Abstract of the article. Here we can place more info.

# Introduction

Research idea:

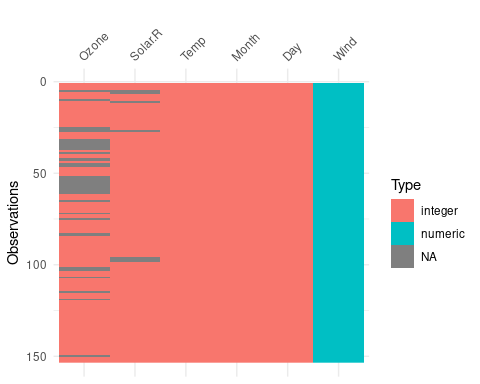
Context: An organization membership located in Portugal. The organization offers an annual membership for the members, the service subscription has several payment options:

* Men with a annual fee of 10€
* Women annual fee of 6€
* Correspondent fee 6€
* Retired fee 5€
* Student fee 2.5€
* under-14 fee 1€

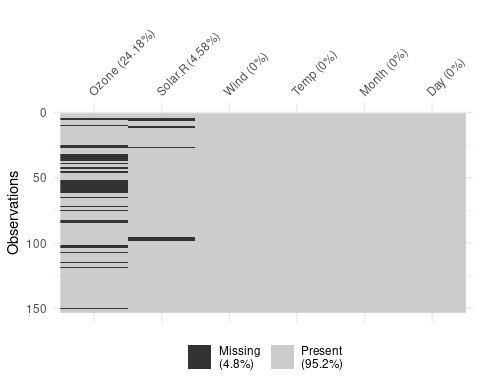
library(dplyr)  
library(dlookr)  
library(ggplot2)  
  
#eda\_report(nlswork,output\_dir =   
# "C:/Users/mangelo.EEG/Documents/GitHub/prjs/reports/",  
# output\_file = "eda\_report.pdf")  
  
## The data  
  
names(airquality)

## [1] "Ozone" "Solar.R" "Wind" "Temp" "Month" "Day"

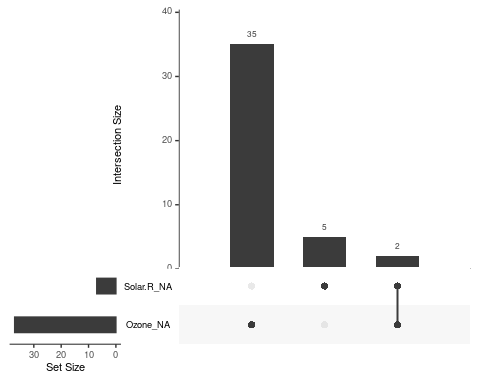
#summary(nlswork)  
  
## Missing values  
library(visdat)  
vis\_dat(airquality)



library(naniar)  
vis\_miss(airquality)



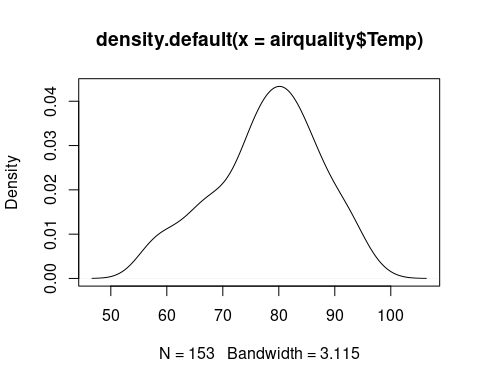
gg\_miss\_upset(airquality)



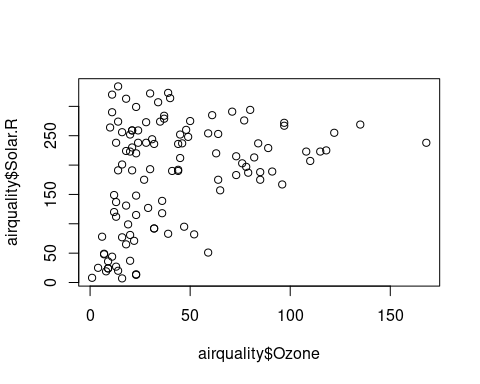
## GRAPHS  
dplyr::glimpse(cars$Ozone)

