



# R : Data manipulation and exploration

Sławek Staworko

University of Lille

2020



Factors and Data frames

Data manipulation

Data exploration

Data transformation



# Factors and Data frames

We shall collect the following information for each respondent

- ▶ **Sex:** categorical variable with M for male and F for female;
- ▶ **Annual Salary:** quantitative variable (numeric);
- ▶ **Hours per week:** quantitative variable (numeric);
- ▶ **Education level:** categorical variable with possible values N for none, P for primary, H for high school, B for bachelor, M for masters, and D for doctorate.
- ▶ **Age:** quantitative variable.



## 5 respondents

1. A 32 years old male with high-school diploma, earning \$30K and working 38 hours per week.
2. A 41 y/o female with master degree, earning \$40K and working 45h/w.
3. A 45 y/o male with doctorate degree, earning \$55K and working 38h/w.
4. A 24 y/o male with bachelor degree, earning \$38K and working 40h/w.
5. A female (age undisclosed) with bachelor degree, earning \$35K and working 35h/w.

# How to store survey answers?



Using tabular representation (data.frame)

Sex	Salary	Hours	Edu	Age
M	30000	38	H	32
F	40000	45	M	41
M	55000	38	D	45
M	38000	40	B	24
F	35000	35	B	NA

Internally represented with a list of columns

```
s ← list()
s$Sex ← c('M', 'F', 'M', 'M', 'F')
s$Salary ← c(30e3, 40e3, 55e3, 38e3, 35e3)
s$Hours ← c(38, 45, 38, 40, 35)
s$Edu ← c('H', 'M', 'D', 'B', 'B')
s$Age ← c(32, 41, 45, 24, NA)
survey ← data.frame(s)
```

**Factors** are used for representing categorical values

- ▶ categorical values range over a set of labels, in R called **levels**
- ▶ each label is assigned a unique consecutive integer
- ▶ this integer is used internally to represent the label

```
s$Sex ← factor(c('M','F','M','M','F'))
```

```
class(s$Sex) ↦ factor
```

```
typeof(s$Sex) ↦ integer
```

```
levels(s$Sex) ↦ 'F' 'M'
```

```
as.integer(s$Sex) ↦ 2 1 2 2 1
```

```
as.character(s$Sex) ↦ 'M' 'F' 'M' 'M' 'F'
```

## Ordered factors

- ▶ Labels of a factor can have a predefined order
- ▶ which can be used to compare different labels of the same factor

```
s$Edu ← factor(  
  x=c('H', 'M', 'D', 'B', 'B'),  
  levels=c('N', 'P', 'H', 'B', 'M', 'D'),  
  ordered=TRUE  
)  
as.integer(s$Edu) ↪ 3 5 6 4 4  
s$Edu > 'B' ↪ FALSE TRUE TRUE FALSE FALSE  
mean(s$Salary[s$Edu > 'B']) ↪ 47500
```



## Data frame have to be carefully created

- ▶ columns need to have the same length
- ▶ character vectors are converted to unordered factors

```
survey ← data.frame(  
  Sex=c('M', 'F', 'M', 'M', 'F'),  
  Salary=c(30000, 40000, 55000, 38000, 35000),  
  Hours=c(38, 45, 38, 40, 35),  
  Edu=factor(  
    x=c('H', 'M', 'D', 'B', 'B'),  
    levels=c('N', 'P', 'H', 'B', 'M', 'D'),  
    ordered=TRUE  
  ),  
  Age=c(32, 41, 45, 24, NA)  
)
```



# Data manipulation

## Factor labels (levels) need to be carefully manipulated

- ▶ Labels can be changed all at once

```
levels(survey$Edu) ← c('none', 'high.school',  
                        'bachelor', 'master', 'doctorate')
```

- ▶ or one by one

```
levels(survey$Sex)[levels(survey$Sex=='M')] ← 'male'  
levels(survey$Sex)[levels(survey$Sex=='F')] ← 'female'
```

- ▶ Reordering is tricky and it's best to create a new factor

```
survey$Edu ← factor(c('H', 'M', 'D', 'B', 'B'))  
as.integer(survey$Edu) ↦ 3 4 2 1 1  
survey$Edu ← factor(survey$Edu,  
                    levels=c('N', 'P', 'H', 'B', 'M', 'D'), ordered=T)  
as.integer(survey$Edu) ↦ 3 5 6 4 4
```

## Data frames as lists

- ▶ `survey$Salary`  $\equiv$  `survey[['Salary']]`  
returns the salary column (a vector)
- ▶ `survey$Salary[2]`  $\equiv$  `survey[['Salary']][2]`  
returns the salary of the 2nd survey respondent
- ▶ `survey[c('Edu', 'Salary')]`  $\equiv$  `survey[c(4,2)]`  
returns a data frame with the columns Edu and Salary
- ▶ `length(survey)`  
returns the length of the list, the number of columns

## Data frames as matrices

- ▶ `survey[4, 'Salary']`  $\equiv$  `survey[4, 2]`  
returns the salary of the 4th respondent
- ▶ `survey[survey$Sex=='M', 'Salary']`  
returns salaries of all men in the survey (a vector)
- ▶ `survey[survey$Sex=='F', ]`  
returns survey responses from women (a data frame)

