LAB: Bivariate analysis

2024-01-31

- **M1 MIDS & MFA**
- **Université Paris Cité**
- Année 2023-2024
- Course Homepage

- Moodle

```
to_be_loaded <- c("tidyverse",
                  "glue",
                  "magrittr",
                  "lobstr",
                  "arrow",
                  "ggforce",
                  "vcd",
                  "ggmosaic",
                  "httr",
                  "cowplot",
                  "patchwork"
)

for (pck in to_be_loaded) {
  if (!require(pck, character.only = T)) {
    install.packages(pck, repos="http://cran.rstudio.com/")
    stopifnot(require(pck, character.only = T))
  }
}
```

Objectives

In Exploratory analysis of tabular data, bivariate analysis is the second step. It consists in exploring, summarizing, visualizing pairs of columns of a dataset.

Bivariate techniques depend on the types of columns we are facing.

For *numerical/numerical* samples

- Scatter plots

- Smoothed lineplots (for example linear regression)
- 2-dimensional density plots

For *categorical/categorical* samples : mosaicplots and variants

For *numerical/categorical* samples

- Boxplots per group
- Histograms per group
- Density plots per group
- Quantile-Quantile plots

Dataset

Once again we rely on the Census dataset.

> Since 1948, the US Census Bureau carries out a monthly Current Population Survey, collecting data concerning residents aged above 15 from 150000 households. This survey is one of the most important sources of information concerning the american workforce. Data reported in file `Recensement.txt` originate from the 2012 census.

Load the data into the session environment and call it `df`. Take advantage of the fact that we saved the result of our data wrangling job in a self-documented file format. Download a `parquet` file from the following URL:

`https://stephane-v-boucheron.fr/data/Recensement.parquet`

Use `httr::GET()` and `WriteBin()`.

Solution

```
# arrow::write_parquet(df, "")
```

```
fname <- "Recensement.parquet"
datapath <- "./DATA"
fpath <- paste(datapath, fname, sep="/")

if (!file.exists(fpath)) {
  tryCatch(expr = {
    url <- 'https://stephane-v-boucheron.fr/data/Recensement.parquet'

    rep <- httr::GET(url)
    stopifnot(rep$status_code==200)

    con <- file(fpath, open="wb")
    writeBin(rep$content, con)
    close(con)
  }, warning = function(w) {
    glue("Successful download but {w}")
  }, error = function(e) {
    stop("Houston, we have a problem!")    # error-handler-code
  }, finally = {
    if (exists("con") && isOpen(con)){
      close(con)
    }
  }
  )
}

df <- arrow::read_parquet(fpath)
```

```
df |>
  glimpse()
## Rows: 599
## Columns: 11
## $ AGE         <dbl> 58, 40, 29, 33, 31, 19, 64, 23, 47, 66, 26, 23, 54, 44, 56,~
## $ SEXE        <fct> F, M, M, M, M, M, F, F, M, F, M, F, F, F, F, F, F, M, M, F,~
## $ REGION      <fct> NE, W, S, NE, W, NW, S, NE, NW, S, NE, NE, W, NW, S, S, NW,~
## $ STAT_MARI   <fct> C, M, C, D, M, C, M, C, M, D, M, C, M, C, M, C, S, M, S, C,~
```

Categorical/Categorical pairs

```
df |>
  select(where(is.factor)) |>
  head()
```

```
# A tibble: 6 x 9
  SEXE  REGION STAT_MARI SYNDICAT CATEGORIE  NIV_ETUDES NB_PERS NB_ENF REV_FOYER
  <fct> <fct>  <fct>     <fct>    <fct>      <fct>      <fct>   <fct>  <fct>
1 F     NE     C         non      "Administ~ Bachelor   2       0      [35000-4~
2 M     W      M         non      "Building~ 12 years ~ 2       0      [17500-2~
3 M     S      C         non      "Administ~ Associate~ 2       0      [75000-1~
4 M     NE     D         oui      "Services" 12 years ~ 4       1      [17500-2~
5 M     W      M         non      "Services" 9 years s~ 8       1      [75000-1~
6 M     NW     C         non      "Services" 12 years ~ 6       0      [1e+05-1~
```

Explore the connection between `CATEGORIE` and `SEX`. Compute the 2-ways contingency table using `table()`, and `count()` from `dplyr`.

