Welcome to my portofolio

Lamyae el Mahboud

2023-05-15

Contents

Chapter 1

My portofolio

1.1 introduction

Welcome to my portofolio. My name is Lamyae and im studying at the Utrecht University of Applied Sciences.

In this github page you'll find different skills.

Chapter 2

C. elegans plate experiment analysis

2.1 Introduction

In this exercise we will read and analyze a data file. While analyzing the data we come across a few issues we try to fix. The data we're gonna use in this exercise was derived from an experiment in which adult C.elegans were exposed to varying concentrations of different compounds. The variables RawData (the outcome - number of offspring counted as an integer value, after incubation time), compName (the generic name of the compound/chemical), the compConcentration (the concentration of the compound), and the expType are the most important variables in this dataset.

After we reviewed the Excel File. We see in the Excel file that there is no data in the colomns 'plateRow' en 'plateColmn'. Some colomns arent nesecerasy like 'expDate', 'expResearcher', 'expTime', 'éxpUnit'. This kind of information is metadata and should not be in your rawdata. It makes the table cluttered and information less easy to find. Some data in colomn is the same for all coditions, like 'compDelivery' and 'bacterialStrain'. This kind of information is metadata and shouldn't be in your table.

after reviewing the file we want to open the excel file in R using the {readxl} package.

```
library(tidyverse) #load in tidyverse package
library(readxl) #load in readxl package

data <- read_excel("CE.LIQ.FLOW.062_Tidydata.xlsx") #load in the data that we want to use for th
```

Before we go ahead an start analyzing our data. We're gonna inspect the following colomns: “Rawdata”, “compName” and “compConcentration”

```
head(data[c('RawData', 'compName', 'compConcentration')]) #we want to see the first fe
```

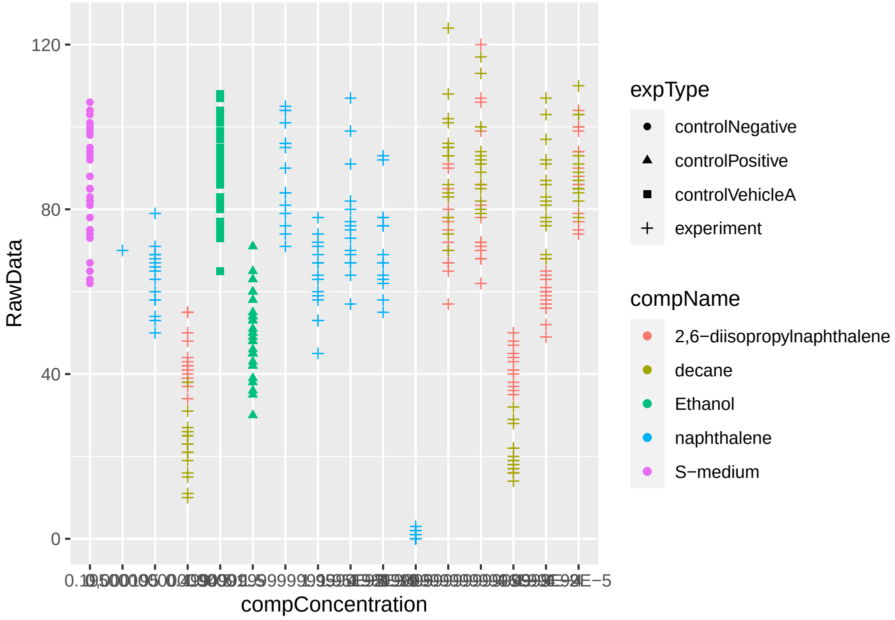
```
## # A tibble: 6 x 3
##   RawData compName                   compConcentration
##     <dbl> <chr>                      <chr>
## 1      44 2,6-diisopropylnaphthalene 4.99
## 2      37 2,6-diisopropylnaphthalene 4.99
## 3      45 2,6-diisopropylnaphthalene 4.99
## 4      47 2,6-diisopropylnaphthalene 4.99
## 5      41 2,6-diisopropylnaphthalene 4.99
## 6      35 2,6-diisopropylnaphthalene 4.99
```

The colomn ‘Rawdata’ is a “dbl” type, this is what we expected. The colomn ‘compName’ is a “chr” type, this is what we expected . The colomn ‘compConcentration’ is a “chr” type, this is not correct since they are only numbers in the colomn.