## Adding new survey entries

- ▶ `survey <- rbind(survey, list('F',28000,42,NA,45))`  
order of elements must follow the order of columns
- ▶ `survey <- rbind(survey,  
list(Sex='M',Salary=25000,Edu='P',Hours=32,Age=30))`  
order is unimportant but labels must agree with column names

## Adding new column (variable)

- ▶ `survey <- cbind(survey,  
Wage=survey$Salary/(52*survey$Hours))`  
add a new column with hourly wage values computed based on salary information

## Dropping a column

- ▶ `survey <- subset(survey, select = -c(6))`

# Ordering a data frame



- ▶ survey is

Sex	Salary	Hours	Edu	Age
M	30000	38	H	32
F	40000	45	M	41
M	55000	38	D	45
M	38000	40	B	24
F	35000	35	B	NA
F	28000	42	NA	45
M	25000	32	P	30

- ▶ `order(survey$Salary)`  $\mapsto$  7 6 1 5 4 2 3  
row numbers in increasing order of salary
- ▶ `order(survey$Salary,decreasing=TRUE)`  $\mapsto$  3 2 4 5 1 6 7  
row numbers in decreasing order of salary

## Ordering a data frame (cont'd.)



► `survey[order(survey$salary, decreasing=TRUE),]`



Sex	Salary	Hours	Edu	Age
M	55000	38	D	45
F	40000	45	M	41
M	38000	40	B	24
F	35000	35	B	NA
M	30000	38	H	32
F	28000	42	NA	45
M	25000	32	P	30





# Data exploration

# Taking a quick glance



- ▶ `ncol(survey)` returns the number of columns
- ▶ `nrow(survey)` returns the number of rows
- ▶ `names(survey)` returns the names of the columns
- ▶ `head(survey,n=6)` returns the first `n` rows
- ▶ `tail(survey,n=6)` returns the last `n` rows
- ▶ `str(survey)` displays the structure of the data frame
- ▶ `summary(survey)` displays the summary of the data frame

# Structure of a data frame



```
str(survey)
```



```
'data.frame': 7 obs. of 5 variables:  
 $ Sex      : Factor w/ 2 levels "M","F": 1 2 1 1 2 2 1  
 $ Salary   : num  30000 40000 55000 38000 35000 28000 25000  
 $ Hours    : num  38 45 48 45 35 42 32  
 $ Edu      : Ord.factor w/ 6 levels "N"<"P"<"H"<"B"<...: 3 5 6 4 4  
 $ Age      : num  32 41 45 24 NA 45 30
```

# Summarizing a data frame



```
summary(survey)
```



```
Sex      Salary      Hours      Edu      Age
M:4  Min.    :25000  Min.    :32.00  N    :0  Min.    :24.00
F:3  1st Qu.:29000  1st Qu.:36.50  P    :1  1st Qu.:30.50
      Median :35000  Median :42.00  H    :1  Median :36.50
      Mean   :35857  Mean   :40.71  B    :2  Mean   :36.17
      3rd Qu.:39000  3rd Qu.:45.00  M    :1  3rd Qu.:44.00
      Max.   :55000  Max.   :48.00  D    :1  Max.   :45.00
                                NA's:1  NA's   :1
```

# Basic statistics of continuous variables



- ▶ `sd(survey$Salary)`  $\mapsto$  10023.78  
standard deviation
- ▶ `var(survey$Age, use="complete")`  $\mapsto$  76.56667  
variance of a variable
- ▶ `cor(survey$Hours,survey$Salary)`  $\mapsto$  0.768371  
correlation between two variables
- ▶ `mean(survey$Salary)`  $\mapsto$  35857.14  
`mean(survey$Age,na.rm=TRUE)`  $\mapsto$  36.16667  
average value
- ▶ `median(survey$Salary)`  $\mapsto$  35000  
median value
- ▶ `quantile(survey$Salary,0.25)`  $\mapsto$  29000  
cut-off value for the 25th percentile of salary values
- ▶ `fivenum(survey$Age)`  $\mapsto$  24.0 30.0 36.5 45.0 45.0  
minimum, lower-hinge, median, upper-hinge, and maximum

# Summarizing categorical variables



```
table(survey$Sex)
```



```
M F  
4 3
```

```
table(survey$Sex,survey$Edu)
```



```
  N P H B M D  
M 0 1 1 1 0 1  
F 0 0 0 1 1 0
```

# Data transformation

```
aggregate(Salary~Sex,survey,median)
```



Sex	Salary
M	34000
F	35000

```
aggregate(cbind(Age,Salary)~Sex+Edu,survey,mean)
```



Sex	Edu	Age	Salary
M	P	30	25000
M	H	32	30000
M	B	24	38000
F	M	41	40000
M	D	45	55000



# Data reshaping (reshape package)



the *long* representation

the *wide* representation

Sex	Salary	Hours	Edu	Age
M	30000	38	H	32
F	40000	45	M	41
M	55000	48	D	45
M	38000	45	B	24
F	35000	35	B	NA

melt



Sex	Edu	variable	value
M	H	Salary	30000
F	M	Salary	40000
M	D	Salary	55000
M	B	Salary	38000
F	B	Salary	35000
M	H	Hours	38
F	M	Hours	45
M	D	Hours	48
M	B	Hours	45
F	B	Hours	35
M	H	Age	32
F	M	Age	41
M	D	Age	45
M	B	Age	24
F	B	Age	NA

cast

