Table of contents

0.1 Introduction

Multiple institutions have been a target of increasingly more disruptive or destructive cyber attacks over the last few years which has lead to government action. The data used in my work was collected yearly by the uk government department for Digital, Culture, Media and Sport (DCMS) with the purpose of helping the government understand the importance cyber security for British institutions, better shape policy regarding cyber security, create schemes to increase awareness for such problems and better protect institutions form cyber security threats. As such this analysis will investigate the relationship between how institutions protect themselves from cyber attacks and the affect of said attacks on these institutions in the last 5 years.

0.2 Objectives

Objectives of this report:

-Creating of a new tidy data set for each of the years including recompiled variables for the management, policing and rules implemented to protect the organisation, the type of attack that affected the institution and its respective outcomes.

-utilize Multiple Imputation by Chained Equations (Mice), to generate missing values to replace the missing data.

-Do hypothesis testing on my new fitted models to compare how the protective measures and attack type affect the outcome of the cyber attack on the institution.

-Mention the limitations of this analysis.

-Conclusion with recommendation for future research.

0.3 Data

Initial data wrangling:

Due to the untidy state of the data collect via the random probability telephone survey, these data sets containing between 421 to 462 variables have to been clean up into 21 easily comparable variables. The clean up process consisted of computing new variables utilizing the multiple subcategories of answers to the survey questions, grouping them into more flexible options while adjusting missing values to allow for such computation. Removing unnecessary data variables that are irrelevant to my hypothesis testing, converting the SPSS labelled data into R data structures to enable imputation of missing values and proper correlations computation and conversion of the appropriate missing value responses to actual missing data.

Initial data analysis:

It can be observed a significant degree of missing data on my data sets coming from multiple sources, the main source of missing data is derived from a limitation of the data collected, the lack of reporting channels in institutions leads to the majority of the missing values that lead to direct missing data in the data sets collected and indirect missing data by

After finishing to tidy the data I have imputed the missing values using Multiple Imputation by Chained Equations on each of the data sets.

using the a mixture of the methods of logical regression in the majority of the variables due to their dichotomous and binary behavior polynomial regression which is more adequate for

##loreg is for dichotomous missing data //dichotomous two mutuallu exclusive groups ##polyreg is for categorical missing data // categorical variable can be one of a limited, and usually fixed number of values aka its discrete

0.4 Limitations

There are multiple limitations to my analysis to be noted. Firstly, the data collected is limited to cyber attacks that were detected, there is variety of attacks that have gone unnoticed and therefore the data has a systematic tendency to underestimate the real level of breach attacks.(Department For Digital 2020) Secondly the missing data generated by imputation is biased since not all data is missing completly at random, mainly due to smaller or less digital institution as they don't have the infrastructure to detect, assess and report cyber attacks.

```r
library(haven)
library(tidyverse)
```

```
-- Attaching packages --------------------------------------- tidyverse 1.3.2 --
v ggplot2 3.3.6      v purrr   0.3.4
v tibble  3.1.8      v dplyr   1.0.10
v tidyr   1.2.1      v stringr 1.4.1
v readr   2.1.3      v forcats 0.5.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

```r
library(dplyr)
library(geometry)
library(formatR)
#install.packages("VIM")
library(Amelia)
```

```
Loading required package: Rcpp
##
## Amelia II: Multiple Imputation
## (Version 1.8.0, built: 2021-05-26)
## Copyright (C) 2005-2022 James Honaker, Gary King and Matthew Blackwell
## Refer to http://gking.harvard.edu/amelia/ for more information
##
```

```r
library(mice)
```

```
Attaching package: 'mice'

The following object is masked from 'package:stats':

    filter

The following objects are masked from 'package:base':

    cbind, rbind
```

```r
library(VIM)
```

```
Warning: package 'VIM' was built under R version 4.2.2
```

```
Loading required package: colorspace
Loading required package: grid
VIM is ready to use.

Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues

Attaching package: 'VIM'

The following object is masked from 'package:datasets':

    sleep
```

```r
library(labelled)
library(GGally)
```

```
Registered S3 method overwritten by 'GGally':
  method from
  +.gg   ggplot2
```

```r
library(mgcv)
```

```
Loading required package: nlme

Attaching package: 'nlme'

The following object is masked from 'package:dplyr':

    collapse

This is mgcv 1.8-40. For overview type 'help("mgcv-package")'.
```

```r
## Very important documentation for the 2018 data set //it is a
## surprise toll that will help us later
technicalAnnex2018 = "https://doc.ukdataservice.ac.uk/doc/8406/mrdoc/pdf/8406_cyber_security_b

## this is the loading the first year of this level of survey data set
## after burning my entire brain, replacing it with the backup one and
## also burning that one I discovered that it is just these lines that
## aren't being formatted in pdf because they are absolutely huge but
## at least it works for the other ones #FicaADica I assume it was
## thanks to formatR ?? I won't bother to redo every single bloody step
## again, enough pain and stack for the day
dataCyberSecuritySurvey2018 = read_spss("C:/AppliedDataScienceAndStatistics/Applied-Data-Scien

## adding the variable year because none of the data sets have any
## proper way to distinguish between the years of each survey
dataCyberSecuritySurvey2018$year = "2018"
```

0.5 Now we do the same for the other years before we merge them

```r
## loading the second year of this level of survey data set
dataCyberSecuritySurvey2019 = read_spss("C:/AppliedDataScienceAndStatistics/Applied-Data-Scien

## adding the variable year because none of the data sets have any
## proper way to distinguish between the years of each survey
```

```r
dataCyberSecuritySurvey2019$year = "2019"
```

```r
## loading the third year of this level of survey data set
dataCyberSecuritySurvey2020 = read_spss("C:/AppliedDataScienceAndStatistics/Applied-Data-Scien

## adding the variable year because none of the data sets have any
## proper way to distinguish between the years of each survey
dataCyberSecuritySurvey2020$year = "2020"
```

```r
## loading the forth year of this level of survey data set
dataCyberSecuritySurvey2021 = read_spss("C:/AppliedDataScienceAndStatistics/Applied-Data-Scien

## adding the variable year because none of the data sets have any
## proper way to distinguish between the years of each survey
dataCyberSecuritySurvey2021$year = "2021"
```

```r
## loading the fifth and final year of this level of survey data set
dataCyberSecuritySurvey2022 = read_spss("C:/AppliedDataScienceAndStatistics/Applied-Data-Scien

## adding the variable year because none of the data sets have any
## proper way to distinguish between the years of each survey
dataCyberSecuritySurvey2022$year = "2022"
```

```r
## Now that we have all data loaded lets start by tidying up data set
## by data set start from 2018


## for some sweet sweet documentation about the questions starting from
## page 26 TODO comment in case of fire or debugging
browseURL(technicalAnnex2018)
```

```r
## This entire code snippet is tidying up the type of organisation for
## the 2018 survey renaming the bloody variables to a more java like
## name

dataCyberSecuritySurvey2018TidyName = rename(dataCyberSecuritySurvey2018,
    instituitionTypes = "samptype")

## if instituitionTypes is 1 it is a business if it is 2 it is a
## charity and in the future 3 is for schools and education


## daily reminder that there is a boolean type but it is called logical
## Numeric -\tSet of all real numbers Integer -\tSet of all integers, Z
## Logical - -\tTRUE and FALSE Complex -\tSet of complex numbers
## Character -\t"a", "b", "c", …, "ç", "#", "~", ….., "1", "2", …etc


## it is a string so lets make it a proper numeric code

dataCyberSecuritySurvey2018TidyName$instituitionTypes = as.integer(dataCyberSecuritySurvey2018


## typex is 1-2 for businesses and 3 for charities so redundant and can
## be removed

dataCyberSecuritySurvey2018TidyName = dataCyberSecuritySurvey2018TidyName %>%
    select(-typex)


## dataCyberSecuritySurvey2018TidyName never forget if R can't show all
## displayed text from a computation it breaks both the rendering and
## ##the refreshing of the rendered code for some reason ¯/_( )_/¯
## future edit anything and everything breaks for no reason at all,
## just kill it and reopen refer to the first NOTE TO SELF for more
## information
```

```r
## see questioner documentation start from page 27

technicalAnnex2019 = "https://assets.publishing.service.gov.uk/government/uploads/system/uploa
## TODO comment in case of fire or debugging
browseURL(technicalAnnex2019)
```

```r
## see questioner documentation start from page 31

technicalAnnex2020 = "https://assets.publishing.service.gov.uk/government/uploads/system/uploa

## TODO comment in case of fire or debugging
browseURL(technicalAnnex2020)
```

```r
## see questioner documentation start from page 28

technicalAnnex2021 = "https://assets.publishing.service.gov.uk/government/uploads/system/uploa

## TODO comment in case of fire or debugging
browseURL(technicalAnnex2021)
```

```r
## see questioner documentation start from page 36

technicalAnnex2022 = "https://assets.publishing.service.gov.uk/government/uploads/system/uploa
```

```r
## TODO comment in case of fire or debugging
browseURL(technicalAnnex2022)

## trying not to get arrested for DDoSing the uk goverment by making a
## request to all the pdfs after rendering the page for the nth because
## I can't code nor debug (challenge impossible) bonus points if I get
## an exeter ip banned because of it
```

```r
## time to recycle the code for the 2018 survey that gets a 'neat' code
## of the institution types

## This entire code snippet is tidying up the type of organisation for
## the 2019 survey renaming the bloody variables to a more java like
## name

dataCyberSecuritySurvey2019TidyName = rename(dataCyberSecuritySurvey2019,
    instituitionTypes = "samptype")

dataCyberSecuritySurvey2019TidyName$instituitionTypes = as.integer(dataCyberSecuritySurvey2019

str(dataCyberSecuritySurvey2019TidyName$instituitionTypes)
```

```
int [1:2080] 1 1 1 1 1 1 1 1 1 1 ...
```

```r
## typex is redundant be we already have an indentifies for each type
## of institution and can be removed same for questtype since this
## questioner has more redundancy than amazon and google data centers
## combined

dataCyberSecuritySurvey2019TidyName = dataCyberSecuritySurvey2019TidyName %>%
    select(-one_of("typex", "questtype"))
```

```r
## I continue to save the planet by recycling as much as I can, mostly
## recycled code from the previous snippet today though this time we do
## have the concept of education institutions as our code just annoy me
## after I thought they should be converted to boolean like a getter in
## java

dataCyberSecuritySurvey2020TidyName = rename(dataCyberSecuritySurvey2020,
    instituitionTypes = "samptype")

dataCyberSecuritySurvey2020TidyName$instituitionTypes = as.integer(dataCyberSecuritySurvey2020

str(dataCyberSecuritySurvey2020TidyName$instituitionTypes)
```

```
int [1:1900] 1 1 1 1 1 1 1 1 1 1 ...
```

```r
## typex is redundant be we already have an indentifies for each type
## of institution and can be removed same for questtype since this
## questioner has more redundancy than amazon and google data centers
## combined

dataCyberSecuritySurvey2020TidyName = dataCyberSecuritySurvey2020TidyName %>%
    select(-one_of("typex", "questtype"))
```

```r
## saving the planet one recycled snippet of code at a time


dataCyberSecuritySurvey2021TidyName = rename(dataCyberSecuritySurvey2021,
    instituitionTypes = "samptype")

dataCyberSecuritySurvey2021TidyName$instituitionTypes = as.integer(dataCyberSecuritySurvey2021

str(dataCyberSecuritySurvey2021TidyName$instituitionTypes)
```

```
int [1:2284] 1 1 1 1 1 1 1 1 1 1 ...
```

```r
## typex is redundant be we already have an indentifies for each type
## of institution and can be removed same for questtype since this
## questioner has more redundancy than amazon and google data centers
## combined

dataCyberSecuritySurvey2021TidyName = dataCyberSecuritySurvey2021TidyName %>%
    select(-one_of("typex", "questtype"))
```

```r
## this comment was already dealt by the garbage collector unlike the
## previous ones

dataCyberSecuritySurvey2022TidyName = rename(dataCyberSecuritySurvey2022,
    instituitionTypes = "samptype")

dataCyberSecuritySurvey2022TidyName$instituitionTypes = as.integer(dataCyberSecuritySurvey2022

str(dataCyberSecuritySurvey2022TidyName$instituitionTypes)
```

```
int [1:2157] 1 1 1 1 1 1 1 1 1 1 ...
```

```r
## questtype is redundant be we already have an indentifies for each
## type of institution and can be removed

dataCyberSecuritySurvey2022TidyName = dataCyberSecuritySurvey2022TidyName %>%
    select(-questtype)
```

```
## now that we have started the data wrangling we will categorize all
## institutions by size remember that for some wicked reason they use
## -97 for missing values for anything without a proper missing value
## code for each question I will start by simply nulling every single
## -97 so we can see how much is missing and then possibly make a table
## with custom missing values for each like I did in C (remember to
## start from -1000 to -1999 like standard ACLs)


############################################## 2018
############################################## ###############################################
numberOfCycles = length(dataCyberSecuritySurvey2018TidyName$sizea)

dataCyberSecuritySurvey2018TidyNameSize = dataCyberSecuritySurvey2018TidyName

## apparently we have to be careful because an already inserted NA on
## the variable breaks the
for (i in 1:numberOfCycles) {
    if (dataCyberSecuritySurvey2018TidyNameSize$sizea[i] == -97) {
        dataCyberSecuritySurvey2018TidyNameSize$sizea[i] = NA
    }
    if (dataCyberSecuritySurvey2018TidyNameSize$sizeb[i] == -97) {
        dataCyberSecuritySurvey2018TidyNameSize$sizeb[i] = NA
    }

}

############################################## 2019
############################################## ###############################################

numberOfCycles = length(dataCyberSecuritySurvey2019TidyName$sizea)

dataCyberSecuritySurvey2019TidyNameSize = dataCyberSecuritySurvey2019TidyName

## apparently we have to be careful because an already inserted NA on
## the variable breaks the
for (i in 1:numberOfCycles) {
    if (dataCyberSecuritySurvey2019TidyNameSize$sizea[i] == -97) {
        dataCyberSecuritySurvey2019TidyNameSize$sizea[i] = NA
    }
    if (dataCyberSecuritySurvey2019TidyNameSize$sizeb[i] == -97) {
        dataCyberSecuritySurvey2019TidyNameSize$sizeb[i] = NA
    }

}

############################################## 2020
############################################## ###############################################
```

```
numberOfCycles = length(dataCyberSecuritySurvey2020TidyName$sizea)

dataCyberSecuritySurvey2020TidyNameSize = dataCyberSecuritySurvey2020TidyName

## apparently we have to be careful because an already inserted NA on
## the variable breaks the
for (i in 1:numberOfCycles) {
    if (dataCyberSecuritySurvey2020TidyNameSize$sizea[i] == -97) {
        dataCyberSecuritySurvey2020TidyNameSize$sizea[i] = NA
    }
    if (dataCyberSecuritySurvey2020TidyNameSize$sizeb[i] == -97) {
        dataCyberSecuritySurvey2020TidyNameSize$sizeb[i] = NA
    }
}

############################################## 2021
############################################## ###############################################

numberOfCycles = length(dataCyberSecuritySurvey2021TidyName$sizea)

dataCyberSecuritySurvey2021TidyNameSize = dataCyberSecuritySurvey2021TidyName

## apparently we have to be careful because an already inserted NA on
## the variable breaks the
for (i in 1:numberOfCycles) {
    if (dataCyberSecuritySurvey2021TidyNameSize$sizea[i] == -97) {
        dataCyberSecuritySurvey2021TidyNameSize$sizea[i] = NA
    }
    if (dataCyberSecuritySurvey2021TidyNameSize$sizeb[i] == -97) {
        dataCyberSecuritySurvey2021TidyNameSize$sizeb[i] = NA
    }
}

############################################## 2022
############################################## ###############################################

numberOfCycles = length(dataCyberSecuritySurvey2022TidyName$sizea)

dataCyberSecuritySurvey2022TidyNameSize = dataCyberSecuritySurvey2022TidyName

## apparently we have to be careful because an already inserted NA on
## the variable breaks the
for (i in 1:numberOfCycles) {
    if (dataCyberSecuritySurvey2022TidyNameSize$sizea[i] == -97) {
        dataCyberSecuritySurvey2022TidyNameSize$sizea[i] = NA
    }
    if (dataCyberSecuritySurvey2022TidyNameSize$sizeb[i] == -97) {
        dataCyberSecuritySurvey2022TidyNameSize$sizeb[i] = NA
    }
}
```

```
## we don't need neither the combined regions (since those are for
## business analyzes and we don't do those) same for sector_comb1 and
## 2.

