water fluoridation ~ fluorosis caries 12 years_2003_2012_2019

Namhuynh

Preapre data from sav file

```
# 12 year-old
data_12 <- read.spss ("~/Library/Mobile Documents/com~apple~CloudDocs/Nam/New project/20 years of water
```

```
## re-encoding from latin1
```

```
data_12a <- data_12[-c(1,15,20,22,23,24 )]
colnames(data_12a)[c(3,13,14,15,16,17,18)] <- c("fluoride_condition","caries","fluorosis_teeth","DEANinc
colnames(data_12a)[c(5:12)] <- c("DT","MT","FT","DMFT","DS","MS","FS","DMFS")

data_12a$code <- as.character(data_12a$code)
data_12a$year <- as.factor(data_12a$year)
firstup <- function(x) {
  x <- tolower(x)
  substr(x, 1, 1) <- toupper(substr(x, 1, 1))
  x
}
data_12a$DEANindex <- firstup(data_12a$DEANindex)
#data_12a$DEANindex[is.na(data_12a$DEANindex)] <- c("Normal")
data_12a$DEANindex <- recode_factor(data_12a$DEANindex, "Normal" = "Normal", "Questinable" = "Questional
                                    "Moderate" = "Moderate", "Serve" = "Severe")

#data_12a$fluoride_condition[is.na(data_12a$fluoride_condition)] <- c("khong fluor hoa")
data_12a$fluoride_condition <- recode_factor(data_12a$fluoride_condition, `khong fluor hoa` = "none_flu
                                             `Fluor hoa khong on dinh` = "unstable_fluoridation",
                                             `Fluor hoa on dinh` = "stable_fluoridation")

data_12a$gender <- recode_factor(data_12a$gender, `NAM` = "male", `NU` = "female")
data_12a$caries <- recode_factor(data_12a$caries,
                      `CO`="TRUE",`KHONG`="FALSE")%>%as.logical()
#data_12a$fluorosis[is.na(data_12a$fluorosis)] <- c("no")
data_12a$fluorosis <- recode_factor(data_12a$fluorosis,
                      `yes`="TRUE",`no`="FALSE")%>%as.logical()
data_12a$age <- c("12")
data_12a$age <- as.factor(data_12a$age)

# There were F+ in 1990
data_12a$fluoride_concentration <-  as.factor(ifelse(data_12a$year == 2012 & data_12a$area == "F+", "0.5
                                                     ifelse(data_12a$year == 2003 & data_12a$area == "F
```

```
# 2019
data_2019 <- read.spss ("~/Library/Mobile Documents/com~apple~CloudDocs/Nam/New project/20 years of wat
```

```
## Warning in read.spss("~/Library/Mobile Documents/com~apple~CloudDocs/Nam/New
## project/20 years of water fluoridation/Data/DATA2019/DATATPHCM1215tuoi2019-
## Phantich-WFProject.sav", : ~/Library/Mobile Documents/com~apple~CloudDocs/Nam/
## New project/20 years of water fluoridation/Data/DATA2019/DATATPHCM1215tuoi2019-
## Phantich-WFProject.sav: Long string value labels record found (record type 7,
## subtype 21), but ignored
```

```
## re-encoding from latin1
```

```
data_2019 <- data_2019[-c(1,3,6)]
colnames(data_2019)[c(4,5,9,10)] <- c("DEANindex","fluorosis_teeth","DMFT","caries")
data_2019$DEANindex <- firstup(data_2019$DEANindex)
data_2019$DEANindex <- recode_factor(data_2019$DEANindex, "Normal" = "Normal", "Questionnable" = "Questi
                                     "Moderate" = "Moderate", "Serve" = "Severe")
data_2019$district <- NA
data_2019$fluoride_condition <- NA
data_2019$year <- 2019
data_2019$year <- as.factor(data_2019$year)
data_2019$fluorosis <- as.factor(ifelse(data_2019$DEANindex == "Normal","FALSE","TRUE"))%>%as.logical()
data_2019$fluoride_concentration <-  as.factor(ifelse(data_2019$area == "F+", "0.5_ppm","0_ppm"))
data_2019 <- data_2019[c(1,16,17,2,6:9,11:14,10,5,4,18,15,19,3,20)]
data_2019$age <- recode_factor(data_2019$age, "12 year-old" = 12, "15 year-old" = 15)
data_2019$gender <- recode_factor(data_2019$gender, "NAM       " = "male","Nam       "="male",
                                  "NU        " = "female","2         "="female")
data_2019$caries <- recode_factor(data_2019$caries,
                      `Yes`="TRUE",`No`="FALSE")%>%as.logical()
data_2019$fluorosis_teeth <- ifelse(data_2019$DEANindex == "Normal",0,data_2019$fluorosis_teeth)

# 12 year old, 2019
data_12_2019 <- subset(data_2019, data_2019$age == "12")

# merge data
data <- rbind(data_12a, data_12_2019)
```

bar_missing

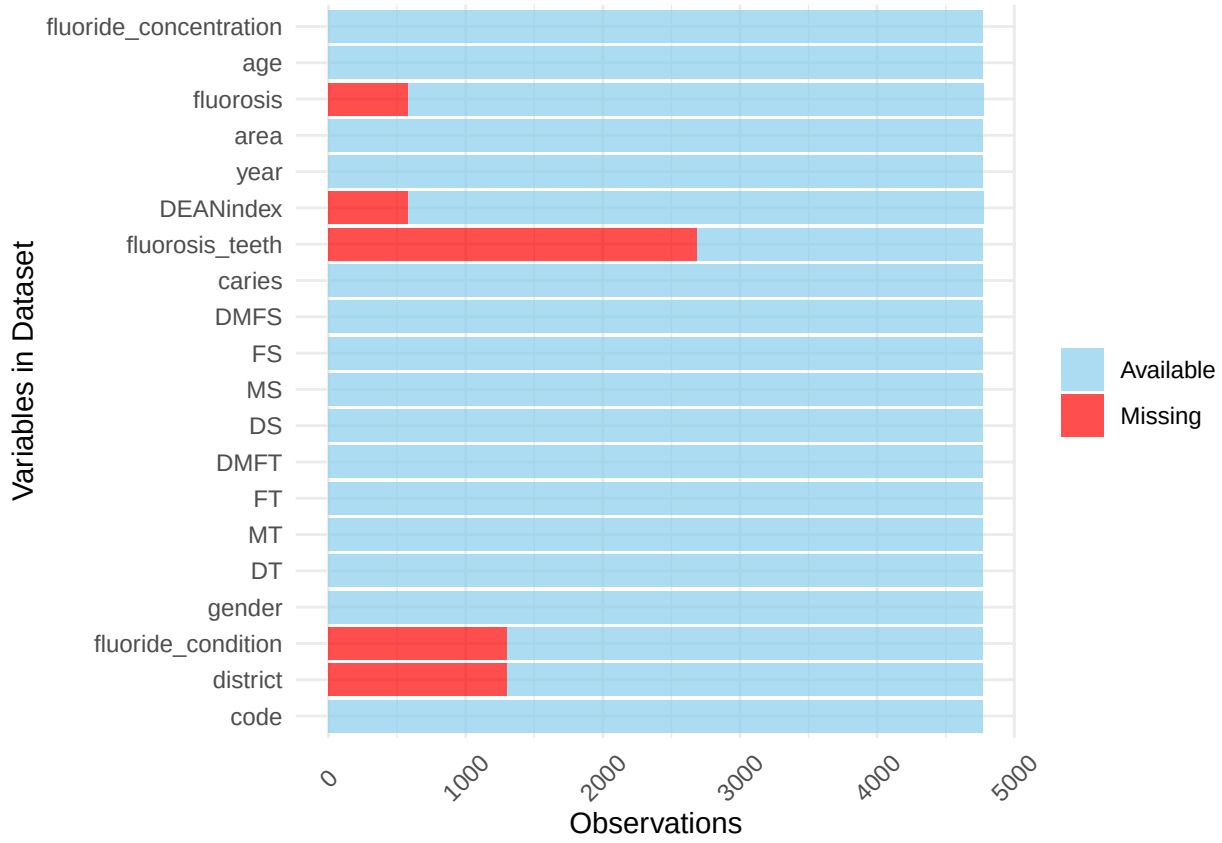
```
bar_missing <- function(x){
  require(reshape2)
  x %>%
    is.na %>%
    melt %>%
    ggplot(data = .,
           aes(x = Var2)) +
    geom_bar(aes(y=(..count..),fill=value),alpha=0.7)+
    scale_fill_manual(values=c("skyblue","red"),
                      name = "",
```

```
                      labels = c("Available","Missing"))+
    theme_minimal()+
    theme(axis.text.x = element_text(angle=45, vjust=0.5)) +
    labs(x = "Variables in Dataset",
         y = "Observations")+coord_flip()
}

bar_missing(data)
```

```
## Loading required package: reshape2
```

```
##
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
##
##     smiths
```



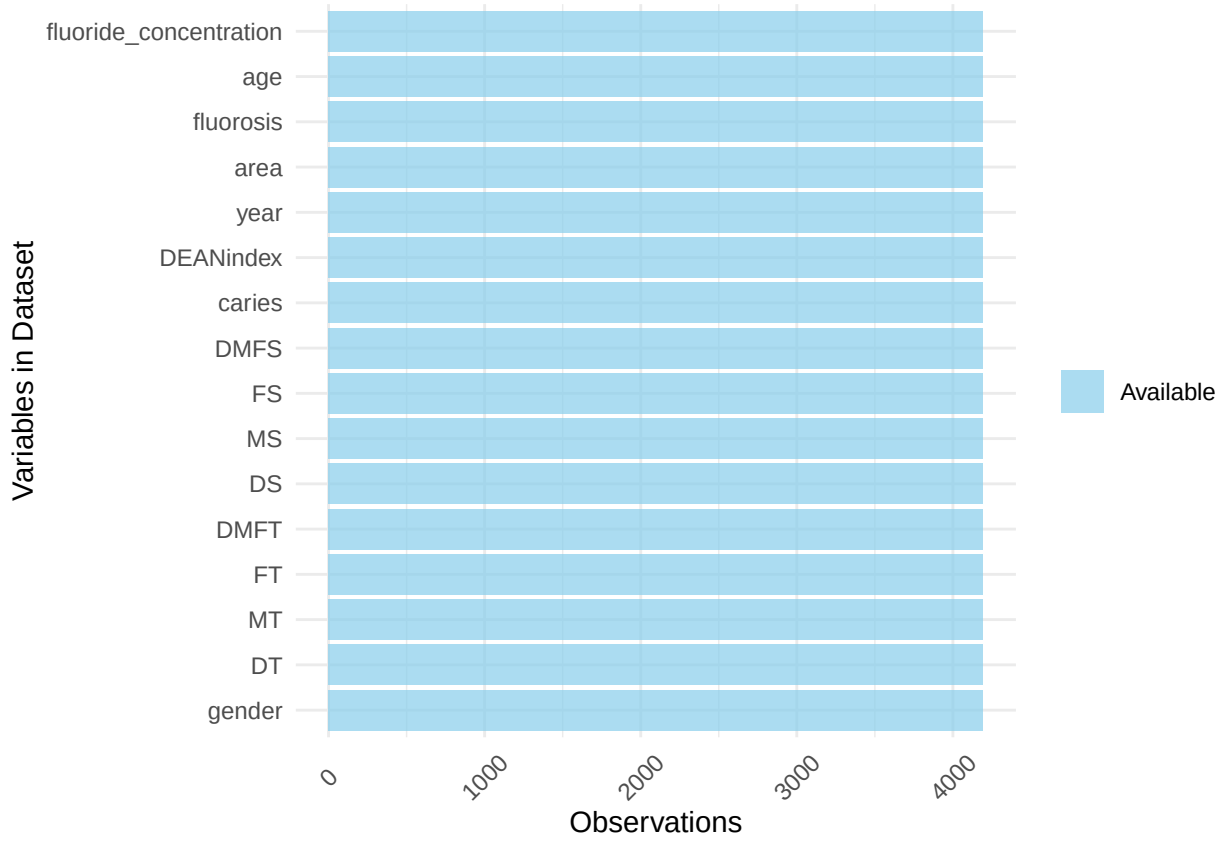
Prepare data w/o NA or unnecessary variables

percent <- list() for (i in 1:ncol(data_b)) { percent[[i]] <- tabyl(data_b[[i]], sort = F) print(i) print(percent[[i]]) }

