Nanodrop_data_management

Fay

2023-03-26

Libraries

```
# install libraries
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.2.1
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(XML)
```

```
## Warning: package 'XML' was built under R version 4.2.3
```

```
library(methods)
library(plyr)
```

```
## Warning: package 'plyr' was built under R version 4.2.3
```

```
## ------------------------------------------------------------------------------
```

```
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
```

```
## ------------------------------------------------------------------------------
```

```
##
## Attaching package: 'plyr'
```

```
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
library(readr)
```

```
## Warning: package 'readr' was built under R version 4.2.1
```

```
library(ggplot2)
```

Read the tsv nanodrop and write them as csv

David

```
#read tsv table
David <- read_tsv("~/GitHub/Namibia_project/Data/Nanodrop_measurements/David_Nanodrop_DNA-Extractions.ts

# remove ffirst column
David <-  David[,-1]

#add column sample type
David <- David %>%
    mutate(animal = "David")

#change the column names

write.csv(David,
          "~/GitHub/Namibia_project/Data/Nanodrop_measurements/CSV/David.csv",
          row.names = FALSE)
```

Düppel

```
#read tsv table
Duppel_1 <- read_tsv("~/GitHub/Namibia_project/Data/Nanodrop_measurements/Düppel_26022023_2.tsv")

# remove ffirst column
Duppel_1 <-  Duppel_1[,-1]


#change the column names

write.csv(Duppel_1,
          "~/GitHub/Namibia_project/Data/Nanodrop_measurements/CSV/Düppel_26022023.csv",
          row.names = FALSE)
```

Rodents

```
#read tsv table
Rodent_1 <- read_tsv("~/GitHub/Namibia_project/Data/Nanodrop_measurements/Rodents_26032023.tsv")
Rodent_2 <- read_tsv("~/GitHub/Namibia_project/Data/Nanodrop_measurements/Rodents_27032023.tsv")

#combine the measuerements
Rodent <- rbind(Rodent_1,Rodent_2)

# remove first column
Rodent <-  Rodent[,-1]

#change the column names

write.csv(Rodent_1,
          "~/GitHub/Namibia_project/Data/Nanodrop_measurements/CSV/Rodents_26032023.csv",
          row.names = FALSE)
```