## region_comb? throw it in the trash. sector_comb1? throw it in the
## trash. sector_comb2? throw it in the trash.

############################################## 2018
############################################## ###############################################

dataCyberSecuritySurvey2018TidyNameSize = dataCyberSecuritySurvey2018TidyNameSize %>%
    select(-one_of("region_comb", "sector_comb1", "sector_comb2"))

############################################## 2019
############################################## ###############################################

dataCyberSecuritySurvey2019TidyNameSize = dataCyberSecuritySurvey2019TidyNameSize %>%
    select(-one_of("region_comb", "sector_comb2"))

############################################## 2020
############################################## ###############################################

dataCyberSecuritySurvey2020TidyNameSize = dataCyberSecuritySurvey2020TidyNameSize %>%
    select(-one_of("region_comb", "sector_comb2"))

############################################## 2021
############################################## ###############################################

dataCyberSecuritySurvey2021TidyNameSize = dataCyberSecuritySurvey2021TidyNameSize %>%
    select(-one_of("region_comb", "sector_comb2"))

############################################## 2022
############################################## ###############################################

dataCyberSecuritySurvey2022TidyNameSize = dataCyberSecuritySurvey2022TidyNameSize %>%
    select(-one_of("region_comb", "sector_comb2"))
```

```
## removing social media questions that are irrelevant because they are
## absolutely terrible metrics to understand the digitalization of an
## institution Note to self: if I have time get all of these type of
## functions in try catch because them breaking up with the select
## error is no good and it makes me cry every time I have to manually
## run a part of the snippet and see which is one the bad one
## https://r-lang.com/r-trycatch-function/ ## #FicaADica

############################################## 2018
############################################## ###############################################

dataCyberSecuritySurvey2018TidyNameSize = dataCyberSecuritySurvey2018TidyNameSize %>%
    select(-(online1:online11))

############################################## 2019
############################################## ###############################################

dataCyberSecuritySurvey2019TidyNameSize = dataCyberSecuritySurvey2019TidyNameSize %>%
    select(-(online1:online11))

############################################## 2020
############################################## ###############################################

dataCyberSecuritySurvey2020TidyNameSize = dataCyberSecuritySurvey2020TidyNameSize %>%
    select(-(online1:online11))

############################################## 2021
############################################## ###############################################

dataCyberSecuritySurvey2021TidyNameSize = dataCyberSecuritySurvey2021TidyNameSize %>%
    select(-(online1:online11))

############################################## 2022
############################################## ###############################################

dataCyberSecuritySurvey2022TidyNameSize = dataCyberSecuritySurvey2022TidyNameSize %>%
    select(-(online1:online14))
```

```
## removing the question about the mobile usage because it also is a
## terrible indicator of a company digitalization

############################################## 2018
############################################## ###############################################

dataCyberSecuritySurvey2018TidyNameSize = dataCyberSecuritySurvey2018TidyNameSize %>%
    select(-mobile)

############################################## 2019
############################################## ###############################################

dataCyberSecuritySurvey2019TidyNameSize = dataCyberSecuritySurvey2019TidyNameSize %>%
    select(-mobile)
```

```
############################################## 2020
############################################## ###############################################

dataCyberSecuritySurvey2020TidyNameSize = dataCyberSecuritySurvey2020TidyNameSize %>%
    select(-mobile)

############################################## 2021
############################################## ###############################################

dataCyberSecuritySurvey2021TidyNameSize = dataCyberSecuritySurvey2021TidyNameSize %>%
    select(-mobile)

############################################## 2022
############################################## ###############################################

dataCyberSecuritySurvey2022TidyNameSize = dataCyberSecuritySurvey2022TidyNameSize %>%
    select(-mobile)
```

```
## question about the attitude and outsourcing of cyber security have
## been removed the the surveys starting from 2020 so it doesn't make
## sense to keep them in the 2018 and 2019 data set

## I will start doing some proper garbage collection and this time I am
## not just taking myself out I will only ever have the original data
## and the most recent modified one


############################################## 2018
############################################## ###############################################

dataCyberSecuritySurvey2018TidyNameSize = dataCyberSecuritySurvey2018TidyNameSize %>%
    select(-(outsource:attitude4))

############################################## 2019
############################################## ###############################################

dataCyberSecuritySurvey2019TidyNameSize = dataCyberSecuritySurvey2019TidyNameSize %>%
    select(-(outsource:attitude4))
```

```
## since we want to have access to some proper data we will tidy the
## questions about how big of a priority is cyber security and how
## often are the higher ups updated about it this could really use some
## try catches because for the some weird reason -97 == NA does not
## return true or false, this is like javascript levels of bad

## also excepting this very first one the removals will be at the end
```

```r
## so they are computed as if they were a transaction because try and
## catch is a lie to sell more lines of codes

## thanks to a blessing for our lord not finding the object only gives
## a warning and we ignore those as long as it still lets run the rest
## of the code

rm(dataCyberSecuritySurvey2018TidyName)
rm(dataCyberSecuritySurvey2019TidyName)
rm(dataCyberSecuritySurvey2020TidyName)
rm(dataCyberSecuritySurvey2021TidyName)
rm(dataCyberSecuritySurvey2022TidyName)

############################################## 2018
############################################## ###############################################
numberOfCycles = length(dataCyberSecuritySurvey2018TidyNameSize$priority)

dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSize

## apparently we have to be careful because an already inserted NA on
## the variable breaks the
for (i in 1:numberOfCycles) {
    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$priority[i] == -97) {
        dataCyberSecuritySurvey2018TidyNameSizeCyber$priority[i] = NA
    }
    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$update[i] == -97) {
        dataCyberSecuritySurvey2018TidyNameSizeCyber$update[i] = NA
    }

}

############################################## 2019
############################################## ###############################################

numberOfCycles = length(dataCyberSecuritySurvey2019TidyNameSize$priority)

dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSize


## apparently we have to be careful because an already inserted NA on
## the variable breaks the
for (i in 1:numberOfCycles) {
    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$priority[i] == -97) {
        dataCyberSecuritySurvey2019TidyNameSizeCyber$priority[i] = NA
    }
    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$update[i] == -97) {
        dataCyberSecuritySurvey2019TidyNameSizeCyber$update[i] = NA
    }

}

############################################## 2020
############################################## ###############################################

numberOfCycles = length(dataCyberSecuritySurvey2020TidyNameSize$priority)

dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSize


## apparently we have to be careful because an already inserted NA on
## the variable breaks the
for (i in 1:numberOfCycles) {
    if (dataCyberSecuritySurvey2020TidyNameSizeCyber$priority[i] == -97) {
        dataCyberSecuritySurvey2020TidyNameSizeCyber$priority[i] = NA
    }
    if (dataCyberSecuritySurvey2020TidyNameSizeCyber$update[i] == -97) {
        dataCyberSecuritySurvey2020TidyNameSizeCyber$update[i] = NA
    }
}

############################################## 2021
############################################## ###############################################

numberOfCycles = length(dataCyberSecuritySurvey2021TidyNameSize$priority)

dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSize


## apparently we have to be careful because an already inserted NA on
## the variable breaks the
for (i in 1:numberOfCycles) {
    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$priority[i] == -97) {
        dataCyberSecuritySurvey2021TidyNameSizeCyber$priority[i] = NA
    }
    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$update[i] == -97) {
        dataCyberSecuritySurvey2021TidyNameSizeCyber$update[i] = NA
    }
}

############################################## 2022
############################################## ###############################################

numberOfCycles = length(dataCyberSecuritySurvey2022TidyNameSize$priority)
```

```
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSize


## apparently we have to be careful because an already inserted NA on
## the variable breaks the
for (i in 1:numberOfCycles) {
    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$priority[i] == -97) {
        dataCyberSecuritySurvey2022TidyNameSizeCyber$priority[i] = NA
    }
    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$update[i] == -97) {
        dataCyberSecuritySurvey2022TidyNameSizeCyber$update[i] = NA
    }
}

## garbage man? Well, of course I know him. He is me.

rm(dataCyberSecuritySurvey2018TidyNameSize)
rm(dataCyberSecuritySurvey2019TidyNameSize)
rm(dataCyberSecuritySurvey2020TidyNameSize)
rm(dataCyberSecuritySurvey2021TidyNameSize)
rm(dataCyberSecuritySurvey2022TidyNameSize)
```

```
## questions about reason of investment in cybersecuirty were removed
## form the pre-pilot survey in 2020

############################################## 2018
############################################## ###############################################

dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(reason1:reason27))

############################################## 2019
############################################## ###############################################

dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-(reason1:reason28))

## the rest were already deleted
```

```
## removing the cyber security insurance claims because they don't give
## us relevant data to what we are analyzing in the data set pro tip:
## having insurance does not make you more or less likely to be
## targeted nor does it change the costs of the attack

############################################## 2018
############################################## ###############################################
```

```
dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(insurex:noinsure19))

############################################## 2019
############################################## ###############################################

dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-(insurex:noinsure19))

############################################## 2020
############################################## ###############################################

dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
    select(-(insurex:claim))

############################################## 2021
############################################## ###############################################

dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-(insurex:claim))

############################################## 2022
############################################## ###############################################

dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(insurex:claim))
```

```
## we are removing the questions about asking for info, advice,
## guidance about cyber security or government schemes

############################################## 2018
############################################## ###############################################

dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(info1:trainwho7))

############################################## 2019
############################################## ###############################################

dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-(info1:trainwho7))

############################################## 2020
############################################## ###############################################

dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
    select(-(info1:scheme5))
```

```
############################################## 2021
############################################## ###############################################

dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-(info1:scheme5))

############################################## 2022
############################################## ###############################################

## here we have more that we need to remove because they introduced
## questions similar to the ones that we are removing but with the
## government as a source for reason to make implementations which we
## are also not interested in

dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(info1:govtact44))
```

0.6 Management - now is when the policies and procedures are evaluated

```
## manage 1 - Board members/trustees with responsibility for cyber
## security manage 2 - outsourcing cyber security manage 3 - formal
## policy or policies in place covering cyber security risks manage 4 -
## Business Continuity Plan manage 5 - Staff members whose job role
## includes information security or governance //it stopped being used
## after the 2020 survey manage 6 - don't know/missing data manage 7 -
## absolutely nothings, good luck have fun (rip bozzo) manage 8 -
## written list of what is critical to protect (only exists in the
## survey of 2022 not to be used)

## altura de me desemerdar que esta aqui esta mesmo grossa não faz
## frio, nem orvalho, está a chover para caralho converting the final
## value to a collection so I can append all the values //facepalm this
## is where the coping begins, thank goodness no one will ever know
## what I had done here before refactoring and optimising the code

## managementContinuity - there is a business continuity plan (manage
## 4) or there are formal policies implemented (outcome 3)
## managementCyber - board members or trustees have cyber security
## responsibilities (manage 1) or cyber security is being outsourced
## (manage 2) (also known as the at least they tried but no matter how
## funny it is terrible variable name)

##################################################### 2018
##################################################### ########################################
```

```
numberOfCycles = length(dataCyberSecuritySurvey2018TidyNameSizeCyber$manage1)

dataCyberSecuritySurvey2018TidyNameSizeCyber$managementContinuity = 0
dataCyberSecuritySurvey2018TidyNameSizeCyber$managementCyber = 0


for (i in 1:numberOfCycles) {

    ## at least this time I am not starting from the last so I don't
    ## have to wait the 10 minutes for my computer to fry some eggs
    ## while it compiles

    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$manage1[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 1 or 2

        dataCyberSecuritySurvey2018TidyNameSizeCyber$managementCyber[i] = 1

    }

    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$manage3[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 3 or 4

        dataCyberSecuritySurvey2018TidyNameSizeCyber$managementContinuity[i] = 1

    }

    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$manage6[i] == 1) {

        dataCyberSecuritySurvey2018TidyNameSizeCyber$managementCyber[i] = NA
        dataCyberSecuritySurvey2018TidyNameSizeCyber$managementContinuity[i] = NA

    }

}



##################################################### 2019
##################################################### ########################################


numberOfCycles = length(dataCyberSecuritySurvey2019TidyNameSizeCyber$manage1)

dataCyberSecuritySurvey2019TidyNameSizeCyber$managementContinuity = 0
dataCyberSecuritySurvey2019TidyNameSizeCyber$managementCyber = 0
```

```
for (i in 1:numberOfCycles) {

    ## at least this time I am not starting from the last so I don't
    ## have to wait the 10 minutes for my computer to fry some eggs
    ## while it compiles

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$manage1[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 1 or 2

        dataCyberSecuritySurvey2019TidyNameSizeCyber$managementCyber[i] = 1

    }

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$manage3[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 3 or 4

        dataCyberSecuritySurvey2019TidyNameSizeCyber$managementContinuity[i] = 1

    }

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$manage6[i] == 1) {

        dataCyberSecuritySurvey2019TidyNameSizeCyber$managementCyber[i] = NA
        dataCyberSecuritySurvey2019TidyNameSizeCyber$managementContinuity[i] = NA

    }
}



##################################################### 2020
##################################################### ########################################


numberOfCycles = length(dataCyberSecuritySurvey2020TidyNameSizeCyber$manage1)

dataCyberSecuritySurvey2020TidyNameSizeCyber$managementContinuity = 0
dataCyberSecuritySurvey2020TidyNameSizeCyber$managementCyber = 0


for (i in 1:numberOfCycles) {

    ## at least this time I am not starting from the last so I don't
    ## have to wait the 10 minutes for my computer to fry some eggs
    ## while it compiles

    if (dataCyberSecuritySurvey2020TidyNameSizeCyber$manage1[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 1 or 2

        dataCyberSecuritySurvey2020TidyNameSizeCyber$managementCyber[i] = 1

    }

    if (dataCyberSecuritySurvey2020TidyNameSizeCyber$manage3[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 3 or 4

        dataCyberSecuritySurvey2020TidyNameSizeCyber$managementContinuity[i] = 1

    }

    if (dataCyberSecuritySurvey2020TidyNameSizeCyber$manage6[i] == 1) {

        dataCyberSecuritySurvey2020TidyNameSizeCyber$managementCyber[i] = NA
        dataCyberSecuritySurvey2020TidyNameSizeCyber$managementContinuity[i] = NA

    }
}


##################################################### 2021
##################################################### ########################################


numberOfCycles = length(dataCyberSecuritySurvey2021TidyNameSizeCyber$manage1)

dataCyberSecuritySurvey2021TidyNameSizeCyber$managementContinuity = 0
dataCyberSecuritySurvey2021TidyNameSizeCyber$managementCyber = 0


for (i in 1:numberOfCycles) {

    ## at least this time I am not starting from the last so I don't
    ## have to wait the 10 minutes for my computer to fry some eggs
    ## while it compiles

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$manage1[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 1 or 2
        dataCyberSecuritySurvey2021TidyNameSizeCyber$managementCyber[i] = 1

    }

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$manage3[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 3 or 4

        dataCyberSecuritySurvey2021TidyNameSizeCyber$managementContinuity[i] = 1

    }

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$manage6[i] == 1) {

        dataCyberSecuritySurvey2021TidyNameSizeCyber$managementCyber[i] = NA
        dataCyberSecuritySurvey2021TidyNameSizeCyber$managementContinuity[i] = NA

    }
}



##################################################### 2022
##################################################### ########################################



numberOfCycles = length(dataCyberSecuritySurvey2022TidyNameSizeCyber$manage1)

dataCyberSecuritySurvey2022TidyNameSizeCyber$managementContinuity = 0
dataCyberSecuritySurvey2022TidyNameSizeCyber$managementCyber = 0


for (i in 1:numberOfCycles) {

    dataCyberSecuritySurvey2022TidyNameSizeCyber$manage1[i] = replace_na(dataCyberSecuritySurv
        -10001)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$manage2[i] = replace_na(dataCyberSecuritySurv
        -10001)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$manage3[i] = replace_na(dataCyberSecuritySurv
        -10001)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$manage4[i] = replace_na(dataCyberSecuritySurv
        -10001)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$manage5[i] = replace_na(dataCyberSecuritySurv
        -10001)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$manage6[i] = replace_na(dataCyberSecuritySurv
        -10001)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$manage7[i] = replace_na(dataCyberSecuritySurv
        -10001)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$manage8[i] = replace_na(dataCyberSecuritySurv
        -10001)

    ## at least this time I am not starting from the last so I don't
    ## have to wait the 10 minutes for my computer to fry some eggs
    ## while it compiles

    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$manage1[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 1 or 2

        dataCyberSecuritySurvey2022TidyNameSizeCyber$managementCyber[i] = 1

    } else if (dataCyberSecuritySurvey2022TidyNameSizeCyber$manage1[i] == -10001 &&
        dataCyberSecuritySurvey2022TidyNameSizeCyber$manage2[i] == -10001) {

        dataCyberSecuritySurvey2022TidyNameSizeCyber$managementCyber[i] = NA
    }

    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$manage3[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 3 or 4

        dataCyberSecuritySurvey2022TidyNameSizeCyber$managementContinuity[i] = 1

    } else if (dataCyberSecuritySurvey2022TidyNameSizeCyber$manage3[i] == -10001 &&
        dataCyberSecuritySurvey2022TidyNameSizeCyber$manage4[i] == -10001) {

        dataCyberSecuritySurvey2022TidyNameSizeCyber$managementCyber[i] = NA
    }

    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$manage6[i] == 1) {

        dataCyberSecuritySurvey2022TidyNameSizeCyber$managementCyber[i] = NA
        dataCyberSecuritySurvey2022TidyNameSizeCyber$managementContinuity[i] = NA

    }
}



dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(manage1:manage7))
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-(manage1:manage7))
```

```
dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
    select(-(manage1:manage7))
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-(manage1:manage7))
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(manage1:manage8))
```

```
## now we are removing the reasons why they don't have the appropriate
## measures because we are more interested in the questions about
## security after these ones also it was deleted after 2019

## that is a catch 22
dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(nopol1:nopol22))
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-(nopol1:nopol22))
```

```
## sadly all the questions about measures done in the last 12 moths
## have changes quite a bit during the years which makes it impossible
## to have a good year to year analysis when we aren't comparing the
## same thing

dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(ident1:ident8))
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-(ident1:ident8))
dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
    select(-(ident1:ident11))
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-(ident1:ident11))
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(ident1:ident7))

## this only exists in 2022 so it makes no sense to look at
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(comply1:audit))
```

0.7 Rules for company policies

```
## This is where the fun begins with some proper policies simping for
## incremental backups


## rule 1 - applying software updates rule 2 - up to date maleware
## protection rule 3 - well configured firewalls rule 4 - proper
## permission configuration rule 5 - monitoring user activity rule 6 -
## encrypting personal data // only used in 2018 rule 7 - security
## controls on company devices rule 8 - only allowing access from
## company devices rule 9 - segregated guest wireless / so basically a
## DMZ rule 10 - don't know rule 11 - none (YOLO) rule 12 - strong
## passwords //only used in 2018 rule 13 - backup data to the cloud
## (diskette robots in data center go brrrrrrr) rule 14 - backup the
## data to another place that isn't the cloud rule 15 - storing and
## moving data/files securely //wasn't used in 2018 rule 16 - 2 factor
## authentication // only used in 2019 rule 17 - policy for strong
## passwords //not til 2020 rule 18 - VPN (virtual private network)
## //only in 2022 rule 19 - phishing procedure // only in 2022 rule 20
## - authentication when accessing the network // only in 2022


## TODO TODO TODO TODO unduck this mess as well because I just don't
## know anymore good news is that I have a solution, bad news is that
## it is not a perfect solution at least I won't be able to cause a
## stack overflow because with a precision of 53 bits, and represents
## to that precision a range of absolute values from about 2e-308 to
## 2e+308


## Rules grouping for optimisation

## TODO might have too much in common and separate both security confs
## and updating with .baks

## rulesUpdating - keeping spftware and maleware protection up to date
## (rule 1 and 2) and baking up information ( rule 13, 14 and 15(not in
## 2018 ) ) rulesSecurityConfigs - well configured firewalls and
## permission (rule 3 and 4), DMZ (rule 9) and strong passwords (rule
## 17 not in 2018) rulesUserControl - monitoring user activity (rule 5)
## as well as good security control and access control (rule 7 and rule
## 8)


###################################################### 2018
###################################################### ########################################



numberOfCycles = length(dataCyberSecuritySurvey2018TidyNameSizeCyber$rules1)

dataCyberSecuritySurvey2018TidyNameSizeCyber$rulesUpdating = 0
dataCyberSecuritySurvey2018TidyNameSizeCyber$rulesSecurityConfigs = 0
dataCyberSecuritySurvey2018TidyNameSizeCyber$rulesUserControl = 0
```