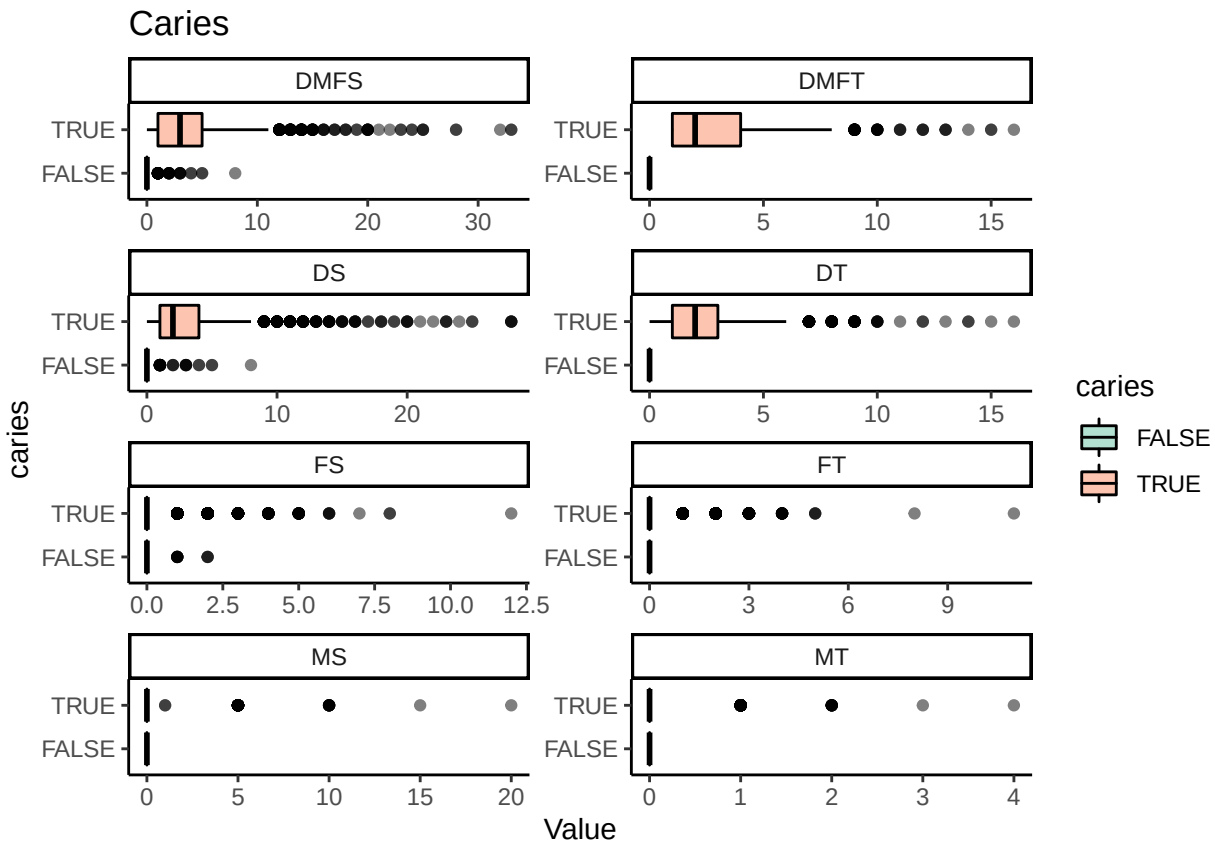
```
library(janitor)
```

```
##
## Attaching package: 'janitor'

## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

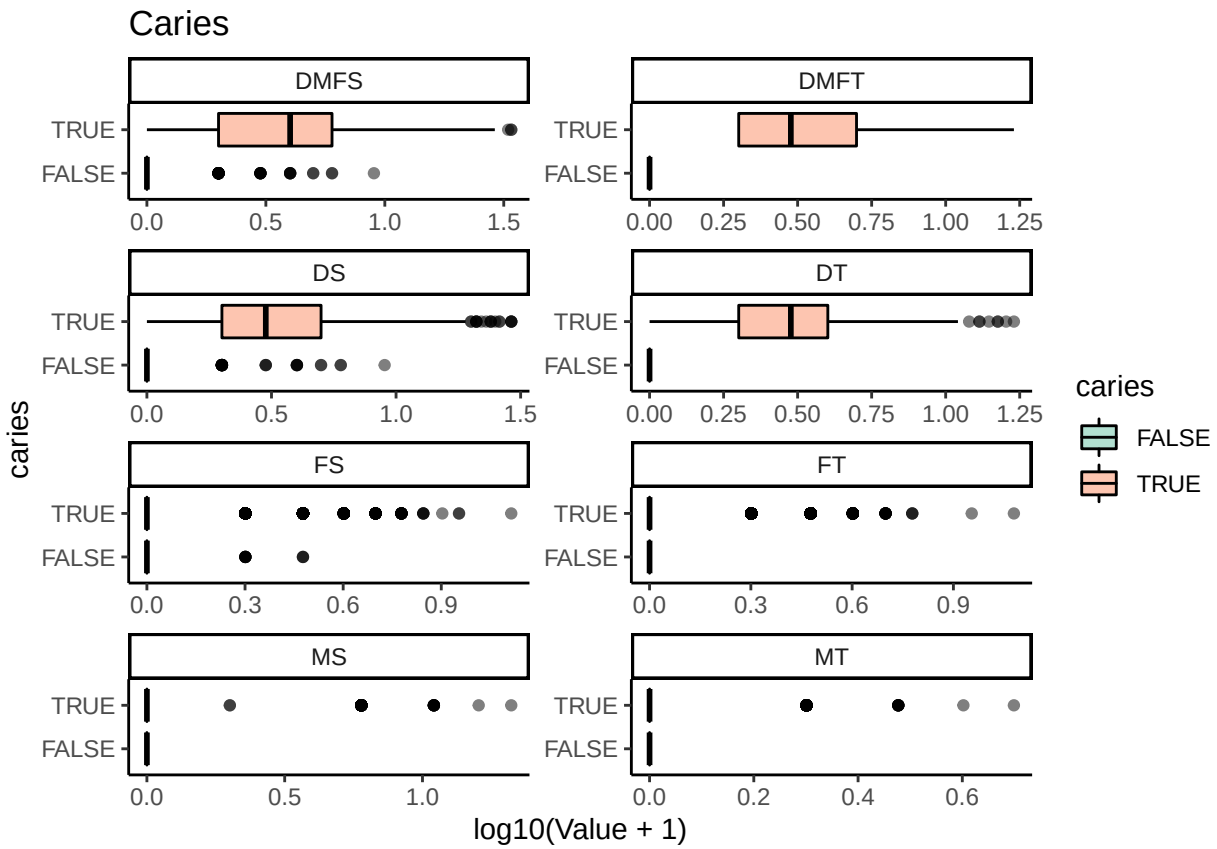
```
library(RColorBrewer)

#remove 1990
data_b <- data[-c(1:3,14)]
data_b <- subset(data_b, !data_b$year == 1990)
bar_missing(data_b)
```



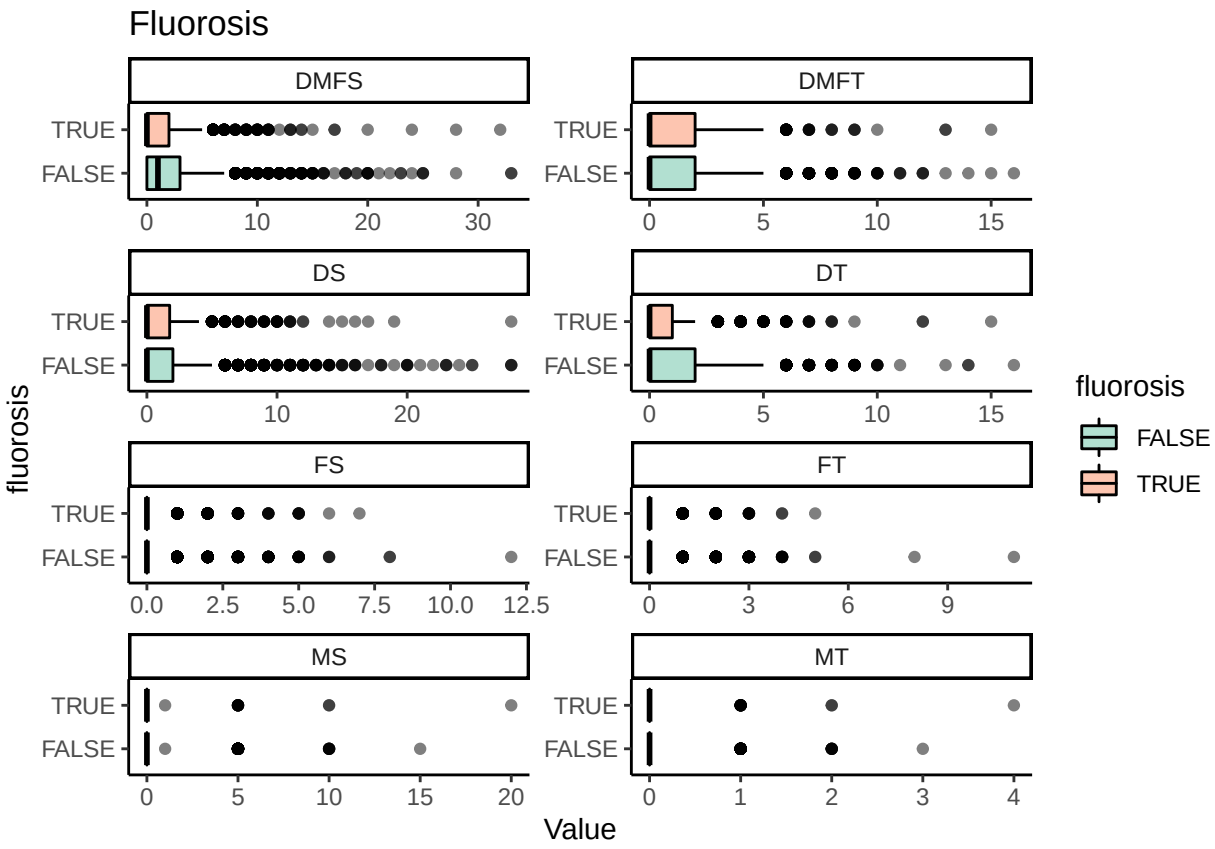

```
data_b %>%gather(c(DT:DMFS),
                 key = "Parameter",
                 value="Value")%>%
  ggplot(aes(x=caries, y=Value,fill=caries))+
  geom_boxplot(alpha=0.5,col="black")+
  ggtitle("Caries")+
  facet_wrap(~Parameter,ncol=2,scales = "free")+
  coord_flip()+
  scale_fill_manual(values = brewer.pal(n = 4, name = "Set2"))+ theme_classic()
```



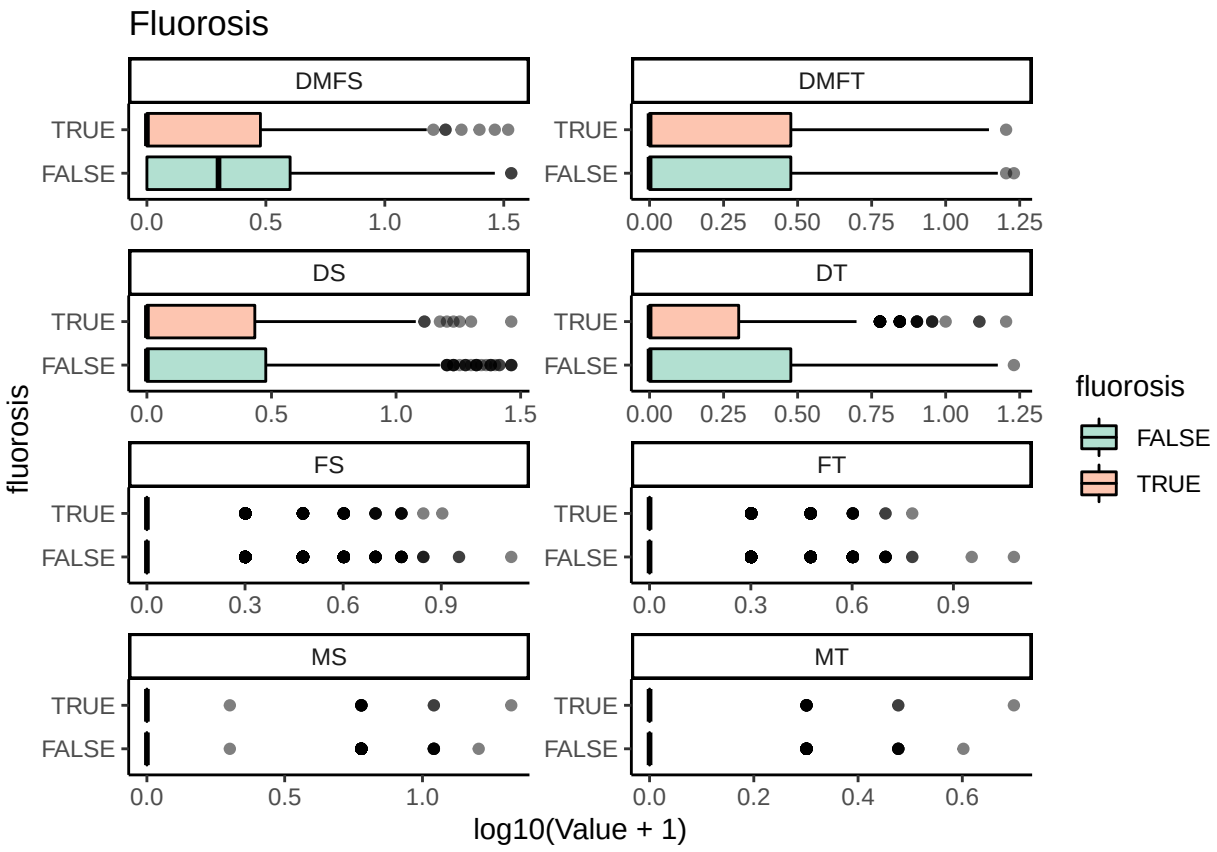

```
#transformation
data_b %>%gather(c(DT:DMFS),
                 key = "Parameter",
                 value="Value")%>%
  ggplot(aes(x=caries, y=log10(Value+1),fill=caries))+
  geom_boxplot(alpha=0.5,col="black")+
  ggtitle("Caries")+
  facet_wrap(~Parameter,ncol=2,scales = "free")+
  coord_flip()+
  scale_fill_manual(values = brewer.pal(n = 4, name = "Set2"))+ theme_classic()
```



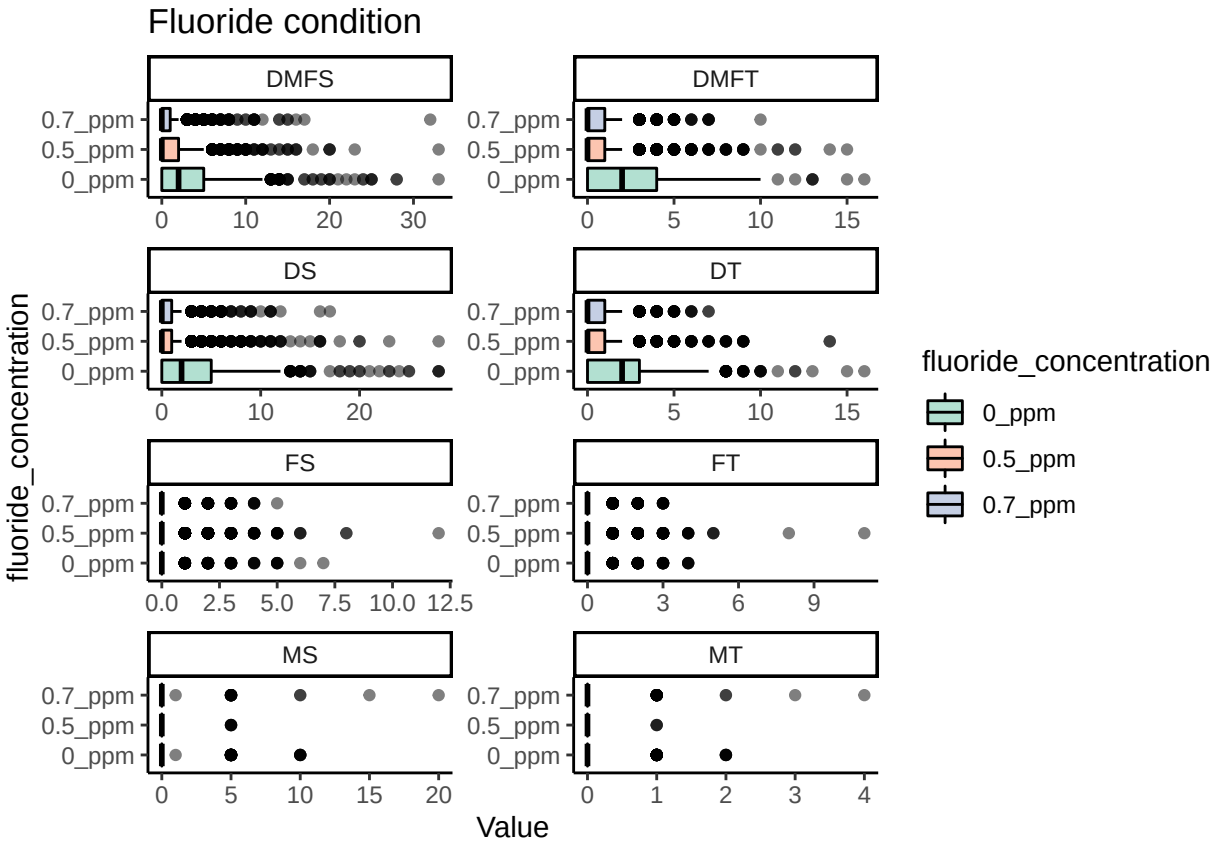

