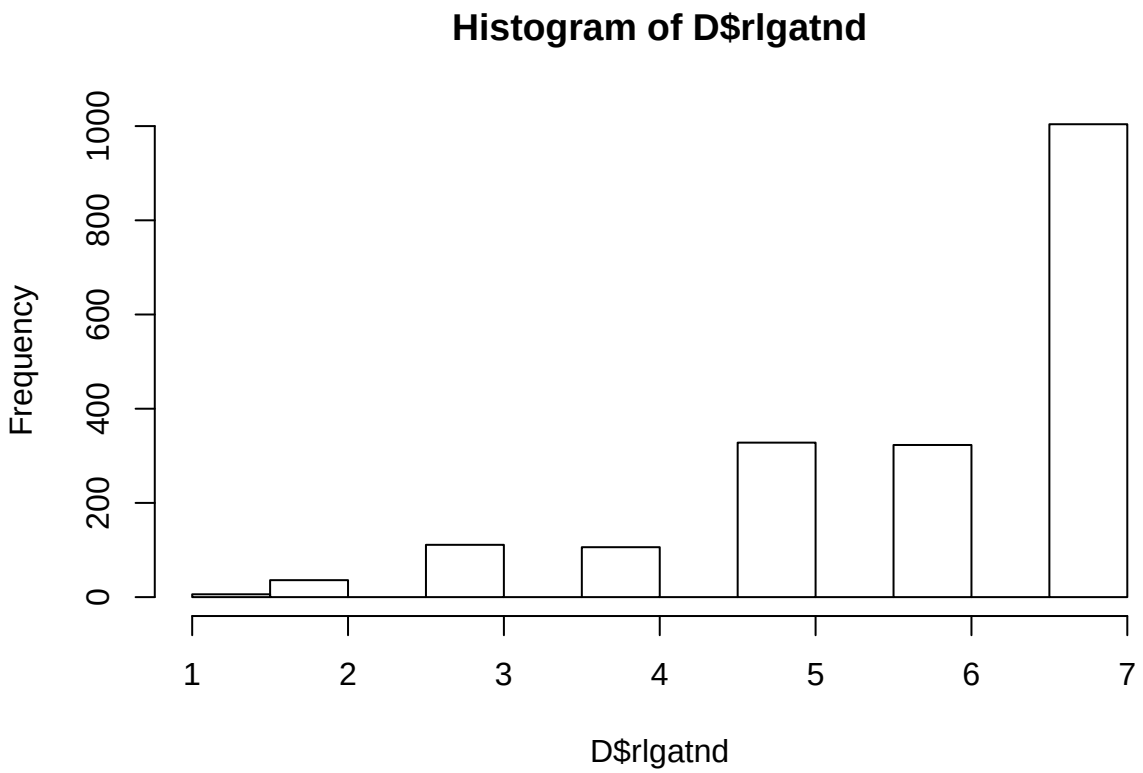
Now we can plot d to see the distribution:

```
plot(den)
```



Density functions are, however, only meaningful for continuous data. In the cases of categorical or ordinal data it is more meaningful to look at a histogram. To plot a histogram in R, we simply say:

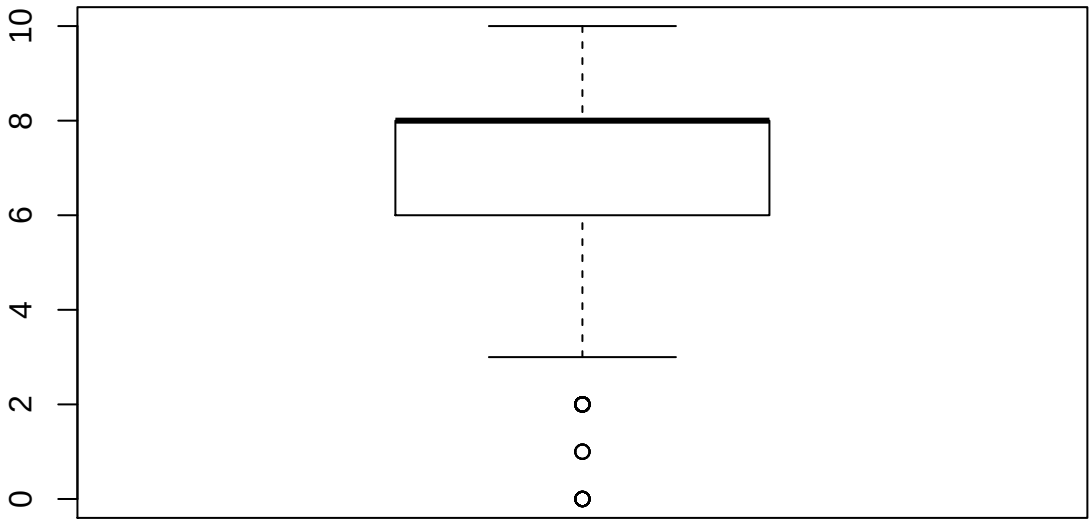
```
hist(D$rlgatnd)
```



Another useful descriptive tool is a boxplot. A boxplot shows us the median, the quartiles, and the maximum