Use `tibble::as_tibble()` to transform the output of `table()` into a dataframe/tibble.

Use `tidyr::pivot_wider()` so as to obtain a wide (but messy) tibble with the same the same shape as the output of `table()`. Can you spot a difference?

Solution

```
tb <- df |>
  dplyr::select(CATEGORIE, SEXE) |>
  table()

tb
```

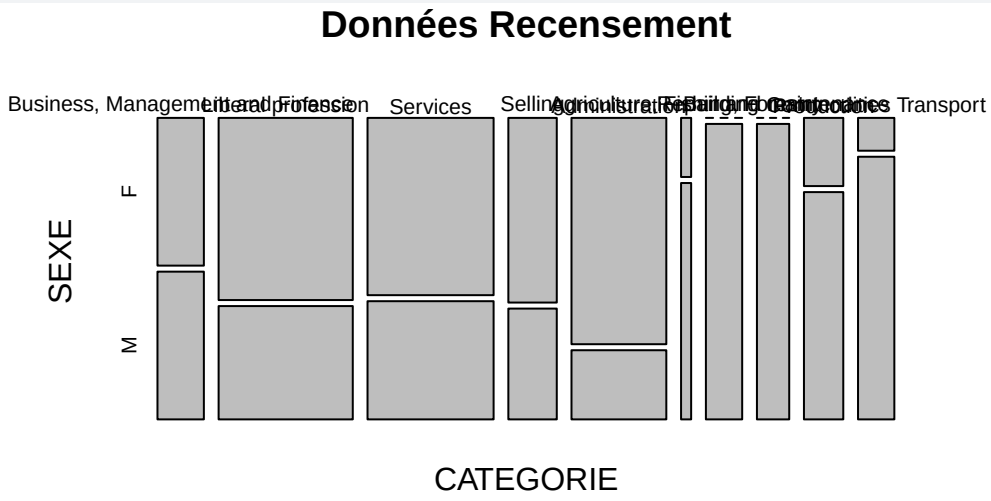
```
                                SEXE
CATEGORIE                        F  M
  Business, Management and Finance 23 23
  Liberal profession              82 51
  Services                        75 50
  Selling                         30 18
  Administration                  72 22
  Agriculture, Fishing, Forestry   2  8
  Building                         0 36
  Repair and maintenance           0 32
  Production                       9 30
  Commodities Transport            4 32
```

```
tb2 <- df |>
  count(CATEGORIE, SEXE)

tb2
```