```
data_b %>%gather(c(DT:DMFS),
                 key = "Parameter",
                 value="Value")%>%
  ggplot(aes(x=fluorosis, y=Value,fill=fluorosis))+
  geom_boxplot(alpha=0.5,col="black")+
  ggtitle("Fluorosis")+
  facet_wrap(~Parameter,ncol=2,scales = "free")+
  coord_flip()+
  scale_fill_manual(values = brewer.pal(n = 4, name = "Set2"))+ theme_classic()
```



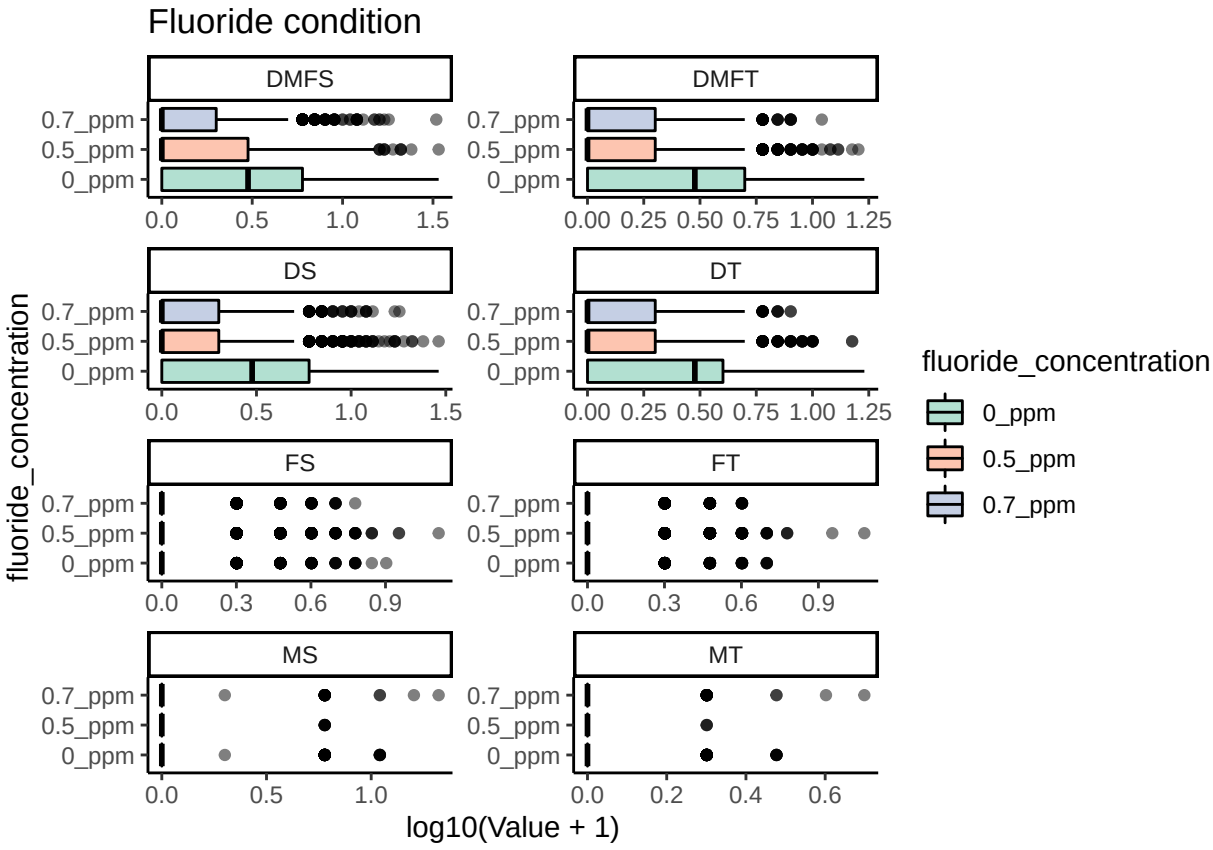

```
#transformation
data_b %>%gather(c(DT:DMFS),
                 key = "Parameter",
                 value="Value")%>%
  ggplot(aes(x=fluorosis, y=log10(Value+1),fill=fluorosis))+
  geom_boxplot(alpha=0.5,col="black")+
  ggtitle("Fluorosis")+
  facet_wrap(~Parameter,ncol=2,scales = "free")+
  coord_flip()+
  scale_fill_manual(values = brewer.pal(n = 4, name = "Set2"))+ theme_classic()
```




```
data_b %>%gather(c(DT:DMFS),
                 key = "Parameter",
                 value="Value")%>%
  ggplot(aes(x=fluoride_concentration, y=Value,fill=fluoride_concentration))+
  geom_boxplot(alpha=0.5,col="black"
               )+
  ggtitle("Fluoride condition")+
  facet_wrap(~Parameter,ncol=2,scales = "free")+
  coord_flip()+
  scale_fill_manual(values = brewer.pal(n = 4, name = "Set2"))+ theme_classic()
```




```
#transformation
data_b %>%gather(c(DT:DMFS),
                 key = "Parameter",
                 value="Value")%>%
  ggplot(aes(x=fluoride_concentration, y=log10(Value+1),fill=fluoride_concentration))+
  geom_boxplot(alpha=0.5,col="black"
               )+
  ggtitle("Fluoride condition")+
  facet_wrap(~Parameter,ncol=2,scales = "free")+
  coord_flip()+
  scale_fill_manual(values = brewer.pal(n = 4, name = "Set2"))+ theme_classic()
```



k

```
#FS, FT, MS, MT are too low -> remove
#area is the same with fluoride_concentration
#remove age
data_b1 <- data_b[-c(3,4,7,8,13,15)]

#transformation
data_b2 <- data_b1
data_b2[,c(2:5)] <- log10(data_b1[,c(2:5)]+1)

scaled_data_b <- data_b2 %>%as.data.frame()
scaled_data_b[,c(2:5)] <- scale(scaled_data_b[,c(2:5)])

#remove the diagnosis variables
X_mat <- scaled_data_b %>% select(-c("caries","fluorosis"))

Es <- numeric(10)
for(i in 1:10){
  kpres <- kproto(X_mat,
                  k = i, nstart = 5,
                  lambda = lambdaest(X_mat),
                  verbose = FALSE)
  Es[i] <- kpres$tot.withinss}
```

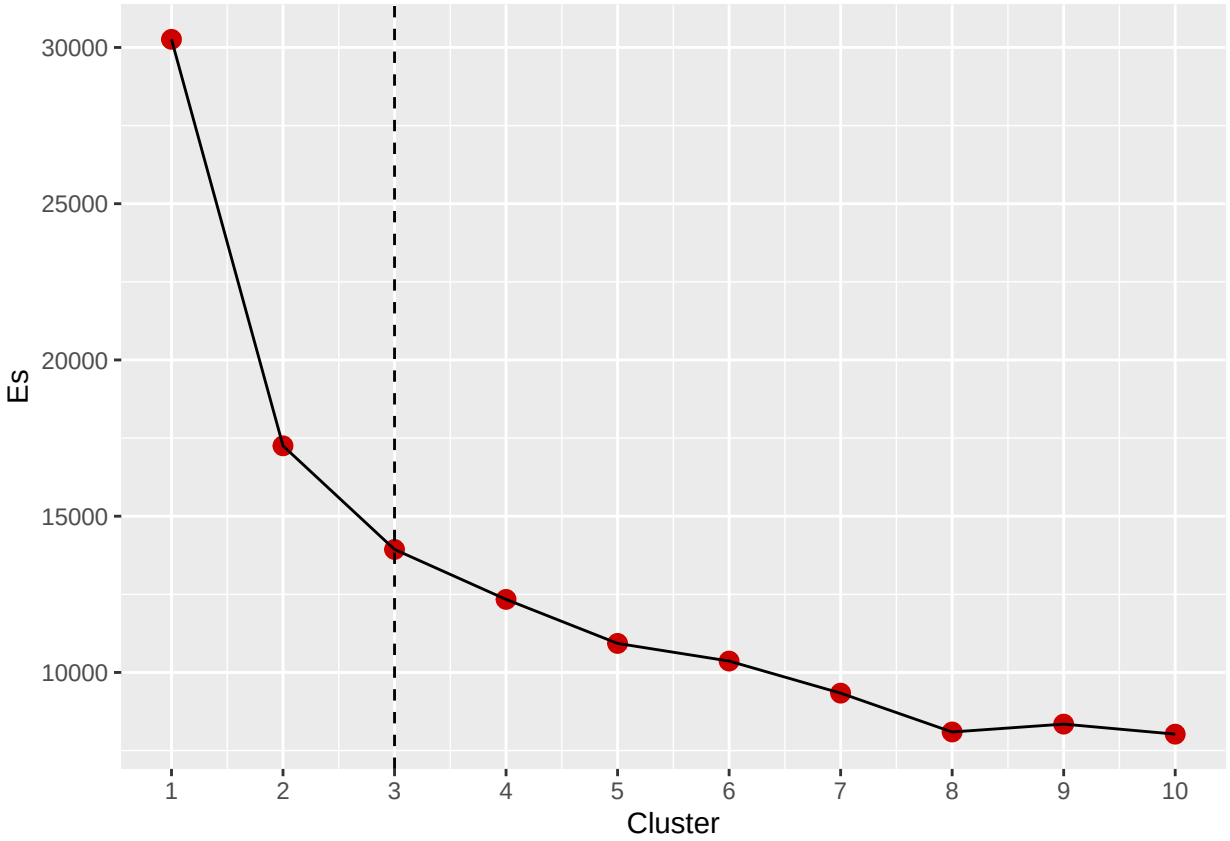
```
## Numeric variances:
##   DT DMFT   DS DMFS
```

```
##    1    1    1    1
## Average numeric variance: 1
##
## Heuristic for categorical variables: (method = 1)
##                 gender              DEANindex                   year
##              0.4986491              0.4218530              0.6120228
## fluoride_concentration
##              0.6311979
## Average categorical variation: 0.5409307
##
## Estimated lambda: 1.848666
##
## Numeric variances:
##   DT DMFT   DS DMFS
##    1    1    1    1
## Average numeric variance: 1
##
## Heuristic for categorical variables: (method = 1)
##                 gender              DEANindex                   year
##              0.4986491              0.4218530              0.6120228
## fluoride_concentration
##              0.6311979
## Average categorical variation: 0.5409307
##
## Estimated lambda: 1.848666
##
## Numeric variances:
##   DT DMFT   DS DMFS
##    1    1    1    1
## Average numeric variance: 1
##
## Heuristic for categorical variables: (method = 1)
##                 gender              DEANindex                   year
##              0.4986491              0.4218530              0.6120228
## fluoride_concentration
##              0.6311979
## Average categorical variation: 0.5409307
##
## Estimated lambda: 1.848666
##
## Numeric variances:
##   DT DMFT   DS DMFS
##    1    1    1    1
## Average numeric variance: 1
##
## Heuristic for categorical variables: (method = 1)
##                 gender              DEANindex                   year
##              0.4986491              0.4218530              0.6120228
## fluoride_concentration
##              0.6311979
## Average categorical variation: 0.5409307
##
## Estimated lambda: 1.848666
##
```

```
## Numeric variances:
##   DT DMFT   DS DMFS
##    1    1    1    1
## Average numeric variance: 1
##
## Heuristic for categorical variables: (method = 1)
##                 gender              DEANindex                   year
##              0.4986491              0.4218530              0.6120228
## fluoride_concentration
##              0.6311979
## Average categorical variation: 0.5409307
##
## Estimated lambda: 1.848666
##
## Numeric variances:
##   DT DMFT   DS DMFS
##    1    1    1    1
## Average numeric variance: 1
##
## Heuristic for categorical variables: (method = 1)
##                 gender              DEANindex                   year
##              0.4986491              0.4218530              0.6120228
## fluoride_concentration
##              0.6311979
## Average categorical variation: 0.5409307
##
## Estimated lambda: 1.848666
##
## Numeric variances:
##   DT DMFT   DS DMFS
##    1    1    1    1
## Average numeric variance: 1
##
## Heuristic for categorical variables: (method = 1)
##                 gender              DEANindex                   year
##              0.4986491              0.4218530              0.6120228
## fluoride_concentration
##              0.6311979
## Average categorical variation: 0.5409307
##
## Estimated lambda: 1.848666
##
## Numeric variances:
##   DT DMFT   DS DMFS
##    1    1    1    1
## Average numeric variance: 1
##
## Heuristic for categorical variables: (method = 1)
##                 gender              DEANindex                   year
##              0.4986491              0.4218530              0.6120228
## fluoride_concentration
##              0.6311979
## Average categorical variation: 0.5409307
##
```

