DASC-5301-ASSIGNMENT

NIVAS

2024-02-07

```
library(purrr)
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
require(stats)
require(graphics)
library(datasets)
library(ggplot2)
library(dplyr)
```

```
##
## 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(psych)
```

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library("ggpubr")
library(zoo)
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✔ forcats   1.0.0     ✔ stringr   1.5.1
## ✔ lubridate 1.9.3     ✔ tibble    3.2.1
## ✔ readr     2.1.5     ✔ tidyr     1.3.1
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ psych::%+%()    masks ggplot2::%+%()
## ✖ psych::alpha()  masks ggplot2::alpha()
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
mydata<-datasets::swiss
head(mydata)
```

```
##              Fertility Agriculture Examination Education Catholic
## Courtelary        80.2        17.0          15        12     9.96
## Delemont          83.1        45.1           6         9    84.84
## Franches-Mnt      92.5        39.7           5         5    93.40
## Moutier           85.8        36.5          12         7    33.77
## Neuveville        76.9        43.5          17        15     5.16
## Porrentruy        76.1        35.3           9         7    90.57
##              Infant.Mortality
## Courtelary               22.2
## Delemont                 22.2
## Franches-Mnt             20.2
## Moutier                  20.3
## Neuveville               20.6
## Porrentruy               26.6
```

```
describe(mydata)
```

```
##                  vars  n  mean    sd median trimmed   mad   min   max range
## Fertility           1 47 70.14 12.49  70.40   70.66 10.23 35.00  92.5 57.50
## Agriculture         2 47 50.66 22.71  54.10   51.16 23.87  1.20  89.7 88.50
## Examination         3 47 16.49  7.98  16.00   16.08  7.41  3.00  37.0 34.00
## Education           4 47 10.98  9.62   8.00    9.38  5.93  1.00  53.0 52.00
## Catholic            5 47 41.14 41.70  15.14   39.12 18.65  2.15 100.0 97.85
## Infant.Mortality    6 47 19.94  2.91  20.00   19.98  2.82 10.80  26.6 15.80
##                   skew kurtosis   se
## Fertility        -0.46     0.26 1.82
## Agriculture      -0.32    -0.89 3.31
## Examination       0.45    -0.14 1.16
## Education         2.27     6.14 1.40
## Catholic          0.48    -1.67 6.08
## Infant.Mortality -0.33     0.78 0.42
```

```
# find location of missing values
missing_value <- which(is.na(mydata))
sprintf("Position of missing values : %d", missing_value)
```

```
## character(0)
```

```
paste(which(is.na(mydata)))
```

```
## character(0)
```

```
# count total missing values
sum(is.na(mydata))
```

```
## [1] 0
```

```
sprintf("Sum of missing values : %d", sum(is.na(mydata)))
```

```
## [1] "Sum of missing values : 0"
```

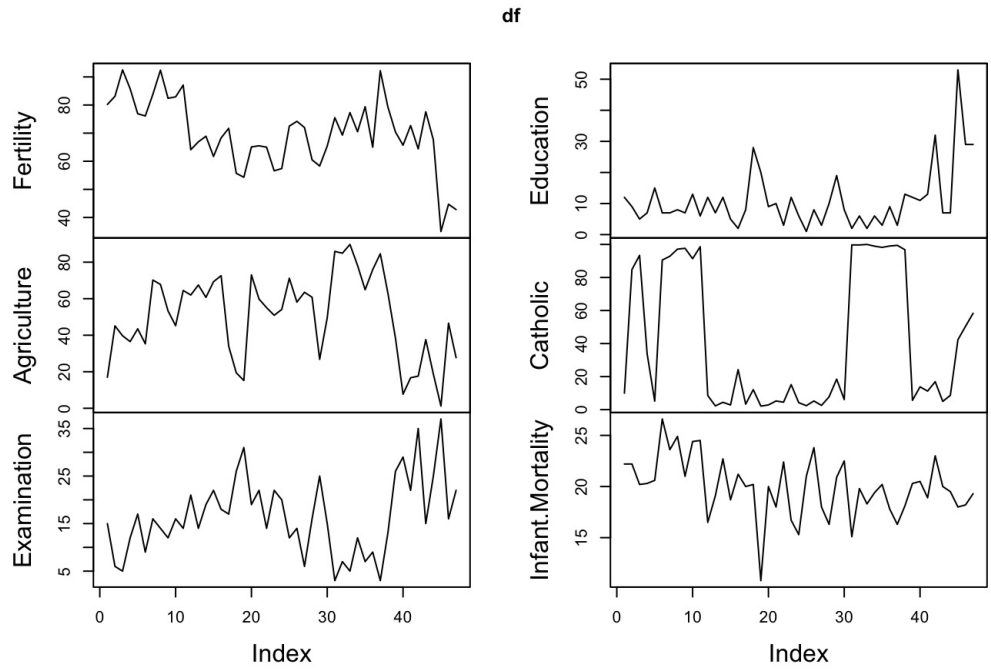
```
#Shape of Swiss Data
glimpse(mydata)
```