```
for (i in 1:numberOfCycles) {

    ## the code is now be faster and other hilarious jokes you can tell
    ## yourself

    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$rules1[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2018TidyNameSizeCyber$rules13[i] == 1 ||
        dataCyberSecuritySurvey2018TidyNameSizeCyber$rules14[i] == 1  #dataCyberSecuritySurvey
) {
        ## either 1,2,13,14 and 15 after 2018

        dataCyberSecuritySurvey2018TidyNameSizeCyber$rulesUpdating[i] = 1

    }
    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$rules3[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2018TidyNameSizeCyber$rules9[i] == 1  #dataCyberSecuritySu
) {
        ## either 3,4,9 and 17 after 2018

        dataCyberSecuritySurvey2018TidyNameSizeCyber$rulesSecurityConfigs[i] = 1

    }

    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$rules5[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2018TidyNameSizeCyber$rules8[i] == 1) {
        ## either 5,7,8

        dataCyberSecuritySurvey2018TidyNameSizeCyber$rulesUserControl[i] = 1

    }

    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$rules10[i] == 1) {

        dataCyberSecuritySurvey2018TidyNameSizeCyber$rulesUserControl[i] = NA
        dataCyberSecuritySurvey2018TidyNameSizeCyber$rulesSecurityConfigs[i] = NA
        dataCyberSecuritySurvey2018TidyNameSizeCyber$rulesUpdating[i] = NA

    }


}


###################################################### 2019
###################################################### ########################################
```

```
numberOfCycles = length(dataCyberSecuritySurvey2019TidyNameSizeCyber$rules1)

dataCyberSecuritySurvey2019TidyNameSizeCyber$rulesUpdating = 0
dataCyberSecuritySurvey2019TidyNameSizeCyber$rulesSecurityConfigs = 0
dataCyberSecuritySurvey2019TidyNameSizeCyber$rulesUserControl = 0


for (i in 1:numberOfCycles) {

    ## the code is now be faster and other hilarious jokes you can tell
    ## yourself

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$rules1[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2019TidyNameSizeCyber$rules13[i] == 1 ||
        dataCyberSecuritySurvey2019TidyNameSizeCyber$rules14[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 1,2,13,14 and 15

        dataCyberSecuritySurvey2019TidyNameSizeCyber$rulesUpdating[i] = 1

    }
    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$rules3[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2019TidyNameSizeCyber$rules9[i] == 1  #dataCyberSecuritySu
) {
        ## either 3,4,9 and 17 after 2019

        dataCyberSecuritySurvey2019TidyNameSizeCyber$rulesSecurityConfigs[i] = 1

    }

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$rules5[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2019TidyNameSizeCyber$rules8[i] == 1) {
        ## either 5,7,8

        dataCyberSecuritySurvey2019TidyNameSizeCyber$rulesUserControl[i] = 1

    }

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$rules10[i] == 1) {

        dataCyberSecuritySurvey2019TidyNameSizeCyber$rulesUserControl[i] = NA
        dataCyberSecuritySurvey2019TidyNameSizeCyber$rulesSecurityConfigs[i] = NA
        dataCyberSecuritySurvey2019TidyNameSizeCyber$rulesUpdating[i] = NA

    }
```

```
}



##################################################### 2020
##################################################### ##########################################



numberOfCycles = length(dataCyberSecuritySurvey2020TidyNameSizeCyber$rules1)

dataCyberSecuritySurvey2020TidyNameSizeCyber$rulesUpdating = 0
dataCyberSecuritySurvey2020TidyNameSizeCyber$rulesSecurityConfigs = 0
dataCyberSecuritySurvey2020TidyNameSizeCyber$rulesUserControl = 0


for (i in 1:numberOfCycles) {

    ## the code is now be faster and other hilarious jokes you can tell
    ## yourself

    if (dataCyberSecuritySurvey2020TidyNameSizeCyber$rules1[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2020TidyNameSizeCyber$rules13[i] == 1 ||
        dataCyberSecuritySurvey2020TidyNameSizeCyber$rules14[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 1,2,13,14 and 15

        dataCyberSecuritySurvey2020TidyNameSizeCyber$rulesUpdating[i] = 1

    }
    if (dataCyberSecuritySurvey2020TidyNameSizeCyber$rules3[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2020TidyNameSizeCyber$rules9[i] == 1 ||
        dataCyberSecuritySurvey2020TidyNameSizeCyber$rules17[i] == 1) {
        ## either 3,4,9 and 17

        dataCyberSecuritySurvey2020TidyNameSizeCyber$rulesSecurityConfigs[i] = 1

    }

    if (dataCyberSecuritySurvey2020TidyNameSizeCyber$rules5[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2020TidyNameSizeCyber$rules8[i] == 1) {
        ## either 5,7,8

        dataCyberSecuritySurvey2020TidyNameSizeCyber$rulesUserControl[i] = 1

    }
    if (dataCyberSecuritySurvey2020TidyNameSizeCyber$rules10[i] == 1) {

        dataCyberSecuritySurvey2020TidyNameSizeCyber$rulesUserControl[i] = NA
        dataCyberSecuritySurvey2020TidyNameSizeCyber$rulesSecurityConfigs[i] = NA
        dataCyberSecuritySurvey2020TidyNameSizeCyber$rulesUpdating[i] = NA

    }


}

##################################################### 2021
##################################################### ########################################


numberOfCycles = length(dataCyberSecuritySurvey2021TidyNameSizeCyber$rules1)

dataCyberSecuritySurvey2021TidyNameSizeCyber$rulesUpdating = 0
dataCyberSecuritySurvey2021TidyNameSizeCyber$rulesSecurityConfigs = 0
dataCyberSecuritySurvey2021TidyNameSizeCyber$rulesUserControl = 0


for (i in 1:numberOfCycles) {

    ## the code is now be faster and other hilarious jokes you can tell
    ## yourself

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$rules1[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2021TidyNameSizeCyber$rules13[i] == 1 ||
        dataCyberSecuritySurvey2021TidyNameSizeCyber$rules14[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 1,2,13,14 and 15

        dataCyberSecuritySurvey2021TidyNameSizeCyber$rulesUpdating[i] = 1

    }
    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$rules3[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2021TidyNameSizeCyber$rules9[i] == 1 ||
        dataCyberSecuritySurvey2021TidyNameSizeCyber$rules17[i] == 1) {
        ## either 3,4,9 and 17

        dataCyberSecuritySurvey2021TidyNameSizeCyber$rulesSecurityConfigs[i] = 1

    }

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$rules5[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2021TidyNameSizeCyber$rules8[i] == 1) {
```

```
        ## either 5,7,8

        dataCyberSecuritySurvey2021TidyNameSizeCyber$rulesUserControl[i] = 1

    }

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$rules10[i] == 1) {

        dataCyberSecuritySurvey2021TidyNameSizeCyber$rulesUserControl[i] = NA
        dataCyberSecuritySurvey2021TidyNameSizeCyber$rulesSecurityConfigs[i] = NA
        dataCyberSecuritySurvey2021TidyNameSizeCyber$rulesUpdating[i] = NA

    }


}


##################################################### 2022
##################################################### ########################################


numberOfCycles = length(dataCyberSecuritySurvey2022TidyNameSizeCyber$rules1)

dataCyberSecuritySurvey2022TidyNameSizeCyber$rulesUpdating = 0
dataCyberSecuritySurvey2022TidyNameSizeCyber$rulesSecurityConfigs = 0
dataCyberSecuritySurvey2022TidyNameSizeCyber$rulesUserControl = 0


for (i in 1:numberOfCycles) {

    ## oh my ducking god just why, this error message was worse then
    ## c++ apparently if you have a missing value on an if with more
    ## than 1 parameter it shows that the missing value error is on the
    ## first parameter so in this case was rule 5, i swear not even
    ## with enough crying and praying I would have gotten there, this
    ## bull crap

    dataCyberSecuritySurvey2022TidyNameSizeCyber$rules8[i] = replace_na(dataCyberSecuritySurve
        -10004)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$rules9[i] = replace_na(dataCyberSecuritySurve
        -10004)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$rules12[i] = replace_na(dataCyberSecuritySurv
        -10004)

    ## the code is now be faster and other hilarious jokes you can tell
    ## yourself
    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$rules1[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2022TidyNameSizeCyber$rules13[i] == 1 ||
        dataCyberSecuritySurvey2022TidyNameSizeCyber$rules14[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 1,2,13,14 and 15

        dataCyberSecuritySurvey2022TidyNameSizeCyber$rulesUpdating[i] = 1

    }
    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$rules3[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2022TidyNameSizeCyber$rules9[i] == 1 ||
        dataCyberSecuritySurvey2022TidyNameSizeCyber$rules17[i] == 1) {
        ## either 3,4,9 and 17

        dataCyberSecuritySurvey2022TidyNameSizeCyber$rulesSecurityConfigs[i] = 1

    }

    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$rules5[i] == 1 || dataCyberSecuritySurvey
        1 || dataCyberSecuritySurvey2022TidyNameSizeCyber$rules8[i] == 1) {
        ## either 5,7,8

        dataCyberSecuritySurvey2022TidyNameSizeCyber$rulesUserControl[i] = 1

    }

    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$rules10[i] == 1) {

        dataCyberSecuritySurvey2022TidyNameSizeCyber$rulesUserControl[i] = NA
        dataCyberSecuritySurvey2022TidyNameSizeCyber$rulesSecurityConfigs[i] = NA
        dataCyberSecuritySurvey2022TidyNameSizeCyber$rulesUpdating[i] = NA

    }


}


## now we can remove all those rules columns that we are no longer
## using

dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(rules1:rules14))
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-(rules1:rules16))
dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
    select(-(rules1:rules17))
```

```
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-(rules1:rules17))
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(rules1:rules20))
```

0.8 Policies

```
## we do a little policing but sadly not the one QoS type on cisco
## servers to be fair it would be as painful to debug


## policy 1 - what can be stored in the removable devices policy 2 -
## remote working policy 3 - what staff are permitted to do on your
## organisations IT devices policy 4 - use of personally-owned devices
## for business activities policy 5 - Use of new digital technologies
## such as cloud computing (seriously what the hell is this question
## smh) policy 6 - data classification policy 7 - a Document Management
## System policy 8 - don't know (estudasses) policy 9 - none of these
## (YOLO) policies 10,11 and 12 were only made in 2022 but since they
## started using policy 11 and 12 instead of the policy 6 and 7 they
## will replace them policy 11 - SaS (software as a service) policy 12
## - how to store data




## TODO TODO TODO TODO unduck this mess as well because I just don't
## know anymore good news is that I have a solution, bad news is that
## it is not a perfect solution at least I won't be able to cause a
## stack overflow because with a precision of 53 bits, and represents
## to that precision a range of absolute values from about 2e-308 to
## 2e+308

## even better news I have a better solution that will make the code
## run with two legs instead of just half a leg

## policyStaffAccess - staff who is allowed to work remotely (policy
## 2), policing of what staff are permited to do om company devices
## (policy 3) and cloud computing (policy 5) policyData -
## classification of data (policy 6) and document management system
## (policy 7) policyPrivate - staff is not allowed to work on personal
## devices (policy 4) and cannot just store anything on removable
## devices (policy 1)
```

```
##################################################### 2018
##################################################### ########################################

## at this point I don't know what my code is more, poorly optimized,
## spaghetti or just straight up cringe

numberOfCycles = length(dataCyberSecuritySurvey2018TidyNameSizeCyber$policy1)

dataCyberSecuritySurvey2018TidyNameSizeCyber$policyStaffAccess = 0
dataCyberSecuritySurvey2018TidyNameSizeCyber$policyData = 0
dataCyberSecuritySurvey2018TidyNameSizeCyber$policyPrivate = 0


for (i in 1:numberOfCycles) {

    dataCyberSecuritySurvey2018TidyNameSizeCyber$policy1[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2018TidyNameSizeCyber$policy2[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2018TidyNameSizeCyber$policy3[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2018TidyNameSizeCyber$policy4[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2018TidyNameSizeCyber$policy5[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2018TidyNameSizeCyber$policy6[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2018TidyNameSizeCyber$policy7[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2018TidyNameSizeCyber$policy8[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2018TidyNameSizeCyber$policy9[i] = replace_na(dataCyberSecuritySurv
        -10005)

    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$policy2[i] == 1 || dataCyberSecuritySurve
        1 || dataCyberSecuritySurvey2018TidyNameSizeCyber$policy5[i] == 1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2018TidyNameSizeCyber$policyStaffAccess[i] = 1

    } else if (dataCyberSecuritySurvey2018TidyNameSizeCyber$policy2[i] == -10005 &&
        dataCyberSecuritySurvey2018TidyNameSizeCyber$policy3[i] == -10005 &&
        dataCyberSecuritySurvey2018TidyNameSizeCyber$policy5[i] == -10005) {

        dataCyberSecuritySurvey2018TidyNameSizeCyber$policyStaffAccess[i] = NA

    }
    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$policy6[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2018TidyNameSizeCyber$policyData[i] = 1

    } else if (dataCyberSecuritySurvey2018TidyNameSizeCyber$policy6[i] == -10005 &&
        dataCyberSecuritySurvey2018TidyNameSizeCyber$policy7[i] == -10005) {

        dataCyberSecuritySurvey2018TidyNameSizeCyber$policyData[i] = NA

    }

    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$policy1[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2018TidyNameSizeCyber$policyPrivate[i] = 1

    } else if (dataCyberSecuritySurvey2018TidyNameSizeCyber$policy1[i] == -10005 &&
        dataCyberSecuritySurvey2018TidyNameSizeCyber$policy4[i] == -10005) {

        dataCyberSecuritySurvey2018TidyNameSizeCyber$policyPrivate[i] = NA

    }

    if (dataCyberSecuritySurvey2018TidyNameSizeCyber$policy8[i] == 1) {
        ## NA

        dataCyberSecuritySurvey2018TidyNameSizeCyber$policyPrivate[i] = NA
        dataCyberSecuritySurvey2018TidyNameSizeCyber$policyData[i] = NA
        dataCyberSecuritySurvey2018TidyNameSizeCyber$policyStaffAccess[i] = NA

    }

}


##################################################### 2019
##################################################### ########################################



## at this point I don't know what my code is more, poorly optimized,
## spaghetti or just straight up cringe

numberOfCycles = length(dataCyberSecuritySurvey2019TidyNameSizeCyber$policy1)
```

```
dataCyberSecuritySurvey2019TidyNameSizeCyber$policyStaffAccess = 0
dataCyberSecuritySurvey2019TidyNameSizeCyber$policyData = 0
dataCyberSecuritySurvey2019TidyNameSizeCyber$policyPrivate = 0


for (i in 1:numberOfCycles) {

    dataCyberSecuritySurvey2019TidyNameSizeCyber$policy1[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$policy2[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$policy3[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$policy4[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$policy5[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$policy6[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$policy7[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$policy8[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$policy9[i] = replace_na(dataCyberSecuritySurv
        -10005)

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$policy2[i] == 1 || dataCyberSecuritySurve
        1 || dataCyberSecuritySurvey2019TidyNameSizeCyber$policy5[i] == 1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2019TidyNameSizeCyber$policyStaffAccess[i] = 1

    } else if (dataCyberSecuritySurvey2019TidyNameSizeCyber$policy2[i] == -10005 &&
        dataCyberSecuritySurvey2019TidyNameSizeCyber$policy3[i] == -10005 &&
        dataCyberSecuritySurvey2019TidyNameSizeCyber$policy5[i] == -10005) {

        dataCyberSecuritySurvey2019TidyNameSizeCyber$policyStaffAccess[i] = NA

    }

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$policy6[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2019TidyNameSizeCyber$policyData[i] = 1

    } else if (dataCyberSecuritySurvey2019TidyNameSizeCyber$policy6[i] == -10005 &&
        dataCyberSecuritySurvey2019TidyNameSizeCyber$policy7[i] == -10005) {

        dataCyberSecuritySurvey2019TidyNameSizeCyber$policyData[i] = NA

    }

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$policy1[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2019TidyNameSizeCyber$policyPrivate[i] = 1

    } else if (dataCyberSecuritySurvey2019TidyNameSizeCyber$policy1[i] == -10005 &&
        dataCyberSecuritySurvey2019TidyNameSizeCyber$policy4[i] == -10005) {

        dataCyberSecuritySurvey2019TidyNameSizeCyber$policyPrivate[i] = NA

    }

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$policy8[i] == 1) {
        ## NA

        dataCyberSecuritySurvey2019TidyNameSizeCyber$policyPrivate[i] = NA
        dataCyberSecuritySurvey2019TidyNameSizeCyber$policyData[i] = NA
        dataCyberSecuritySurvey2019TidyNameSizeCyber$policyStaffAccess[i] = NA

    }

}


##################################################### 2020
##################################################### ########################################