Merging the tables

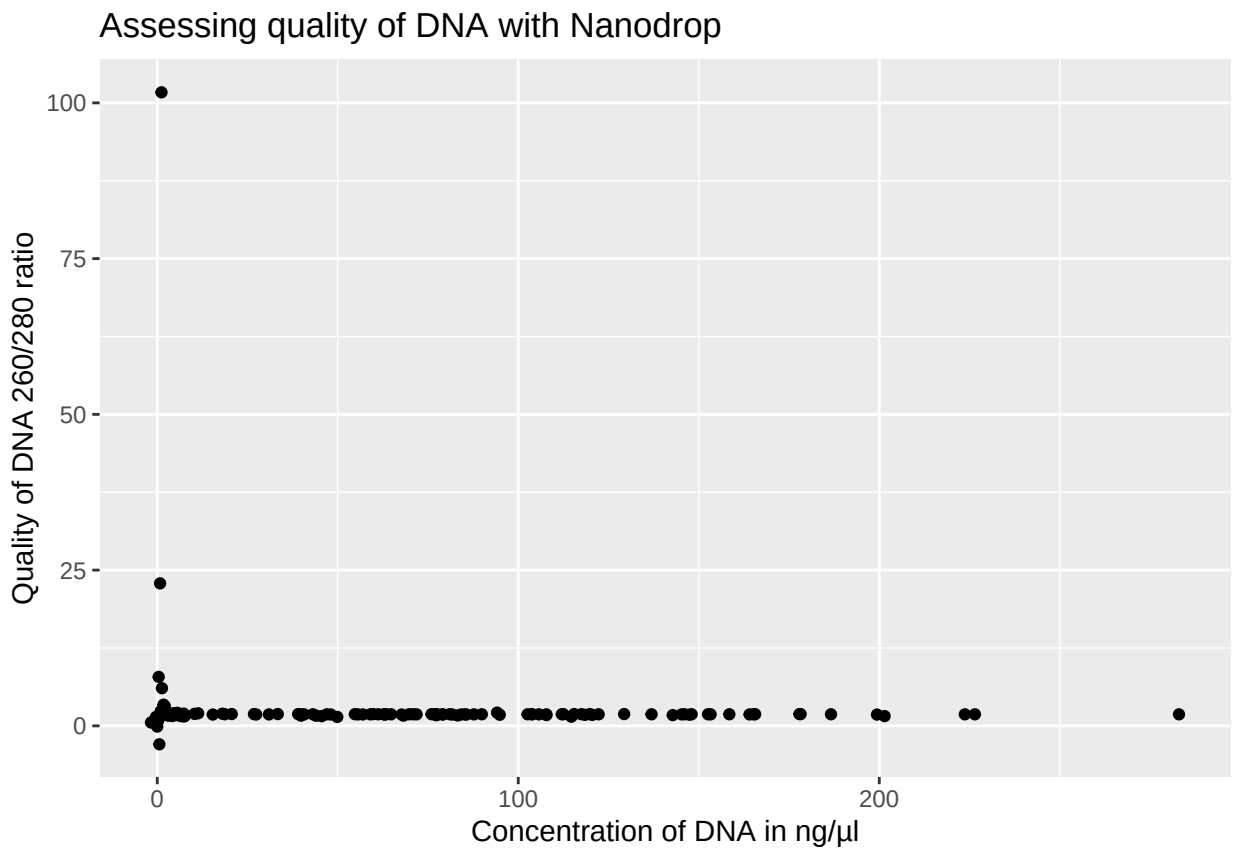
```
Nanodrop <- rbind(Duppel_1, Rodent)


# remove spaces
colnames(Nanodrop) <- gsub(" ", "_", colnames(Nanodrop))


Nanodrop <- Nanodrop %>%
    dplyr::rename(Quality_260_280 = '260/280',
                  Quality_260_230 = '260/230')
```