```
## Rows: 47
## Columns: 6
## $ Fertility        <dbl> 80.2, 83.1, 92.5, 85.8, 76.9, 76.1, 83.8, 92.4, 82.4,…
## $ Agriculture      <dbl> 17.0, 45.1, 39.7, 36.5, 43.5, 35.3, 70.2, 67.8, 53.3,…
## $ Examination      <int> 15, 6, 5, 12, 17, 9, 16, 14, 12, 16, 14, 21, 14, 19, …
## $ Education        <int> 12, 9, 5, 7, 15, 7, 7, 8, 7, 13, 6, 12, 7, 12, 5, 2, …
## $ Catholic         <dbl> 9.96, 84.84, 93.40, 33.77, 5.16, 90.57, 92.85, 97.16,…
## $ Infant.Mortality <dbl> 22.2, 22.2, 20.2, 20.3, 20.6, 26.6, 23.6, 24.9, 21.0,…
```

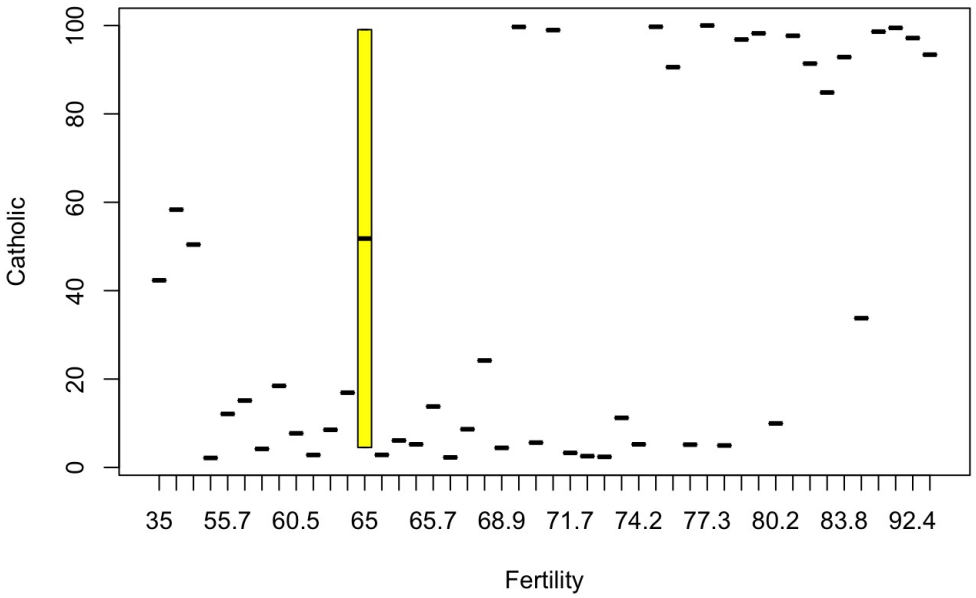
```
dim(mydata)
```