```
## Estimated lambda: 1.848666
##
## Numeric variances:
##   DT DMFT   DS DMFS
##    1    1    1    1
## Average numeric variance: 1
##
## Heuristic for categorical variables: (method = 1)
##                  gender              DEANindex                   year
##               0.4986491              0.4218530              0.6120228
## fluoride_concentration
##               0.6311979
## Average categorical variation: 0.5409307
##
## Estimated lambda: 1.848666
##
## Numeric variances:
##   DT DMFT   DS DMFS
##    1    1    1    1
## Average numeric variance: 1
##
## Heuristic for categorical variables: (method = 1)
##                  gender              DEANindex                   year
##               0.4986491              0.4218530              0.6120228
## fluoride_concentration
##               0.6311979
## Average categorical variation: 0.5409307
##
## Estimated lambda: 1.848666
```

```
tibble(Cluster = c(1:10), Es = Es) %>%
  ggplot(aes(x = Cluster, y = Es)) +
  geom_point(size = 3,
             col ="red3") +
  geom_path() +
  geom_vline(xintercept = 3,
             linetype = 2)+
  scale_x_continuous(breaks = c(1:10))
```



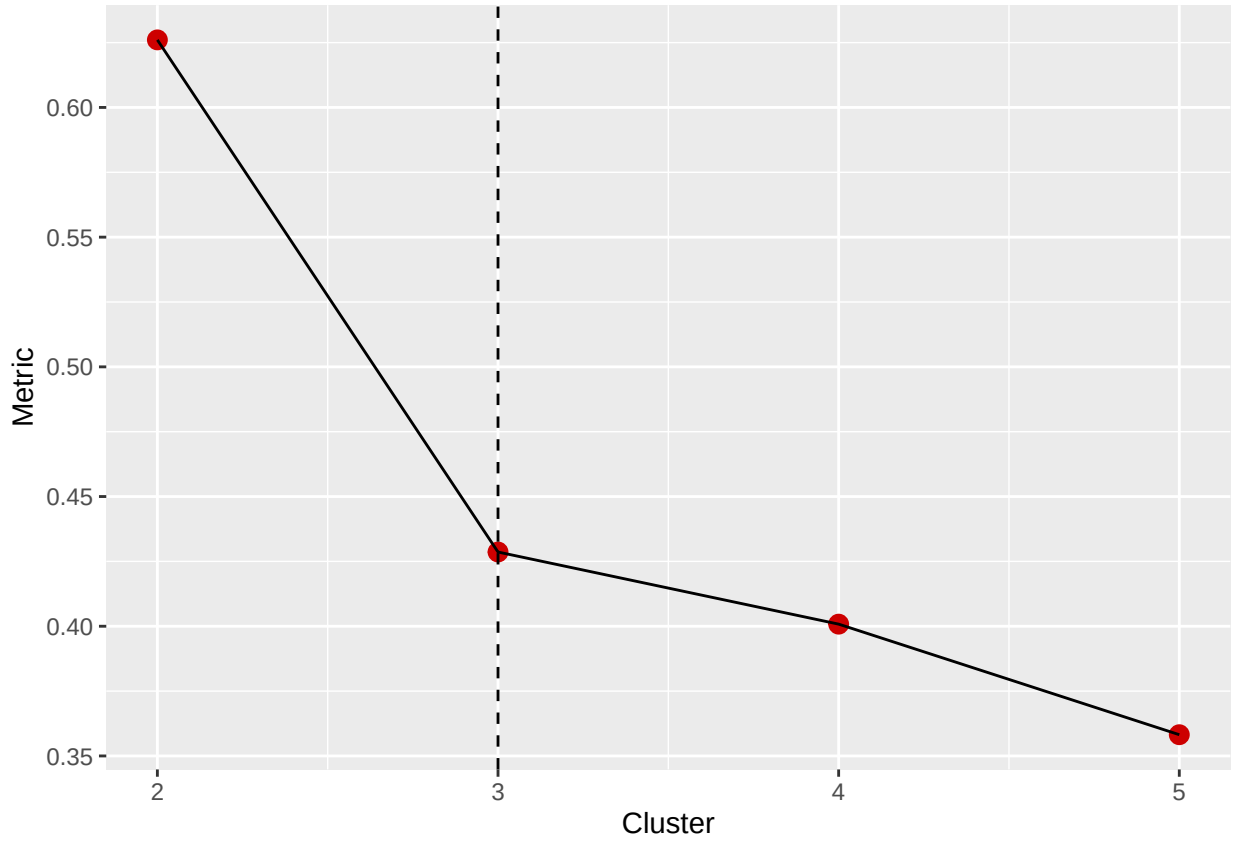
Clustering

```
k_opt1 <- validation_kproto(method="silhouette", data=X_mat, lambda=lambdaest(X_mat), k=2:5,
                            kp_obj="optimal", nstart = 5, verbose = FALSE)
```

```
## Numeric variances:
##   DT DMFT   DS DMFS
##    1    1    1    1
## Average numeric variance: 1
##
## Heuristic for categorical variables: (method = 1)
##                  gender              DEANindex                   year
##               0.4986491              0.4218530             0.6120228
## fluoride_concentration
##               0.6311979
## Average categorical variation: 0.5409307
##
## Estimated lambda: 1.848666
```

```
saveRDS(k_opt1,"~/Library/Mobile Documents/com~apple~CloudDocs/Nam/New project/20 years of water fluori

k_opt1 <- readRDS("~/Library/Mobile Documents/com~apple~CloudDocs/Nam/New project/20 years of water flu
```

```
#
tibble(Cluster = c(2:5),
       Metric = as.vector(k_opt1$indices)) %>%
  ggplot(aes(x = Cluster,
             y = Metric)) +
  geom_point(size = 3,
             col ="red3") +
  geom_path() +
  geom_vline(xintercept = 3,
             linetype = 2)+
  scale_x_continuous(breaks = c(2:5))
```




```
#k=3
kpres <- kproto(x = X_mat,
                k = 3,
                lambda = lambdaest(X_mat), nstart = 5)
```

```
## Numeric variances:
##   DT DMFT   DS DMFS
##    1    1    1    1
## Average numeric variance: 1
##
## Heuristic for categorical variables: (method = 1)
##                 gender              DEANindex                   year
```

```
##              0.4986491              0.4218530              0.6120228
## fluoride_concentration
##              0.6311979
## Average categorical variation: 0.5409307
##
## Estimated lambda: 1.848666
##
## # NAs in variables:
##                 gender                     DT                   DMFT
##                      0                      0                      0
##                     DS                   DMFS              DEANindex
##                      0                      0                      0
##                   year fluoride_concentration
##                      0                      0
## 0 observation(s) with NAs.
##
## # NAs in variables:
##                 gender                     DT                   DMFT
##                      0                      0                      0
##                     DS                   DMFS              DEANindex
##                      0                      0                      0
##                   year fluoride_concentration
##                      0                      0
## 0 observation(s) with NAs.
##
## # NAs in variables:
##                 gender                     DT                   DMFT
##                      0                      0                      0
##                     DS                   DMFS              DEANindex
##                      0                      0                      0
##                   year fluoride_concentration
##                      0                      0
## 0 observation(s) with NAs.
##
## # NAs in variables:
##                 gender                     DT                   DMFT
##                      0                      0                      0
##                     DS                   DMFS              DEANindex
##                      0                      0                      0
##                   year fluoride_concentration
##                      0                      0
## 0 observation(s) with NAs.
##
## # NAs in variables:
##                 gender                     DT                   DMFT
##                      0                      0                      0
##                     DS                   DMFS              DEANindex
##                      0                      0                      0
##                   year fluoride_concentration
##                      0                      0
## 0 observation(s) with NAs.
```

```
kpres
```

```
## Numeric predictors: 4
## Categorical predictors: 4
## Lambda: 1.848666
##
## Number of Clusters: 3
## Cluster sizes: 1345 1841 1008
## Within cluster error: 6219.54 4860.473 2856.019
##
## Cluster prototypes:
##   gender         DT       DMFT         DS       DMFS DEANindex year
## 1 female  1.2625421  1.2281913  1.2575552  1.2259739    Normal 2012
## 2   male -0.6298630 -0.6119791 -0.6229753 -0.6065016    Normal 2012
## 3 female -0.5342672 -0.5210951 -0.5401926 -0.5281403    Normal 2003
##   fluoride_concentration
## 1                  0_ppm
## 2                0.5_ppm
## 3                0.7_ppm
```

```
summary(kpres)
```

```
## gender
##
## cluster  male female
##       1 0.400  0.600
##       2 0.569  0.431
##       3 0.400  0.600
##
## ------------------------------------------------------------
## DT
##         Min.    1st Qu.     Median       Mean    3rd Qu.      Max.
## 1 -0.7715461  0.9163978  1.3584019  1.2625421  1.7012470 3.5814956
## 2 -0.7715461 -0.7715461 -0.7715461 -0.6298630 -0.7715461 1.3584019
## 3 -0.7715461 -0.7715461 -0.7715461 -0.5342672 -0.7715461 0.9163978
##
## ------------------------------------------------------------
## DMFT
##         Min.    1st Qu.     Median       Mean    3rd Qu.     Max.
## 1 -0.8536977  0.7652349  1.1891678  1.2281913  1.5179957 3.321371
## 2 -0.8536977 -0.8536977 -0.8536977 -0.6119791 -0.8536977 1.786668
## 3 -0.8536977 -0.8536977 -0.8536977 -0.5210951  0.1677351 1.189168
##
## ------------------------------------------------------------
## DS
##         Min.    1st Qu.     Median       Mean    3rd Qu.     Max.
## 1 -0.7507042  0.6763264  1.0500080  1.2575552  1.7769148 3.623208
## 2 -0.7507042 -0.7507042 -0.7507042 -0.6229753 -0.7507042 1.339858
## 3 -0.7507042 -0.7507042 -0.7507042 -0.5401926 -0.7507042 1.050008
##
## ------------------------------------------------------------
## DMFS
##          Min.    1st Qu.     Median       Mean     3rd Qu.     Max.
## 1  0.02846731  0.5294851  1.1606928  1.2259739  1.57645871 3.529361
## 2 -0.82802821 -0.8280282 -0.8280282 -0.6065016 -0.82802821 1.576459
## 3 -0.82802821 -0.8280282 -0.8280282 -0.5281403  0.02846731 2.597954
```

```
##
## ------------------------------------------------------------------
## DEANindex
##
## cluster Normal Questionable Very mild  Mild Moderate Severe
##       1  0.793        0.033     0.100 0.051    0.022  0.001
##       2  0.762        0.017     0.137 0.069    0.015  0.000
##       3  0.653        0.061     0.135 0.086    0.062  0.004
##
## ------------------------------------------------------------------
## year
##
## cluster  1990  2003  2012  2019
##       1 0.000 0.294 0.501 0.205
##       2 0.000 0.033 0.777 0.190
##       3 0.000 0.898 0.004 0.098
##
## ------------------------------------------------------------------
## fluoride_concentration
##
## cluster 0_ppm 0.5_ppm 0.7_ppm
##       1 0.529   0.322   0.149
##       2 0.152   0.848   0.000
##       3 0.161   0.043   0.797
##
## ------------------------------------------------------------------
```

```
saveRDS(kpres,"~/Library/Mobile Documents/com~apple~CloudDocs/Nam/New project/20 years of water fluorida
kpres <- readRDS("~/Library/Mobile Documents/com~apple~CloudDocs/Nam/New project/20 years of water fluo
```

Plot

```
valid_df <- data_b1 %>% mutate(Cluster = as.factor(kpres$cluster))

#Community Fluorosis Index (CFI)
valid_df$DEANweight <- ifelse(valid_df$DEANindex == "Normal", 0,
                              ifelse(valid_df$DEANindex == "Questionable",0.5,
                                     ifelse(valid_df$DEANindex == "Very mild",1,
                                            ifelse(valid_df$DEANindex == "Mild",2,
                                                   ifelse(valid_df$DEANindex == "Moderate",3,4)))))

valid_df$CFI <- ifelse(valid_df$Cluster == 1, mean(subset(valid_df, valid_df$Cluster == 1)$DEANweight),
                              ifelse(valid_df$Cluster == 2,mean(subset(valid_df, valid_df$Cluster ==
                                     ifelse(valid_df$Cluster == 3, mean(subset(valid_df, valid_df$Clu
                                            mean(subset(valid_df, valid_df$Cluster == 4)$DEANweight)


valid_df$Significance <- as.factor(ifelse(valid_df$CFI <= 0.4 , "Negative", "Border line"))

# Table of CFI
```

```r
# Whole population: "Negative"
mean(valid_df$DEANweight)
```

```
## [1] 0.3655222
```

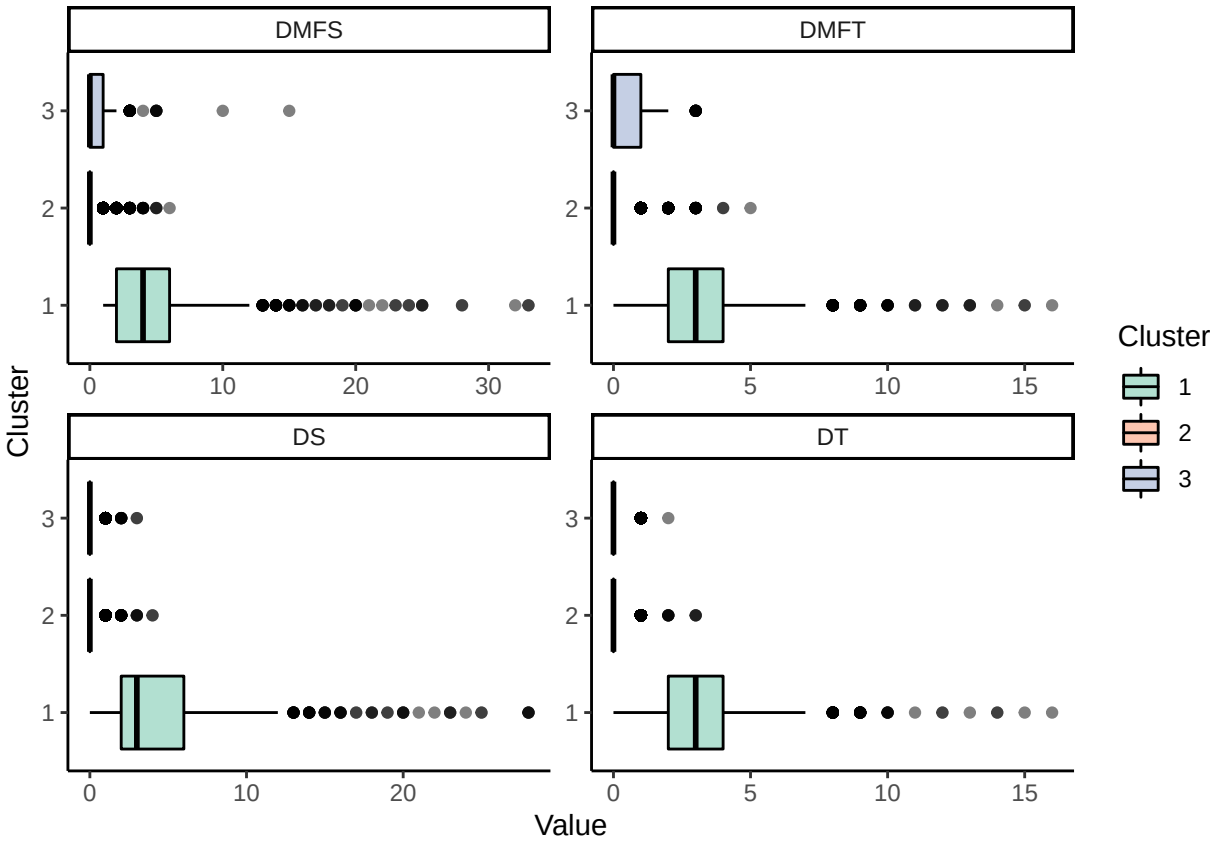
```r
# For each cluster
table(valid_df$Cluster, valid_df$CFI)
```