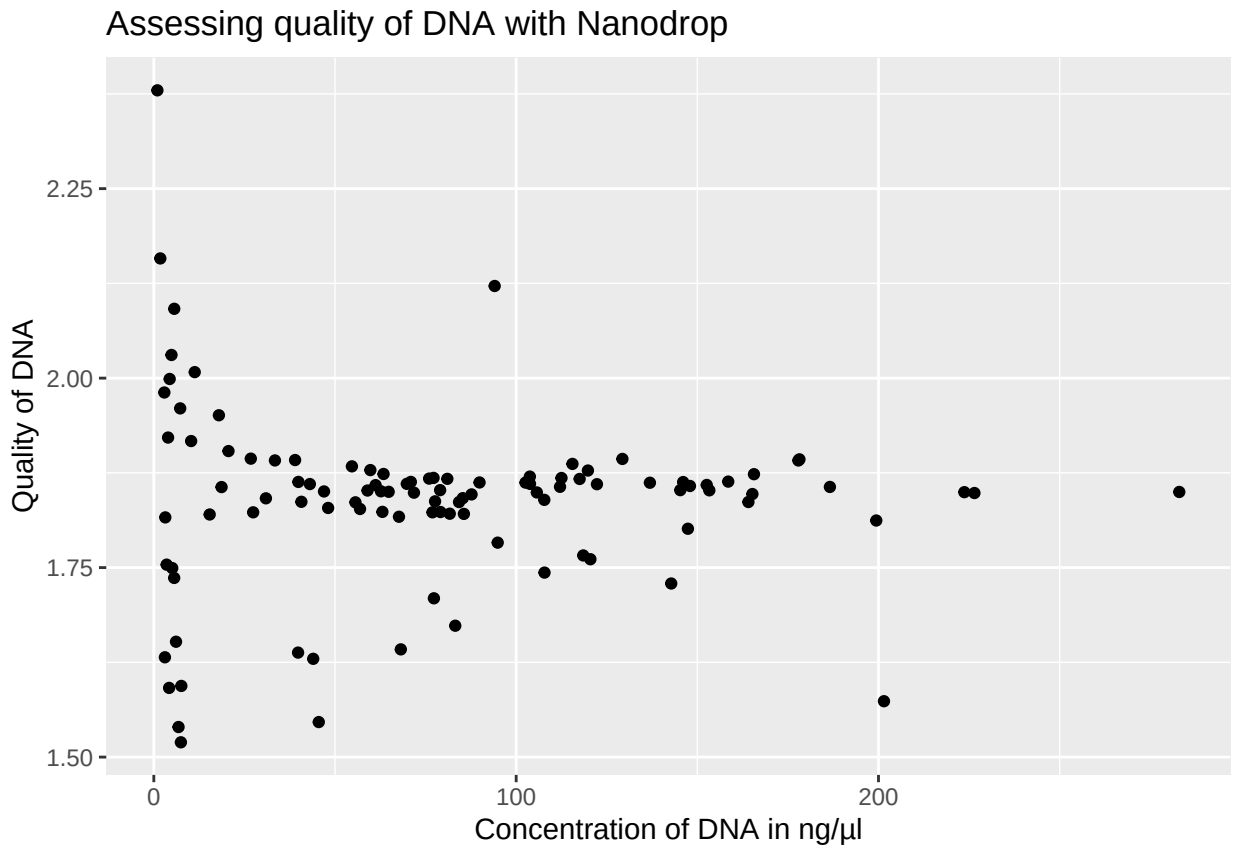
Assessing the quality of the DNA

```
ggplot(Nanodrop, aes(x = Nucleic_Acid, y = Quality_260_280)) +
    geom_jitter() +
    labs(x = "Concentration of DNA in ng/µl", y = "Quality of DNA 260/280 ratio",
         title = "Assessing quality of DNA with Nanodrop")
```



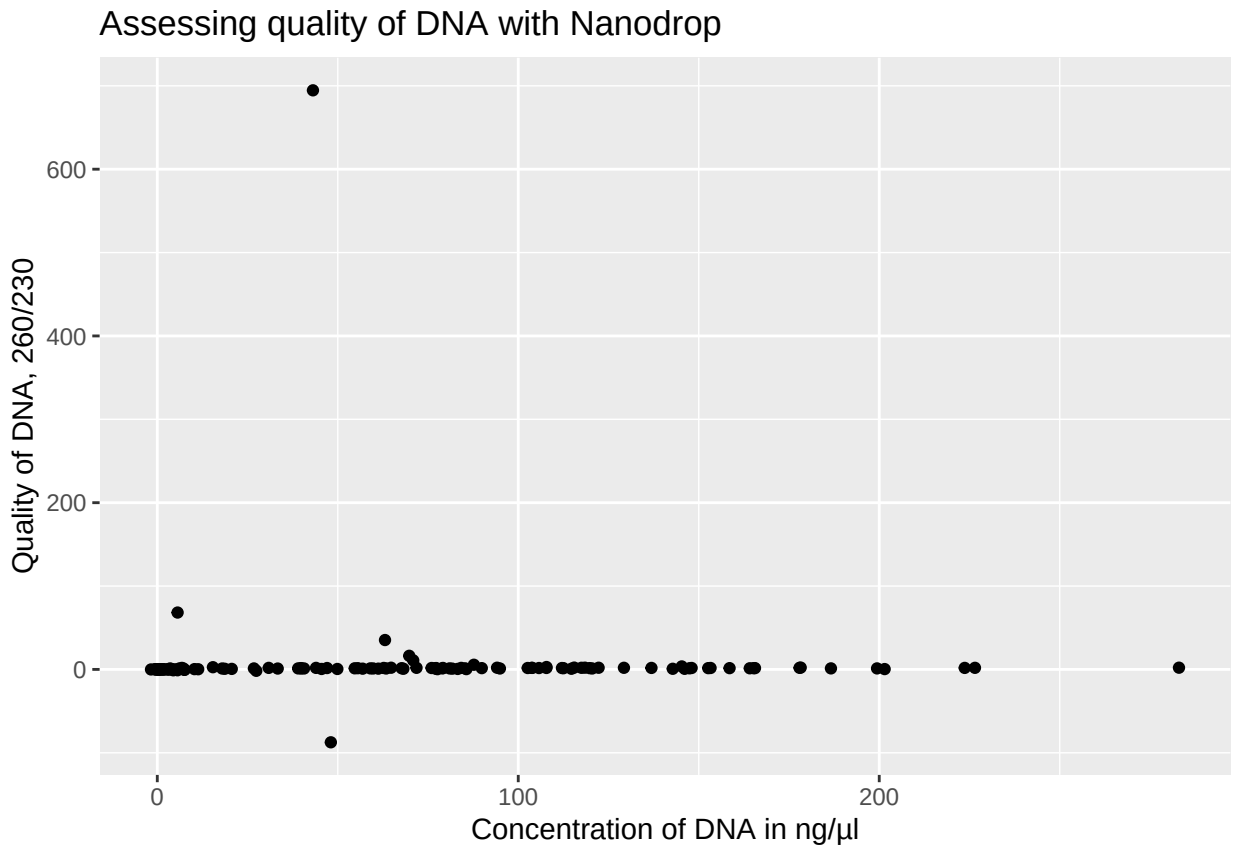
Let's filter out the samples with eextremely high ratios

```
ggplot(Nanodrop %>%
    filter(Quality_260_280 < 2.5,  Quality_260_280 > 1.5 ),
    aes(x = Nucleic_Acid, y = Quality_260_280)) +
    geom_jitter() +
    labs(x = "Concentration of DNA in ng/µl", y = "Quality of DNA",
         title = "Assessing quality of DNA with Nanodrop")
```