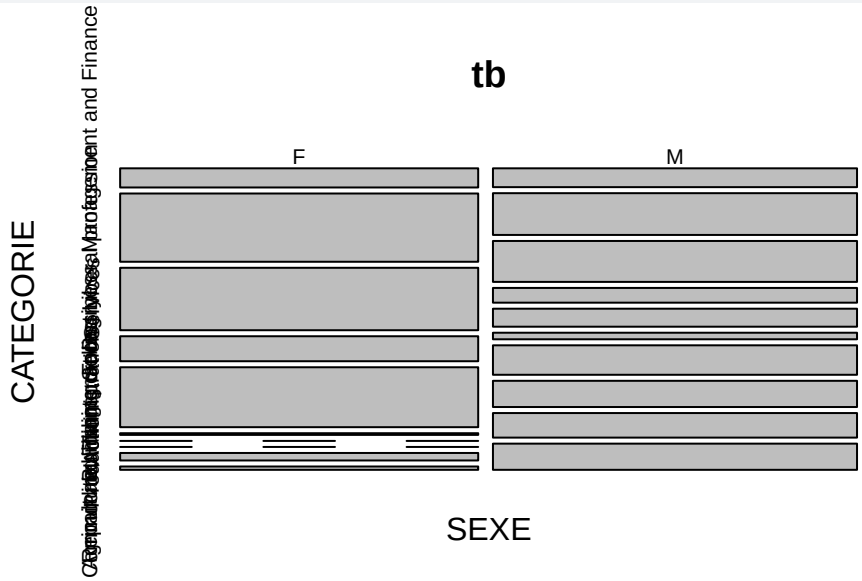
```
# A tibble: 18 x 3
   CATEGORIE                          SEXE      n
   <fct>                              <fct> <int>
 1 "Business, Management and Finance" F        23
 2 "Business, Management and Finance" M        23
 3 "Liberal profession"               F        82
 4 "Liberal profession"               M        51
 5 "Services"                         F        75
 6 "Services"                         M        50
 7 "Selling"                          F        30
 8 "Selling"                          M        18
 9 "Administration"                   F        72
10 "Administration"                   M        22
```

Use `mosaicplot()` from base `R` to visualize the contingency table.

```
mosaicplot(~ CATEGORIE + SEXE,
           tb,
           main="Données Recensement")
```



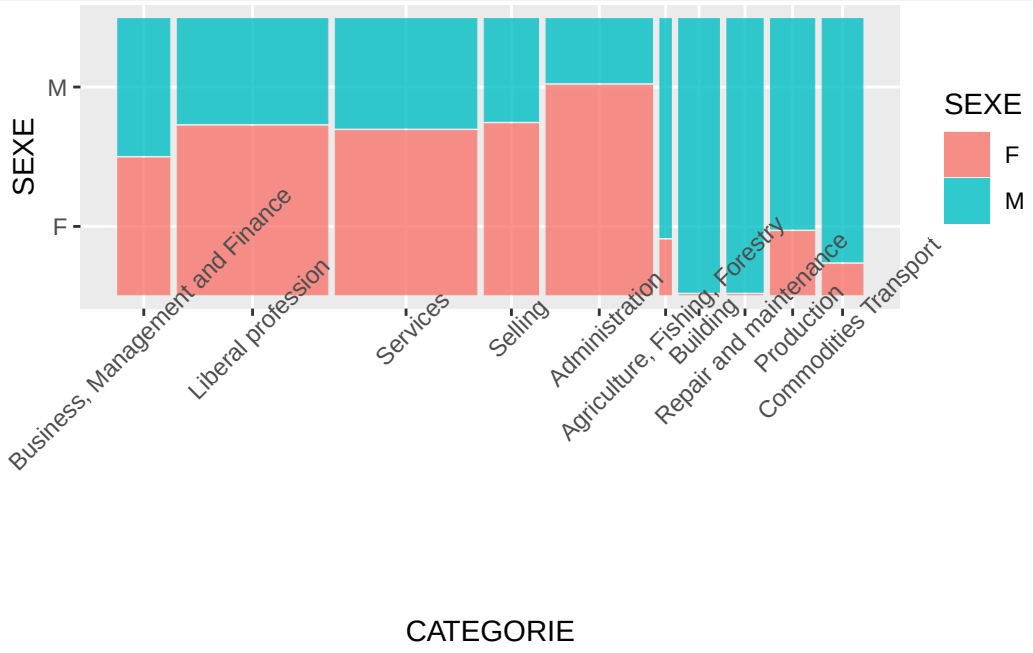

```
mosaicplot(~ SEXE + CATEGORIE, tb)
```



Use `geom_mosaic` from `ggmosaic` to visualize the contingency table

- Make the plot as readable as possible
- Reorder `CATEGORIE` acccording to counts

```r
rot_x_text <- theme(axis.text.x = element_text(angle = 45))

df |>
  ggplot() +
  geom_mosaic(aes(x=product(SEXE, CATEGORIE), fill=SEXE)) +
  rot_x_text
```



- Collapse rare levels of `CATEGORIE` (consider that a level is rare if it has less than 40 occurrences). Use tools from `forcats`.