```
##
##     0.288104089219331 0.32753938077132 0.538194444444444
##   1              1345                0                 0
##   2                 0             1841                 0
##   3                 0                0              1008
```

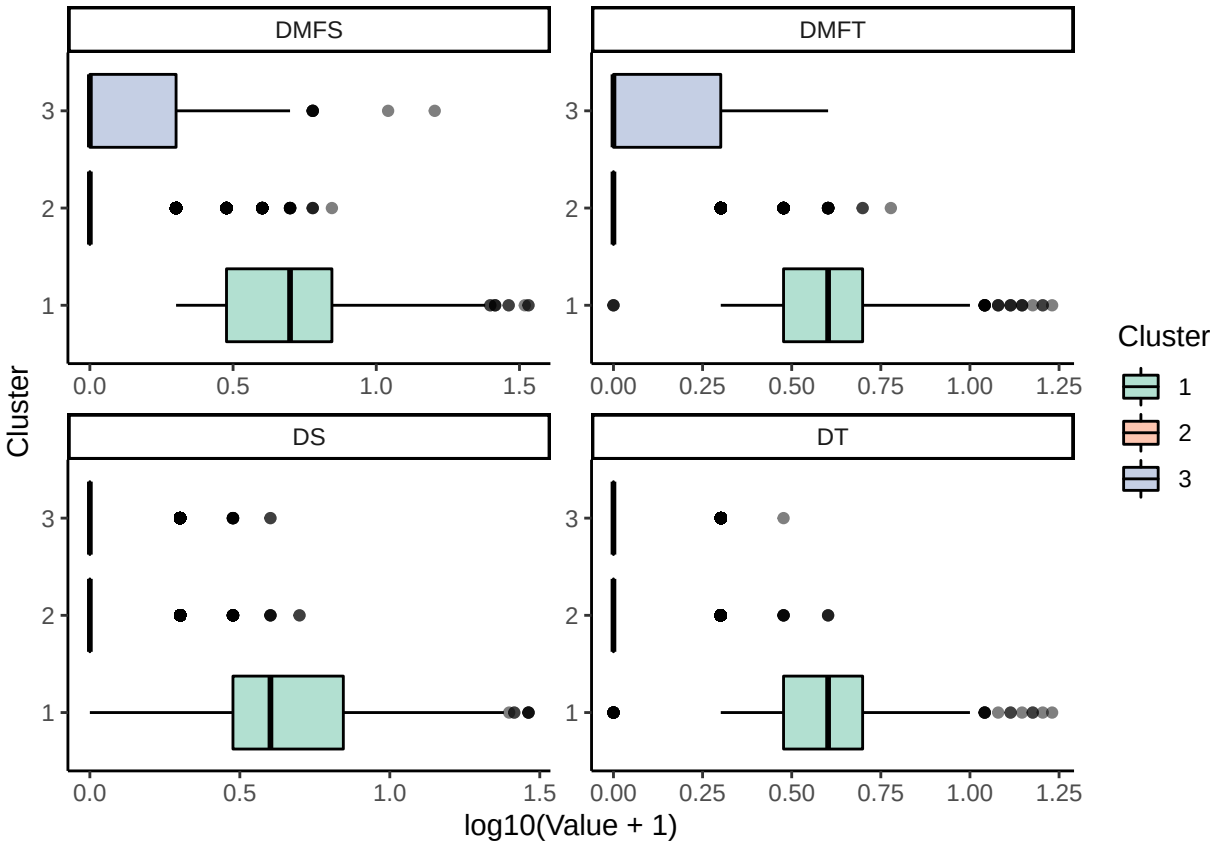
```r
table(valid_df$Cluster, valid_df$Significance)
```

```
##
##     Border line Negative
##   1           0     1345
##   2           0     1841
##   3        1008        0
```

```r
valid_df %>%gather(c(DT:DMFS),
                   key = "Parameter",
                   value="Value")%>%
  ggplot(aes(x = Cluster, y=Value, fill = Cluster))+
  geom_boxplot(alpha=0.5,col="black")+
  facet_wrap(~Parameter,ncol=2,scales = "free")+
  coord_flip()+
  scale_fill_manual(values = brewer.pal(n = 4, name = "Set2"))+ theme_classic()
```



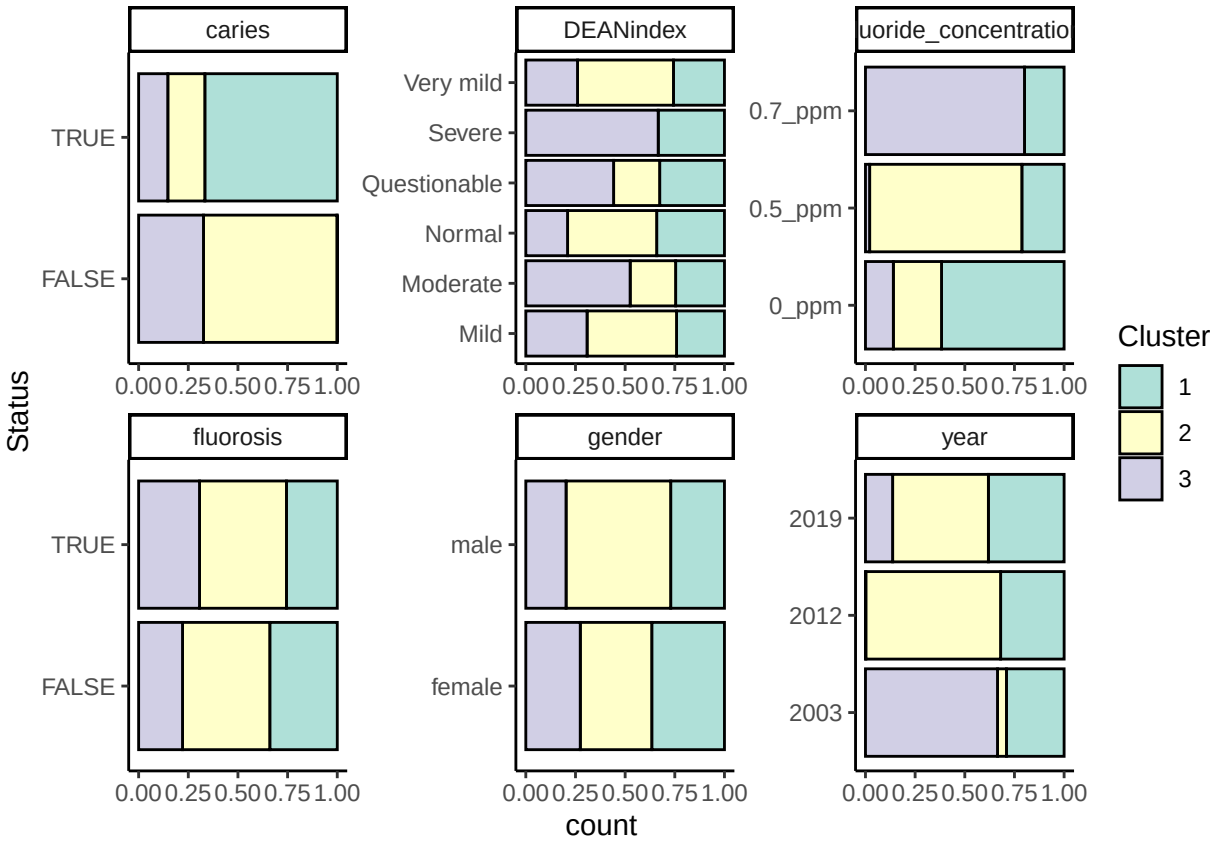

```
#transformation
valid_df %>%gather(c(DT:DMFS),
                   key = "Parameter",
                   value="Value")%>%
  ggplot(aes(x = Cluster, y=log10(Value+1), fill = Cluster))+
  geom_boxplot(alpha=0.5,col="black")+
  facet_wrap(~Parameter,ncol=2,scales = "free")+
  coord_flip()+
  scale_fill_manual(values = brewer.pal(n = 4, name = "Set2"))+ theme_classic()
```



Plot

```
valid_df %>% gather(gender, fluoride_concentration, caries, fluorosis, DEANindex, year,
                    key="Factor",value="Status")%>%
  ggplot(aes(x=Status,fill=Cluster))+
  geom_bar(stat = "count",position="fill",show.legend = T,alpha=0.7,col="black")+
  coord_flip()+
  facet_wrap(~Factor,ncol=3,scales = "free")+
  scale_fill_manual(values = brewer.pal(n = 4, name = "Set3"))+ theme_classic()
```

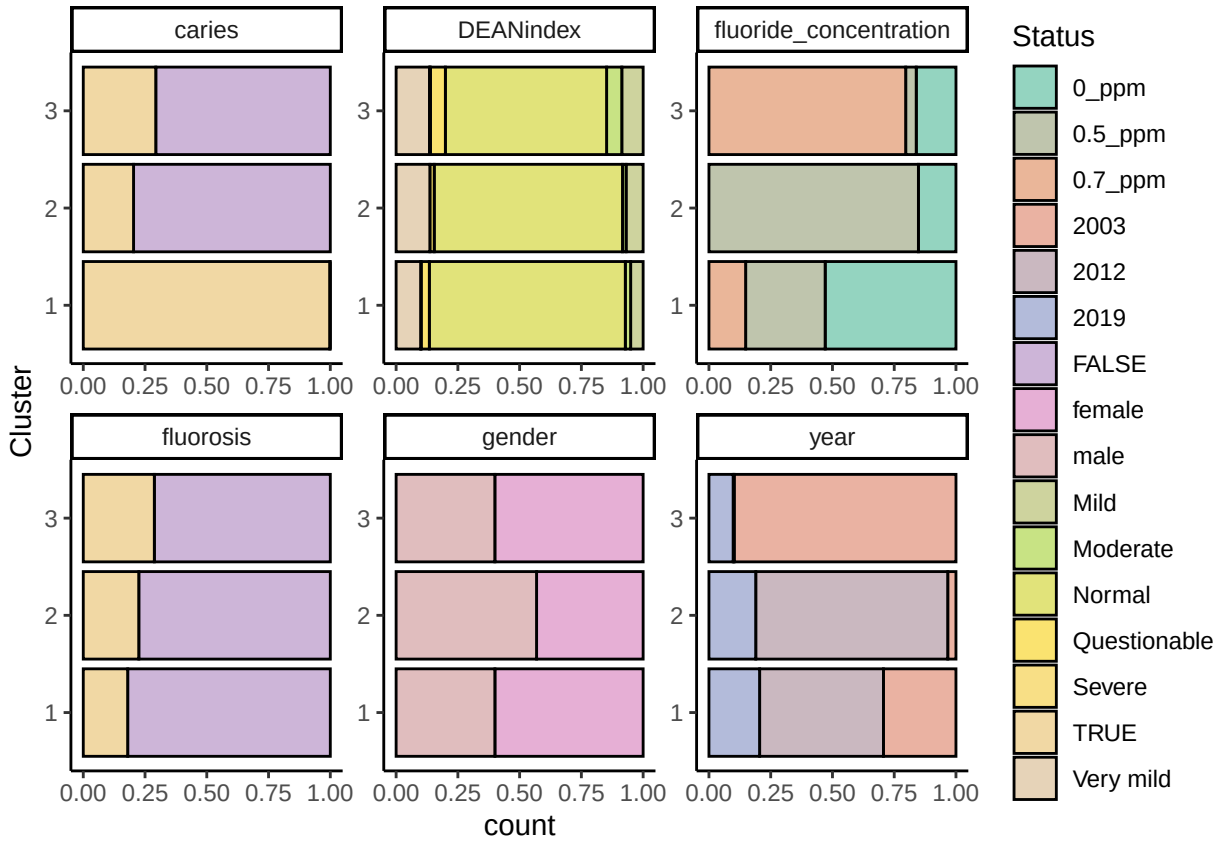
```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```




```
valid_df %>% gather(gender, fluoride_concentration, caries, fluorosis, DEANindex, year,
                    key="Factor",value="Status")%>%
  ggplot(aes(x=Cluster,fill= Status))+
  geom_bar(stat = "count",position="fill",show.legend = T,alpha=0.7,col="black")+
  coord_flip()+
  facet_wrap(~Factor,ncol=3,scales = "free")+
  scale_fill_manual(values = colorRampPalette(brewer.pal(n = 12, name = "Set2"))(18)) + theme_classic()
```

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```

```
## Warning in brewer.pal(n = 12, name = "Set2"): n too large, allowed maximum for palette Set2 is 8
## Returning the palette you asked for with that many colors
```



#grid.col = c(1 ="#f7286d", 2 ="#1faae0", TRUE= "#2968c2", FALSE= "#97c425",)