## NULL

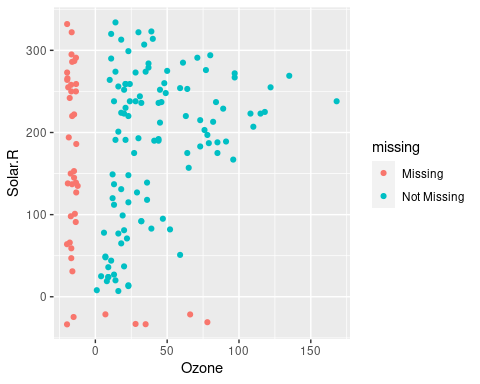
d <- density(airquality$Temp)  
plot(d)



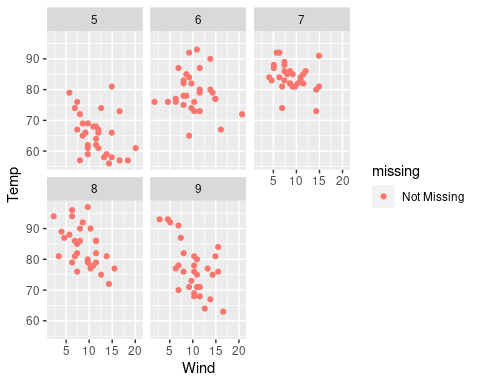
plot(airquality$Ozone, airquality$Solar.R)



ggplot(airquality, aes(x = Ozone, y = Solar.R)) +  
geom\_miss\_point()



ggplot(airquality, aes(x = Wind, y = Temp)) +  
geom\_miss\_point() +  
facet\_wrap(vars(Month))



stats <- summary(airquality$Temp)  
stats

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 56.00 72.00 79.00 77.88 85.00 97.00

describe(airquality)

## # A tibble: 6 × 26  
## variable n na mean sd se\_mean IQR skewness kurtosis p00  
## <chr> <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Ozone 116 37 42.1 33.0 3.06 45.2 1.24 1.29 1   
## 2 Solar.R 146 7 186. 90.1 7.45 143 -0.428 -0.968 7   
## 3 Wind 153 0 9.96 3.52 0.285 4.1 0.348 0.111 1.7  
## 4 Temp 153 0 77.9 9.47 0.765 13 -0.378 -0.404 56   
## 5 Month 153 0 6.99 1.42 0.115 2 -0.00239 -1.30 5   
## 6 Day 153 0 15.8 8.86 0.717 15 0.00265 -1.20 1   
## # … with 16 more variables: p01 <dbl>, p05 <dbl>, p10 <dbl>, p20 <dbl>,  
## # p25 <dbl>, p30 <dbl>, p40 <dbl>, p50 <dbl>, p60 <dbl>, p70 <dbl>,  
## # p75 <dbl>, p80 <dbl>, p90 <dbl>, p95 <dbl>, p99 <dbl>, p100 <dbl>

# Experimental Results

## Data description

## [1] "Sócio" "dataAdesao" "ano"   
## [4] "dataNascimento" "idade" "sexo"   
## [7] "estadoCivil" "categoria" "quotaMensal"   
## [10] "profissao" "codPostal" "ultimaQuota"   
## [13] "ultimoPagamento" "valorTotal" "totalJogos"   
## [16] "jogosEpoca" "diasUltimoPagamento" "mesesUP"   
## [19] "abandonou" "anosSocio" "idaEstadio"   
## [22] "escaloesTotalJogos" "mes"

## [1] "num\_socio" "dt\_inscription"   
## [3] "year" "birth\_date"   
## [5] "age" "sex"   
## [7] "marital\_status" "category"   
## [9] "monthly\_fee" "occupation"   
## [11] "zip\_code" "dt\_last\_invoice"   
## [13] "dt\_last\_payment" "total\_amount"   
## [15] "total\_matches" "season\_matches"   
## [17] "days\_since\_last\_payment" "months\_since\_last\_payment"  
## [19] "dropout" "years\_membership"   
## [21] "stadium\_access" "quart\_stadium\_entries"   
## [23] "inscription\_month"