Solution

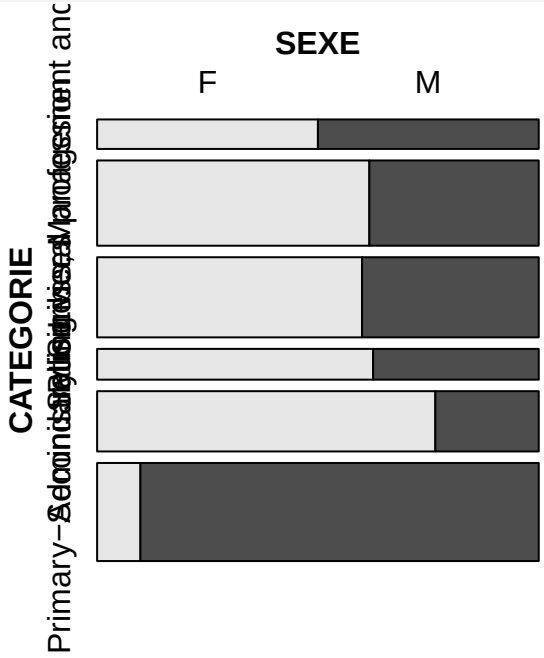
```
df |>
  count(CATEGORIE) |>
  arrange(desc(n))
```

```
# A tibble: 10 x 2
   CATEGORIE                              n
   <fct>                              <int>
 1 "Liberal profession"                 133
 2 "Services"                           125
 3 "Administration"                      94
 4 "Selling"                             48
 5 "Business, Management and Finance"    46
 6 "Production"                          39
 7 "Building "                           36
 8 "Commodities Transport"               36
 9 "Repair and maintenance"              32
10 "Agriculture, Fishing, Forestry"      10
```

```
rare_categories <- df |>
  count(CATEGORIE) |>
  filter(n<=40)

rare_categories
```

```
# A tibble: 5 x 2
```

```
vcd::mosaic(formula=SEXE~CATEGORIE,
            data=table(select(df, CATEGORIE, SEXE)))
```



Testing association

Chi-square independence/association test

```r
test_1 <- df |>
  select(CATEGORIE, SEXE) |>
  table() |>
  chisq.test()

test_1
```

```
    Pearson's Chi-squared test

data:  table(select(df, CATEGORIE, SEXE))
X-squared = 140.67, df = 5, p-value < 2.2e-16
```

```r
test_1 |>
  broom::tidy()
```

```
# A tibble: 1 x 4
  statistic  p.value parameter method
      <dbl>    <dbl>     <int> <chr>
1      141. 1.29e-28         5 Pearson's Chi-squared test
```

The Chi-square statistics can be computed from the contingeny table

```r
rowcounts <- apply(tb, MARGIN = 1, FUN = sum)
colcounts <- apply(tb, MARGIN = 2, FUN = sum)

expected <- (rowcounts %*% t(colcounts))/sum(colcounts)

norm((tb - expected) / sqrt(expected), type = "F")^2
```

```
[1] 140.6717
```