We’re gonna make a scatterplot for the data. As you can see in the code below we put the “compConcentration” on the x-axis, the “Rawdata” counts on the y-axis. For each level in “compName” we assign a colour and for each level in “expType” we assign a different symbol.

```
library(tidyverse) #load in the tidyverse data

ggplot(data = data, aes(x = compConcentration, y = RawData )) +
  geom_point(aes(color = compName,
                 shape = expType)) #scatterplot for the different compounds and the va
```



We just made a scatterplot with the data from the excel file we imported earlier. Each compName has its own colour and each expType has its own shape, but we see one big problem. As we can see is the x-axis not very readable, actually not at all!

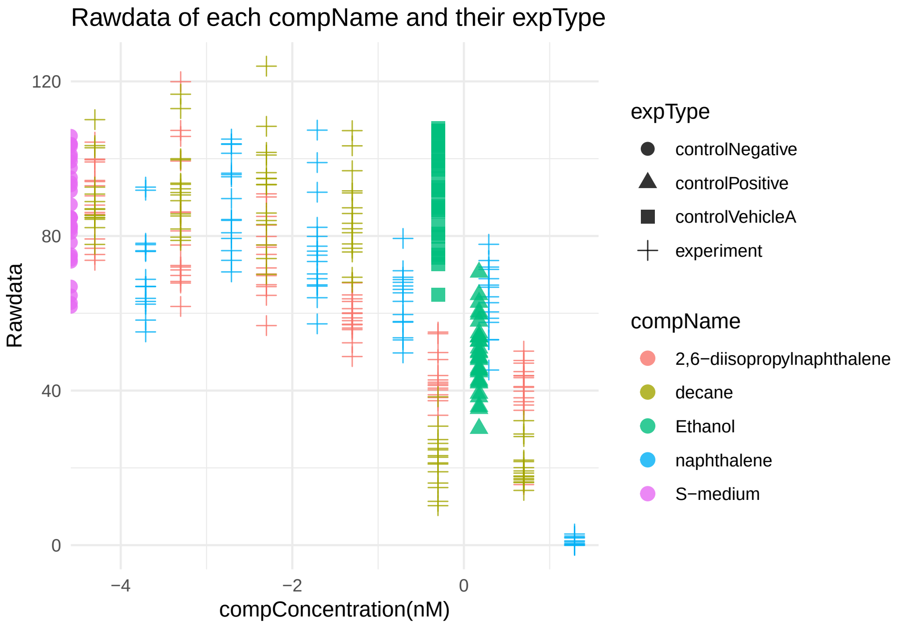
when we imported the Excel file, the colomn ‘compConcentration’ had a charachter type. Because of this every single charachter in each cell was shown on the x-axis and the numbers weren’t rounded. This explains why the x-axis wasn’t readable.

To correct this mistake, we’re gonna change the type of the colomn “compConcentration” from “chr” to “dbl”.

```
data$compConcentration <- as.numeric(data$compConcentration) #change the type of the colomn from

library(tidyverse) #load in tidyverse package

ggplot(data = data, aes(x = log10(compConcentration), y = RawData)) +
  geom_jitter(aes(color = compName,
                  shape = expType),
              size = 3,alpha = 0.8) +
  labs(title = "Rawdata of each compName and their expType",
       y = "Rawdata",
       x = "compConcentration(nM)") +
  theme_minimal() #scatterplot of the data with fixed x-axis and jitter
```



As we van see the x-axis is now readable.

We see in the scatterplot that the positive control for this experiments is Ethanol. The negative control for this experiment is S-medium.

if we really want to compare the different between the compund and experiments type. We need to normalize the data for the Negative control. We normalize the data by setting the mean value for the “controlNegative” to 1. All the other values are expressed as a fraction thereof.