## at this point I don't know what my code is more, poorly optimized,
## spaghetti or just straight up cringe

numberOfCycles = length(dataCyberSecuritySurvey2020TidyNameSizeCyber$policy1)

dataCyberSecuritySurvey2020TidyNameSizeCyber$policyStaffAccess = 0
dataCyberSecuritySurvey2020TidyNameSizeCyber$policyData = 0
dataCyberSecuritySurvey2020TidyNameSizeCyber$policyPrivate = 0


for (i in 1:numberOfCycles) {
```

```
dataCyberSecuritySurvey2020TidyNameSizeCyber$policy1[i] = replace_na(dataCyberSecuritySurv
    -10005)
dataCyberSecuritySurvey2020TidyNameSizeCyber$policy2[i] = replace_na(dataCyberSecuritySurv
    -10005)
dataCyberSecuritySurvey2020TidyNameSizeCyber$policy3[i] = replace_na(dataCyberSecuritySurv
    -10005)
dataCyberSecuritySurvey2020TidyNameSizeCyber$policy4[i] = replace_na(dataCyberSecuritySurv
    -10005)
dataCyberSecuritySurvey2020TidyNameSizeCyber$policy5[i] = replace_na(dataCyberSecuritySurv
    -10005)
dataCyberSecuritySurvey2020TidyNameSizeCyber$policy6[i] = replace_na(dataCyberSecuritySurv
    -10005)
dataCyberSecuritySurvey2020TidyNameSizeCyber$policy7[i] = replace_na(dataCyberSecuritySurv
    -10005)
dataCyberSecuritySurvey2020TidyNameSizeCyber$policy8[i] = replace_na(dataCyberSecuritySurv
    -10005)
dataCyberSecuritySurvey2020TidyNameSizeCyber$policy9[i] = replace_na(dataCyberSecuritySurv
    -10005)

if (dataCyberSecuritySurvey2020TidyNameSizeCyber$policy2[i] == 1 || dataCyberSecuritySurve
    1 || dataCyberSecuritySurvey2020TidyNameSizeCyber$policy5[i] == 1) {
    ## either 2,3,4

    dataCyberSecuritySurvey2020TidyNameSizeCyber$policyStaffAccess[i] = 1

} else if (dataCyberSecuritySurvey2020TidyNameSizeCyber$policy2[i] == -10005 &&
    dataCyberSecuritySurvey2020TidyNameSizeCyber$policy3[i] == -10005 &&
    dataCyberSecuritySurvey2020TidyNameSizeCyber$policy5[i] == -10005) {

    dataCyberSecuritySurvey2020TidyNameSizeCyber$policyStaffAccess[i] = NA

}

if (dataCyberSecuritySurvey2020TidyNameSizeCyber$policy6[i] == 1 || dataCyberSecuritySurve
    1) {
    ## either 2,3,4

    dataCyberSecuritySurvey2020TidyNameSizeCyber$policyData[i] = 1

} else if (dataCyberSecuritySurvey2020TidyNameSizeCyber$policy6[i] == -10005 &&
    dataCyberSecuritySurvey2020TidyNameSizeCyber$policy7[i] == -10005) {

    dataCyberSecuritySurvey2020TidyNameSizeCyber$policyData[i] = NA

}

if (dataCyberSecuritySurvey2020TidyNameSizeCyber$policy1[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2020TidyNameSizeCyber$policyPrivate[i] = 1

    } else if (dataCyberSecuritySurvey2020TidyNameSizeCyber$policy1[i] == -10005 &&
        dataCyberSecuritySurvey2020TidyNameSizeCyber$policy4[i] == -10005) {

        dataCyberSecuritySurvey2020TidyNameSizeCyber$policyPrivate[i] = NA

    }

    if (dataCyberSecuritySurvey2020TidyNameSizeCyber$policy8[i] == 1) {
        ## NA

        dataCyberSecuritySurvey2020TidyNameSizeCyber$policyPrivate[i] = NA
        dataCyberSecuritySurvey2020TidyNameSizeCyber$policyData[i] = NA
        dataCyberSecuritySurvey2020TidyNameSizeCyber$policyStaffAccess[i] = NA

    }

}

##################################################### 2021
##################################################### ########################################



## at this point I don't know what my code is more, poorly optimized,
## spaghetti or just straight up cringe

numberOfCycles = length(dataCyberSecuritySurvey2021TidyNameSizeCyber$policy1)

dataCyberSecuritySurvey2021TidyNameSizeCyber$policyStaffAccess = 0
dataCyberSecuritySurvey2021TidyNameSizeCyber$policyData = 0
dataCyberSecuritySurvey2021TidyNameSizeCyber$policyPrivate = 0


for (i in 1:numberOfCycles) {

    dataCyberSecuritySurvey2021TidyNameSizeCyber$policy1[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$policy2[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$policy3[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$policy4[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$policy5[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$policy6[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$policy7[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$policy8[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$policy9[i] = replace_na(dataCyberSecuritySurv
        -10005)

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$policy2[i] == 1 || dataCyberSecuritySurve
        1 || dataCyberSecuritySurvey2021TidyNameSizeCyber$policy5[i] == 1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2021TidyNameSizeCyber$policyStaffAccess[i] = 1

    } else if (dataCyberSecuritySurvey2021TidyNameSizeCyber$policy2[i] == -10005 &&
        dataCyberSecuritySurvey2021TidyNameSizeCyber$policy3[i] == -10005 &&
        dataCyberSecuritySurvey2021TidyNameSizeCyber$policy5[i] == -10005) {

        dataCyberSecuritySurvey2021TidyNameSizeCyber$policyStaffAccess[i] = NA

    }

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$policy6[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2021TidyNameSizeCyber$policyData[i] = 1

    } else if (dataCyberSecuritySurvey2021TidyNameSizeCyber$policy6[i] == -10005 &&
        dataCyberSecuritySurvey2021TidyNameSizeCyber$policy7[i] == -10005) {

        dataCyberSecuritySurvey2021TidyNameSizeCyber$policyData[i] = NA

    }

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$policy1[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2021TidyNameSizeCyber$policyPrivate[i] = 1

    } else if (dataCyberSecuritySurvey2021TidyNameSizeCyber$policy1[i] == -10005 &&
        dataCyberSecuritySurvey2021TidyNameSizeCyber$policy4[i] == -10005) {
        dataCyberSecuritySurvey2021TidyNameSizeCyber$policyPrivate[i] = NA

    }

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$policy8[i] == 1) {
        ## NA

        dataCyberSecuritySurvey2021TidyNameSizeCyber$policyPrivate[i] = NA
        dataCyberSecuritySurvey2021TidyNameSizeCyber$policyData[i] = NA
        dataCyberSecuritySurvey2021TidyNameSizeCyber$policyStaffAccess[i] = NA

    }

}


##################################################### 2022
##################################################### ########################################


## at this point I don't know what my code is more, poorly optimized,
## spaghetti or just straight up cringe

numberOfCycles = length(dataCyberSecuritySurvey2022TidyNameSizeCyber$policy1)

dataCyberSecuritySurvey2022TidyNameSizeCyber$policyStaffAccess = 0
dataCyberSecuritySurvey2022TidyNameSizeCyber$policyData = 0
dataCyberSecuritySurvey2022TidyNameSizeCyber$policyPrivate = 0


for (i in 1:numberOfCycles) {

    dataCyberSecuritySurvey2022TidyNameSizeCyber$policy1[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$policy2[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$policy3[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$policy4[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$policy5[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$policy11[i] = replace_na(dataCyberSecuritySur
        -10005)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$policy12[i] = replace_na(dataCyberSecuritySur
        -10005)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$policy8[i] = replace_na(dataCyberSecuritySurv
        -10005)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$policy9[i] = replace_na(dataCyberSecuritySurv
        -10005)

    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$policy2[i] == 1 || dataCyberSecuritySurve
        1 || dataCyberSecuritySurvey2022TidyNameSizeCyber$policy5[i] == 1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2022TidyNameSizeCyber$policyStaffAccess[i] = 1

    } else if (dataCyberSecuritySurvey2022TidyNameSizeCyber$policy2[i] == -10005 &&
        dataCyberSecuritySurvey2022TidyNameSizeCyber$policy3[i] == -10005 &&
        dataCyberSecuritySurvey2022TidyNameSizeCyber$policy5[i] == -10005) {

        dataCyberSecuritySurvey2022TidyNameSizeCyber$policyStaffAccess[i] = NA

    }

    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$policy11[i] == 1 ||
        dataCyberSecuritySurvey2022TidyNameSizeCyber$policy12[i] == 1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2022TidyNameSizeCyber$policyData[i] = 1

    } else if (dataCyberSecuritySurvey2022TidyNameSizeCyber$policy11[i] ==
        -10005 && dataCyberSecuritySurvey2022TidyNameSizeCyber$policy12[i] ==
        -10005) {

        dataCyberSecuritySurvey2022TidyNameSizeCyber$policyData[i] = NA

    }

    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$policy1[i] == 1 || dataCyberSecuritySurve
        1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2022TidyNameSizeCyber$policyPrivate[i] = 1

    } else if (dataCyberSecuritySurvey2022TidyNameSizeCyber$policy1[i] == -10005 &&
        dataCyberSecuritySurvey2022TidyNameSizeCyber$policy4[i] == -10005) {

        dataCyberSecuritySurvey2022TidyNameSizeCyber$policyPrivate[i] = NA

    }

    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$policy8[i] == 1) {
        ## NA
```

```
        dataCyberSecuritySurvey2022TidyNameSizeCyber$policyPrivate[i] = NA
        dataCyberSecuritySurvey2022TidyNameSizeCyber$policyData[i] = NA
        dataCyberSecuritySurvey2022TidyNameSizeCyber$policyStaffAccess[i] = NA

    }

}


## another day of garbage collection of unused columns

dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(policy1:policy9))
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-(policy1:policy9))
dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
    select(-(policy1:policy9))
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-(policy1:policy9))
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(policy1:policy12))
```

```
## taking care of the columns that are only in the 2018 survey

dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(doc1:doc6))

## removing the question about if they know about the 10 steps for
## cyber security Spoiler alert: knowing about it doesnt mean you apply
## it and you can learn about it from other sources either way
## https://www.ncsc.gov.uk/collection/10-steps/risk-management

## same for the next question about the cyber essential scheme

## nevermind they removed all the rest of the questions until business
## standard on 2019 and 2020 (Q43)

dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(tensteps:implemb))
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-(tensteps:implemb))

## TODO: think if I should keep the review of cyber security
## documentation colum removing it since i don't have it on 2018 and I
## don't think I will be using it
```

```r
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-review)
dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
    select(-review)
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-review)
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-review)

## in 2022 they asked some proper questions about cyber security
## training and cyber security strategy that will be removed for lack
## of comparrison with the other years

dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(trained:corprisk))


## Removing the question about cyber security conserns in the suppliers
## because 1 - most institutions evaluated here won't be in a scale
## where that is an important question 2 - if you were a big
## institution you would just have taken of most of the inside managed
## and now would worry about the suppliers on that level you will just
## get multiple suppliers in case your main supplier fails removing the
## SPOF(single point of failure) that way like what would they do
## anyway such a poorly written question, just hire me to write next
## year survey instead

dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-supply)

dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-supply)

dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
    select(-(supplyrisk1:supplyrisk2))

dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-(supplyrisk1:supplyrisk2))

dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(supplyrisk1:supplyrisk2))

## questions about supplier standards were deleted after 2019

dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(adhere1:cloud))
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
```

```
    select(-(adhere1:cloud))

## only asked in 2022 so not relevant for comparrison

dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(barrier1:barrier8))
```

##Type of attacks

```
##type of attacks that targetted the institution
##type 01 - ramsomware infection
##type 02 - spyware, malware or other type of infection
##type 03 - DDOS (distributed denial of service)
##type 04 - hacking online bank accounts
##type 05 - phising - impersonating your organisation
##type 06 - phising - fraudulent emails or website targetting staff
##type 07 - unauthorized access by internal staff members
##type 08 - unauthorized access by outsiders
##type 09 - other type of cyber attacks
##type 10 - don't know
##type 11 - (don't care) none of these
##type 12 - refused to answer
##type 13 - unauthorized access by students (to be merged with type 7) // only used starting f
## type 15 and 16 were only collect in 2022 so not to be compared and type 14 just does not ex

##time to some non-git merges on the variables

##attackInfection - ramsomware infection (type 1), spyware, malware and other types (type 2),
##attackPhising - hacking online bank accounts (type 4), phising - impersonating organisation
##attackBreaching - unauthorized access internal staff (type 7), unauthorized access outsiders
##remember that type 10 and 12 is missing data


##################################################### 2018 ###################################

numberOfCycles = length(dataCyberSecuritySurvey2018TidyNameSizeCyber$type1)

dataCyberSecuritySurvey2018TidyNameSizeCyber$attackInfection = 0
dataCyberSecuritySurvey2018TidyNameSizeCyber$attackPhising = 0
dataCyberSecuritySurvey2018TidyNameSizeCyber$attackBreaching = 0

for (i in 1:numberOfCycles) {


    if(dataCyberSecuritySurvey2018TidyNameSizeCyber$type1[i] == 1 || dataCyberSecuritySurvey20

      dataCyberSecuritySurvey2018TidyNameSizeCyber$attackInfection[i] = 1
    }

      if(dataCyberSecuritySurvey2018TidyNameSizeCyber$type4[i] == 1 || dataCyberSecuritySurvey

      dataCyberSecuritySurvey2018TidyNameSizeCyber$attackPhising[i] = 1

      }

      if(dataCyberSecuritySurvey2018TidyNameSizeCyber$type7[i] == 1 || dataCyberSecuritySurvey

      dataCyberSecuritySurvey2018TidyNameSizeCyber$attackBreaching[i] = 1

      }


      if(
        ##dataCyberSecuritySurvey2018TidyNameSizeCyber$type1[i] == 9 ||
        dataCyberSecuritySurvey2018TidyNameSizeCyber$type10[i] == 1 || dataCyberSecuritySurvey

      dataCyberSecuritySurvey2018TidyNameSizeCyber$attackPhising[i] = NA
      dataCyberSecuritySurvey2018TidyNameSizeCyber$attackBreaching[i] = NA
      dataCyberSecuritySurvey2018TidyNameSizeCyber$attackInfection[i] = NA

      }

      ## basically if we only know they got other type of attacks (policy 9 then we know the t


}


##################################################### 2019 ####################################


numberOfCycles = length(dataCyberSecuritySurvey2019TidyNameSizeCyber$type1)

dataCyberSecuritySurvey2019TidyNameSizeCyber$attackInfection = 0
dataCyberSecuritySurvey2019TidyNameSizeCyber$attackPhising = 0
dataCyberSecuritySurvey2019TidyNameSizeCyber$attackBreaching = 0

for (i in 1:numberOfCycles) {


    if(dataCyberSecuritySurvey2019TidyNameSizeCyber$type1[i] == 1 || dataCyberSecuritySurvey20

      dataCyberSecuritySurvey2019TidyNameSizeCyber$attackInfection[i] = 1
    }

      if(dataCyberSecuritySurvey2019TidyNameSizeCyber$type4[i] == 1 || dataCyberSecuritySurvey

      dataCyberSecuritySurvey2019TidyNameSizeCyber$attackPhising[i] = 1

      }

      if(dataCyberSecuritySurvey2019TidyNameSizeCyber$type7[i] == 1 || dataCyberSecuritySurvey

      dataCyberSecuritySurvey2019TidyNameSizeCyber$attackBreaching[i] = 1

      }


      if(
        ##dataCyberSecuritySurvey2019TidyNameSizeCyber$type1[i] == 9 ||
        dataCyberSecuritySurvey2019TidyNameSizeCyber$type10[i] == 1 || dataCyberSecuritySurvey

      dataCyberSecuritySurvey2019TidyNameSizeCyber$attackPhising[i] = NA
      dataCyberSecuritySurvey2019TidyNameSizeCyber$attackBreaching[i] = NA
      dataCyberSecuritySurvey2019TidyNameSizeCyber$attackInfection[i] = NA

      }

      ## basically if we only know they got other type of attacks (policy 9 then we know the t


}


###################################################### 2020 ####################################


numberOfCycles = length(dataCyberSecuritySurvey2020TidyNameSizeCyber$type1)

dataCyberSecuritySurvey2020TidyNameSizeCyber$attackInfection = 0
dataCyberSecuritySurvey2020TidyNameSizeCyber$attackPhising = 0
dataCyberSecuritySurvey2020TidyNameSizeCyber$attackBreaching = 0

for (i in 1:numberOfCycles) {


    if(dataCyberSecuritySurvey2020TidyNameSizeCyber$type1[i] == 1 || dataCyberSecuritySurvey20

      dataCyberSecuritySurvey2020TidyNameSizeCyber$attackInfection[i] = 1
    }

      if(dataCyberSecuritySurvey2020TidyNameSizeCyber$type4[i] == 1 || dataCyberSecuritySurvey

      dataCyberSecuritySurvey2020TidyNameSizeCyber$attackPhising[i] = 1

      }

      if(dataCyberSecuritySurvey2020TidyNameSizeCyber$type7[i] == 1 || dataCyberSecuritySurvey

      dataCyberSecuritySurvey2020TidyNameSizeCyber$attackBreaching[i] = 1

      }


      if(
        ##dataCyberSecuritySurvey2020TidyNameSizeCyber$type1[i] == 9 ||
        dataCyberSecuritySurvey2020TidyNameSizeCyber$type10[i] == 1 || dataCyberSecuritySurvey

      dataCyberSecuritySurvey2020TidyNameSizeCyber$attackPhising[i] = NA
      dataCyberSecuritySurvey2020TidyNameSizeCyber$attackBreaching[i] = NA
      dataCyberSecuritySurvey2020TidyNameSizeCyber$attackInfection[i] = NA

      }

      ## basically if we only know they got other type of attacks (policy 9 then we know the t


}


##################################################### 2021 ###################################


numberOfCycles = length(dataCyberSecuritySurvey2021TidyNameSizeCyber$type1)

dataCyberSecuritySurvey2021TidyNameSizeCyber$attackInfection = 0
dataCyberSecuritySurvey2021TidyNameSizeCyber$attackPhising = 0
dataCyberSecuritySurvey2021TidyNameSizeCyber$attackBreaching = 0

for (i in 1:numberOfCycles) {


    if(dataCyberSecuritySurvey2021TidyNameSizeCyber$type1[i] == 1 || dataCyberSecuritySurvey20

      dataCyberSecuritySurvey2021TidyNameSizeCyber$attackInfection[i] = 1
```

```
  }

   if(dataCyberSecuritySurvey2021TidyNameSizeCyber$type4[i] == 1 || dataCyberSecuritySurvey

   dataCyberSecuritySurvey2021TidyNameSizeCyber$attackPhising[i] = 1

   }

   if(dataCyberSecuritySurvey2021TidyNameSizeCyber$type7[i] == 1 || dataCyberSecuritySurvey

   dataCyberSecuritySurvey2021TidyNameSizeCyber$attackBreaching[i] = 1

   }


   if(
    ##dataCyberSecuritySurvey2020TidyNameSizeCyber$type1[i] == 9 ||
    dataCyberSecuritySurvey2021TidyNameSizeCyber$type10[i] == 1 || dataCyberSecuritySurvey

   dataCyberSecuritySurvey2021TidyNameSizeCyber$attackPhising[i] = NA
   dataCyberSecuritySurvey2021TidyNameSizeCyber$attackBreaching[i] = NA
   dataCyberSecuritySurvey2021TidyNameSizeCyber$attackInfection[i] = NA