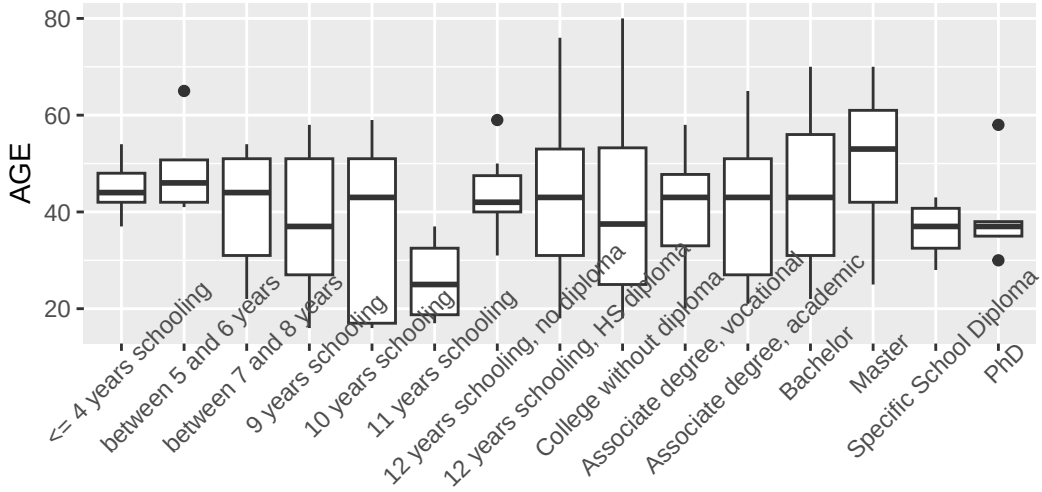
```r
# expected <- (tb |>
#   vcd::independence_table())
```

Categorical/Numerical pairs

Grouped boxplots

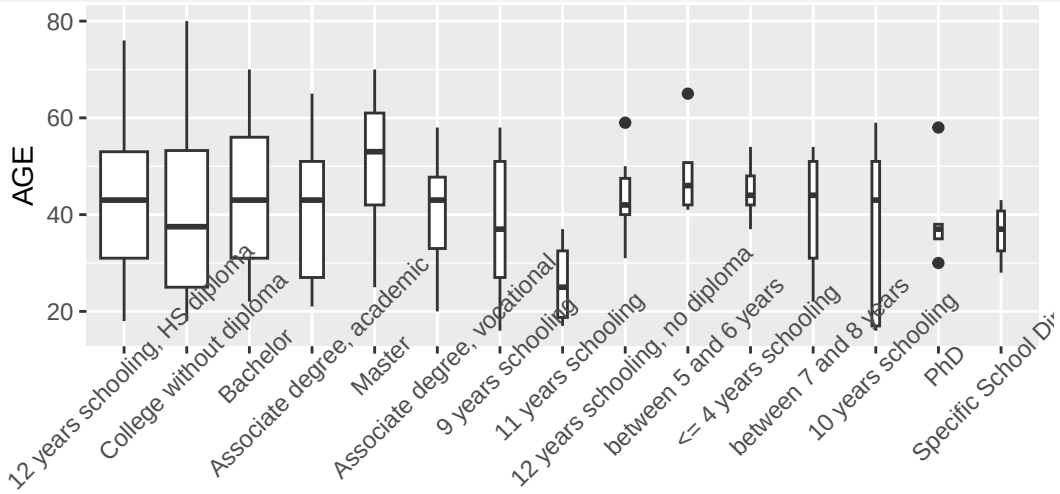
Plot boxplots of `AGE` according to `NIV_ETUDES`

```
df |>
  ggplot() +
  aes(x=NIV_ETUDES, y=AGE) +
  geom_boxplot() +
  rot_x_text
```



NIV_ETUDES

```
df |>
  ggplot() +
  aes(x=fct_infreq(NIV_ETUDES), y=AGE) +
  geom_boxplot(varwidth = T) +
  rot_x_text
```

