Bivariate analysis

2025-01-20

M1 MIDS/MFA/LOGOS
Université Paris Cité
Année 2024
Course Homepage
Moodle

> ! **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.

Setup

```r
stopifnot(
  require(tidyverse),
  require(glue),
  require(magrittr),
  require(lobstr),
  require(arrow),
  require(ggforce),
  require(vcd),
  require(ggmosaic),
  require(httr),
  require(patchwork)
)
```

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

```
fname <- "Recensement.parquet"

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, 59, 51, 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,~
## $ SAL_HOR    <dbl> 13.25, 12.50, 14.00, 10.60, 13.00, 7.00, 19.57, 13.00, 20.1~
## $ SYNDICAT   <fct> non, non, non, oui, non, non, non, non, oui, non, non, non,~
## $ CATEGORIE  <fct> "Administration", "Building ", "Administration", "Services"~
## $ NIV_ETUDES <fct> "Bachelor", "12 years schooling, no diploma", "Associate de~
## $ NB_PERS    <fct> 2, 2, 2, 4, 8, 6, 3, 2, 3, 1, 3, 2, 6, 5, 4, 4, 3, 2, 3, 2,~
## $ NB_ENF     <fct> 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0,~
## $ REV_FOYER  <fct> [35000-40000), [17500-20000), [75000-1e+05), [17500-20000),~

df |>
  head()
## # A tibble: 6 x 11
##     AGE SEXE  REGION STAT_MARI SAL_HOR SYNDICAT CATEGORIE      NIV_ETUDES NB_PERS
##   <dbl> <fct> <fct>  <fct>       <dbl> <fct>    <fct>          <fct>      <fct>
## 1    58 F     NE     C            13.2 non      "Administrat~ Bachelor   2
```

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
```

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

tb2
```

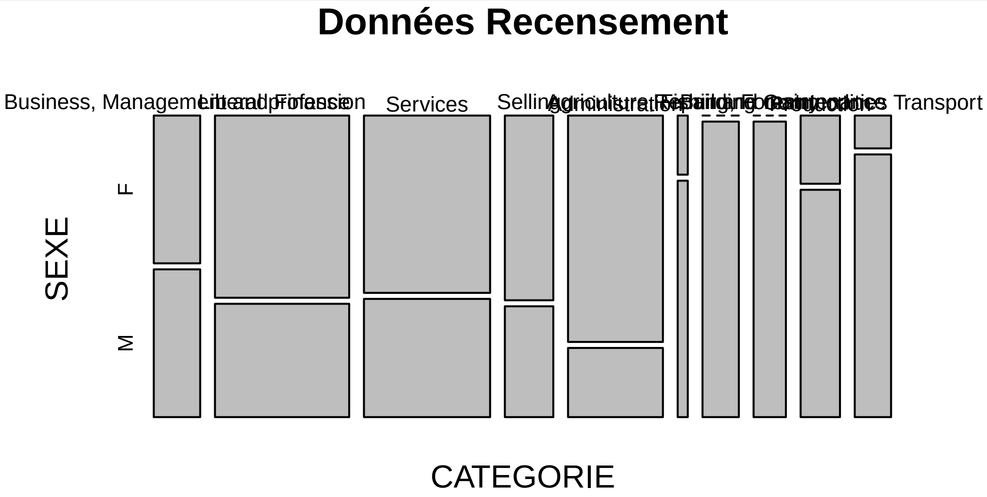
```
# A tibble: 18 x 3
   CATEGORIE                          SEXE      n
   <fct>                              <fct> <int>
 1 "Business, Management and Finance" F        23
 2 "Business, Management and Finance" M        23
 3 "Liberal profession"               F        82
 4 "Liberal profession"               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
11 "Agriculture, Fishing, Forestry"   F         2
12 "Agriculture, Fishing, Forestry"   M         8
13 "Building "                        M        36
14 "Repair and maintenance"           M        32
15 "Production"                       F         9
16 "Production"                       M        30
17 "Commodities Transport"            F         4
18 "Commodities Transport"            M        32
```

```
tb2 |>
  pivot_wider(id_cols=CATEGORIE,
              names_from=SEXE,
              values_from=n)
```