   }

   ## basically if we only know they got other type of attacks (policy 9 then we know the t


}


##################################################### 2022 ###################################

numberOfCycles = length(dataCyberSecuritySurvey2022TidyNameSizeCyber$type1)

dataCyberSecuritySurvey2022TidyNameSizeCyber$attackInfection = 0
dataCyberSecuritySurvey2022TidyNameSizeCyber$attackPhising = 0
dataCyberSecuritySurvey2022TidyNameSizeCyber$attackBreaching = 0

for (i in 1:numberOfCycles) {


  if(dataCyberSecuritySurvey2022TidyNameSizeCyber$type1[i] == 1 || dataCyberSecuritySurvey20

   dataCyberSecuritySurvey2022TidyNameSizeCyber$attackInfection[i] = 1

  }
```

```
      if(dataCyberSecuritySurvey2022TidyNameSizeCyber$type4[i] == 1 || dataCyberSecuritySurvey

      dataCyberSecuritySurvey2022TidyNameSizeCyber$attackPhising[i] = 1

      }

      if(dataCyberSecuritySurvey2022TidyNameSizeCyber$type7[i] == 1 || dataCyberSecuritySurvey

      dataCyberSecuritySurvey2022TidyNameSizeCyber$attackBreaching[i] = 1

      }


      if(
        ##dataCyberSecuritySurvey2020TidyNameSizeCyber$type1[i] == 9 ||
        dataCyberSecuritySurvey2022TidyNameSizeCyber$type10[i] == 1 || dataCyberSecuritySurvey

      dataCyberSecuritySurvey2022TidyNameSizeCyber$attackPhising[i] = NA
      dataCyberSecuritySurvey2022TidyNameSizeCyber$attackBreaching[i] = NA
      dataCyberSecuritySurvey2022TidyNameSizeCyber$attackInfection[i] = NA

      }

      ## basically if we only know they got other type of attacks (policy 9 then we know the t


}


dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>


## i think I will also remove the frequency of the attack since I won't be using them for anyt

dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>

## now to register both outcome and impact
```

0.9 Outcomes - we check for each of the attack outcoems and group them

```
## for the frequency of attacks in the last 12 months I am not sure if
## I am interested in that data TODO I am temporary removing them if I
## can add thhem back if needed (more interested in the outcome of the
## attacks)

## outcomes from the attacks outcome 01 - Software or systems were
## corrupted or damaged (permanent DDOS) outcome 02 - Personal data was
## altered, destroyed or taken outcome 03 - Permanent loss of files
## (other than personal data) outcome 04 - Temporary loss of access to
## files or networks outcome 05 - Lost or stolen assets, trade secrets
## or intellectual property outcome 06 - Money was stolen outcome 07 -
## (DDOS) website or online services were taken down or made slower
## outcome 08 - Lost access to any third-party services you rely on
## outcome 09 - Don't know (NA) outcome 10 - none of these outcome
## 11,12 and 13 are only present in 2022 so we won't use them to make
## comparisons


## as the lord and savior Dr Mark Kelson has preached to me on you
## shall merge the multiple outcomes into smaller ones to have better
## correlations If you are reading this one mark don't forget my extra
## points for that one time I had to turn on the system in the computer
## lab because it was turned off the the professors assistants weren't
## there yet to comment it all its ctr+shift+c #FicaADica

## outcomesData - is a combination of personal data was altered
## (outcome2), temporary or permanent loss of access to files (outcome
## 3 and 4) outcomesDDOS - websites or online service was taken down or
## made slower (outcome 7), lost access to any third party services
## (outcome 8) and software or system corruption and damaged (permanent
## DDOS) (outcome1) outcomesTheft - lost or stolen assets, trade
## secrets or intellectual property (outcome 5) and stolen money
## (outcome 6)




##################################################### 2018
##################################################### ########################################


numberOfCycles = length(dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome1)

dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesData = 0
dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesDDOS = 0
```

```
dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesTheft = 0



for (i in 1:numberOfCycles) {

  dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome1[i] = replace_na(dataCyberSecuritySur
    -10007)
  dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome2[i] = replace_na(dataCyberSecuritySur
    -10007)
  dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome3[i] = replace_na(dataCyberSecuritySur
    -10007)
  dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome4[i] = replace_na(dataCyberSecuritySur
    -10007)
  dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome5[i] = replace_na(dataCyberSecuritySur
    -10007)
  dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome6[i] = replace_na(dataCyberSecuritySur
    -10007)
  dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome7[i] = replace_na(dataCyberSecuritySur
    -10007)
  dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome8[i] = replace_na(dataCyberSecuritySur
    -10007)
  dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome10[i] = replace_na(dataCyberSecuritySu
    -10007)
  dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome9[i] = replace_na(dataCyberSecuritySur
    -10007)

  ## my probably pathetic attempt to optimize my loops to not get a
  ## 10 minute compilation #my toast runs slower than a toaster, and
  ## toasters don't even have legs to run

  if (dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome1[i] == 1 ||
    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome7[i] == 1 ||
    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome8[i] == 1) {
    ## either 1,7,8

    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesDDOS[i] = 1

  } else if (dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome1[i] ==
    -10007 && dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome7[i] ==
    -10007 && dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome8[i] ==
    -10007) {

    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesDDOS[i] = NA
  }

  if (dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome2[i] == 1 ||
    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome3[i] == 1 ||
    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome4[i] == 1) {
    ## either 2,3,4

    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesData[i] = 1

  } else if (dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome2[i] ==
    -10007 && dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome3[i] ==
    -10007 && dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome4[i] ==
    -10007) {

    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesData[i] = NA
  }

  if (dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome5[i] == 1 ||
    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome6[i] == 1) {
    ## either 5 or 6

    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesTheft[i] = 1

  } else if (dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome5[i] ==
    -10007 && dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome6[i] ==
    -10007) {

    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesTheft[i] = NA
  }

  if (dataCyberSecuritySurvey2018TidyNameSizeCyber$outcome9[i] == 1) {

    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesTheft[i] = NA
    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesData[i] = NA
    dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesDDOS[i] = NA
  }

}




##################################################### 2019
##################################################### ########################################


numberOfCycles = length(dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome1)
```

```
dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesData = 0
dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesDDOS = 0
dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesTheft = 0

for (i in 1:numberOfCycles) {

    dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome1[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome2[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome3[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome4[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome5[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome6[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome7[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome8[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome9[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome10[i] = replace_na(dataCyberSecuritySu
        -10007)

    ## my probably pathetic attempt to optimize my loops to not get a
    ## 10 minute compilation #my toast runs slower than a toaster, and
    ## toasters don't even have legs to run

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome1[i] == 1 ||
        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome7[i] == 1 ||
        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome8[i] == 1) {
        ## either 1,7,8

        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesDDOS[i] = 1

    } else if (dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome1[i] ==
        -10007 && dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome7[i] ==
        -10007 && dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome8[i] ==
        -10007) {

        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesDDOS[i] = NA
    }

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome2[i] == 1 ||
        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome3[i] == 1 ||
        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome4[i] == 1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesData[i] = 1

    } else if (dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome2[i] ==
        -10007 && dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome3[i] ==
        -10007 && dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome4[i] ==
        -10007) {

        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesData[i] = NA
    }

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome5[i] == 1 ||
        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome6[i] == 1) {
        ## either 5 or 6

        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesTheft[i] = 1

    } else if (dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome5[i] ==
        -10007 && dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome6[i] ==
        -10007) {

        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesTheft[i] = NA
    }

    if (dataCyberSecuritySurvey2019TidyNameSizeCyber$outcome9[i] == 1) {

        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesTheft[i] = NA
        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesData[i] = NA
        dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesDDOS[i] = NA
    }


}

##################################################### 2020
##################################################### ########################################

numberOfCycles = length(dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome1)

dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesData = 0
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesDDOS = 0
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesTheft = 0

for (i in 1:numberOfCycles) {
```

```
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome1[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome2[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome3[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome4[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome5[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome6[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome7[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome8[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome9[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome10[i] = replace_na(dataCyberSecuritySu
    -10007)

## my probably pathetic attempt to optimize my loops to not get a
## 10 minute compilation #my toast runs slower than a toaster, and
## toasters don't even have legs to run

if (dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome1[i] == 1 ||
    dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome7[i] == 1 ||
    dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome8[i] == 1) {
    ## either 1,7,8

    dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesDDOS[i] = 1

} else if (dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome1[i] ==
    -10007 && dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome7[i] ==
    -10007 && dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome8[i] ==
    -10007) {

    dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesDDOS[i] = NA
}

if (dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome2[i] == 1 ||
    dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome3[i] == 1 ||
    dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome4[i] == 1) {
    ## either 2,3,4

    dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesData[i] = 1
    } else if (dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome2[i] ==
        -10007 && dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome3[i] ==
        -10007 && dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome4[i] ==
        -10007) {

        dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesData[i] = NA
    }

    if (dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome5[i] == 1 ||
        dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome6[i] == 1) {
        ## either 5 or 6

        dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesTheft[i] = 1

    } else if (dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome5[i] ==
        -10007 && dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome6[i] ==
        -10007) {

        dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesTheft[i] = NA
    }
    if (dataCyberSecuritySurvey2020TidyNameSizeCyber$outcome9[i] == 1) {

        dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesTheft[i] = NA
        dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesData[i] = NA
        dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesDDOS[i] = NA
    }

}


##################################################### 2021
##################################################### ########################################


numberOfCycles = length(dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome1)

dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesData = 0
dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesDDOS = 0
dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesTheft = 0

for (i in 1:numberOfCycles) {

    dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome1[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome2[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome3[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome4[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome5[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome6[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome7[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome8[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome9[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome10[i] = replace_na(dataCyberSecuritySu
        -10007)

    ## my probably pathetic attempt to optimize my loops to not get a
    ## 10 minute compilation #my toast runs slower than a toaster, and
    ## toasters don't even have legs to run

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome1[i] == 1 ||
        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome7[i] == 1 ||
        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome8[i] == 1) {
        ## either 1,7,8

        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesDDOS[i] = 1

    } else if (dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome1[i] ==
        -10007 && dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome7[i] ==
        -10007 && dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome8[i] ==
        -10007) {

        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesDDOS[i] = NA
    }

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome2[i] == 1 ||
        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome3[i] == 1 ||
        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome4[i] == 1) {
        ## either 2,3,4

        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesData[i] = 1

    } else if (dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome2[i] ==
        -10007 && dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome3[i] ==
        -10007 && dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome4[i] ==
        -10007) {
        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesData[i] = NA
    }

    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome5[i] == 1 ||
        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome6[i] == 1) {
        ## either 5 or 6

        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesTheft[i] = 1

    } else if (dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome5[i] ==
        -10007 && dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome6[i] ==
        -10007) {

        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesTheft[i] = NA
    }
    if (dataCyberSecuritySurvey2021TidyNameSizeCyber$outcome9[i] == 1) {

        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesTheft[i] = NA
        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesData[i] = NA
        dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesDDOS[i] = NA
    }

}



#################################################### 2022
#################################################### #######################################



numberOfCycles = length(dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome1)

dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesData = 0
dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesDDOS = 0
dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesTheft = 0

for (i in 1:numberOfCycles) {

    dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome1[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome2[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome3[i] = replace_na(dataCyberSecuritySur
        -10007)
    dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome4[i] = replace_na(dataCyberSecuritySur
        -10007)
```

```
dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome5[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome6[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome7[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome8[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome9[i] = replace_na(dataCyberSecuritySur
    -10007)
dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome10[i] = replace_na(dataCyberSecuritySu
    -10007)

## my probably pathetic attempt to optimize my loops to not get a
## 10 minute compilation #my toast runs slower than a toaster, and
## toasters don't even have legs to run if anyone asks yes I
## started from bottom to top but I can and I love see my pc dying
## compiling my poorly optimized code

if (dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome1[i] == 1 ||
    dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome7[i] == 1 ||
    dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome8[i] == 1) {
    ## either 1,7,8

    dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesDDOS[i] = 1

} else if (dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome1[i] ==
    -10007 && dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome7[i] ==
    -10007 && dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome8[i] ==
    -10007) {

    dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesDDOS[i] = NA
}

if (dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome2[i] == 1 ||
    dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome3[i] == 1 ||
    dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome4[i] == 1) {
    ## either 2,3,4

    dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesData[i] = 1

} else if (dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome2[i] ==
    -10007 && dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome3[i] ==
    -10007 && dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome4[i] ==
    -10007) {

    dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesData[i] = NA
    }

    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome5[i] == 1 ||
        dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome6[i] == 1) {
        ## either 5 or 6

        dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesTheft[i] = 1

    } else if (dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome5[i] ==
        -10007 && dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome6[i] ==
        -10007) {

        dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesTheft[i] = NA
    }
    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$outcome9[i] == 1) {

        dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesTheft[i] = NA
        dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesData[i] = NA
        dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesDDOS[i] = NA
    }

}




dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(outcome1:outcome10))
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-(outcome1:outcome10))
dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
    select(-(outcome1:outcome10))
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-(outcome1:outcome10))
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(outcome1:outcome13))
```

```
## we have now finished checking the outcome of these attacks as the
## last relevant parameter we will analyse so we will now cleanse the
## dataset of all the other unnecessary data

## the costs have too much in different and missing data, it is better
## not to use them, no way josé


## removing the costs here as well, not opening another Pandora box so
## close to the deadline
```

```
dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(impact1:weight))
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-(impact1:weight))
dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
    select(-(impact1:weight))
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-(impact1:weight))


dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(impact1:Sum10Steps))


dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-(halfa:weight))


## we also won't be looking at which country inside the uk the
## charities belong

dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-country)
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-country)
dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
    select(-country)
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-country)
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-country)

## we also won't be using the charity income data


dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-income)
dataCyberSecuritySurvey2019TidyNameSizeCyber = dataCyberSecuritySurvey2019TidyNameSizeCyber %>
    select(-one_of("income", "income2"))
dataCyberSecuritySurvey2020TidyNameSizeCyber = dataCyberSecuritySurvey2020TidyNameSizeCyber %>
    select(-one_of("income", "income2"))
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-one_of("income", "income2"))
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-one_of("income", "income2"))

## cleaning 2018 specific
```

```
dataCyberSecuritySurvey2018TidyNameSizeCyber = dataCyberSecuritySurvey2018TidyNameSizeCyber %>
    select(-(charityo:core))

## removing last year specific
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-title)

## hold up why do I bother having typex when any of the others years
## don't dataCyberSecuritySurvey2022TidyNameSizeCyber$typex =
## as.factor(dataCyberSecuritySurvey2022TidyNameSizeCyber$typex)
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-typex)

## recorded time of death 22:53 9/11/2022, cause : realized I had the
## 2021 variable pointing to the 2020 file this entire time this is
## what we call being dumber than a door #estudasses

dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-(title1:title19))
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-(online12:online13))
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-(scheme6:outcome13))
dataCyberSecuritySurvey2021TidyNameSizeCyber = dataCyberSecuritySurvey2021TidyNameSizeCyber %>
    select(-covpri)
```

0.10 Labelling Conversion

```
## Now we have all the data wrangled ready we will preparing for it to
## be imputated


summary(dataCyberSecuritySurvey2018TidyNameSizeCyber)
```

```
     imid         instituitionTypes     sizea            sizeb      
Min.   :100018   Min.   :1.000     Min.   :    2.0   Min.   :1.000  
1st Qu.:122907   1st Qu.:1.000     1st Qu.:    5.0   1st Qu.:1.000  
Median :146281   Median :1.000     Median :   16.0   Median :2.000  
Mean   :154715   Mean   :1.273     Mean   :  292.3   Mean   :2.109  
3rd Qu.:153526   3rd Qu.:2.000     3rd Qu.:   80.0   3rd Qu.:3.000  
Max.   :260174   Max.   :2.000     Max.   :69035.0   Max.   :4.000  
                                   NA's   :58        NA's   :3      
   priority         update          year           managementContinuity
Min.   :1.000   Min.   :1.00   Length:2088        Min.   :0.0000      
1st Qu.:1.000   1st Qu.:3.00   Class :character   1st Qu.:0.0000      
```

```
Median :2.000   Median :4.00   Mode  :character   Median :1.0000  
Mean   :1.894   Mean   :4.18                      Mean   :0.6112  
3rd Qu.:2.000   3rd Qu.:5.00                      3rd Qu.:1.0000  
Max.   :4.000   Max.   :9.00                      Max.   :1.0000  
NA's   :26                                        NA's   :20      
managementCyber rulesUpdating    rulesSecurityConfigs rulesUserControl
Min.   :0.000   Min.   :0.0000   Min.   :0.0000       Min.   :0.0000  
1st Qu.:0.000   1st Qu.:1.0000   1st Qu.:1.0000       1st Qu.:1.0000  
Median :1.000   Median :1.0000   Median :1.0000       Median :1.0000  
Mean   :0.648   Mean   :0.9597   Mean   :0.9448       Mean   :0.8579  
3rd Qu.:1.000   3rd Qu.:1.0000   3rd Qu.:1.0000       3rd Qu.:1.0000  
Max.   :1.000   Max.   :1.0000   Max.   :1.0000       Max.   :1.0000  
NA's   :20      NA's   :5        NA's   :5            NA's   :5       
policyStaffAccess   policyData     policyPrivate    attackInfection 
Min.   :0.0000    Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
1st Qu.:1.0000    1st Qu.:0.0000   1st Qu.:1.0000   1st Qu.:0.0000  
Median :1.0000    Median :1.0000   Median :1.0000   Median :0.0000  
Mean   :0.9543    Mean   :0.7437   Mean   :0.8436   Mean   :0.2184  
3rd Qu.:1.0000    3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:0.0000  
Max.   :1.0000    Max.   :1.0000   Max.   :1.0000   Max.   :1.0000  
NA's   :1257      NA's   :1257     NA's   :1257     NA's   :23      
attackPhising    attackBreaching   outcomesData     outcomesDDOS   
Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000  
Median :0.0000   Median :0.0000   Median :0.0000   Median :0.0000  
Mean   :0.4203   Mean   :0.0954   Mean   :0.2675   Mean   :0.2624  
3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:1.0000  
Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000  
NA's   :23       NA's   :23       NA's   :1101     NA's   :1101    
outcomesTheft   
Min.   :0.0000  
1st Qu.:0.0000  
Median :0.0000  
Mean   :0.0648  
3rd Qu.:0.0000  
Max.   :1.0000  
NA's   :1101    
```

```
summary(dataCyberSecuritySurvey2019TidyNameSizeCyber)
```

```
     imid        instituitionTypes     sizea             sizeb     
Min.   :100008   Min.   :1.000     Min.   :    2.0   Min.   :1.00  
1st Qu.:105163   1st Qu.:1.000     1st Qu.:    5.0   1st Qu.:1.00  
Median :110652   Median :1.000     Median :   14.0   Median :2.00  
Mean   :152875   Mean   :1.224     Mean   :  193.4   Mean   :2.03  
3rd Qu.:115574   3rd Qu.:1.000     3rd Qu.:   72.0   3rd Qu.:3.00  
Max.   :401174   Max.   :2.000     Max.   :34000.0   Max.   :4.00  
```