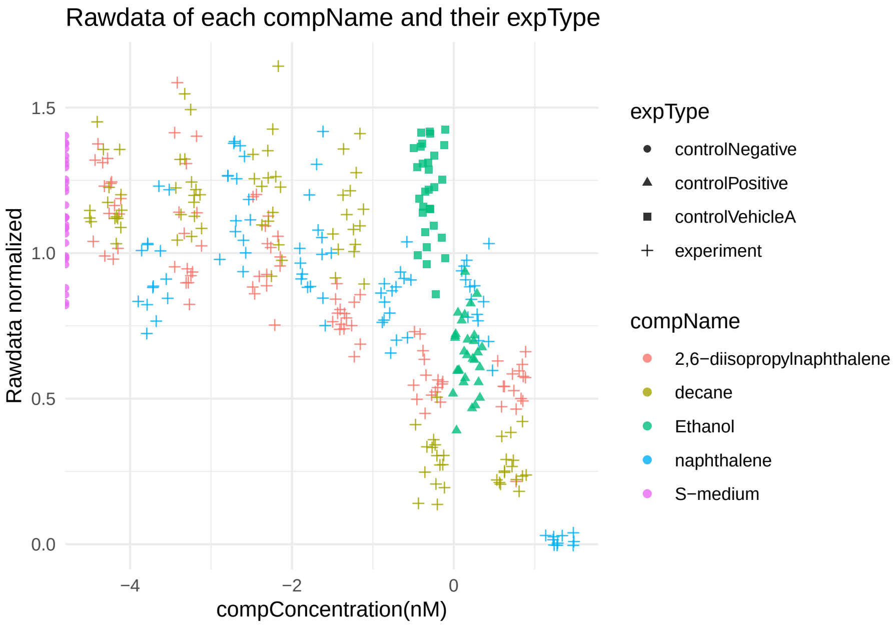
```
library(tidyverse)

data_df <- as.data.frame(data)

controlnegative_mean <- mean(data_df$RawData[data_df$compVehicle == "controlNegative"]

data_df$data_norm <- data_df$RawData/controlnegative_mean

ggplot(data = data_df, aes(x = log10(compConcentration), y = data_norm)) +
  geom_jitter(aes(color = compName,
                  shape = expType),
              size = 1.5,alpha = 0.8, width = 0.2) +
  labs(title = "Rawdata of each compName and their expType",
       y = "Rawdata normalized",
       x = "compConcentration(nM)") +
  theme_minimal() #scatterplot of the data with fixed x-axis and jitter
```



In the dataset we used for this analysis, each compound has his own concentrations. If we want make the data more comparable we have to normalize the data.

2.2 analyzing plan

If we want to analyze the data to learn whether there is indeed an effect of different concentrations on offspring count and whether the different compounds have a different curve(ic50), First we need to get the data and prepare it. We did this already as you can see above.

after preparing the data we will perform the shapiro.test for every compound to check if the data (Rawdata) is normally distributed. All compounds need to have a p-value greater than 0.05.

To test if there is a significant difference between compound concentrations we can use the one sample t-test for each compound individually. To do this we have to filter the table for the compound we want to test first before we perform the test.

To compare the IC50 of each compound we can use the {drc} package to visualize the different curves for each compound.

Chapter 3

Open Peer Review

3.1 Introduction

We're gonna identify reproducibility issues in a scientific publication. We use the criteria for reproduciblity to see if our article is reproducible or not. The scientific article that we use can be found here.

3.2 Clarification

We've read the article and scored the article based on the Transparency Criteria in table 1. For each "Yes" in "Availability" the article scores one point. As we can see in table 1 the article scores four points out of eight. This article is not reproducible.

The authors give us a statement why they started this research in the last paragraph. Although they show all their results, there is no raw data available

Table 3.1: Table1: The Transparency Criteria scores of the scientific article

transparency Criteria	Available
Study Purpose	Yes
Data Availability Statement	No
Data Location	No
Study Location	No
Author Review	Yes
Ethics Statement	No
Funcing statement	Yes
Code Availability	No

of the western blot or the qPCR.

The data of this study cant be accessed. There is no file or link with the raw or processed data available, only results.

The paper doesn't provide the exact study location in the methods section or else where. However it does give us a few laboratories where the research is done, like the Jackson Laboratory and the Bartek Laboratory. But since these laboratories have multiple locations we still don't know where they performed the experiments.

In the paper is an e-mail adress of one of the authors. There are no contact information of the other authors. The paper doesn't give us a statement within the paper indicating any ethical concerns. The paper does give us a statement within the paper indicating that the authors received funding for their research. In the last paragraph under Results and Discussion the authors give use a summary of all their findings.

The codes that the researchers used in this experiment are not available for the reader.