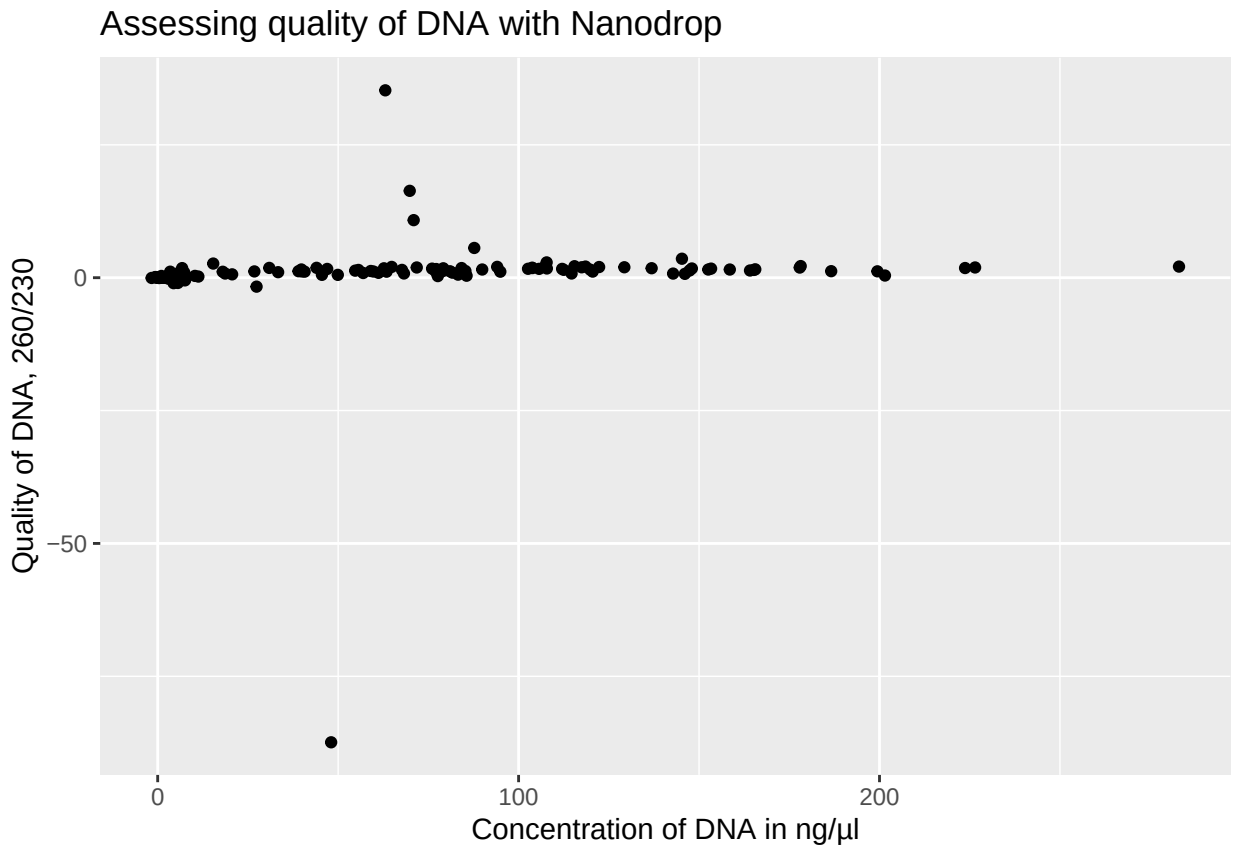


Assesing the quality of 260/230

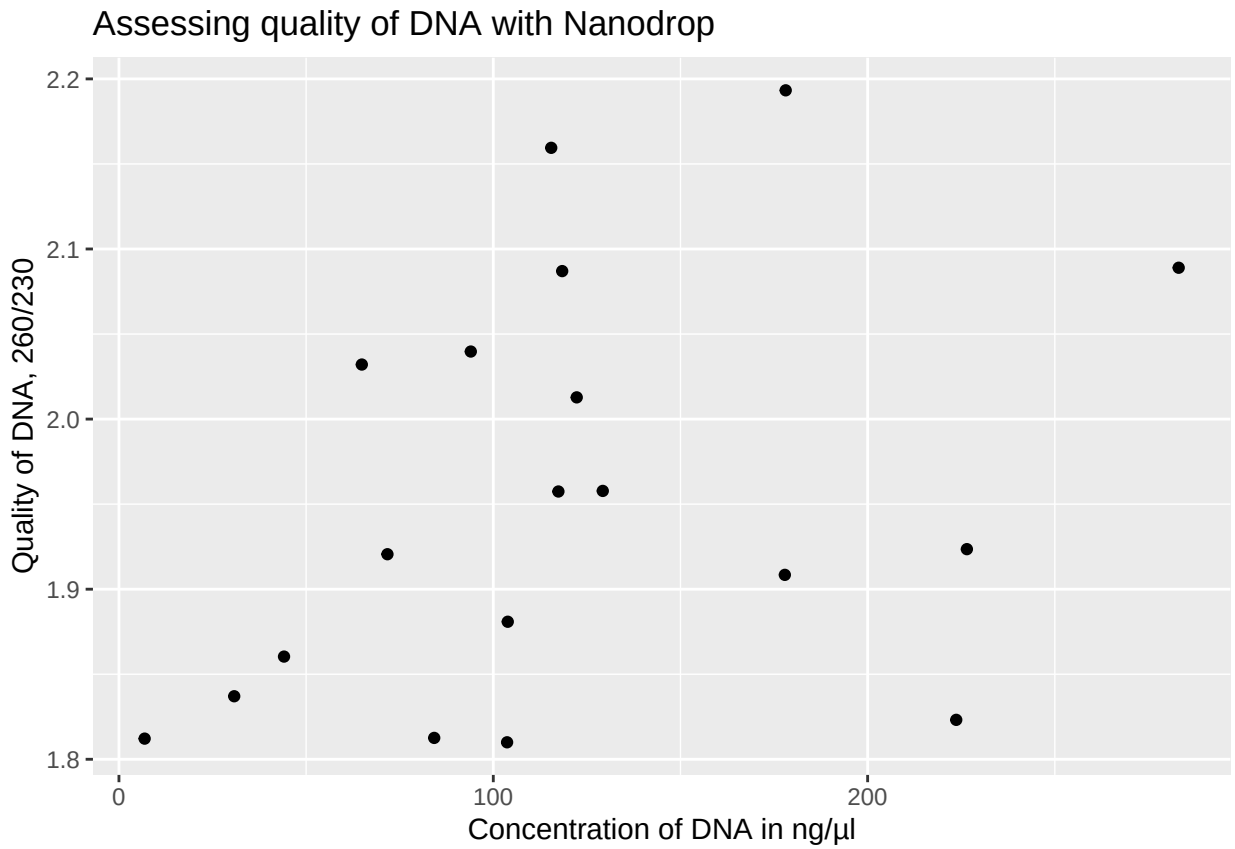
```
ggplot(Nanodrop,
    aes(x = Nucleic_Acid, y = Quality_260_230)) +
    geom_jitter() +
    labs(x = "Concentration of DNA in ng/µl", y = "Quality of DNA, 260/230",
         title = "Assessing quality of DNA with Nanodrop")
```




```
ggplot(Nanodrop %>% filter(Quality_260_230 < 60),
    aes(x = Nucleic_Acid, y = Quality_260_230)) +
    geom_jitter() +
    labs(x = "Concentration of DNA in ng/µl", y = "Quality of DNA, 260/230",
         title = "Assessing quality of DNA with Nanodrop")
```




```
ggplot(Nanodrop %>%
    filter(Quality_260_230 < 2.5, Quality_260_230 > 1.8),
    aes(x = Nucleic_Acid, y = Quality_260_230)) +
    geom_jitter() +
    labs(x = "Concentration of DNA in ng/µl", y = "Quality of DNA, 260/230",
         title = "Assessing quality of DNA with Nanodrop")
```