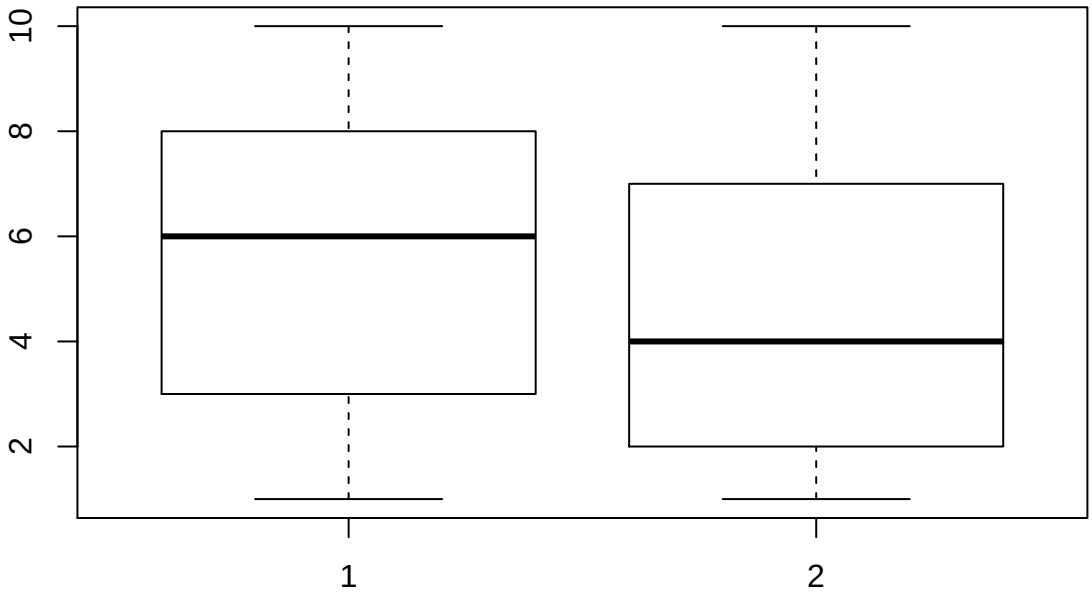
and minimum of a variable. In R, say:

```
boxplot(D$happy)
```



Boxplots are particularly useful for comparing the distributions of certain subsets of a given variable. Say that we are interested in seeing the different income distributions of men and women. We can do this by looking at a boxplot of income by gender:

```
boxplot(D$hinctnta ~ D$gndr)
```



Tables

Finally, it is very useful to organize our data into a table. A two-way table arranges the values of one variable by the values of another. Such organization is of course only meaningful for categorical or ordinal data, not for continuous variables. Let's make a table summarizing the vote for different parties by gender (note, we are working with the variable *party* created in the previous lesson dealing with Operations in R!):

```
table(D$gndr, D$party)
```

```
##
##     Left Ecolo Soc Center Gaulists Front
##   1   44    33 147     39      149    66
##   2   28    54 188     25      154    60
```

This, however, creates a table with raw numbers, which is not very useful. To do this comparison meaningfully, we must compare proportional data. In R we first create a raw table of vote by rich:

```
party.table<-table(D$gndr,D$party)
```

Now we use the prop.table command to create proportions:

```
prop.table(party.table) # gives us the proportions by all cells
```

```
##
##           Left      Ecolo        Soc     Center   Gaulists      Front
##   1 0.04457953 0.03343465 0.14893617 0.03951368 0.15096251 0.06686930
##   2 0.02836879 0.05471125 0.19047619 0.02532928 0.15602837 0.06079027
```

```
prop.table(party.table,1) #gives us the proportions by rows
```

```
##
##           Left      Ecolo        Soc     Center   Gaulists      Front
##   1 0.09205021 0.06903766 0.30753138 0.08158996 0.31171548 0.13807531
##   2 0.05500982 0.10609037 0.36935167 0.04911591 0.30255403 0.11787819
```

```
prop.table(party.table,2) #gives us the proportions by columns
```

```
##
##          Left     Ecolo       Soc    Center  Gaulists     Front
##   1 0.6111111 0.3793103 0.4388060 0.6093750 0.4917492 0.5238095
##   2 0.3888889 0.6206897 0.5611940 0.3906250 0.5082508 0.4761905
```

In order to test whether there is a difference gender support across parties, we can ask R to provide the χ^2 test:

```
library(MASS) #load the appropriate library
chisq.test(party.table) #perform the test
```

```
##
##  Pearson's Chi-squared test
##
## data:  party.table
## X-squared = 16.115, df = 5, p-value = 0.006522
```

Given the low p-value, we reject the hypothesis that there is no difference between men's and women's party support. Gender seems to map onto party preferences.