3.3 the second peer review

h and i. https://osf.io/87mpk

j. This code is analyzing two countries, the USA and India.

k. This codes scores a two out of five on readability. the codes contain a 'header', but it could be more informative. A header like "Donation" doesn't give the reader enough information.

l. before we could run the code, we hade to download a dataset named "data.clean.txt".

```
library(rethinking)

d<-read.table("data.clean.txt",sep="\t",header=T,stringsAsFactors = F)
names(d)

dat <- list(
  C=d$CountC,

  A = standardize(d$Age),
  G = as.integer(as.factor(d$Gender)),  #1 Man, 2 Woman
  CD = as.integer(d$COND),
  Mfm = standardize(d$MAC.fam),
  Mg = standardize(d$MAC.gro),
  Mr = standardize(d$MAC.rec),
  Mh = standardize(d$MAC.her),
  Md = standardize(d$MAC.def),
  Mfi = standardize(d$MAC.fai),
  Mp = standardize(d$MAC.pro),
  P = standardize(d$Precaution),
  S = standardize(d$Prosociality),
  D = standardize(d$donation),

  Dgr=standardize(d$danger.for.participant),

  CC = ifelse(d$COND==1,0,ifelse(d$COND==2,0,ifelse(d$COND==3,0,ifelse(d$COND==4,standardize(d$MA

)

set.seed(42)
m1 <- ulam(
  alist(

    D ~ dnorm(muD,sigmaD),

    muD<-aG[G]+bA*A+bP*P+bS*S+aC[CD]+bCon*CC+bFam*Mfm+bGro*Mg+bRec*Mr+bHer*Mh+bDef*Md+bFai*Mfi+bP

    #Donation
    aG[G]~dnorm(0,0.2),
    bA~dnorm(0,0.5),

    bP~dnorm(0,0.5),
    bS~dnorm(0,0.5),

    aC[CD]~dnorm(0,0.2),

    bCon~dnorm(0,0.5),
    bFam~dnorm(0,0.5),
    bGro~dnorm(0,0.5),
    bRec~dnorm(0,0.5),
    bHer~dnorm(0,0.5),
    bDef~dnorm(0,0.5),
    bFai~dnorm(0,0.5),
    bPro~dnorm(0,0.5),

    bDg~dnorm(0,0.5),


    #Model of precaution and prosociality
    P ~ dnorm(muP,sigmaP),
    S ~ dnorm(muS,sigmaS),
    muP<-aGP[G]+bAP*A+aCP[CD]+bConP*CC+bFamP*Mfm+bGroP*Mg+bRecP*Mr+bHerP*Mh+bDefP*Md+b

    muS<-aGS[G]+bAS*A+aCS[CD]+bConS*CC+bFamS*Mfm+bGroS*Mg+bRecS*Mr+bHerS*Mh+bDefS*Md+b

    #Priors
    #Precaution
    aGP[G]~dnorm(0,0.2),
    bAP~dnorm(0,0.5),

    aCP[CD]~dnorm(0,0.2),

    bConP~dnorm(0,0.5),
    bFamP~dnorm(0,0.5),
    bGroP~dnorm(0,0.5),
    bRecP~dnorm(0,0.5),
    bHerP~dnorm(0,0.5),
    bDefP~dnorm(0,0.5),
    bFaiP~dnorm(0,0.5),
    bProP~dnorm(0,0.5),

    bDgP~dnorm(0,0.5),

    #ProSociality
    aGS[G]~dnorm(0,0.2),
    bAS~dnorm(0,0.5),

    aCS[CD]~dnorm(0,0.2),

    bConS~dnorm(0,0.5),
    bFamS~dnorm(0,0.5),
    bGroS~dnorm(0,0.5),
    bRecS~dnorm(0,0.5),
    bHerS~dnorm(0,0.5),
    bDefS~dnorm(0,0.5),
    bFaiS~dnorm(0,0.5),
    bProS~dnorm(0,0.5),

    bDgS~dnorm(0,0.5),

    #sigmas
    sigmaD~dexp(1),
    sigmaP~dexp(1),
    sigmaS~dexp(1),

    #Models of MAC dimensions
    Mfm ~ dnorm(mu_Fam,sigma_Fam),
    Mg ~ dnorm(mu_Gro,sigma_Gro),
    Mr ~ dnorm(mu_Rec,sigma_Rec),
    Mh ~ dnorm(mu_Her,sigma_Her),
    Md ~ dnorm(mu_Def,sigma_Def),
    Mfi ~ dnorm(mu_Fai,sigma_Fai),
    Mp ~ dnorm(mu_Pro,sigma_Pro),

    mu_Fam<-aG_Fam[G]+bAge_Fam*A,
    mu_Gro<-aG_Gro[G]+bAge_Gro*A,
    mu_Rec<-aG_Rec[G]+bAge_Rec*A,
    mu_Her<-aG_Her[G]+bAge_Her*A,
    mu_Def<-aG_Def[G]+bAge_Def*A,
    mu_Fai<-aG_Fai[G]+bAge_Fai*A,
    mu_Pro<-aG_Pro[G]+bAge_Pro*A,

    #priors of MAC intercepts and slopes
    aG_Fam[G]~dnorm(0,0.2),
    aG_Gro[G]~dnorm(0,0.2),
    aG_Rec[G]~dnorm(0,0.2),
    aG_Her[G]~dnorm(0,0.2),
    aG_Def[G]~dnorm(0,0.2),
    aG_Fai[G]~dnorm(0,0.2),
    aG_Pro[G]~dnorm(0,0.2),

    bAge_Fam~dnorm(0,0.5),
    bAge_Gro~dnorm(0,0.5),
    bAge_Rec~dnorm(0,0.5),
    bAge_Her~dnorm(0,0.5),
    bAge_Def~dnorm(0,0.5),
    bAge_Fai~dnorm(0,0.5),
    bAge_Pro~dnorm(0,0.5),

    #sigmas
    sigma_Fam~dexp(1),
    sigma_Gro~dexp(1),
    sigma_Rec~dexp(1),
    sigma_Her~dexp(1),
    sigma_Def~dexp(1),
    sigma_Fai~dexp(1),
    sigma_Pro~dexp(1),

    #model of how dangerous COVID is perceived for the participant
    Dgr ~ dnorm(mu_Dang,sigma_Dang),
    mu_Dang<-aG_Dang[G]+bAge_Dang*A,
    #priors
    aG_Dang[G]~dnorm(0,0.2),
    bAge_Dang~dnorm(0,0.5),

    sigma_Dang~dexp(1)

    ) , data=dat, chains=4 , cores=4 , log_lik=TRUE ,iter = 5000,control=list(max_treed

#Sumarize the model
precis(m1,depth=2)

#Sample posetrior and prior for graphical comparison
post1<-extract.samples(m1)

set.seed(42)
prio1<-extract.prior(m1,n=10000)

save.image(file="posterior_samples_single.RData")
```