```
## [1] 47  6
```

```
df <- zoo(mydata)
plot(df)
```



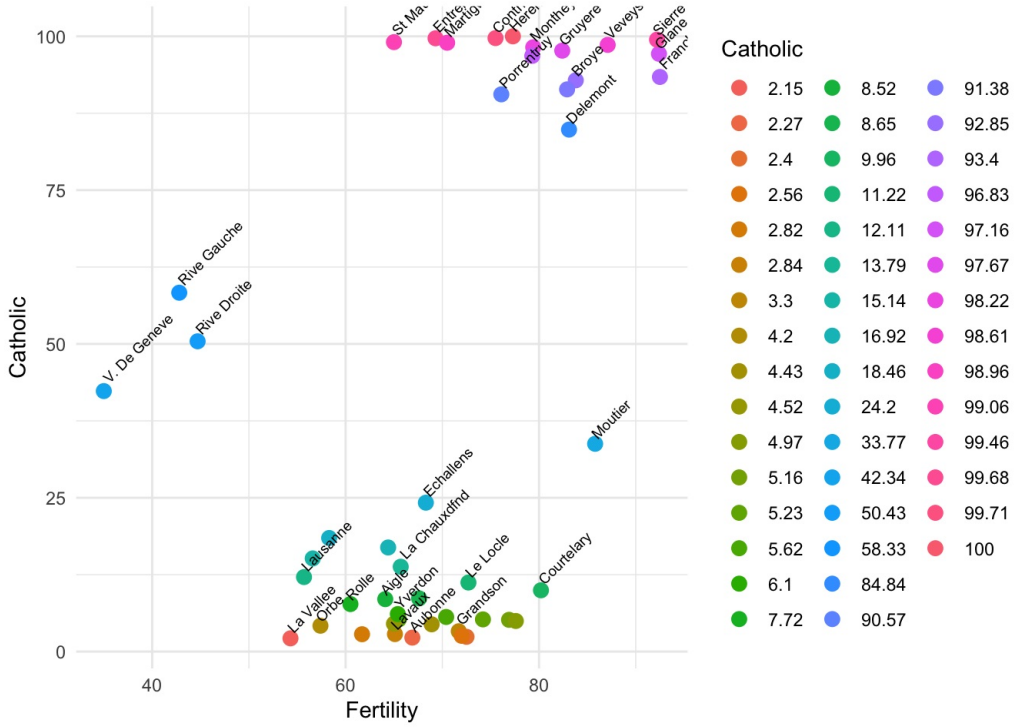

```
# Comparing Catholic and Fertility Attribute in Swiss Dataset
boxplot(Catholic~Fertility, data=mydata, col = "yellow")
```