## tibble [25,316 × 14] (S3: tbl\_df/tbl/data.frame)  
## $ year : num [1:25316] 1944 1944 1945 1945 1945 ...  
## $ age : num [1:25316] 83 88 73 97 97 91 88 95 88 78 ...  
## $ sex : chr [1:25316] "M" "M" "M" "M" ...  
## $ marital\_status : chr [1:25316] "casado" "solteiro" "nao definido" "casado" ...  
## $ monthly\_fee : num [1:25316] 10 10 10 5 10 5 5 5 10 10 ...  
## $ total\_amount : num [1:25316] 1906 1906 1553 790 1466 ...  
## $ total\_matches : num [1:25316] 0 0 0 0 0 20 74 0 154 0 ...  
## $ season\_matches : num [1:25316] 0 0 0 0 0 0 0 0 6 0 ...  
## $ months\_since\_last\_payment: num [1:25316] 3 3 36 8 35 4 41 40 4 2 ...  
## $ dropout : num [1:25316] 0 0 1 0 1 0 1 1 0 0 ...  
## $ years\_membership : num [1:25316] 74 74 73 73 73 73 73 73 73 72 ...  
## $ stadium\_access : num [1:25316] 0 0 0 0 0 1 1 0 1 0 ...  
## $ quart\_stadium\_entries : chr [1:25316] "ate 1" "ate 1" "ate 1" "ate 1" ...  
## $ inscription\_month : num [1:25316] 10 10 8 9 9 12 1 1 2 4 ...

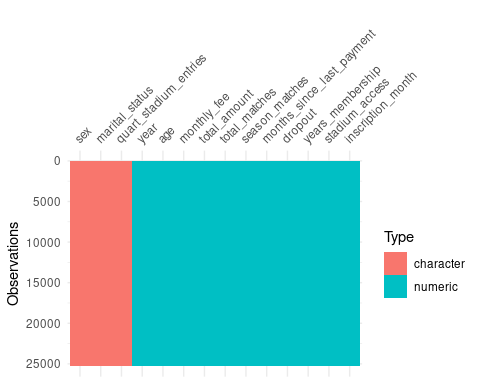


Table @ref(tab:tab1cars) shows data’s summary statistics.[[1]](#footnote-29) stargazer() is and excellent solution to export outputs.

Teste @ref(tab:summarytable)

from pysurvival.utils.display import correlation\_matrix  
import pandas as pd  
import numpy as np  
  
col = ['sex','marital\_status','quart\_stadium\_entries']  
  
df\_members = r.df\_members #copy r dataframe to python  
  
df\_members = pd.get\_dummies(df\_members, columns=col,drop\_first=True)  
  
# Creating the time and event columns  
time\_column = 'years\_membership'  
event\_column = 'dropout'  
  
# Extracting the features  
features = np.setdiff1d(df\_members.columns, [time\_column, event\_column] ).tolist()  
  
  
correlation\_matrix(df\_members, figure\_size=(10,10), text\_fontsize=6)

The average age in our data is 27.3.

# Tables

R Markdown PDF is now able to produce good tables with our output. For stargazer the label is contained in the function, while for kable it’s contained in the chunk name.

## stargazer(): Summary and regression tables

Table @ref(tab2) reports regression outputs. Name the models as you can refer to their names in the text (M1, M2, M3).