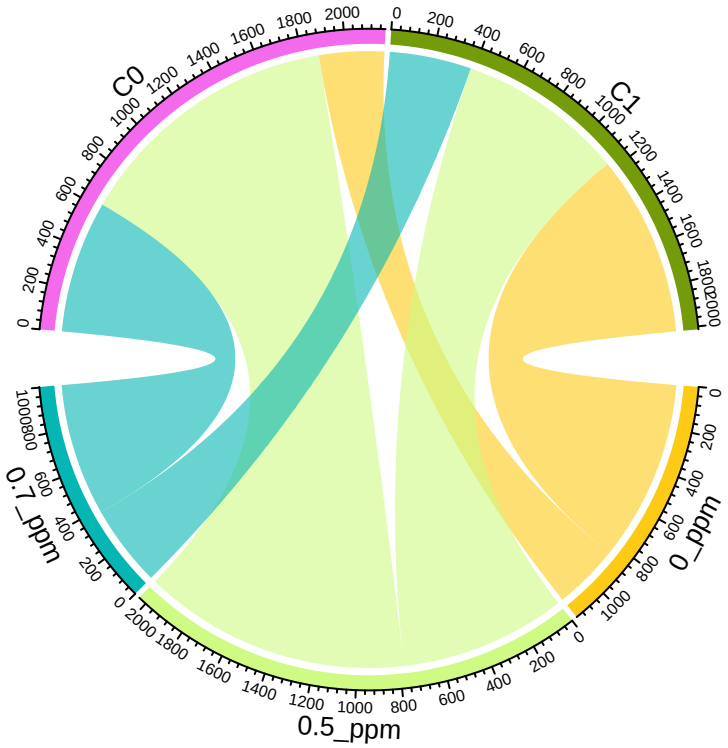
```
library(circlize)
```

```
## ========================================
## circlize version 0.4.13
## CRAN page: https://cran.r-project.org/package=circlize
## Github page: https://github.com/jokergoo/circlize
## Documentation: https://jokergoo.github.io/circlize_book/book/
##
## If you use it in published research, please cite:
## Gu, Z. circlize implements and enhances circular visualization
##   in R. Bioinformatics 2014.
##
## This message can be suppressed by:
##   suppressPackageStartupMessages(library(circlize))
## ========================================
```

```
#caries, fluoride_concentration
xtb_y <- valid_df %>%
    mutate(Id=rownames(valid_df))%>%
    gather(caries,
           key="Pathology",
           value="Diagnosis")%>%
  group_by(fluoride_concentration,Diagnosis,Pathology)%>%
  summarise(frequency = n())
```

```
## `summarise()` has grouped output by 'fluoride_concentration', 'Diagnosis'. You can override using the
```

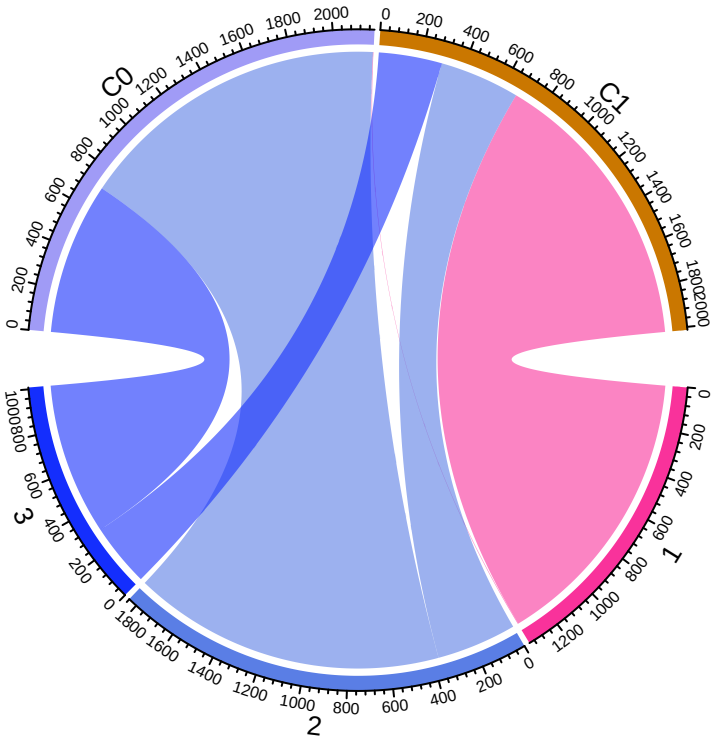
```
xtb_y$Diagnosis <- ifelse(xtb_y$Pathology == "caries" & xtb_y$Diagnosis == "FALSE", "C0",
                          ifelse(xtb_y$Pathology == "caries" & xtb_y$Diagnosis == "TRUE", "C1",
                                 ifelse(xtb_y$Pathology == "fluorosis" & xtb_y$Diagnosis == "FALSE", "F
xtb_y <- xtb_y%>%.[,c(1,2,4)]

chordDiagram(as.data.frame(xtb_y),
             transparency = 0.4,
#             grid.col = grid.col,
             column.col ="black")
```




```
#caries, Cluster
xtb_y <- valid_df %>%
    mutate(Id=rownames(valid_df))%>%
    gather(caries,
           key="Pathology",
           value="Diagnosis")%>%
  group_by(Cluster,Diagnosis,Pathology)%>%
  summarise(frequency = n())
```

```
## `summarise()` has grouped output by 'Cluster', 'Diagnosis'. You can override using the `.groups` argu
```

```
xtb_y$Diagnosis <- ifelse(xtb_y$Pathology == "caries" & xtb_y$Diagnosis == "FALSE", "C0",
                          ifelse(xtb_y$Pathology == "caries" & xtb_y$Diagnosis == "TRUE", "C1",
                                 ifelse(xtb_y$Pathology == "fluorosis" & xtb_y$Diagnosis == "FALSE", "F
xtb_y <- xtb_y%>%.[,c(1,2,4)]

chordDiagram(as.data.frame(xtb_y),
             transparency = 0.4,
 #            grid.col = grid.col,
             column.col ="black")
```

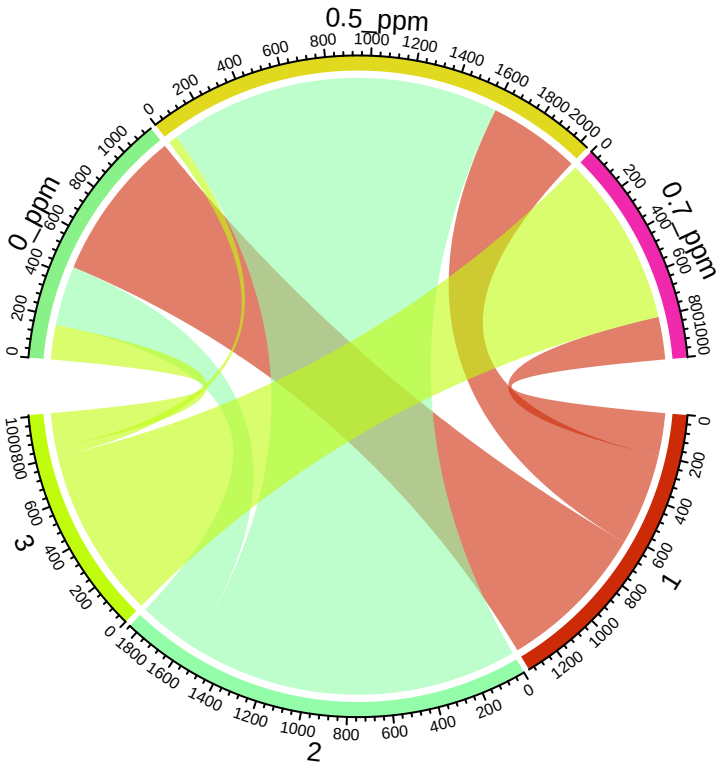



```
#fluoride_concentration, Cluster
xtb_y <- valid_df %>%
    mutate(Id=rownames(valid_df))%>%
    gather(fluoride_concentration,
           key="Treatment",
           value="Dose")%>%
  group_by(Cluster,Dose,Treatment)%>%
  summarise(frequency = n())
```

```
## `summarise()` has grouped output by 'Cluster', 'Dose'. You can override using the `.groups` argument
```

```
xtb_y <- xtb_y%>%.[,c(1,2,4)]

chordDiagram(as.data.frame(xtb_y),
```

```
             transparency = 0.4,
#             grid.col = grid.col,
             column.col ="black")
```

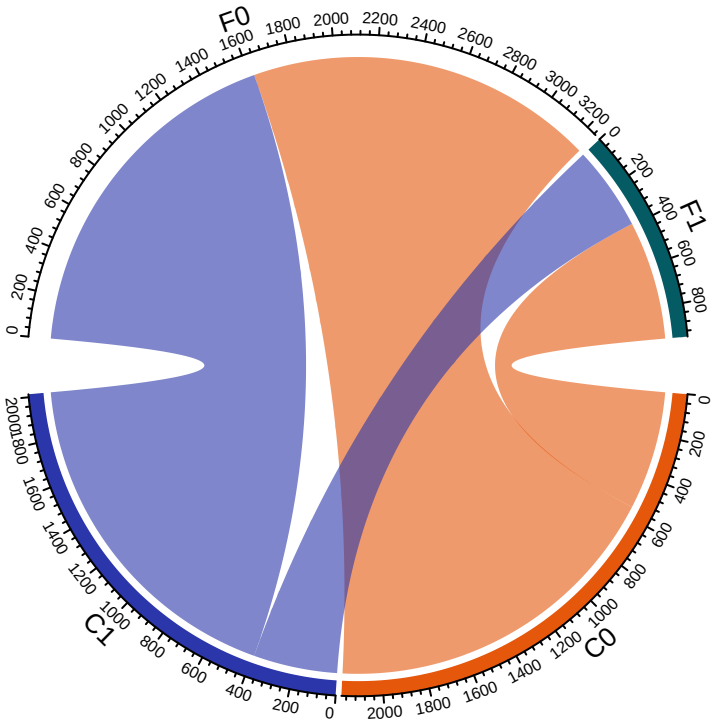



```
#fluorosis, caries
xtb_y = valid_df %>%
    mutate(Id=rownames(valid_df))%>%
    gather(fluorosis,
           key="Pathology",
           value="Diagnosis")%>%
  group_by(caries,Diagnosis,Pathology)%>%
  summarise(frequency = n())
```

```
## `summarise()` has grouped output by 'caries', 'Diagnosis'. You can override using the `.groups` argu
```

```
xtb_y$Diagnosis <- ifelse(xtb_y$Pathology == "caries" & xtb_y$Diagnosis == "FALSE", "C0",
                          ifelse(xtb_y$Pathology == "caries" & xtb_y$Diagnosis == "TRUE", "C1",
                                 ifelse(xtb_y$Pathology == "fluorosis" & xtb_y$Diagnosis == "FALSE", "F
xtb_y$caries <- ifelse(xtb_y$caries == "FALSE", "C0", "C1")

xtb_y <- xtb_y%>%.[,c(1,2,4)]

chordDiagram(as.data.frame(xtb_y),
             transparency = 0.4,
```

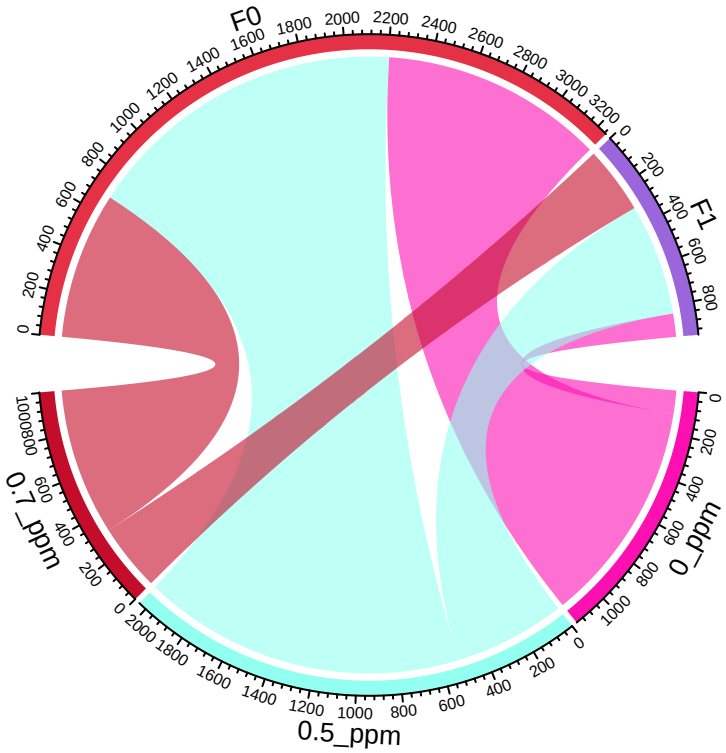
```
#             grid.col = grid.col,
            column.col ="black")
```




```
#fluorosis, fluoride_concentration
xtb_y = valid_df %>%
    mutate(Id=rownames(valid_df))%>%
    gather(fluorosis,
           key="Pathology",
           value="Diagnosis")%>%
  group_by(fluoride_concentration,Diagnosis,Pathology)%>%
  summarise(frequency = n())
```

```
## `summarise()` has grouped output by 'fluoride_concentration', 'Diagnosis'. You can override using the
```

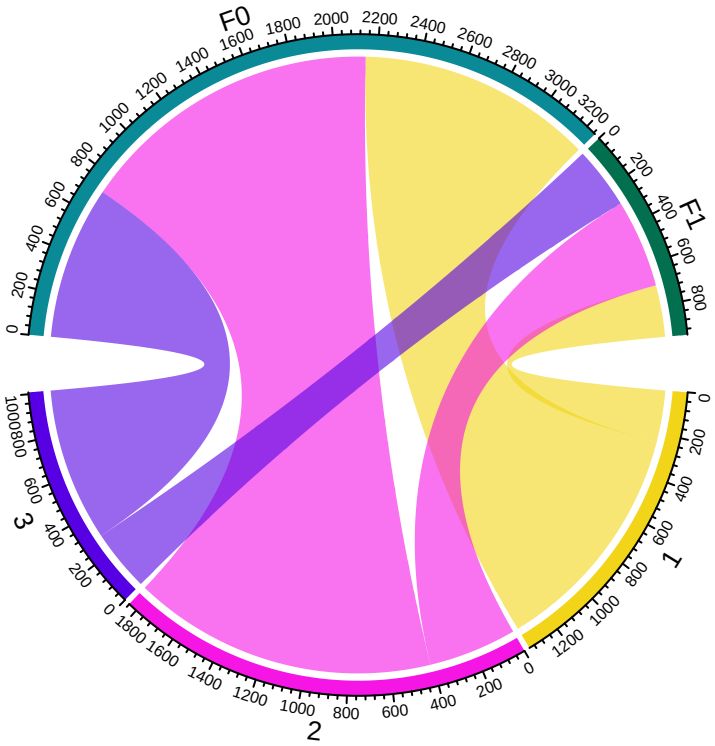
```
xtb_y$Diagnosis <- ifelse(xtb_y$Pathology == "caries" & xtb_y$Diagnosis == "FALSE", "C0",
                          ifelse(xtb_y$Pathology == "caries" & xtb_y$Diagnosis == "TRUE", "C1",
                                 ifelse(xtb_y$Pathology == "fluorosis" & xtb_y$Diagnosis == "FALSE", "F

xtb_y <- xtb_y%>%.[,c(1,2,4)]

chordDiagram(as.data.frame(xtb_y),
             transparency = 0.4,
 #            grid.col = grid.col,
             column.col ="black")
```




```
#fluorosis, Cluster
xtb_y = valid_df %>%
    mutate(Id=rownames(valid_df))%>%
    gather(fluorosis,
           key="Pathology",
           value="Diagnosis")%>%
  group_by(Cluster,Diagnosis,Pathology)%>%
  summarise(frequency = n())
```

```
## `summarise()` has grouped output by 'Cluster', 'Diagnosis'. You can override using the `.groups` argu
```