```
cor(mydata$Catholic,mydata$Fertility)
```

```
## [1] 0.4636847
```

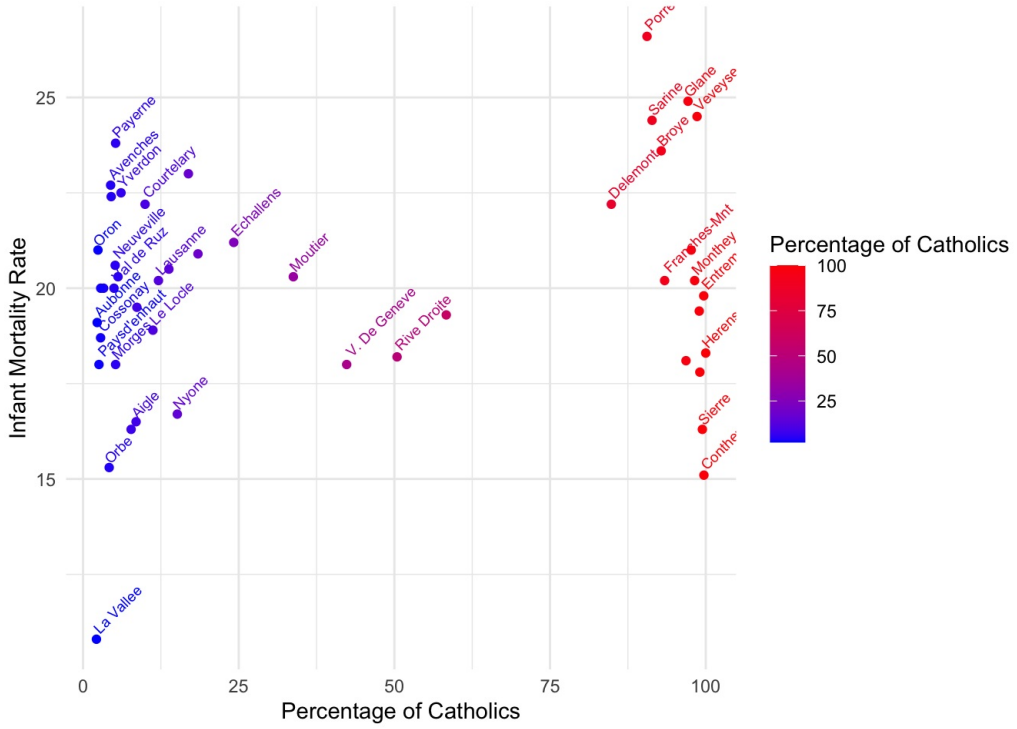
```
ggplot(mydata, aes(x = Fertility, y = Catholic)) +
  geom_point(aes(color = factor(Catholic)), size = 3) +
  geom_text(aes(label = row.names(mydata)), size = 2.5, angle = 45, vjust = -0.10, hjust = -0.10, check_overlap =
TRUE)+
  scale_color_discrete(name = "Catholic") +
  labs(x = "Fertility", y = "Catholic") +
  theme_minimal()
```




```
# Comparing Catholic and Infant.Mortality Attribute in Swiss Dataset
cor(mydata$Catholic,mydata$Infant.Mortality)
```

```
## [1] 0.1754959
```

```
ggplot(mydata, aes(x = Catholic, y = Infant.Mortality, color = Catholic)) +
  geom_point() +
  geom_text(aes(label = row.names(mydata)), size = 2.5, angle = 45, vjust = -0.10, hjust = -0.10, check_overlap =
TRUE) +
  scale_color_gradient(low = "blue", high = "red") +  # Move this line before geom_text()
  labels(x = "Percentage of Catholics", y = "Infant Mortality Rate", color = "Percentage of Catholics") +
  theme_minimal()
```



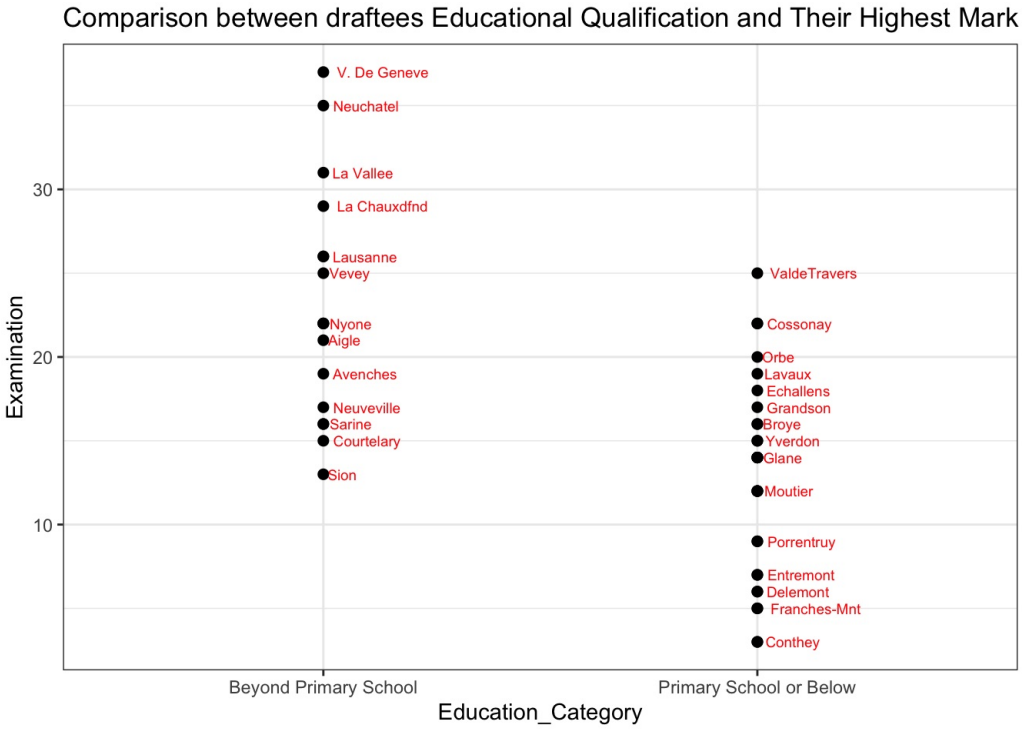