```
                                   NA's   :68        NA's   :4
   priority         update          year           managementContinuity
Min.   :1.000   Min.   :1.000   Length:2080        Min.   :0.0000
1st Qu.:1.000   1st Qu.:3.000   Class :character   1st Qu.:0.0000
Median :2.000   Median :4.000   Mode  :character   Median :1.0000
Mean   :1.699   Mean   :4.254                      Mean   :0.6495
3rd Qu.:2.000   3rd Qu.:5.000                      3rd Qu.:1.0000
Max.   :4.000   Max.   :8.000                      Max.   :1.0000
NA's   :32      NA's   :110                        NA's   :23
managementCyber  rulesUpdating    rulesSecurityConfigs rulesUserControl
Min.   :0.0000   Min.   :0.0000   Min.   :0.0000       Min.   :0.0000
1st Qu.:0.0000   1st Qu.:1.0000   1st Qu.:1.0000       1st Qu.:1.0000
Median :1.0000   Median :1.0000   Median :1.0000       Median :1.0000
Mean   :0.6918   Mean   :0.9701   Mean   :0.9591       Mean   :0.8729
3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000       3rd Qu.:1.0000
Max.   :1.0000   Max.   :1.0000   Max.   :1.0000       Max.   :1.0000
NA's   :23       NA's   :3        NA's   :3            NA's   :3
policyStaffAccess   policyData     policyPrivate    attackInfection
Min.   :0.0000    Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
1st Qu.:1.0000    1st Qu.:1.0000   1st Qu.:1.0000   1st Qu.:0.0000
Median :1.0000    Median :1.0000   Median :1.0000   Median :0.0000
Mean   :0.9602    Mean   :0.8133   Mean   :0.8663   Mean   :0.1516
3rd Qu.:1.0000    3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:0.0000
Max.   :1.0000    Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
NA's   :1100      NA's   :1100     NA's   :1100     NA's   :28
attackPhising   attackBreaching   outcomesData     outcomesDDOS
Min.   :0.000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
1st Qu.:0.000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000
Median :0.000   Median :0.0000   Median :0.0000   Median :0.0000
Mean   :0.366   Mean   :0.0653   Mean   :0.2037   Mean   :0.2195
3rd Qu.:1.000   3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.0000
Max.   :1.000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
NA's   :28      NA's   :28       NA's   :1260     NA's   :1260
outcomesTheft
Min.   :0.0000
1st Qu.:0.0000
Median :0.0000
Mean   :0.0646
3rd Qu.:0.0000
Max.   :1.0000
NA's   :1260
```

```
summary(dataCyberSecuritySurvey2020TidyNameSizeCyber)
```

```
     imid        instituitionTypes     sizea              sizeb
Min.   :100059   Min.   :1.00      Min.   :     2.0   Min.   :1.000
1st Qu.:135624   1st Qu.:1.00      1st Qu.:     5.0   1st Qu.:1.000
```

```
Median :169565   Median :1.00      Median :    16.0   Median :2.000  
Mean   :212340   Mean   :1.39      Mean   :   577.8   Mean   :2.131  
3rd Qu.:231585   3rd Qu.:2.00      3rd Qu.:    84.0   3rd Qu.:3.000  
Max.   :600019   Max.   :3.00      Max.   :300000.0   Max.   :4.000  
                                   NA's   :79         NA's   :4      
   priority         update          year           managementContinuity
Min.   :1.000   Min.   :1.000   Length:1900        Min.   :0.0000      
1st Qu.:1.000   1st Qu.:3.000   Class :character   1st Qu.:0.0000      
Median :1.000   Median :4.000   Mode  :character   Median :1.0000      
Mean   :1.626   Mean   :4.275                      Mean   :0.7164      
3rd Qu.:2.000   3rd Qu.:5.000                      3rd Qu.:1.0000      
Max.   :4.000   Max.   :8.000                      Max.   :1.0000      
NA's   :38      NA's   :193                        NA's   :24          
managementCyber  rulesUpdating    rulesSecurityConfigs rulesUserControl
Min.   :0.0000   Min.   :0.0000   Min.   :0.000        Min.   :0.0000  
1st Qu.:0.0000   1st Qu.:1.0000   1st Qu.:1.000        1st Qu.:1.0000  
Median :1.0000   Median :1.0000   Median :1.000        Median :1.0000  
Mean   :0.7111   Mean   :0.9746   Mean   :0.973        Mean   :0.8864  
3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.000        3rd Qu.:1.0000  
Max.   :1.0000   Max.   :1.0000   Max.   :1.000        Max.   :1.0000  
NA's   :24       NA's   :8        NA's   :8            NA's   :8       
policyStaffAccess   policyData     policyPrivate    attackInfection 
Min.   :0.0000    Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
1st Qu.:1.0000    1st Qu.:1.0000   1st Qu.:1.0000   1st Qu.:0.0000  
Median :1.0000    Median :1.0000   Median :1.0000   Median :0.0000  
Mean   :0.9536    Mean   :0.8087   Mean   :0.8617   Mean   :0.1337  
3rd Qu.:1.0000    3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:0.0000  
Max.   :1.0000    Max.   :1.0000   Max.   :1.0000   Max.   :1.0000  
NA's   :823       NA's   :823      NA's   :823      NA's   :37      
attackPhising    attackBreaching    outcomesData     outcomesDDOS  
Min.   :0.0000   Min.   :0.00000   Min.   :0.000   Min.   :0.000  
1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.000   1st Qu.:0.000  
Median :1.0000   Median :0.00000   Median :0.000   Median :0.000  
Mean   :0.5153   Mean   :0.06817   Mean   :0.168   Mean   :0.112  
3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:0.000   3rd Qu.:0.000  
Max.   :1.0000   Max.   :1.00000   Max.   :1.000   Max.   :1.000  
NA's   :37       NA's   :37        NA's   :900     NA's   :900    
outcomesTheft  
Min.   :0.000  
1st Qu.:0.000  
Median :0.000  
Mean   :0.062  
3rd Qu.:0.000  
Max.   :1.000  
NA's   :900    
```

```
summary(dataCyberSecuritySurvey2021TidyNameSizeCyber)
```

```
    imid           instituitionTypes     sizea             sizeb
Length:2284        Min.   :1.000      Min.   :    0.0    Min.   :1.000
Class :character   1st Qu.:1.000      1st Qu.:    5.0    1st Qu.:1.000
Mode  :character   Median :1.000      Median :   17.0    Median :2.000
                   Mean   :1.532      Mean   :  222.7    Mean   :2.143
                   3rd Qu.:2.000      3rd Qu.:  100.0    3rd Qu.:3.000
                   Max.   :3.000      Max.   :30000.0    Max.   :4.000
                                      NA's   :83         NA's   :8
   priority         update          year           managementContinuity
Min.   :1.000   Min.   :1.000   Length:2284        Min.   :0.0000
1st Qu.:1.000   1st Qu.:3.000   Class :character   1st Qu.:0.0000
Median :2.000   Median :4.000   Mode  :character   Median :1.0000
Mean   :1.687   Mean   :4.084                      Mean   :0.6433
3rd Qu.:2.000   3rd Qu.:5.000                      3rd Qu.:1.0000
Max.   :4.000   Max.   :8.000                      Max.   :1.0000
NA's   :28      NA's   :136                        NA's   :55
managementCyber  rulesUpdating    rulesSecurityConfigs rulesUserControl
Min.   :0.0000   Min.   :0.0000   Min.   :0.0000       Min.   :0.0000
1st Qu.:0.0000   1st Qu.:1.0000   1st Qu.:1.0000       1st Qu.:1.0000
Median :1.0000   Median :1.0000   Median :1.0000       Median :1.0000
Mean   :0.7048   Mean   :0.9455   Mean   :0.9455       Mean   :0.8524
3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000       3rd Qu.:1.0000
Max.   :1.0000   Max.   :1.0000   Max.   :1.0000       Max.   :1.0000
NA's   :55       NA's   :8        NA's   :8            NA's   :8
policyStaffAccess   policyData   policyPrivate   attackInfection
Min.   :0.0000    Min.   :0      Min.   :0.000   Min.   :0.0000
1st Qu.:1.0000    1st Qu.:0      1st Qu.:1.000   1st Qu.:0.0000
Median :1.0000    Median :0      Median :1.000   Median :0.0000
Mean   :0.9521    Mean   :0      Mean   :0.863   Mean   :0.1058
3rd Qu.:1.0000    3rd Qu.:0      3rd Qu.:1.000   3rd Qu.:0.0000
Max.   :1.0000    Max.   :0      Max.   :1.000   Max.   :1.0000
NA's   :1072      NA's   :1072   NA's   :1072    NA's   :34
attackPhising    attackBreaching    outcomesData     outcomesDDOS
Min.   :0.0000   Min.   :0.00000   Min.   :0.0000   Min.   :0.0000
1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.0000
Median :0.0000   Median :0.00000   Median :0.0000   Median :0.0000
Mean   :0.4324   Mean   :0.04267   Mean   :0.1097   Mean   :0.1213
3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:0.0000   3rd Qu.:0.0000
Max.   :1.0000   Max.   :1.00000   Max.   :1.0000   Max.   :1.0000
NA's   :34       NA's   :34        NA's   :1245     NA's   :1245
outcomesTheft
Min.   :0.0000
1st Qu.:0.0000
Median :0.0000
Mean   :0.0549
3rd Qu.:0.0000
Max.   :1.0000
NA's   :1245
```

```r
summary(dataCyberSecuritySurvey2022TidyNameSizeCyber)
```

```
    imid           instituitionTypes     sizea              sizeb      
Length:2157        Min.   :1.000     Min.   :     2.0   Min.   :1.000  
Class :character   1st Qu.:1.000     1st Qu.:     5.0   1st Qu.:1.000  
Mode  :character   Median :1.000     Median :    19.0   Median :2.000  
                   Mean   :1.641     Mean   :   688.2   Mean   :2.142  
                   3rd Qu.:2.000     3rd Qu.:   100.0   3rd Qu.:3.000  
                   Max.   :3.000     Max.   :450000.0   Max.   :4.000  
                                     NA's   :86         NA's   :9      
   priority         update          year           managementContinuity
Min.   :1.00   Min.   :1.00   Length:2157        Min.   :0.0000      
1st Qu.:1.00   1st Qu.:3.00   Class :character   1st Qu.:0.0000      
Median :1.00   Median :4.00   Mode  :character   Median :1.0000      
Mean   :1.63   Mean   :4.31                      Mean   :0.6089      
3rd Qu.:2.00   3rd Qu.:5.00                      3rd Qu.:1.0000      
Max.   :4.00   Max.   :8.00                      Max.   :1.0000      
NA's   :32     NA's   :122                       NA's   :63          
managementCyber   rulesUpdating    rulesSecurityConfigs rulesUserControl
Min.   :0.0000   Min.   :0.0000   Min.   :0.0000       Min.   :0.0000  
1st Qu.:0.0000   1st Qu.:1.0000   1st Qu.:1.0000       1st Qu.:1.0000  
Median :1.0000   Median :1.0000   Median :1.0000       Median :1.0000  
Mean   :0.5263   Mean   :0.9584   Mean   :0.9496       Mean   :0.8319  
3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000       3rd Qu.:1.0000  
Max.   :1.0000   Max.   :1.0000   Max.   :1.0000       Max.   :1.0000  
NA's   :63       NA's   :16       NA's   :16           NA's   :16      
policyStaffAccess   policyData      policyPrivate    attackInfection
Min.   :0.0000    Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
1st Qu.:1.0000    1st Qu.:1.0000   1st Qu.:1.0000   1st Qu.:0.0000  
Median :1.0000    Median :1.0000   Median :1.0000   Median :0.0000  
Mean   :0.9395    Mean   :0.8932   Mean   :0.8461   Mean   :0.1133  
3rd Qu.:1.0000    3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:0.0000  
Max.   :1.0000    Max.   :1.0000   Max.   :1.0000   Max.   :1.0000  
NA's   :1033      NA's   :1033     NA's   :1033     NA's   :30      
attackPhising     attackBreaching    outcomesData      outcomesDDOS   
Min.   :0.0000   Min.   :0.00000   Min.   :0.0000   Min.   :0.0000  
1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.0000  
Median :0.0000   Median :0.00000   Median :0.0000   Median :0.0000  
Mean   :0.4664   Mean   :0.04184   Mean   :0.1212   Mean   :0.1442  
3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:0.0000   3rd Qu.:0.0000  
Max.   :1.0000   Max.   :1.00000   Max.   :1.0000   Max.   :1.0000  
NA's   :30       NA's   :30        NA's   :1117     NA's   :1117    
outcomesTheft   
Min.   :0.0000  
1st Qu.:0.0000  
Median :0.0000  
Mean   :0.0423  
3rd Qu.:0.0000  
```

```
 Max.   :1.0000  
 NA's   :1117    
```

```
## I want to cry but at least I have a solution that will just take me
## a few more hours to implement we will have to delabbel and then
## clean most of the labels from the maybe I over reacted a bit, maybe
```

```
######################################################## 2022
######################################################## ###########################################

## perfect factorizationa and numeric conversion removing all the
## problematic labels for imputation
dataCyberSecuritySurvey2022TidyNameSizeCyber$instituitionTypes = as.factor(dataCyberSecuritySu

dataCyberSecuritySurvey2022TidyNameSizeCyber$sizea = as.numeric(dataCyberSecuritySurvey2022Tid

dataCyberSecuritySurvey2022TidyNameSizeCyber$sizeb = as.factor(dataCyberSecuritySurvey2022Tidy

dataCyberSecuritySurvey2022TidyNameSizeCyber$priority = as.factor(dataCyberSecuritySurvey2022T

dataCyberSecuritySurvey2022TidyNameSizeCyber$update = as.factor(dataCyberSecuritySurvey2022Tid

dataCyberSecuritySurvey2022TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit

dataCyberSecuritySurvey2022TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit

dataCyberSecuritySurvey2022TidyNameSizeCyber$year = as.numeric(dataCyberSecuritySurvey2022Tidy

dataCyberSecuritySurvey2022TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit
dataCyberSecuritySurvey2022TidyNameSizeCyber$managementCyber = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2022TidyNameSizeCyber$rulesUpdating = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2022TidyNameSizeCyber$rulesUserControl = as.factor(dataCyberSecuritySur
dataCyberSecuritySurvey2022TidyNameSizeCyber$rulesSecurityConfigs = as.factor(dataCyberSecurit
dataCyberSecuritySurvey2022TidyNameSizeCyber$policyStaffAccess = as.factor(dataCyberSecuritySu
dataCyberSecuritySurvey2022TidyNameSizeCyber$policyData = as.factor(dataCyberSecuritySurvey202
dataCyberSecuritySurvey2022TidyNameSizeCyber$policyPrivate = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2022TidyNameSizeCyber$attackInfection = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2022TidyNameSizeCyber$attackPhising = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2022TidyNameSizeCyber$attackBreaching = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesData = as.factor(dataCyberSecuritySurvey2
dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesDDOS = as.factor(dataCyberSecuritySurvey2
dataCyberSecuritySurvey2022TidyNameSizeCyber$outcomesTheft = as.factor(dataCyberSecuritySurvey


str(dataCyberSecuritySurvey2022TidyNameSizeCyber)
```

```
tibble [2,157 x 21] (S3: tbl_df/tbl/data.frame)
```

```
$ imid                 : chr [1:2157] "191876DGRU" "149212ENSE" "134827SVYQ" "169906ZTQZ" ...
 ..- attr(*, "label")= chr "Unique ID not linked to IDBR or any other sample frames"
 ..- attr(*, "format.spss")= chr "A30"
 ..- attr(*, "display_width")= int 10
$ instituitionTypes    : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 1 1 1 1 ...
$ sizea                : num [1:2157] 3 4 3 8 6 2 5 4 4 5 ...
$ sizeb                : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...
$ priority             : Factor w/ 4 levels "1","2","3","4": 1 2 2 1 2 1 1 1 2 2 ...
$ update               : Factor w/ 8 levels "1","2","3","4",..: 7 1 3 5 6 7 6 6 3 1 ...
$ year                 : num [1:2157] 2022 2022 2022 2022 2022 ...
$ managementContinuity: Factor w/ 2 levels "0","1": 2 2 1 2 1 1 2 2 1 1 ...
$ managementCyber      : Factor w/ 2 levels "0","1": 2 2 1 1 1 2 1 2 2 1 ...
$ rulesUpdating        : Factor w/ 2 levels "0","1": 2 2 2 2 NA 2 2 1 2 2 ...
$ rulesSecurityConfigs: Factor w/ 2 levels "0","1": 2 2 2 2 NA 2 2 2 2 2 ...
$ rulesUserControl     : Factor w/ 2 levels "0","1": 2 1 1 2 NA 2 2 1 1 1 ...
$ policyStaffAccess    : Factor w/ 2 levels "0","1": 1 1 NA 2 NA NA 2 1 NA NA ...
$ policyData           : Factor w/ 2 levels "0","1": 2 1 NA 2 NA NA 2 2 NA NA ...
$ policyPrivate        : Factor w/ 2 levels "0","1": 2 1 NA 2 NA NA 2 1 NA NA ...
$ attackInfection      : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
$ attackPhising        : Factor w/ 2 levels "0","1": 1 1 1 2 1 2 2 1 1 1 ...
$ attackBreaching      : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
$ outcomesData         : Factor w/ 2 levels "0","1": NA NA NA 1 NA 1 1 NA NA NA ...
$ outcomesDDOS         : Factor w/ 2 levels "0","1": NA NA NA 1 NA 1 1 NA NA NA ...
$ outcomesTheft        : Factor w/ 2 levels "0","1": NA NA NA 1 NA 1 1 NA NA NA ...
```

```
####################################################### 2021
####################################################### #######################################