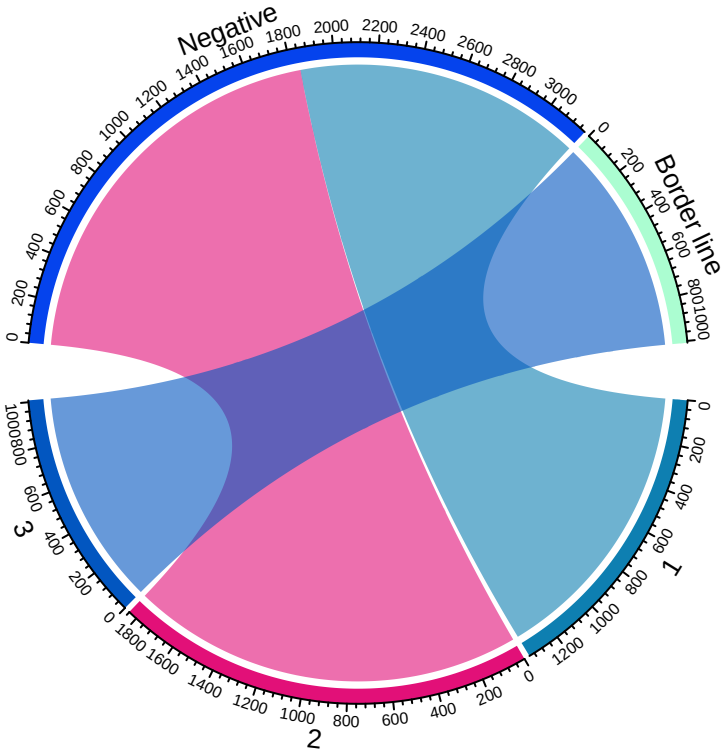
```
xtb_y$Diagnosis <- ifelse(xtb_y$Pathology == "caries" & xtb_y$Diagnosis == "FALSE", "C0",
                          ifelse(xtb_y$Pathology == "caries" & xtb_y$Diagnosis == "TRUE", "C1",
                                 ifelse(xtb_y$Pathology == "fluorosis" & xtb_y$Diagnosis == "FALSE", "F0

xtb_y <- xtb_y%>%.[,c(1,2,4)]

chordDiagram(as.data.frame(xtb_y),
             transparency = 0.4,
 #            grid.col = grid.col,
             column.col ="black")
```




```
#CFI Public Health Significance, Cluster
xtb_y = valid_df %>%
    mutate(Id=rownames(valid_df))%>%
    gather(Significance,
           key="Pathology",
           value="Diagnosis")%>%
  group_by(Cluster,Diagnosis,Pathology)%>%
  summarise(frequency = n())
```

```
## `summarise()` has grouped output by 'Cluster', 'Diagnosis'. You can override using the `.groups` argu
```

```
xtb_y <- xtb_y%>%.[,c(1,2,4)]

chordDiagram(as.data.frame(xtb_y),
             transparency = 0.4,
 #            grid.col = grid.col,
             column.col ="black")
```

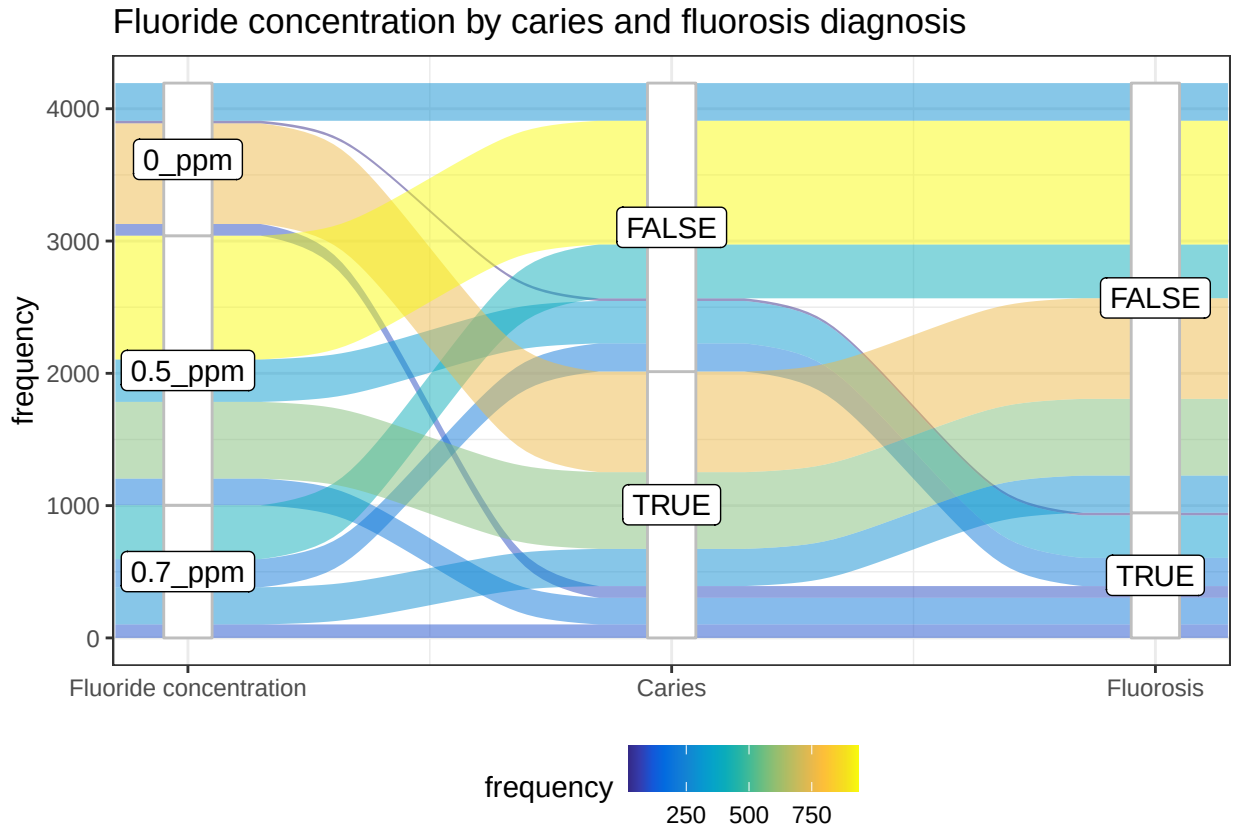



```
library(ggalluvial)
library(pals)
#
xtb_y2 <- valid_df %>%
  dplyr::group_by(caries, fluorosis, fluoride_concentration)%>%
  summarise(frequency = n())
```

```
## `summarise()` has grouped output by 'caries', 'fluorosis'. You can override using the `.groups` argu
```

```
xtb_y2 %>% ggplot(aes(y = frequency,
                 axis1= fluoride_concentration,
                 axis2 = caries,
                 axis3= fluorosis,
                 )) +
  geom_alluvium(aes(fill = frequency), width = 0.3) +
  geom_stratum(width = 1/10,
               fill = "white",
               color = "grey") +
  geom_label(stat = "stratum",
             infer.label = TRUE) +
  scale_x_continuous(breaks = 1:3,
                     labels = c("Fluoride concentration", "Caries", "Fluorosis")) +
  scale_fill_gradientn(colours = pals::parula(n=500))+
  theme_bw()+theme(legend.position="bottom") +
  ggtitle("Fluoride concentration by caries and fluorosis diagnosis")
```

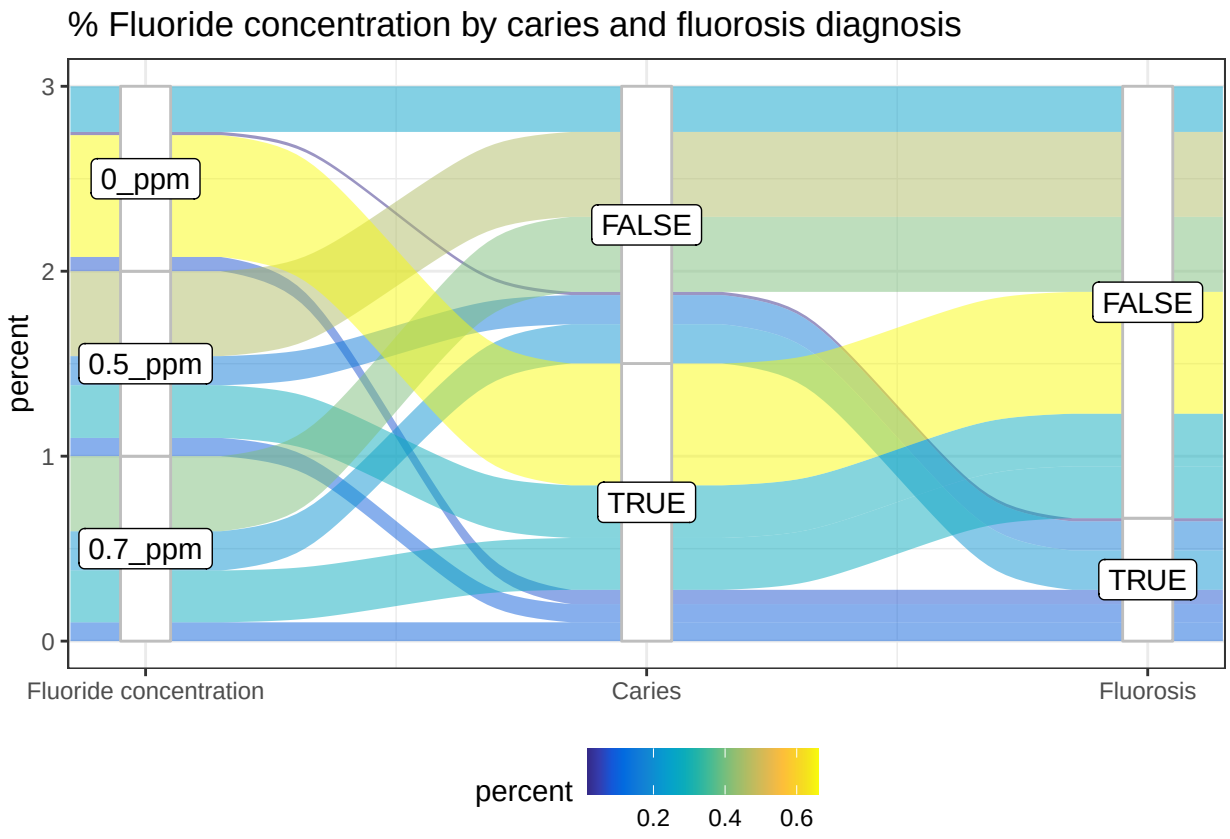
```
## Warning in to_lodes_form(data = data, axes = axis_ind, discern =
## params$discern): Some strata appear at multiple axes.

## Warning in to_lodes_form(data = data, axes = axis_ind, discern =
## params$discern): Some strata appear at multiple axes.

## Warning in to_lodes_form(data = data, axes = axis_ind, discern =
## params$discern): Some strata appear at multiple axes.

## Warning: The parameter 'infer.label' is deprecated.
## Use 'aes(label = after_stat(stratum))'.
```




```
#
xtb_y2 <- xtb_y2 %>%
  dplyr::group_by(fluoride_concentration)%>% mutate(percent = frequency/sum(frequency))
xtb_y2$sum <- xtb_y2$frequency/xtb_y2$percent

xtb_y2 %>% ggplot(aes(y = percent,
                      axis1= fluoride_concentration,
                      axis2 = caries,
                      axis3= fluorosis,
                      )) +
  geom_alluvium(aes(fill = percent), width = 0.3) +
  geom_stratum(width = 1/10,
               fill = "white",
```

```
              color = "grey") +
  geom_label(stat = "stratum",
             infer.label = TRUE) +
  scale_x_continuous(breaks = 1:3,
                     labels = c("Fluoride concentration", "Caries", "Fluorosis")) +
  scale_fill_gradientn(colours = pals::parula(n=500))+
  theme_bw()+theme(legend.position="bottom") +
  ggtitle("% Fluoride concentration by caries and fluorosis diagnosis")
```

```
## Warning in to_lodes_form(data = data, axes = axis_ind, discern =
## params$discern): Some strata appear at multiple axes.

## Warning in to_lodes_form(data = data, axes = axis_ind, discern =
## params$discern): Some strata appear at multiple axes.

## Warning in to_lodes_form(data = data, axes = axis_ind, discern =
## params$discern): Some strata appear at multiple axes.

## Warning: The parameter `infer.label` is deprecated.
## Use `aes(label = after_stat(stratum))`.
```




```
#
xtb_y3 <- valid_df %>%
  dplyr::group_by(caries, fluorosis, fluoride_concentration, Cluster)%>%
  summarise(frequency = n())
```

```
## `summarise()` has grouped output by 'caries', 'fluorosis', 'fluoride_concentration'. You can overrid
```