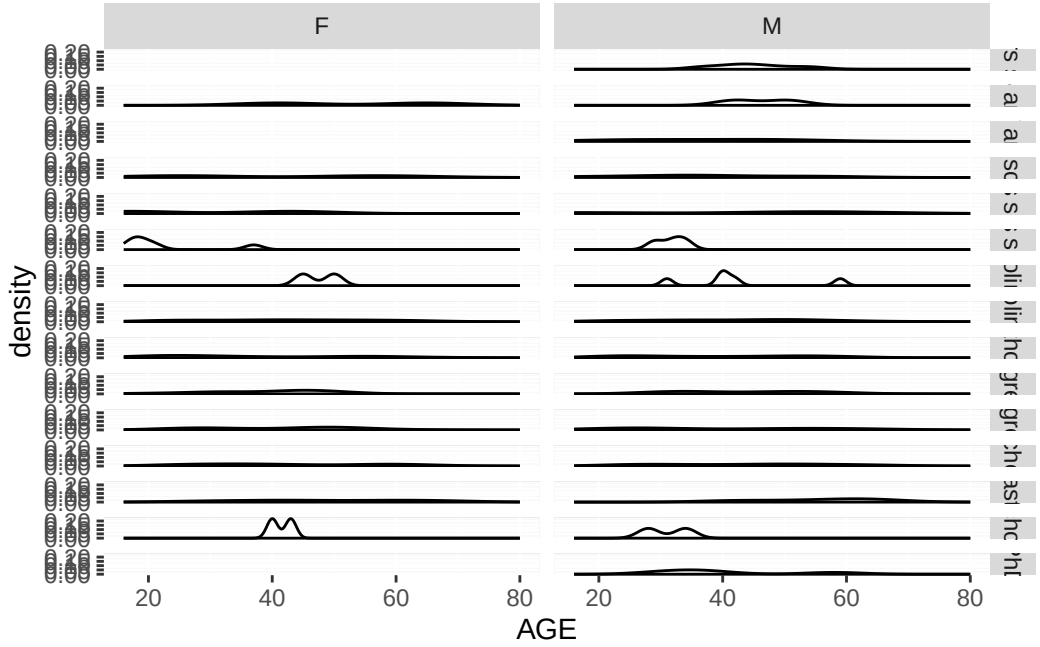


fct_infreq(NIV_ETUDES)

Draw density plots of `AGE`, facet by `NIV_ETUDES` and `SEXE`

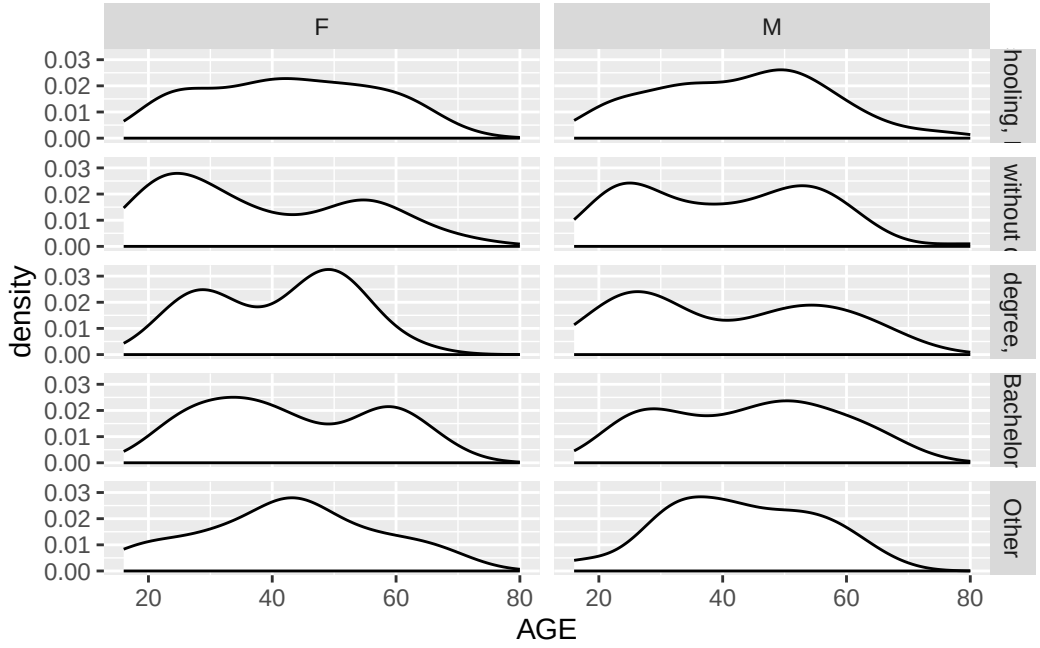
```
p <- df |>
  ggplot() +
  aes(x=AGE) +
  stat_density(fill="white", color="black") +
  facet_grid(rows=vars(NIV_ETUDES),
             cols=vars(SEXE))

p
```

```
Warning: Groups with fewer than two data points have been dropped.
Groups with fewer than two data points have been dropped.
Warning: Removed 2 rows containing missing values (`position_stack()`).
```