```
# Create a new variable to categorize Examination scores
cor(mydata$Education,mydata$Examination)
```

```
## [1] 0.6984153
```

```
partial_data <- mydata %>%
  select(
    Education = 'Education',
    Examination = 'Examination'
  )
partial_data$Education_Category <- ifelse(partial_data$Education > mean(partial_data$Education), "Beyond Primary
School", "Primary School or Below")
mean(partial_data$Education)
```

```
## [1] 10.97872
```

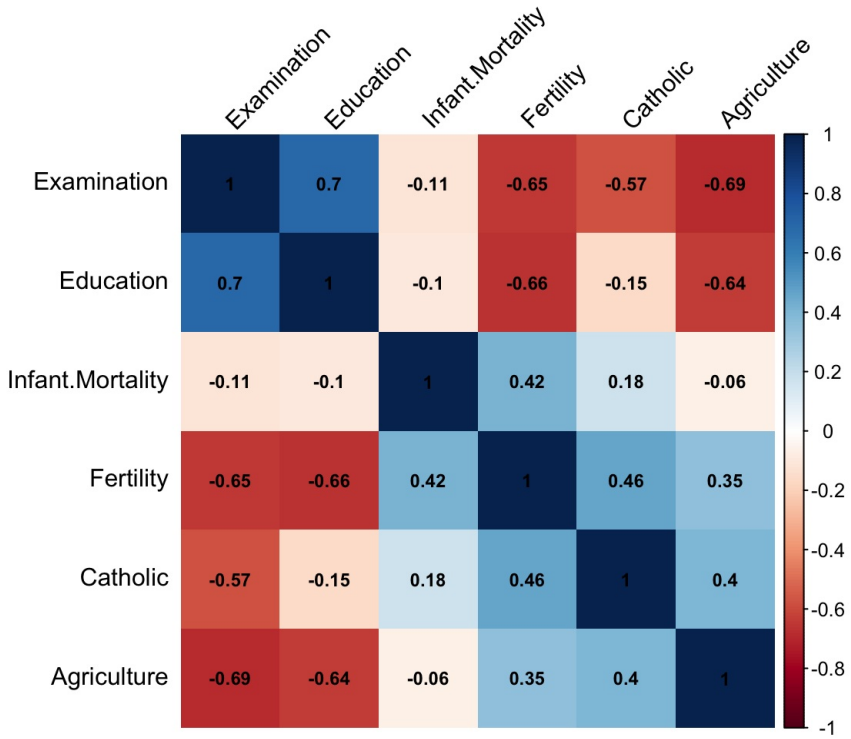
```
# Create a bar plot
theme_set(theme_bw())
ggplot(partial_data, aes(x = Examination, y = Education_Category, label = row.names(mydata))) +
  geom_point(stat = 'identity', fill = "black", size = 2) +
  geom_text(color = "red", size = 2.5, hjust = -0.15, check_overlap = TRUE) +
  labs(title = "Comparison between draftees Educational Qualification and Their Highest Mark", ) +
  coord_flip()
```