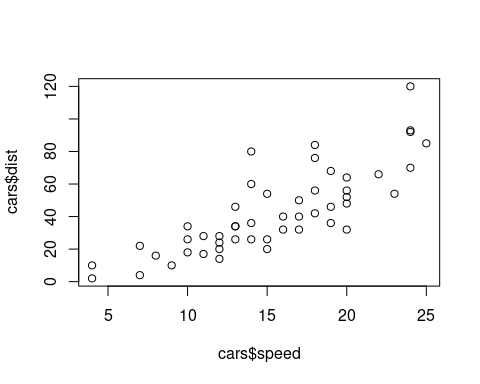
library(stargazer)  
model1 <- lm(speed ~ dist, data = cars)  
model2 <- lm(speed ~ dist, data = cars)  
model3 <- lm(dist ~ speed, data = cars)  
stargazer(model1, model2, model3,  
 title = "Regression table with stargazer",  
 label = "tab2",  
 table.placement = "h",  
 column.labels = c("M1", "M2", "M3"),  
 model.numbers = FALSE,  
 header = FALSE)

# Figures

## Graphs with R

You can insert figures like this. One would like to produce and insert them on the fly in the .rmd file. Figure @ref(fig:fig-1) is such an example.

plot(cars$speed, cars$dist)



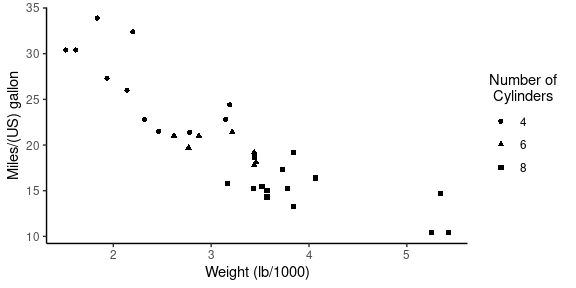
Scatterplot of Speed and Distance

However, in some cases it does not work.

## Example: ggplot2 graphs

See the ggplot2 output reported in Figure @ref(fig:fig-2).

mtcars$cyl <- as.factor(mtcars$cyl) # Convert cyl to factor  
library(ggplot2)  
ggplot(mtcars, aes(x = wt, y = mpg, shape = cyl)) + geom\_point() +  
 labs(x = "Weight (lb/1000)", y = "Miles/(US) gallon",  
 shape = "Number of \n Cylinders") + theme\_classic()



Miles per gallon according to the weight

## Another example using Plotly

With Plotly we can produce interactive graphs which play well, for example, once can embedded in html webpages (drop by [here](https://paulcbauer.shinyapps.io/visualizing-causal-scenarios/) for an example). One can insert this type of graphs in R Markdown PDF using Orca (it generates static images from Plotly graphs). Go [here](https://github.com/plotly/orca#installation) to check how to install it. See Figure @ref(fig:fig-3) for an example.

library(plotly)  
p <- plot\_ly(cars, type = "scatter", mode = "markers",  
 x = ~speed,  
 y = ~dist)  
#Sys.setenv('MAPBOX\_TOKEN' = '12423423') # set arbitrary token  
#orca(p, "logs/plotly-plot.pdf")

# Lets create a value for example  
  
media <- mean(cars$speed)

The criminal rate is 15.4%o.

# Miguel’s tests

## R

Example of an equation

*Example of a matrix*

or

See Equation @ref(eq:binom).

library(stargazer)  
stargazer(cars,  
 title = "Summary 24",  
 label = "tab24",  
 table.placement = "ht",  
 header = FALSE)

# Final remarks

Check the replication package for Bonhomme, Lamadon and Manresa (2019): <https://github.com/tlamadon/blm-replicate>

# References

# Appendix: Chunk options

## Software versioning

### R

cat(paste("#", capture.output(sessionInfo()), "\n", collapse = ""))