Collapse rare levels of `NIV_ETUDES` and replay.

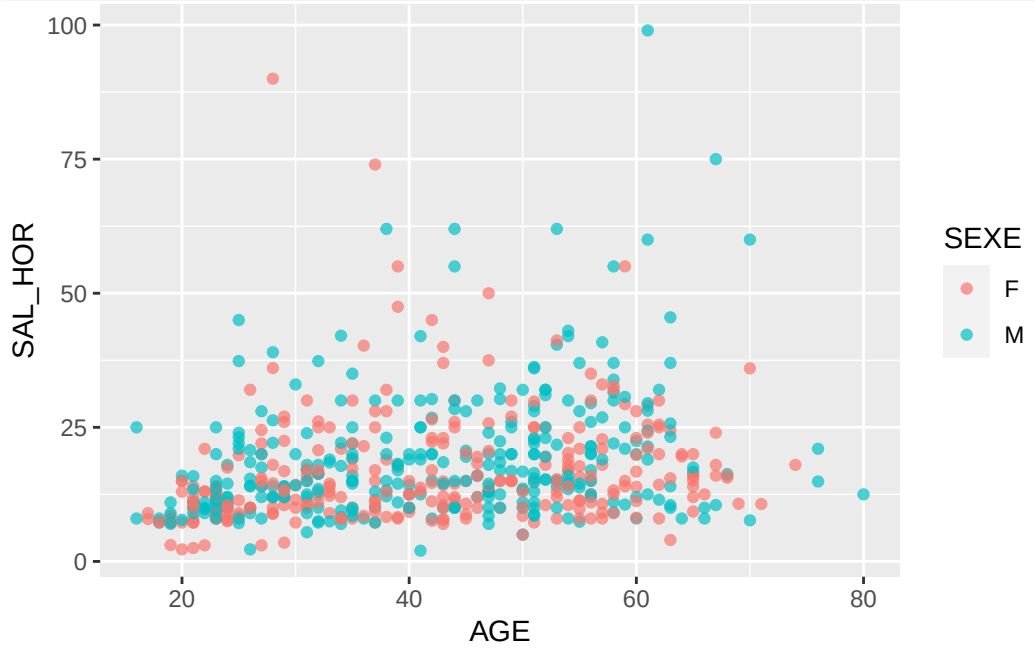
```
p %+% (df |>
  mutate(NIV_ETUDES = fct_lump_min(NIV_ETUDES, min=30)) )
```



Numerical/Numerical pairs

Make a scatterplot of `SAL_HOR`with respect to `AGE`

```r
df |>
  ggplot() +
  aes(x=AGE, y=SAL_HOR, color=SEXE) +
  geom_point(alpha=.7)
```



`pairs` from base `R`

`ggpairs()`

Useful links

- rmarkdown
- dplyr
- ggplot2
- *R Graphic Cookbook*. Winston Chang. O' Reilly.
- A blog on ggplot object
- `skimr`
- `vcd`
- `ggmosaic`
- `ggforce`
- `arrow`
- `httr`