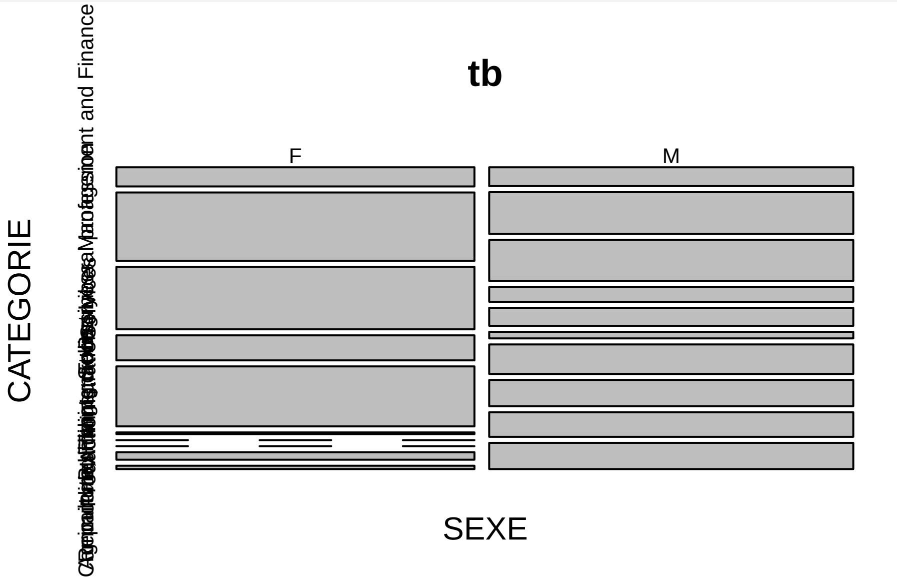
```
# A tibble: 10 x 3
   CATEGORIE                              F     M
   <fct>                              <int> <int>
 1 "Business, Management and Finance"    23    23
 2 "Liberal profession"                  82    51
 3 "Services"                            75    50
 4 "Selling"                             30    18
 5 "Administration"                      72    22
 6 "Agriculture, Fishing, Forestry"       2     8
 7 "Building "                           NA    36
 8 "Repair and maintenance"              NA    32
 9 "Production"                           9    30
10 "Commodities Transport"                4    32
```

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

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



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



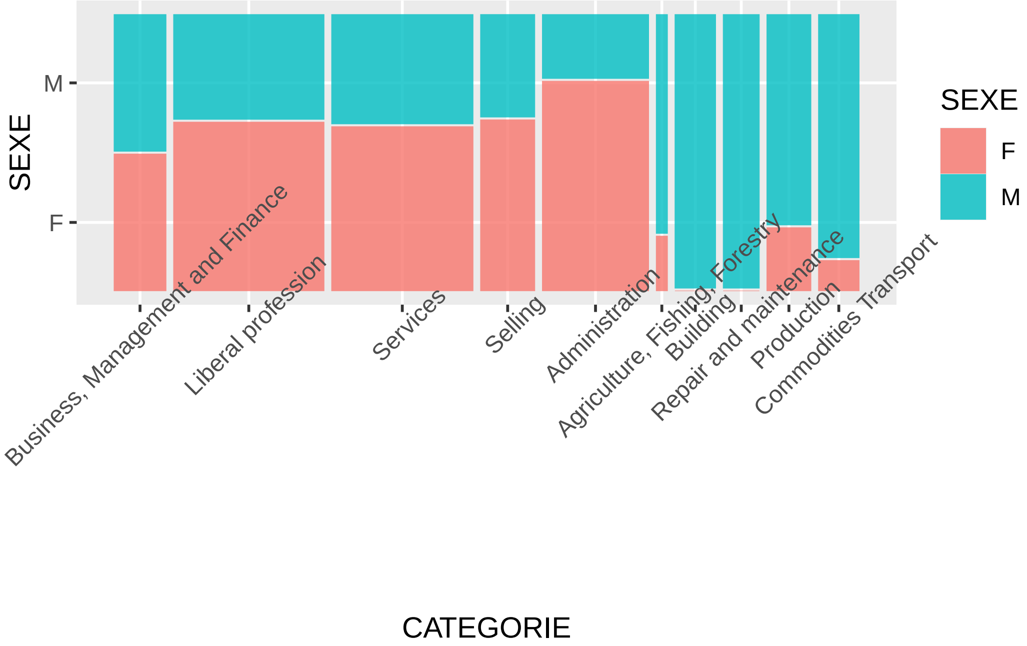
Use `geom_mosaic` from `ggmosaic` to visualize the contingency table

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

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

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

```
Warning: The `scale_name` argument of `continuous_scale()` is deprecated as of ggplot2
3.5.0.
Warning: The `trans` argument of `continuous_scale()` is deprecated as of ggplot2 3.5.
i Please use the `transform` argument instead.
Warning: `unite_()` was deprecated in tidyr 1.2.0.
i Please use `unite()` instead.
i The deprecated feature was likely used in the ggmosaic package.
  Please report the issue at <https://github.com/haleyjeppson/ggmosaic>.
```