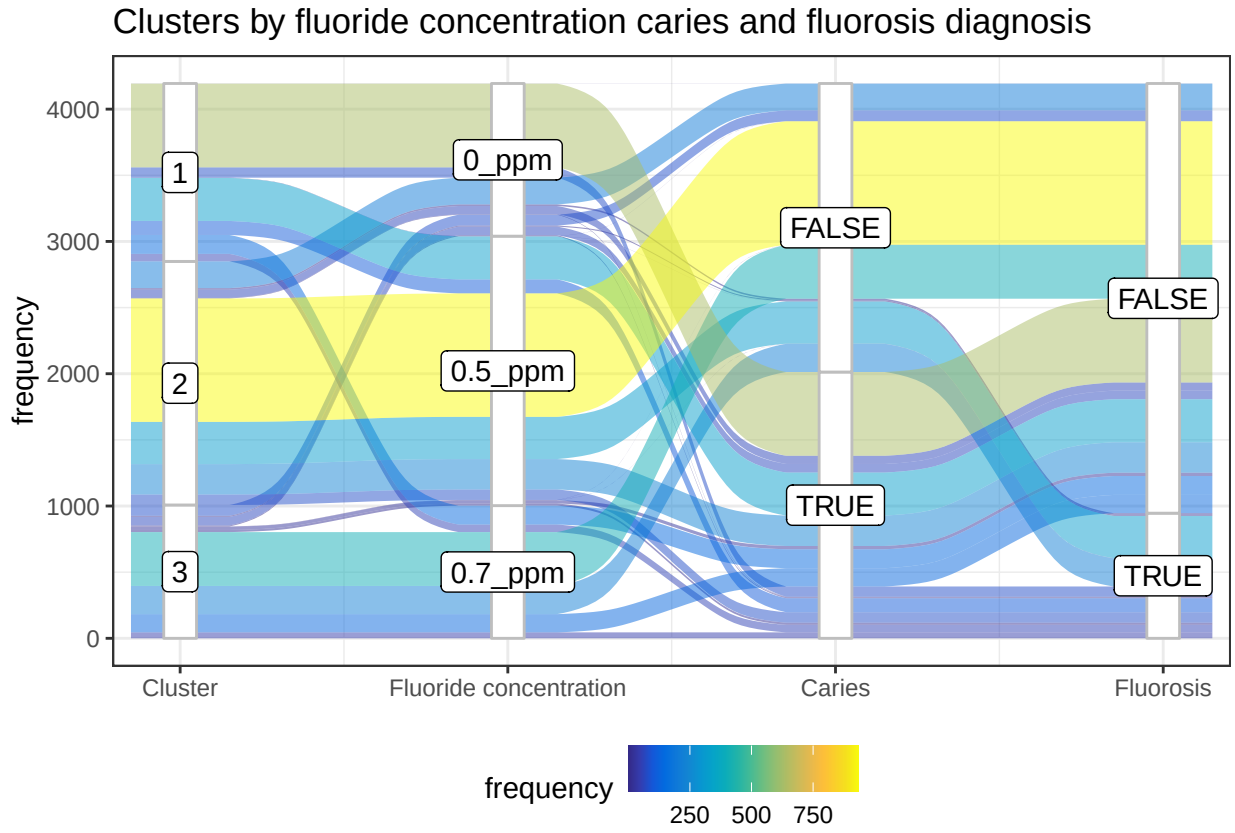
```
xtb_y3 %>% ggplot(aes(y = frequency,
                 axis1 = Cluster,
                 axis2= fluoride_concentration,
                 axis3 = caries,
                 axis4= fluorosis,
                 )) +
  geom_alluvium(aes(fill = frequency), width = 0.3) +
  geom_stratum(width = 1/10,
               fill = "white",
               color = "grey") +
  geom_label(stat = "stratum",
             infer.label = TRUE) +
  scale_x_continuous(breaks = 1:4,
                     labels = c("Cluster", "Fluoride concentration", "Caries", "Fluorosis")) +
  scale_fill_gradientn(colours = pals::parula(n=500))+
  theme_bw()+theme(legend.position="bottom") +
  ggtitle("Clusters by fluoride concentration caries and fluorosis diagnosis")
```

```
## Warning in to_lodes_form(data = data, axes = axis_ind, discern =
## params$discern): Some strata appear at multiple axes.
```

```
## Warning in to_lodes_form(data = data, axes = axis_ind, discern =
## params$discern): Some strata appear at multiple axes.
```

```
## Warning in to_lodes_form(data = data, axes = axis_ind, discern =
## params$discern): Some strata appear at multiple axes.
```

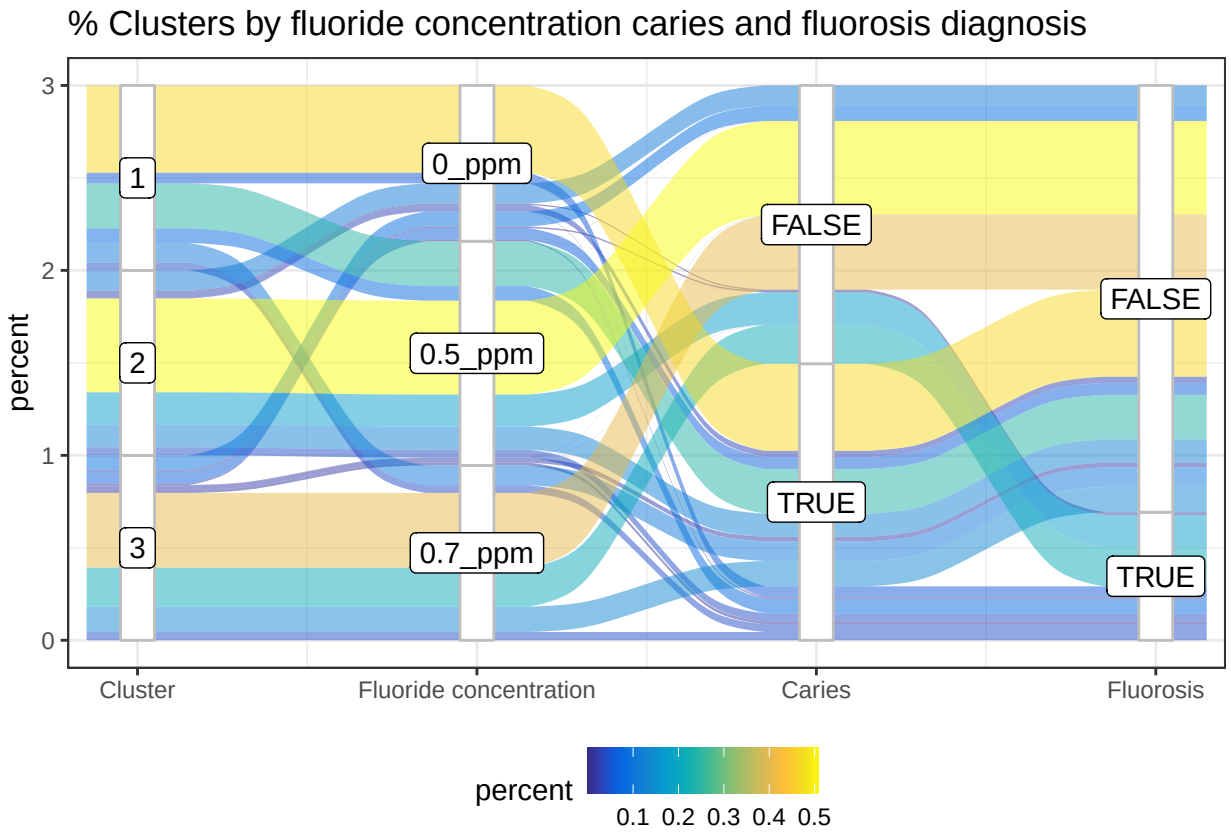
```
## Warning: The parameter `infer.label` is deprecated.
## Use `aes(label = after_stat(stratum))`.
```




```
#
xtb_y3 <- xtb_y3 %>%
  dplyr::group_by(Cluster)%>% mutate(percent = frequency/sum(frequency))
xtb_y3$sum <- xtb_y3$frequency/xtb_y3$percent

xtb_y3 %>% ggplot(aes(y = percent,
                  axis1 = Cluster,
                  axis2= fluoride_concentration,
                  axis3 = caries,
                  axis4= fluorosis,
                  )) +
  geom_alluvium(aes(fill = percent), width = 0.3) +
  geom_stratum(width = 1/10,
               fill = "white",
               color = "grey") +
  geom_label(stat = "stratum",
             infer.label = TRUE) +
  scale_x_continuous(breaks = 1:4,
                     labels = c("Cluster", "Fluoride concentration", "Caries", "Fluorosis")) +
  scale_fill_gradientn(colours = pals::parula(n=500))+
  theme_bw()+theme(legend.position="bottom") +
  ggtitle("% Clusters by fluoride concentration caries and fluorosis diagnosis")
```

```
## Warning in to_lodes_form(data = data, axes = axis_ind, discern =
## params$discern): Some strata appear at multiple axes.
```

```
## Warning in to_lodes_form(data = data, axes = axis_ind, discern =
## params$discern): Some strata appear at multiple axes.

## Warning in to_lodes_form(data = data, axes = axis_ind, discern =
## params$discern): Some strata appear at multiple axes.

## Warning: The parameter 'infer.label' is deprecated.
## Use 'aes(label = after_stat(stratum))'.
```



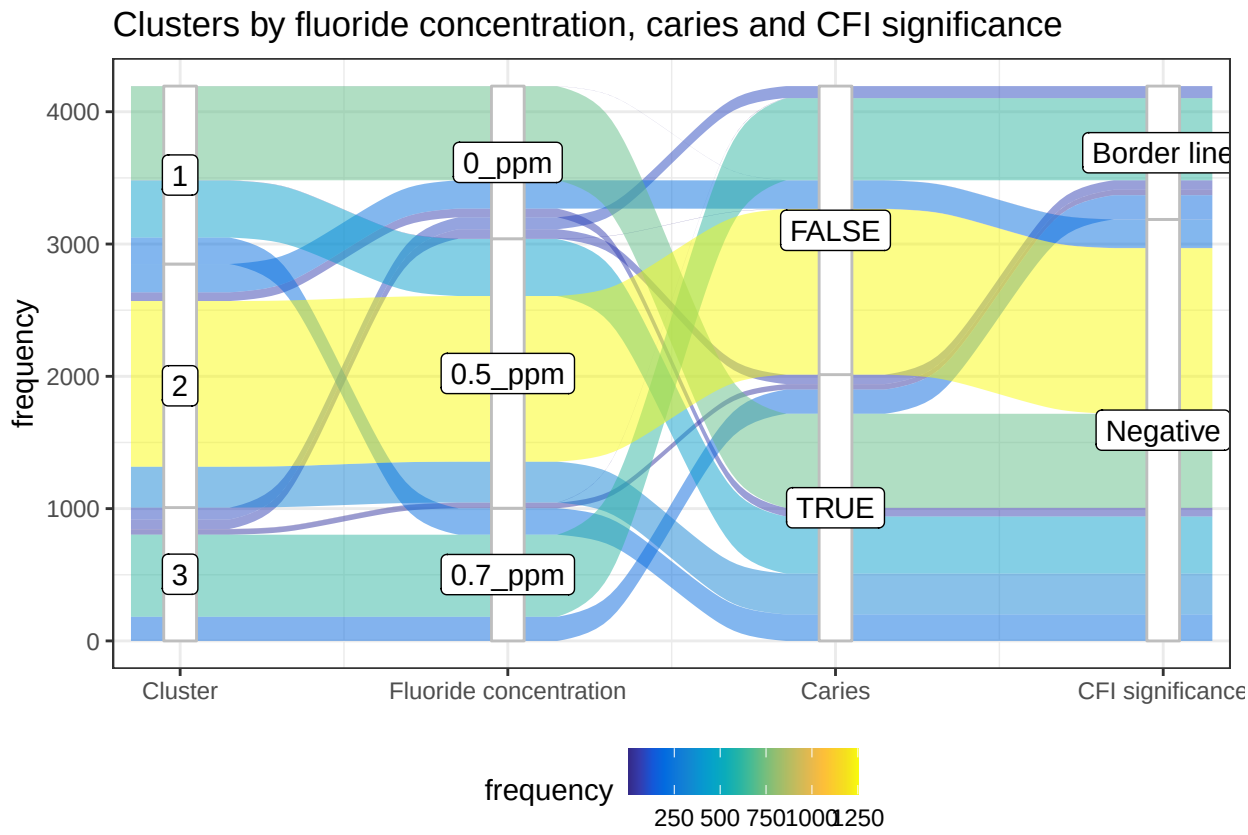

```
#
xtb_y3 <- valid_df %>%
  dplyr::group_by(caries, Significance, fluoride_concentration, Cluster)%>%
  summarise(frequency = n())
```

```
## 'summarise()' has grouped output by 'caries', 'Significance', 'fluoride_concentration'. You can overi
```

```
xtb_y3 %>% ggplot(aes(y = frequency,
                      axis1 = Cluster,
                      axis2= fluoride_concentration,
                      axis3 = caries,
                      axis4= Significance,
                      )) +
  geom_alluvium(aes(fill = frequency), width = 0.3) +
  geom_stratum(width = 1/10,
```

```
                fill = "white",
                color = "grey") +
  geom_label(stat = "stratum",
             infer.label = TRUE) +
  scale_x_continuous(breaks = 1:4,
                     labels = c("Cluster", "Fluoride concentration", "Caries", "CFI significance")) +
  scale_fill_gradientn(colours = pals::parula(n=500))+
  theme_bw()+theme(legend.position="bottom") +
  ggtitle("Clusters by fluoride concentration, caries and CFI significance")
```

```
## Warning: The parameter 'infer.label' is deprecated.
## Use 'aes(label = after_stat(stratum))'.
```




```
#
xtb_y3 <- xtb_y3 %>%
  dplyr::group_by(Cluster)%>% mutate(percent = frequency/sum(frequency))
xtb_y3$sum <- xtb_y3$frequency/xtb_y3$percent

xtb_y3 %>% ggplot(aes(y = percent,
                  axis1 = Cluster,
                  axis2= fluoride_concentration,
                  axis3 = caries,
                  axis4= Significance,
                  )) +
  geom_alluvium(aes(fill = percent), width = 0.3) +
```

```
  geom_stratum(width = 1/10,
               fill = "white",
               color = "grey") +
  geom_label(stat = "stratum",
             infer.label = TRUE) +
  scale_x_continuous(breaks = 1:4,
                     labels = c("Cluster", "Fluoride concentration", "Caries", "CFI significance")) +
  scale_fill_gradientn(colours = pals::parula(n=500))+
  theme_bw()+theme(legend.position="bottom") +
  ggtitle("% Clusters by fluoride concentration, caries and CFI significance")
```

```
## Warning: The parameter 'infer.label' is deprecated.
## Use 'aes(label = after_stat(stratum))'.
```