Select the samples having the golden ratios

```
Nanodrop %>%
    filter( Quality_260_280 < 2.5,  Quality_260_280 > 1.5 )
```

```
## # A tibble: 104 x 11
##    Sampl~1 User_~2 Date_~3 Nucle~4 Unit  A260_~5 A280_~6 Quali~7 Quali~8 Sampl~9
##    <chr>   <chr>   <chr>     <dbl> <chr>   <dbl>   <dbl>   <dbl>   <dbl> <chr>
##  1 S01     fay_w   26/03/~     1   "ng/~   0.019   0.008    2.38    0.33 DNA
##  2 S11     fay_w   26/03/~     3.1 "ng/~   0.062   0.038    1.63    0.24 DNA
##  3 S12     fay_w   26/03/~    10.3 "ng/~   0.206   0.108    1.92    0.34 DNA
##  4 S28     fay_w   26/03/~     1.8 "ng/~   0.037   0.017    2.16    0.03 DNA
##  5 S42     fay_w   26/03/~    30.9 "ng/~   0.618   0.336    1.84    1.84 DNA
##  6 BW24    fay_w   26/03/~    18.7 "ng/~   0.374   0.201    1.86    0.78 DNA
##  7 BW40    fay_w   26/03/~    63.4 "ng/~   1.27    0.68     1.87    1.13 DNA
##  8 BW44    fay_w   26/03/~    11.3 "ng/~   0.226   0.112    2.01    0.21 DNA
##  9 C01     fay_w   26/03/~     4.4 "ng/~   0.088   0.044    2      -1.03 DNA
## 10 C03     fay_w   26/03/~     6.8 "ng/~   0.135   0.088    1.54    1.81 DNA
## # ... with 94 more rows, 1 more variable: Factor <dbl>, and abbreviated
## #   variable names 1: Sample_ID, 2: User_name, 3: Date_and_Time,
## #   4: Nucleic_Acid, 5: `A260_(Abs)`, 6: `A280_(Abs)`, 7: Quality_260_280,
## #   8: Quality_260_230, 9: Sample_Type
```