While running the cod ewe come across a few problems and bugs we need to fix.

the first problem is that R didn't recognize rethinking as a package, I found a person on Stackoverflow with the same problem and he recommended me this:

```
install.packages("rethinking",
                 repos=c(cran="https://cloud.r-project.org",
                         rethinking="http://xcelab.net/R"))
```

The second problem was that Rstudio couldn't find the function "standerdize". A search on google gave me this:.

```
if(!require('robustHD')) {
  install.packages('robustHD')
  library('robustHD')
}
```

The next problem is that R didn't recognize the function "ulam". This is weird since it is part of the "rethinking" package. i tried fixing it with the code below, but R doesn't recognize "rethinking" as a package. R cant recognize this package because there is no version of this package available for the version of R that's been used for this analyse.

I tried something else and i found a quick installion for the 'rethinking' package. This worked and the function "ulam" got recognized.

```
devtools::install_github("stan-dev/cmdstanr")

install.packages(c("coda","mvtnorm","devtools","loo","dagitty"))
```

```
devtools::install_github("rmcelreath/rethinking")
```

The next struggle we come across is that the CmdStan path has not been set. To fix this we have to give R the location of the CmdStan installion. To find the location we use the "cmdstan_default_install_path()" function. After i set the path with set_cmdstan_path, R couldn't find the directory. Unfortnetly, this problem i couln't fix.

```
library(rethinking)

if(!require('robustHD')) {
  install.packages('robustHD')
  library('robustHD')
}


d<-read.table("data.clean.txt",sep="\t",header=T,stringsAsFactors = F)
names(d)

dat <- list(
  C=d$CountC,

  A = standardize(d$Age),
  G = as.integer(as.factor(d$Gender)),  #1 Man, 2 Woman
  CD = as.integer(d$COND),
  Mfm = standardize(d$MAC.fam),
  Mg = standardize(d$MAC.gro),
  Mr = standardize(d$MAC.rec),
  Mh = standardize(d$MAC.her),
  Md = standardize(d$MAC.def),
  Mfi = standardize(d$MAC.fai),
  Mp = standardize(d$MAC.pro),
  P = standardize(d$Precaution),
  S = standardize(d$Prosociality),
  D = standardize(d$donation),

  Dgr=standardize(d$danger.for.participant),

  CC = ifelse(d$COND==1,0,ifelse(d$COND==2,0,ifelse(d$COND==3,0,ifelse(d$COND==4,standardize(d$MA

)

set.seed(42)
m1 <- ulam(
  set_cmdstan_path("C:/Users/Lamya/OneDrive/Documenten/.cmdstan"),
  alist(
    D ~ dnorm(muD,sigmaD),

    muD<-aG[G]+bA*A+bP*P+bS*S+aC[CD]+bCon*CC+bFam*Mfm+bGro*Mg+bRec*Mr+bHer*Mh+bDef*Md+

    #Donation
    aG[G]~dnorm(0,0.2),
    bA~dnorm(0,0.5),

    bP~dnorm(0,0.5),
    bS~dnorm(0,0.5),

    aC[CD]~dnorm(0,0.2),

    bCon~dnorm(0,0.5),
    bFam~dnorm(0,0.5),