## # R version 4.1.0 (2021-05-18)   
## # Platform: x86\_64-pc-linux-gnu (64-bit)   
## # Running under: Ubuntu 20.04.2 LTS   
## #   
## # Matrix products: default   
## # BLAS: /usr/lib/x86\_64-linux-gnu/openblas-pthread/libblas.so.3   
## # LAPACK: /home/sobreiro/miniconda3/envs/survival/lib/libmkl\_intel\_lp64.so   
## #   
## # locale:   
## # [1] en\_US.UTF8   
## #   
## # attached base packages:   
## # [1] stats graphics grDevices utils datasets methods base   
## #   
## # other attached packages:   
## # [1] plotly\_4.9.4.1 kableExtra\_1.3.4 gtsummary\_1.4.2 readxl\_1.3.1   
## # [5] stargazer\_5.2.2 naniar\_0.6.1 visdat\_0.5.3 ggplot2\_3.3.5   
## # [9] dlookr\_0.4.5 dplyr\_1.0.7 rmarkdown\_2.9 nvimcom\_0.9-115   
## #   
## # loaded via a namespace (and not attached):   
## # [1] colorspace\_2.0-2 ellipsis\_0.3.2 class\_7.3-19   
## # [4] rio\_0.5.27 htmlTable\_2.2.1 base64enc\_0.1-3   
## # [7] rstudioapi\_0.13 proxy\_0.4-26 farver\_2.1.0   
## # [10] fansi\_0.5.0 mvtnorm\_1.1-2 xml2\_1.3.2   
## # [13] splines\_4.1.0 extrafont\_0.17 libcoin\_1.0-8   
## # [16] knitr\_1.33 Formula\_1.2-4 jsonlite\_1.7.2   
## # [19] gt\_0.3.0 Rttf2pt1\_1.3.8 cluster\_2.1.2   
## # [22] png\_0.1-7 compiler\_4.1.0 httr\_1.4.2   
## # [25] backports\_1.2.1 lazyeval\_0.2.2 assertthat\_0.2.1   
## # [28] Matrix\_1.3-4 cli\_3.0.1 hrbrthemes\_0.8.0   
## # [31] htmltools\_0.5.1.1 tools\_4.1.0 partykit\_1.2-13   
## # [34] gtable\_0.3.0 glue\_1.4.2 tinytex\_0.32   
## # [37] Rcpp\_1.0.7 carData\_3.0-4 cellranger\_1.1.0   
## # [40] vctrs\_0.3.8 svglite\_2.0.0 extrafontdb\_1.0   
## # [43] crosstalk\_1.1.1 broom.helpers\_1.3.0 inum\_1.0-4   
## # [46] xfun\_0.24 stringr\_1.4.0 openxlsx\_4.2.4   
## # [49] rvest\_1.0.0 lifecycle\_1.0.0 RcmdrMisc\_2.7-1   
## # [52] MASS\_7.3-54 zoo\_1.8-9 scales\_1.1.1   
## # [55] hms\_1.1.0 sandwich\_3.0-1 RColorBrewer\_1.1-2   
## # [58] yaml\_2.2.1 curl\_4.3.2 reticulate\_1.20   
## # [61] gridExtra\_2.3 UpSetR\_1.4.0 gdtools\_0.2.3   
## # [64] rpart\_4.1-15 latticeExtra\_0.6-29 stringi\_1.7.3   
## # [67] highr\_0.9 corrplot\_0.90 nortest\_1.0-4   
## # [70] e1071\_1.7-7 checkmate\_2.0.0 zip\_2.2.0   
## # [73] rlang\_0.4.11 pkgconfig\_2.0.3 systemfonts\_1.0.2   
## # [76] evaluate\_0.14 lattice\_0.20-44 purrr\_0.3.4   
## # [79] htmlwidgets\_1.5.3 labeling\_0.4.2 tidyselect\_1.1.1   
## # [82] bookdown\_0.22 plyr\_1.8.6 magrittr\_2.0.1   
## # [85] R6\_2.5.0 generics\_0.1.0 Hmisc\_4.5-0   
## # [88] DBI\_1.1.1 pillar\_1.6.1 haven\_2.4.1   
## # [91] foreign\_0.8-81 withr\_2.4.2 prettydoc\_0.4.1   
## # [94] survival\_3.2-11 abind\_1.4-5 nnet\_7.3-16   
## # [97] tibble\_3.1.3 crayon\_1.4.1 car\_3.0-11   
## # [100] utf8\_1.2.1 jpeg\_0.1-8.1 grid\_4.1.0   
## # [103] data.table\_1.14.0 forcats\_0.5.1 digest\_0.6.27   
## # [106] webshot\_0.5.2 tidyr\_1.1.3 munsell\_0.5.0   
## # [109] viridisLite\_0.4.0

# or use message() instead of cat()

1. You can reference the table as @ref(tab1cars). [↑](#footnote-ref-29)