```
#Significant Correlation In Swiss Data
round <- round(cor(mydata),
      digits = 2 # rounded to 2 decimals
)
round
```

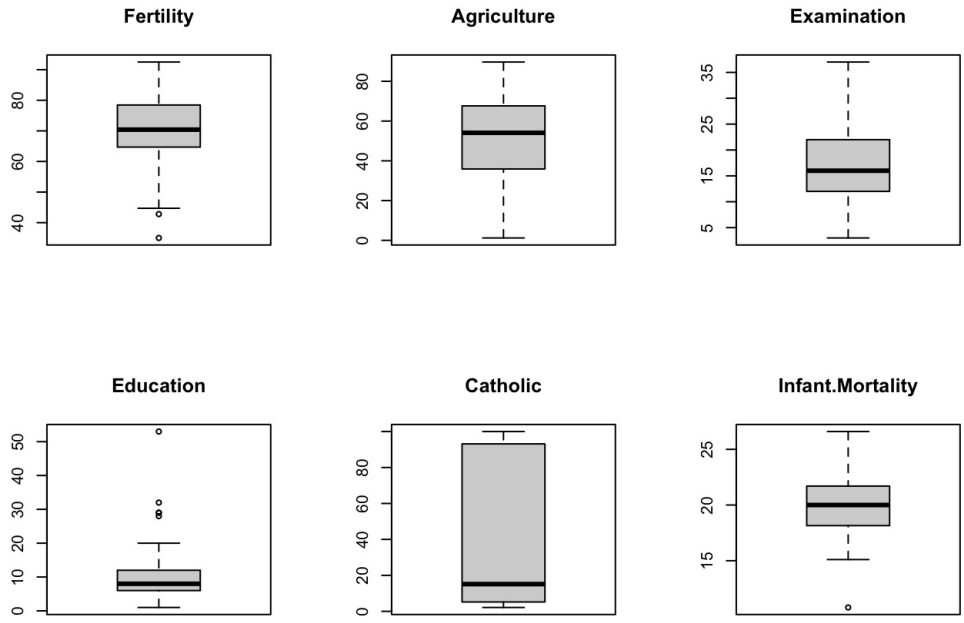
```
##                  Fertility Agriculture Examination Education Catholic
## Fertility             1.00        0.35       -0.65     -0.66     0.46
## Agriculture           0.35        1.00       -0.69     -0.64     0.40
## Examination          -0.65       -0.69        1.00      0.70    -0.57
## Education            -0.66       -0.64        0.70      1.00    -0.15
## Catholic              0.46        0.40       -0.57     -0.15     1.00
## Infant.Mortality      0.42       -0.06       -0.11     -0.10     0.18
##                  Infant.Mortality
## Fertility                    0.42
## Agriculture                 -0.06
## Examination                 -0.11
## Education                   -0.10
## Catholic                     0.18
## Infant.Mortality             1.00
```

```
corrplot(cor(mydata), method="shade",shade.col=NA, tl.col="black", tl.srt=45, addCoef.col="black",
         order = "AOE",
         number.cex=0.75)
```




```
# Outlier from each column

par(mfrow=c(2,3))
for (i in 1:length(mydata)) {
  boxplot(mydata[,i], main=names(mydata[i]), type="l")
}
```




```
par(mfrow=c(1,1))
outliers <- function(dataframe){
  dataframe %>%
    select_if(is.numeric) %>%
    map(~ boxplot.stats(.x)$out)

}
outliers(mydata)
```

```
## $Fertility
## [1] 35.0 42.8
##
## $Agriculture
## numeric(0)
##
## $Examination
## integer(0)
##
## $Education
## [1] 28 32 53 29 29
##
## $Catholic
## numeric(0)
##
## $Infant.Mortality
## [1] 10.8
```