    bGro~dnorm(0,0.5),
    bRec~dnorm(0,0.5),
    bHer~dnorm(0,0.5),
    bDef~dnorm(0,0.5),
    bFai~dnorm(0,0.5),
    bPro~dnorm(0,0.5),

    bDg~dnorm(0,0.5),


    #Model of precaution and prosociality
    P ~ dnorm(muP,sigmaP),
    S ~ dnorm(muS,sigmaS),

    muP<-aGP[G]+bAP*A+aCP[CD]+bConP*CC+bFamP*Mfm+bGroP*Mg+bRecP*Mr+bHerP*Mh+bDefP*Md+b

    muS<-aGS[G]+bAS*A+aCS[CD]+bConS*CC+bFamS*Mfm+bGroS*Mg+bRecS*Mr+bHerS*Mh+bDefS*Md+b

    #Priors
    #Precaution
    aGP[G]~dnorm(0,0.2),
    bAP~dnorm(0,0.5),

    aCP[CD]~dnorm(0,0.2),

    bConP~dnorm(0,0.5),
    bFamP~dnorm(0,0.5),
    bGroP~dnorm(0,0.5),
    bRecP~dnorm(0,0.5),
    bHerP~dnorm(0,0.5),
    bDefP~dnorm(0,0.5),
    bFaiP~dnorm(0,0.5),
    bProP~dnorm(0,0.5),

    bDgP~dnorm(0,0.5),

    #ProSociality
    aGS[G]~dnorm(0,0.2),
    bAS~dnorm(0,0.5),

    aCS[CD]~dnorm(0,0.2),

    bConS~dnorm(0,0.5),
    bFamS~dnorm(0,0.5),
    bGroS~dnorm(0,0.5),
    bRecS~dnorm(0,0.5),
    bHerS~dnorm(0,0.5),
    bDefS~dnorm(0,0.5),
    bFaiS~dnorm(0,0.5),
    bProS~dnorm(0,0.5),

    bDgS~dnorm(0,0.5),

    #sigmas
    sigmaD~dexp(1),
    sigmaP~dexp(1),
    sigmaS~dexp(1),

    #Models of MAC dimensions
    Mfm ~ dnorm(mu_Fam,sigma_Fam),
    Mg ~ dnorm(mu_Gro,sigma_Gro),
    Mr ~ dnorm(mu_Rec,sigma_Rec),
    Mh ~ dnorm(mu_Her,sigma_Her),
    Md ~ dnorm(mu_Def,sigma_Def),
    Mfi ~ dnorm(mu_Fai,sigma_Fai),
    Mp ~ dnorm(mu_Pro,sigma_Pro),

    mu_Fam<-aG_Fam[G]+bAge_Fam*A,
    mu_Gro<-aG_Gro[G]+bAge_Gro*A,
    mu_Rec<-aG_Rec[G]+bAge_Rec*A,
    mu_Her<-aG_Her[G]+bAge_Her*A,
    mu_Def<-aG_Def[G]+bAge_Def*A,
    mu_Fai<-aG_Fai[G]+bAge_Fai*A,
    mu_Pro<-aG_Pro[G]+bAge_Pro*A,
    #priors of MAC intercepts and slopes
    aG_Fam[G]~dnorm(0,0.2),
    aG_Gro[G]~dnorm(0,0.2),
    aG_Rec[G]~dnorm(0,0.2),
    aG_Her[G]~dnorm(0,0.2),
    aG_Def[G]~dnorm(0,0.2),
    aG_Fai[G]~dnorm(0,0.2),
    aG_Pro[G]~dnorm(0,0.2),

    bAge_Fam~dnorm(0,0.5),
    bAge_Gro~dnorm(0,0.5),
    bAge_Rec~dnorm(0,0.5),
    bAge_Her~dnorm(0,0.5),
    bAge_Def~dnorm(0,0.5),
    bAge_Fai~dnorm(0,0.5),
    bAge_Pro~dnorm(0,0.5),

    #sigmas
    sigma_Fam~dexp(1),
    sigma_Gro~dexp(1),
    sigma_Rec~dexp(1),
    sigma_Her~dexp(1),
    sigma_Def~dexp(1),
    sigma_Fai~dexp(1),
    sigma_Pro~dexp(1),

    #model of how dangerous COVID is perceived for the participant
    Dgr ~ dnorm(mu_Dang,sigma_Dang),
    mu_Dang<-aG_Dang[G]+bAge_Dang*A,

    #priors
    aG_Dang[G]~dnorm(0,0.2),
    bAge_Dang~dnorm(0,0.5),

    sigma_Dang~dexp(1)

    ) , data=dat, chains=4 , cores=4 , log_lik=TRUE ,iter = 5000,control=list(max_treed

#Sumarize the model
precis(m1,depth=2)

#Sample posetrior and prior for graphical comparison
post1<-extract.samples(m1)

set.seed(42)
```

```
prio1<-extract.prior(m1,n=10000)

save.image(file="posterior_samples_single.RData")
```