## perfect factorizationa and numeric conversion removing all the
## problematic labels for imputation
dataCyberSecuritySurvey2021TidyNameSizeCyber$institiuitionTypes = as.factor(dataCyberSecuritySu

dataCyberSecuritySurvey2021TidyNameSizeCyber$sizea = as.numeric(dataCyberSecuritySurvey2021Tid

dataCyberSecuritySurvey2021TidyNameSizeCyber$sizeb = as.factor(dataCyberSecuritySurvey2021Tidy

dataCyberSecuritySurvey2021TidyNameSizeCyber$priority = as.factor(dataCyberSecuritySurvey2021T

dataCyberSecuritySurvey2021TidyNameSizeCyber$update = as.factor(dataCyberSecuritySurvey2021Tid

dataCyberSecuritySurvey2021TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit

dataCyberSecuritySurvey2021TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit

dataCyberSecuritySurvey2021TidyNameSizeCyber$year = as.numeric(dataCyberSecuritySurvey2021Tidy

dataCyberSecuritySurvey2021TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit
dataCyberSecuritySurvey2021TidyNameSizeCyber$managementCyber = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2021TidyNameSizeCyber$rulesUpdating = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2021TidyNameSizeCyber$rulesUserControl = as.factor(dataCyberSecuritySur
dataCyberSecuritySurvey2021TidyNameSizeCyber$rulesSecurityConfigs = as.factor(dataCyberSecurit
dataCyberSecuritySurvey2021TidyNameSizeCyber$policyStaffAccess = as.factor(dataCyberSecuritySu
dataCyberSecuritySurvey2021TidyNameSizeCyber$policyData = as.factor(dataCyberSecuritySurvey202
dataCyberSecuritySurvey2021TidyNameSizeCyber$policyPrivate = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2021TidyNameSizeCyber$attackInfection = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2021TidyNameSizeCyber$attackPhising = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2021TidyNameSizeCyber$attackBreaching = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesData = as.factor(dataCyberSecuritySurvey2
dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesDDOS = as.factor(dataCyberSecuritySurvey2
dataCyberSecuritySurvey2021TidyNameSizeCyber$outcomesTheft = as.factor(dataCyberSecuritySurvey


str(dataCyberSecuritySurvey2021TidyNameSizeCyber)
```

```
tibble [2,284 x 21] (S3: tbl_df/tbl/data.frame)
 $ imid                 : chr [1:2284] "182150YKXT" "117166AEHW" "145903ZCHA" "171710NODP" ...
  ..- attr(*, "label")= chr "Unique ID"
  ..- attr(*, "format.spss")= chr "A30"
  ..- attr(*, "display_width")= int 26
 $ instituitionTypes    : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 1 1 1 1 ...
 $ sizea                : num [1:2284] 9 6 2 3 7 2 7 3 6 3 ...
 $ sizeb                : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...
 $ priority             : Factor w/ 4 levels "1","2","3","4": 3 3 1 1 2 2 1 2 1 1 ...
 $ update               : Factor w/ 8 levels "1","2","3","4",..: 4 4 7 5 3 5 3 6 4 8 ...
 $ year                 : num [1:2284] 2021 2021 2021 2021 2021 ...
 $ managementContinuity: Factor w/ 2 levels "0","1": 2 1 1 1 2 1 1 2 2 2 ...
 $ managementCyber      : Factor w/ 2 levels "0","1": 2 2 1 2 2 2 2 2 2 2 ...
 $ rulesUpdating        : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
 $ rulesSecurityConfigs: Factor w/ 2 levels "0","1": 2 2 1 2 2 2 2 2 2 2 ...
 $ rulesUserControl     : Factor w/ 2 levels "0","1": 1 2 2 1 2 2 2 2 2 2 ...
 $ policyStaffAccess    : Factor w/ 2 levels "0","1": 2 NA NA NA 2 NA NA 2 2 2 ...
 $ policyData           : Factor w/ 1 level "0": 1 NA NA NA 1 NA NA 1 1 1 ...
 $ policyPrivate        : Factor w/ 2 levels "0","1": 2 NA NA NA 2 NA NA 2 2 1 ...
 $ attackInfection      : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
 $ attackPhising        : Factor w/ 2 levels "0","1": 2 1 1 1 2 2 1 2 1 2 ...
 $ attackBreaching      : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
 $ outcomesData         : Factor w/ 2 levels "0","1": 1 NA NA NA 1 1 NA 1 NA 1 ...
 $ outcomesDDOS         : Factor w/ 2 levels "0","1": 1 NA NA NA 1 1 NA 1 NA 1 ...
 $ outcomesTheft        : Factor w/ 2 levels "0","1": 1 NA NA NA 2 1 NA 1 NA 1 ...
```

```
###################################################### 2020
###################################################### #######################################

## perfect factorizationa and numeric conversion removing all the
## problematic labels for imputation
```

```
dataCyberSecuritySurvey2020TidyNameSizeCyber$instituitionTypes = as.factor(dataCyberSecuritySu

dataCyberSecuritySurvey2020TidyNameSizeCyber$sizea = as.numeric(dataCyberSecuritySurvey2020Tid

dataCyberSecuritySurvey2020TidyNameSizeCyber$sizeb = as.factor(dataCyberSecuritySurvey2020Tidy

dataCyberSecuritySurvey2020TidyNameSizeCyber$priority = as.factor(dataCyberSecuritySurvey2020T

dataCyberSecuritySurvey2020TidyNameSizeCyber$update = as.factor(dataCyberSecuritySurvey2020Tid

dataCyberSecuritySurvey2020TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit

dataCyberSecuritySurvey2020TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit

dataCyberSecuritySurvey2020TidyNameSizeCyber$year = as.numeric(dataCyberSecuritySurvey2020Tidy

dataCyberSecuritySurvey2020TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit
dataCyberSecuritySurvey2020TidyNameSizeCyber$managementCyber = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2020TidyNameSizeCyber$rulesUpdating = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2020TidyNameSizeCyber$rulesUserControl = as.factor(dataCyberSecuritySur
dataCyberSecuritySurvey2020TidyNameSizeCyber$rulesSecurityConfigs = as.factor(dataCyberSecurit
dataCyberSecuritySurvey2020TidyNameSizeCyber$policyStaffAccess = as.factor(dataCyberSecuritySu
dataCyberSecuritySurvey2020TidyNameSizeCyber$policyData = as.factor(dataCyberSecuritySurvey202
dataCyberSecuritySurvey2020TidyNameSizeCyber$policyPrivate = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2020TidyNameSizeCyber$attackInfection = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2020TidyNameSizeCyber$attackPhising = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2020TidyNameSizeCyber$attackBreaching = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesData = as.factor(dataCyberSecuritySurvey2
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesDDOS = as.factor(dataCyberSecuritySurvey2
dataCyberSecuritySurvey2020TidyNameSizeCyber$outcomesTheft = as.factor(dataCyberSecuritySurvey


str(dataCyberSecuritySurvey2020TidyNameSizeCyber)
```

```
tibble [1,900 x 21] (S3: tbl_df/tbl/data.frame)
 $ imid                : num [1:1900] 1e+05 1e+05 1e+05 1e+05 1e+05 ...
  ..- attr(*, "label")= chr "Unique ID not linked to IDBR"
  ..- attr(*, "format.spss")= chr "F8.2"
  ..- attr(*, "display_width")= int 10
 $ instituitionTypes   : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 1 1 1 1 ...
 $ sizea               : num [1:1900] 2 5 8 2 2 2 3 3 7 8 ...
 $ sizeb               : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...
 $ priority            : Factor w/ 4 levels "1","2","3","4": 1 1 2 3 1 4 1 1 2 1 ...
 $ update              : Factor w/ 8 levels "1","2","3","4",..: 7 6 4 3 7 1 7 5 4 3 ...
 $ year                : num [1:1900] 2020 2020 2020 2020 2020 2020 2020 2020 2020 2020 ...
 $ managementContinuity: Factor w/ 2 levels "0","1": 2 1 1 1 2 1 2 1 2 2 ...
 $ managementCyber     : Factor w/ 2 levels "0","1": 2 2 2 1 2 1 2 2 2 2 ...
 $ rulesUpdating       : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
 $ rulesSecurityConfigs: Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
 $ rulesUserControl    : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
 $ policyStaffAccess   : Factor w/ 2 levels "0","1": 2 NA NA NA NA NA 1 NA NA 2 ...
 $ policyData          : Factor w/ 2 levels "0","1": 2 NA NA NA NA NA 2 NA NA 2 ...
 $ policyPrivate       : Factor w/ 2 levels "0","1": 2 NA NA NA NA NA 1 NA NA 1 ...
 $ attackInfection     : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
 $ attackPhising       : Factor w/ 2 levels "0","1": 2 1 2 1 1 2 2 1 2 1 ...
 $ attackBreaching     : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
 $ outcomesData        : Factor w/ 2 levels "0","1": 1 NA 1 NA NA 1 1 NA 1 NA ...
 $ outcomesDDOS        : Factor w/ 2 levels "0","1": 1 NA 1 NA NA 1 1 NA 1 NA ...
 $ outcomesTheft       : Factor w/ 2 levels "0","1": 1 NA 1 NA NA 1 1 NA 1 NA ...
```

```r
###################################################### 2019
###################################################### #######################################

## perfect factorizationa and numeric conversion removing all the
## problematic labels for imputation
dataCyberSecuritySurvey2019TidyNameSizeCyber$instituitionTypes = as.factor(dataCyberSecuritySu

dataCyberSecuritySurvey2019TidyNameSizeCyber$sizea = as.numeric(dataCyberSecuritySurvey2019Tid

dataCyberSecuritySurvey2019TidyNameSizeCyber$sizeb = as.factor(dataCyberSecuritySurvey2019Tidy

dataCyberSecuritySurvey2019TidyNameSizeCyber$priority = as.factor(dataCyberSecuritySurvey2019T

dataCyberSecuritySurvey2019TidyNameSizeCyber$update = as.factor(dataCyberSecuritySurvey2019Tid

dataCyberSecuritySurvey2019TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit

dataCyberSecuritySurvey2019TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit

dataCyberSecuritySurvey2019TidyNameSizeCyber$year = as.numeric(dataCyberSecuritySurvey2019Tidy

dataCyberSecuritySurvey2019TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit
dataCyberSecuritySurvey2019TidyNameSizeCyber$managementCyber = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2019TidyNameSizeCyber$rulesUpdating = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2019TidyNameSizeCyber$rulesUserControl = as.factor(dataCyberSecuritySur
dataCyberSecuritySurvey2019TidyNameSizeCyber$rulesSecurityConfigs = as.factor(dataCyberSecurit
dataCyberSecuritySurvey2019TidyNameSizeCyber$policyStaffAccess = as.factor(dataCyberSecuritySu
dataCyberSecuritySurvey2019TidyNameSizeCyber$policyData = as.factor(dataCyberSecuritySurvey201
dataCyberSecuritySurvey2019TidyNameSizeCyber$policyPrivate = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2019TidyNameSizeCyber$attackInfection = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2019TidyNameSizeCyber$attackPhising = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2019TidyNameSizeCyber$attackBreaching = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesData = as.factor(dataCyberSecuritySurvey2
dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesDDOS = as.factor(dataCyberSecuritySurvey2
dataCyberSecuritySurvey2019TidyNameSizeCyber$outcomesTheft = as.factor(dataCyberSecuritySurvey
str(dataCyberSecuritySurvey2019TidyNameSizeCyber)
```

```
tibble [2,080 x 21] (S3: tbl_df/tbl/data.frame)
 $ imid                 : num [1:2080] 1e+05 1e+05 1e+05 1e+05 1e+05 ...
  ..- attr(*, "label")= chr "Unique ID not linked to IDBR"
  ..- attr(*, "format.spss")= chr "F8.2"
  ..- attr(*, "display_width")= int 10
 $ instituitionTypes    : Factor w/ 2 levels "1","2": 1 1 1 1 1 1 1 1 1 1 ...
 $ sizea                : num [1:2080] 6 4 4 2 4 5 2 5 5 6 ...
 $ sizeb                : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...
 $ priority             : Factor w/ 4 levels "1","2","3","4": 3 NA 2 3 1 1 1 1 2 1 ...
 $ update               : Factor w/ 8 levels "1","2","3","4",..: 1 NA 1 1 1 4 7 4 7 8 ...
 $ year                 : num [1:2080] 2019 2019 2019 2019 2019 ...
 $ managementContinuity: Factor w/ 2 levels "0","1": 1 1 1 1 2 2 2 1 1 2 ...
 $ managementCyber     : Factor w/ 2 levels "0","1": 1 1 1 1 2 2 2 2 2 2 ...
 $ rulesUpdating       : Factor w/ 2 levels "0","1": 2 2 2 1 2 2 2 2 2 2 ...
 $ rulesSecurityConfigs: Factor w/ 2 levels "0","1": 2 2 2 1 2 2 2 1 1 2 ...
 $ rulesUserControl    : Factor w/ 2 levels "0","1": 1 2 2 1 2 2 2 1 1 2 ...
 $ policyStaffAccess   : Factor w/ 2 levels "0","1": NA NA NA NA 2 NA 2 NA NA 1 ...
 $ policyData          : Factor w/ 2 levels "0","1": NA NA NA NA 2 NA 2 NA NA 1 ...
 $ policyPrivate       : Factor w/ 2 levels "0","1": NA NA NA NA 2 NA 2 NA NA 1 ...
 $ attackInfection     : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 2 1 1 1 ...
 $ attackPhising       : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 2 1 1 1 ...
 $ attackBreaching     : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
 $ outcomesData        : Factor w/ 2 levels "0","1": NA NA 1 NA NA NA 1 NA NA NA ...
 $ outcomesDDOS        : Factor w/ 2 levels "0","1": NA NA 1 NA NA NA 1 NA NA NA ...
 $ outcomesTheft       : Factor w/ 2 levels "0","1": NA NA 1 NA NA NA 1 NA NA NA ...
```

```r
###################################################### 2018
###################################################### #######################################

## perfect factorizationa and numeric conversion removing all the
## problematic labels for imputation
dataCyberSecuritySurvey2018TidyNameSizeCyber$instituitionTypes = as.factor(dataCyberSecuritySu

dataCyberSecuritySurvey2018TidyNameSizeCyber$sizea = as.numeric(dataCyberSecuritySurvey2018Tid

dataCyberSecuritySurvey2018TidyNameSizeCyber$sizeb = as.factor(dataCyberSecuritySurvey2018Tidy

dataCyberSecuritySurvey2018TidyNameSizeCyber$priority = as.factor(dataCyberSecuritySurvey2018T

dataCyberSecuritySurvey2018TidyNameSizeCyber$update = as.factor(dataCyberSecuritySurvey2018Tid

dataCyberSecuritySurvey2018TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit

dataCyberSecuritySurvey2018TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit
dataCyberSecuritySurvey2018TidyNameSizeCyber$year = as.numeric(dataCyberSecuritySurvey2018Tidy

dataCyberSecuritySurvey2018TidyNameSizeCyber$managementContinuity = as.factor(dataCyberSecurit
dataCyberSecuritySurvey2018TidyNameSizeCyber$managementCyber = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2018TidyNameSizeCyber$rulesUpdating = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2018TidyNameSizeCyber$rulesUserControl = as.factor(dataCyberSecuritySur
dataCyberSecuritySurvey2018TidyNameSizeCyber$rulesSecurityConfigs = as.factor(dataCyberSecurit
dataCyberSecuritySurvey2018TidyNameSizeCyber$policyStaffAccess = as.factor(dataCyberSecuritySu
dataCyberSecuritySurvey2018TidyNameSizeCyber$policyData = as.factor(dataCyberSecuritySurvey201
dataCyberSecuritySurvey2018TidyNameSizeCyber$policyPrivate = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2018TidyNameSizeCyber$attackInfection = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2018TidyNameSizeCyber$attackPhising = as.factor(dataCyberSecuritySurvey
dataCyberSecuritySurvey2018TidyNameSizeCyber$attackBreaching = as.factor(dataCyberSecuritySurv
dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesData = as.factor(dataCyberSecuritySurvey2
dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesDDOS = as.factor(dataCyberSecuritySurvey2
dataCyberSecuritySurvey2018TidyNameSizeCyber$outcomesTheft = as.factor(dataCyberSecuritySurvey


str(dataCyberSecuritySurvey2018TidyNameSizeCyber)
```

```
tibble [2,088 x 21] (S3: tbl_df/tbl/data.frame)
 $ imid                : num [1:2088] 100065 100075 100304 100318 100779 ...
  ..- attr(*, "label")= chr "Unique ID not linked to IDBR"
  ..- attr(*, "format.spss")= chr "F8.0"
  ..- attr(*, "display_width")= int 10
 $ instituitionTypes   : Factor w/ 2 levels "1","2": 1 1 1 1 1 1 1 1 1 1 ...
 $ sizea               : num [1:2088] 60 13 4 100 3 70 50 60 80 NA ...
 $ sizeb               : Factor w/ 4 levels "1","2","3","4": 3 2 1 3 1 3 3 3 3 4 ...
 $ priority            : Factor w/ 4 levels "1","2","3","4": 1 1 1 2 2 2 2 2 2 1 ...
 $ update              : Factor w/ 9 levels "1","2","3","4",..: 9 4 9 4 5 9 6 5 3 6 ...
 $ year                : num [1:2088] 2018 2018 2018 2018 2018 ...
 $ managementContinuity: Factor w/ 2 levels "0","1": 2 2 1 2 2 2 2 2 2 2 ...
 $ managementCyber     : Factor w/ 2 levels "0","1": 1 2 2 2 2 2 2 2 1 2 ...
 $ rulesUpdating       : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
 $ rulesSecurityConfigs: Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
 $ rulesUserControl    : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
 $ policyStaffAccess   : Factor w/ 2 levels "0","1": NA NA NA 2 2 2 2 2 NA 2 ...
 $ policyData          : Factor w/ 2 levels "0","1": NA NA NA 2 1 2 2 2 NA 2 ...
 $ policyPrivate       : Factor w/ 2 levels "0","1": NA NA NA 1 2 2 2 2 NA 2 ...
 $ attackInfection     : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
 $ attackPhising       : Factor w/ 2 levels "0","1": 2 2 1 1 1 2 1 2 2 1 ...
 $ attackBreaching     : Factor w/ 2 levels "0","1": 1 1 1 2 1 1 1 1 1 1 ...
 $ outcomesData        : Factor w/ 2 levels "0","1": 1 1 NA 1 NA 1 NA 1 1 NA ...
 $ outcomesDDOS        : Factor w/ 2 levels "0","1": 1 2 NA 1 NA 1 NA 1 1 NA ...
 $ outcomesTheft       : Factor w/ 2 levels "0","1": 1 1 NA 1 NA 1 NA 1 1 NA ...
```

0.11 Data Imputation

```
## remember we must make the imputations before we merge the data of
## all the years


## m= number of imputations maxit = number of iterations

## ppm is for continuous missing data loreg is for dichotomous missing
## data //dichotomous two mutuallu exclusive groups polyreg is for
## categorical missing data // categorical variable can be one of a
## limited, and usually fixed number of values aka its discrete norm is
## Bayesian linear regression without predictive mean matching TODO
## maybe write this one the report

## ok so we should have maxit(number of iterations) equal to number of
## variables because in

## iteration 0 - per imputed dataset at iteration number 0 values are
## randomly drawn iteration 1 - At this step the first variable values
## are set back to missing. Subsequently, a linear regression model is
## applied in the available data iteration 2 - the same procedure is
## repeated for the next variable and so on
```

("Book_MI.knit" 2022)

```
## as we can see in Rubin's works the larger the number of data sets
## the larger the error will meaning a finite number of imputations is
## favorable to an infinite number of imputations RE=1/(1+(FMI/m))
```

(Rubin 1975)

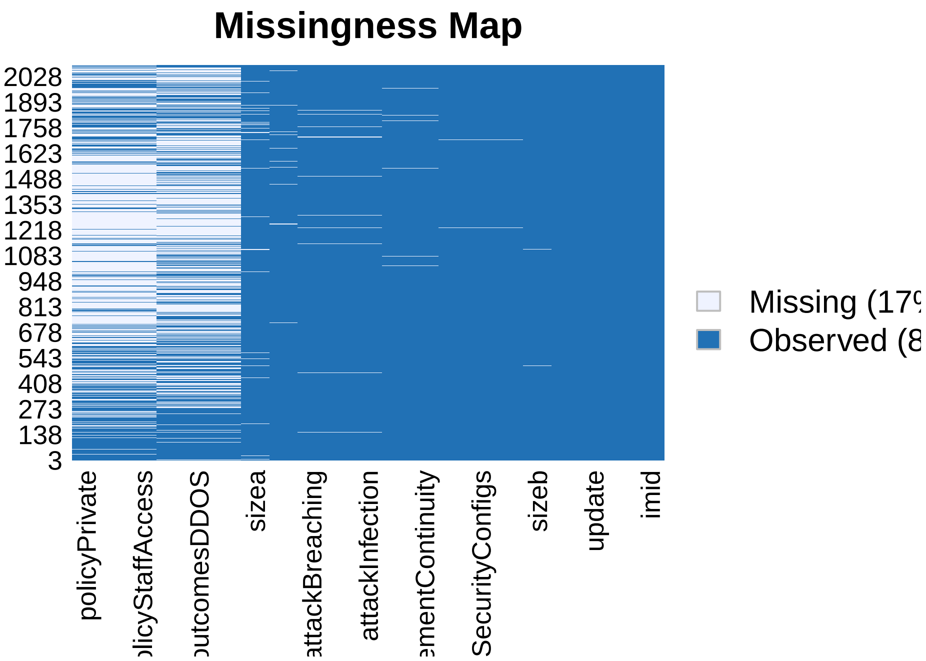
```
## in White, Royston, and Wood book "Multiple imputation using chained
## equations: Issues and guidance for practice' they proposed the rule
## of equating the number of imputations to the percentage of missing
## data in each of the data sets which is what I will be using
```

(**White2010?**)

```
missmap(dataCyberSecuritySurvey2018TidyNameSizeCyber)  ## 17% missing data
```

```
Warning: Unknown or uninitialised column: `arguments`.
Unknown or uninitialised column: `arguments`.


Warning: Unknown or uninitialised column: `imputations`.
```

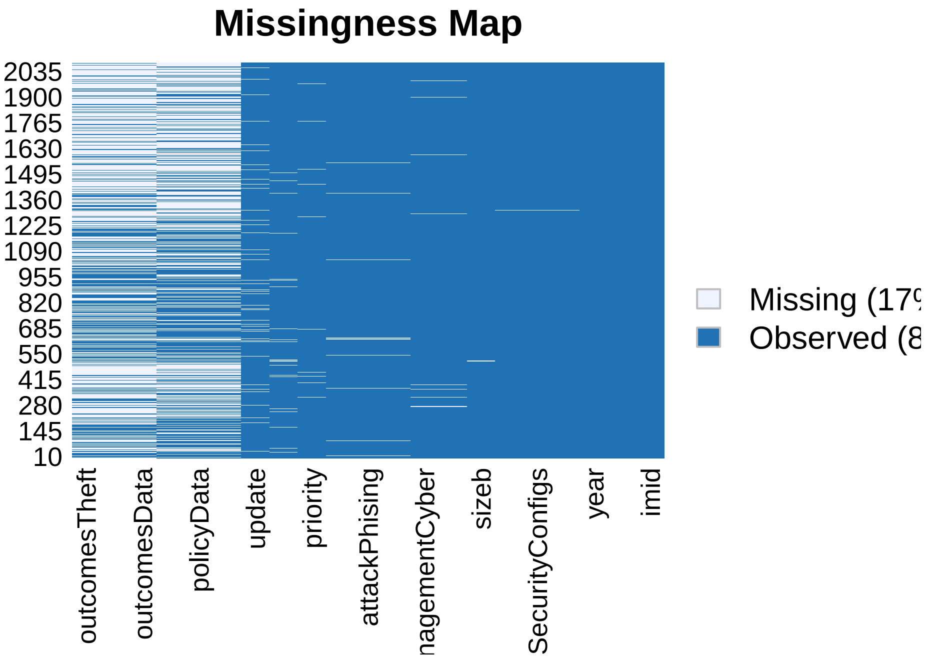


```
missmap(dataCyberSecuritySurvey2019TidyNameSizeCyber)  ## 17% missing data
```

```
Warning: Unknown or uninitialised column: `arguments`.

Warning: Unknown or uninitialised column: `arguments`.

Warning: Unknown or uninitialised column: `imputations`.
```

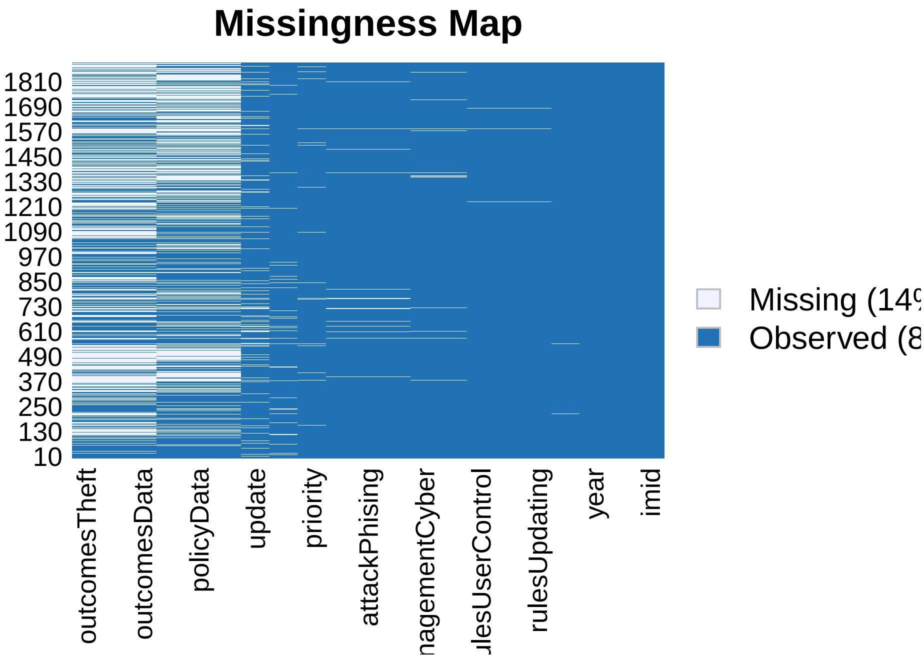


```r
missmap(dataCyberSecuritySurvey2020TidyNameSizeCyber)  ## 14% missing data
```

```
Warning: Unknown or uninitialised column: `arguments`.

Warning: Unknown or uninitialised column: `arguments`.

Warning: Unknown or uninitialised column: `imputations`.
```

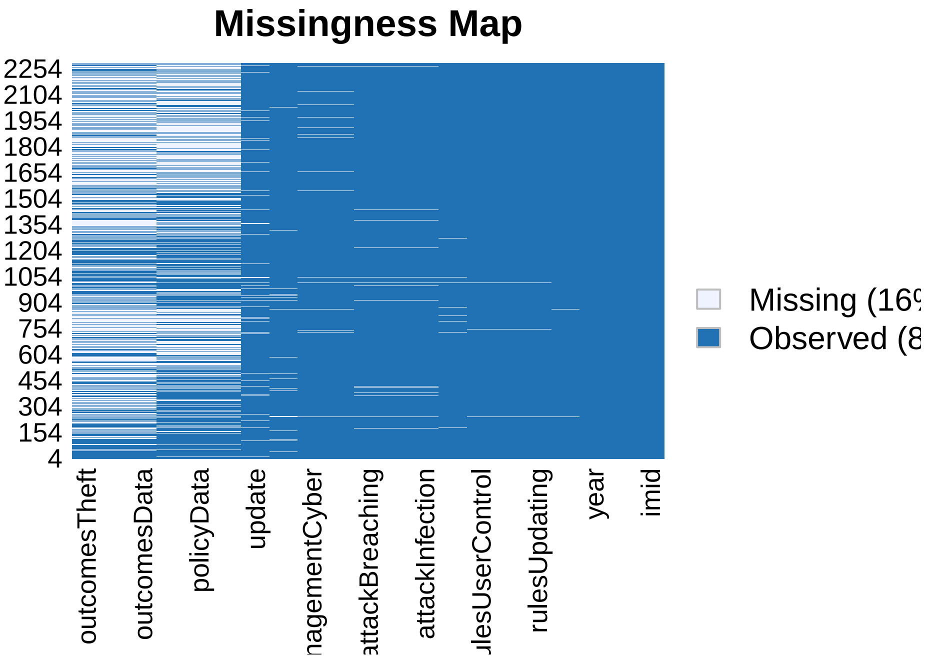


```r
missmap(dataCyberSecuritySurvey2021TidyNameSizeCyber)  ## 16% missing data
```

```
Warning: Unknown or uninitialised column: `arguments`.

Warning: Unknown or uninitialised column: `arguments`.

Warning: Unknown or uninitialised column: `imputations`.
```

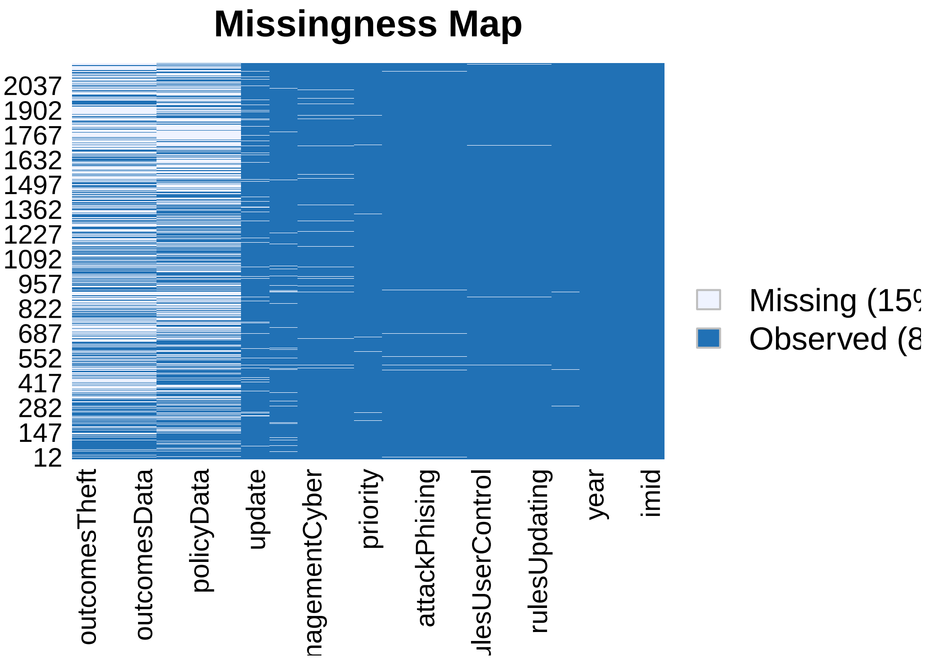


```
missmap(dataCyberSecuritySurvey2022TidyNameSizeCyber)   ## 15% missing data
```

```
Warning: Unknown or uninitialised column: `arguments`.

Warning: Unknown or uninitialised column: `arguments`.

Warning: Unknown or uninitialised column: `imputations`.
```



0.12 Imputed values here

```
## we will release the krakens after all the testing and debuggins
## imp2022 = mice(dataCyberSecuritySurvey2022TidyNameSizeCyber, m=100,
## maxit=21)

## we actually don't want to specify the method here because mice will
## automatically choose between logression and polyregression for the
## values depending on the R data structure

imp2022 = mice(dataCyberSecuritySurvey2022TidyNameSizeCyber, m = 17, maxit = 21)
imp2021 = mice(dataCyberSecuritySurvey2022TidyNameSizeCyber, m = 17, maxit = 21)
imp2020 = mice(dataCyberSecuritySurvey2022TidyNameSizeCyber, m = 14, maxit = 21)
imp2019 = mice(dataCyberSecuritySurvey2022TidyNameSizeCyber, m = 16, maxit = 21)
imp2018 = mice(dataCyberSecuritySurvey2022TidyNameSizeCyber, m = 15, maxit = 21)

imp2022$method

# imp$loggedEvents view(imp$loggedEvents)

## don't forget
## https://stats.stackexchange.com/questions/76488/error-system-is-computationally-singular-wh

## from this day on I will pray and offer a candle to my new god,
## professor Martijn W Heymans
## https://missingdatasolutions.rbind.io/contact/

## truly a blessing from the lord


## the imputed data sets are extracted with the function complete. The
## options action = "long" and include = TRUE returns a dataframe with
## imputed datasets that are stacked with each other imputed datasets
## plus the original dataset (with missings) on top
```

0.13 Fitting the model after the imputations

```
## continuing from here because otherwise the reference won't work


imp2022$predictorMatrix
## the mice library already turned on the relevant variable for the
## imputation modelling turning only the imid(unique id) and year
## variable which is not useful for the imputations done on a year to
## year basis therefore I don't need to tweak it manually using pred
## <-imp$predictorMatrix
## I might disable the size, priority or updates if needed though but
## they seemed to be needed since they make the amount missing values
## depends on these (basically auxiliary variables)


## https://stefvanbuuren.name/fimd/workflow.html // go to 5.1 mice
## documentation is *chef kiss*

# imp$nmis

## the imputations are done so now we will use the with function from
## mice to fit the model

## chose the one with the lowest aic and bic
anova()

## fitting the model to one of the variables of the outcomes, this case
## I will start with the outcome for Data damages since it is the more
## commonly flagged one

## this does not work because it is not finding anything meaning
## between my factors
fit2022 = with(imp2022, lm(outcomesData ~ attackBreaching + attackPhising +
    attackInfection + managementCyber + managementCyber))

fit2022Attacks = with(imp2022, lm(outcomesData ~ attackBreaching + attackPhising +
    attackInfection))

anova(fit2022, fit2022Attacks)

fit2022Attacks = with(dataCyberSecuritySurvey2022TidyNameSizeCyber, lm(outcomesData ~
    attackBreaching))

dataCyberSecuritySurvey2022TidyNameSizeCyber$attackBreaching = as.numeric(dataCyberSecuritySur

for (i in 1:length(dataCyberSecuritySurvey2022TidyNameSizeCyber$attackBreaching[i])) {
    dataCyberSecuritySurvey2022TidyNameSizeCyber$attackBreaching[i] = replace_na(dataCyberSecu
        2)
    if (dataCyberSecuritySurvey2022TidyNameSizeCyber$attackBreaching[i] ==
        1e-07) {
        dataCyberSecuritySurvey2022TidyNameSizeCyber$attackBreaching[i] = 2
    }

}

str(dataCyberSecuritySurvey2022TidyNameSizeCyber)
```

```
dataCyberSecuritySurvey2022TidyNameSizeCyber$attackBreaching = as.factor(dataCyberSecuritySurv

fit2022 = with(imp2022, lm(outcomesTheft ~ attackBreaching + attackPhising +
    attackInfection))

## No meaningfull correlation moment
summary(fit2022)


## est2022=pool(fit2022)

test = dataCyberSecuritySurvey2022TidyNameSizeCyber

dataCyberSecuritySurvey2022TidyNameSizeCyber

str(dataCyberSecuritySurvey2022TidyNameSizeCyber)
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-year)
dataCyberSecuritySurvey2022TidyNameSizeCyber = dataCyberSecuritySurvey2022TidyNameSizeCyber %>
    select(-sizea)

## after fitting the model we will use the pooled imputations using the
## model to make the estimation





# plot(imp2022) stripplot(imp2018)



# complete(imp, action = 'long', include = TRUE)
```

```
1 + 1
```

```
[1] 2
```

```
1 + 1
```

```
[1] 2
```

```
1 + 1
```

```
[1] 2
```

```r
1 + 1
```

```
[1] 2
```

```r
1 + 1
```

```
[1] 2
```

```r
1 + 1
```

```
[1] 2
```

```r
1 + 1
```

```
[1] 2
```

```r
1 + 1
```

```
[1] 2
```

```r
1 + 1
```

```
[1] 2
```

```r
1 + 1
```

```
[1] 2
```

```r
1 + 1
```

```
[1] 2
```

```r
1 + 1
```

```
[1] 2
```

```r
1 + 1
```

```
[1] 2
```

```
1 + 1
```

```
[1] 2
```

```
1 + 1
```

```
[1] 2
```

```
1 + 1
```

```
[1] 2
```

```
## my last brain cells also code is finished compiling
browseURL("https://www.youtube.com/watch?v=M3Keg5XKJ08")

## in case of emergency break this comment
## browseURL('https://youtu.be/eoV7lw7YBG4?t=6')
```

0.14 Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see https://quarto.org.

0.15 Running Code

When you click the **Render** button a document will be generated that includes both content and the output of embedded code. You can embed code like this:

```
1 + 1
```

```
[1] 2
```

You can add options to executable code like this

```
[1] 4
```

The `echo: false` option disables the printing of code (only output is displayed).

"Book_MI.knit." 2022. *Home*. https://bookdown.org/mwheymans/bookmi/.

Department For Digital, Culture. 2020. "Cyber Security Breaches Survey, 2020." UK Data Service. https://doi.org/10.5255/UKDA-SN-8638-1.

Rubin, Donald B. 1975. "Biometrika 63 (3): 581–90." In *Inference and Missing Data*. Verlag nicht ermittelbar.