o. From a scale from 1 (very hard) to 5(very easy). This script gets a 1. It was very hard to read the code without any pseudo coding. It's hard to see what the author wants to achieve with this code. Running the code wasn't an easy job either. A lot of packages needed to be installed in a different way than what the author did. In conclusion, this code is not really reproducible.

Chapter 4

The Guerrila structure

4.1 Why organizing data?

Organizing data is crucial for reproducibility and a good workflow in general. Below us we find an example of a folder containing multiple projects i made. This dataset is orginized by the Guerrilla Analytics Principles. Each project contains a folder "data" and "output". And each folder contains a "README" file about the folder and data. Each "data" folder contains a "rawdata" folder. In this folder is the file with the rawdata. This data will not be used during the experiment, instead we have a copy of the dataset in "data" to prevent errors and false data from happening.

```
## C:/Users/Lamya/OneDrive/Documenten/dsfb2_workflows_portfolio/portofolio_bookdown/Guerrila_stru
## +-- metagenomics
## |   +-- metagenomics_formatief
## |   |   +-- data
## |   |   |   +-- HU2_MOCK2_L001_R1_001.fastq.gz
## |   |   |   +-- HU2_MOCK2_L001_R2_001.fastq.gz
## |   |   |   +-- HU_waternet_MOCK2_composition_copy.csv
## |   |   |   +-- rawdata
## |   |   |   |   \-- HU_waternet_MOCK2_composition.csv
## |   |   |   \-- README.txt
## |   |   +-- output
## |   |   |   +-- metagenomics_formatief.html
## |   |   |   \-- metagenomics_formatief.Rmd
## |   |   \-- README.txt
## |   +-- metagenomics_reader
## |   |   +-- data
## |   |   |   +-- HU1_MOCK1_L001_R1_001.fastq.gz
## |   |   |   +-- HU1_MOCK1_L001_R2_001.fastq.gz
## |   |   |   +-- HU_waternet_MOCK1_composition_copy.csv
## |   |   |   +-- rawdata
## |   |   |   |   +-- HU_waternet_MOCK1_composition.csv
## |   |   |   |   \-- README.txt
## |   |   |   \-- README.txt
## |   |   +-- output
## |   |   |   +-- metagenomics_reader.html
## |   |   |   \-- metagenomics_reader.Rmd
## |   |   \-- README.txt
## |   \-- README.txt
## \-- rna_sequencing
##     +-- README.txt
##     +-- rnaseq_airway
##     |   +-- data
##     |   |   +-- airway_sampledata_copy.csv
##     |   |   +-- bam
##     |   |   |   +-- READMe.txt
##     |   |   |   +-- SRR1039508.bam
##     |   |   |   +-- SRR1039508.bam.indel.vcf
##     |   |   |   \-- SRR1039508.bam.summary
##     |   |   +-- counts
##     |   |   |   +-- README.txt
##     |   |   |   \-- read_counts.rds
##     |   |   +-- fastq
##     |   |   |   +-- SRR1039516_1.fastq.gz
##     |   |   |   +-- SRR1039516_2.fastq.gz
##     |   |   |   +-- SRR1039517_1.fastq.gz
##     |   |   |   \-- SRR1039520_1.fastq.gz
##     |   |   +-- rawdata
##     |   |   |   \-- airway_sampledata.csv
##     |   |   \-- README.txt
##     |   +-- output
##     |   |   \-- fastqc_output
##     |   |       +-- SRR1039508_1_fastqc.html
##     |   |       +-- SRR1039508_1_fastqc.zip
##     |   |       +-- SRR1039508_2_fastqc.html
##     |   |       \-- SRR1039508_2_fastqc.zip
##     |   \-- README.txt
##     +-- rnaseq_ipsc
##     |   +-- data
##     |   |   +-- bam
##     |   |   |   +-- SRR7866687.bam
##     |   |   |   +-- SRR7866687.bam.indel.vcf
##     |   |   |   \-- SRR7866687.bam.summary
##     |   |   +-- counts
##     |   |   |   \-- read_counts.rds
##     |   |   +-- fastq
##     |   |   |   +-- SRR7866693_1.fastq.gz
##     |   |   |   +-- SRR7866693_2.fastq.gz
##     |   |   |   +-- SRR7866694_1.fastq.gz
##     |   |   |   \-- SRR7866694_2.fastq.gz
##     |   |   +-- ipsc_sampledata_copy.csv
##     |   |   +-- rawdata
##     |   |   |   \-- ipsc_sampledata.csv
##     |   |   \-- README.txt
##     |   +-- output
##     |   |   +-- fastqc_output
##     |   |   |   +-- README.txt
##     |   |   |   +-- SRR7866687_1_fastqc.html
##     |   |   |   +-- SRR7866687_1_fastqc.zip
##     |   |   |   +-- SRR7866687_2_fastqc.html
##     |   |   |   \-- SRR7866687_2_fastqc.zip
##     |   |   \-- README.txt
##     |   \-- README.txt
##     \-- rnaseq_onecut
##         +-- data
##         |   +-- bam
##         |   |   +-- SRR7866699.bam
##         |   |   +-- SRR7866699.bam.indel.vcf
##         |   |   \-- SRR7866699.bam.summary
##         |   +-- counts
##         |   |   \-- read_counts_OC3.rds
##         |   +-- fastqc
##         |   |   +-- SRR7866703_1.fastq.gz
##         |   |   +-- SRR7866703_2.fastq.gz
##         |   |   +-- SRR7866704_1.fastq.gz
##         |   |   +-- SRR7866704_2.fastq.gz
##         |   |   +-- SRR7866705_1.fastq.gz
##         |   |   \-- SRR7866706_2.fastq.gz
##         |   +-- onecut_sampledata_OC3_copy.csv
##         |   +-- rawdata
##         |   |   \-- onecut_sampledata_OC3.csv
##         |   \-- README.txt
##         +-- output
##         \-- README.txt
```

Chapter 5

Western Blot analysis using Rstudio

5.1 Introduction

Western Blotting (WB) is a commonly used method in the biological scienes. This technique is used to investigate many features of the protein, ranging from basic protein analysis to disease detection such as cancer and HIV (https://www.future-science.com/doi/10.2144/btn-2022-0003). Western Blotting involves three components to achieve this goal: (1) separating based on size, (2) transferring onto a solid surface, and (3) labeling the protein of interest with specific primary and secondary antibodies for visualization (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3456489/). This results in a gel full of different proteins separated by size. ImageJ is often used to analyze the Western Blot results. A lane is

The technique is highly sensitive and can be used to detect even small amounts of protein. It is often used in research into diseases such as cancer and HIV, where specific proteins can be used as markers for diagnosis and treatment. The next skill im gonna learn is to analyse Western Blots using Rstudio.

5.2 planning

The first thing i need to do is to read about Western blot analysis in Rstudio. I need to look if people made packages and if they succeed. Most of the Western Blot analysis is done with ImageJ, which makes my search for Rcodes a bit difficultier.

- Find articles about photo analysis and Western blot analysis in Rstudio.

- Look for codes that analysist used for Western Blot results.
- Try codes on Western blot photo's.
-

Chapter 6

My CV

hier komt mijn cv