- 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))
```

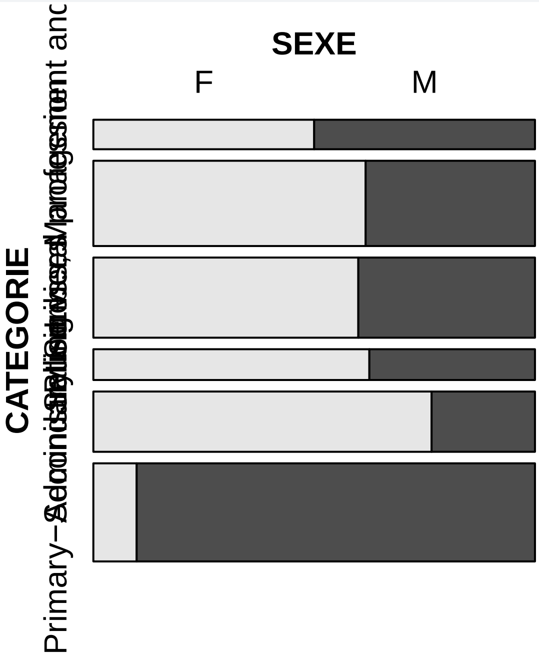
```
# A tibble: 10 x 2
   CATEGORIE                                n
   <fct>                                <int>
 1 "Liberal profession"                   133
 2 "Services"                             125
 3 "Administration"                        94
 4 "Selling"                               48
 5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

```
df <- df |>
  mutate(CATEGORIE=fct_lump_min(CATEGORIE,
                                min=40,
                                other_level = "Primary-Secondary"))
```

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



Testing association

Chi-square independence/association test

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

# test_1

test_1  |>
  broom::tidy()  |>
  knitr::kable()
```

statistic	p.value	parameter	method
140.6717	0	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

expected |>
  as_tibble() |>
  knitr::kable()
```

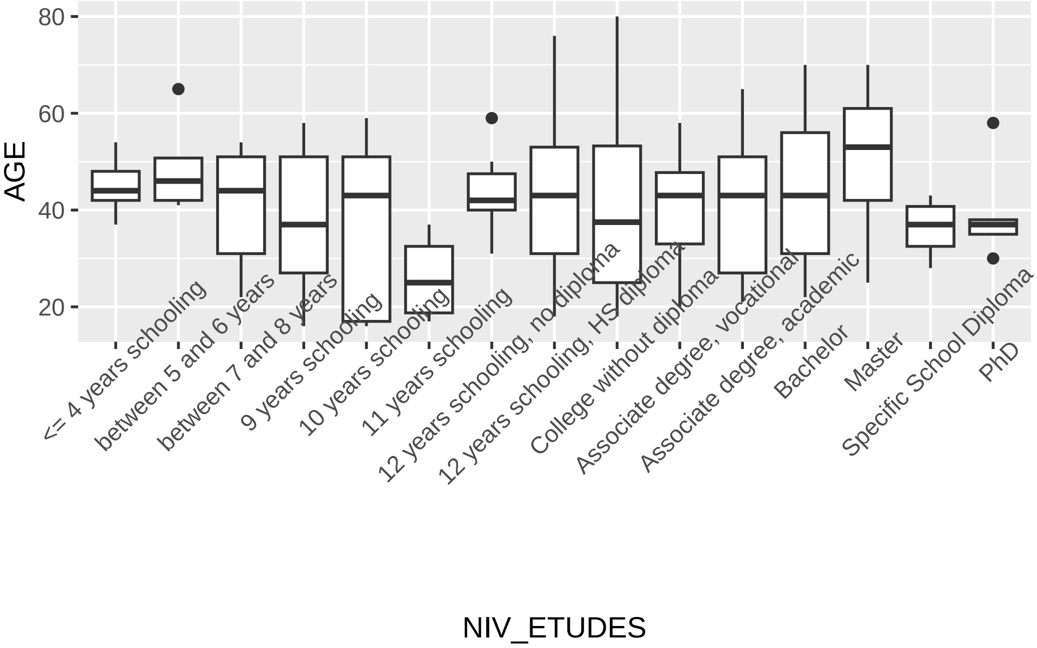
F	M
22.80801	23.19199
65.94491	67.05509
61.97830	63.02170
23.79967	24.20033
46.60768	47.39232
75.86144	77.13856

Categorical/Numerical pairs

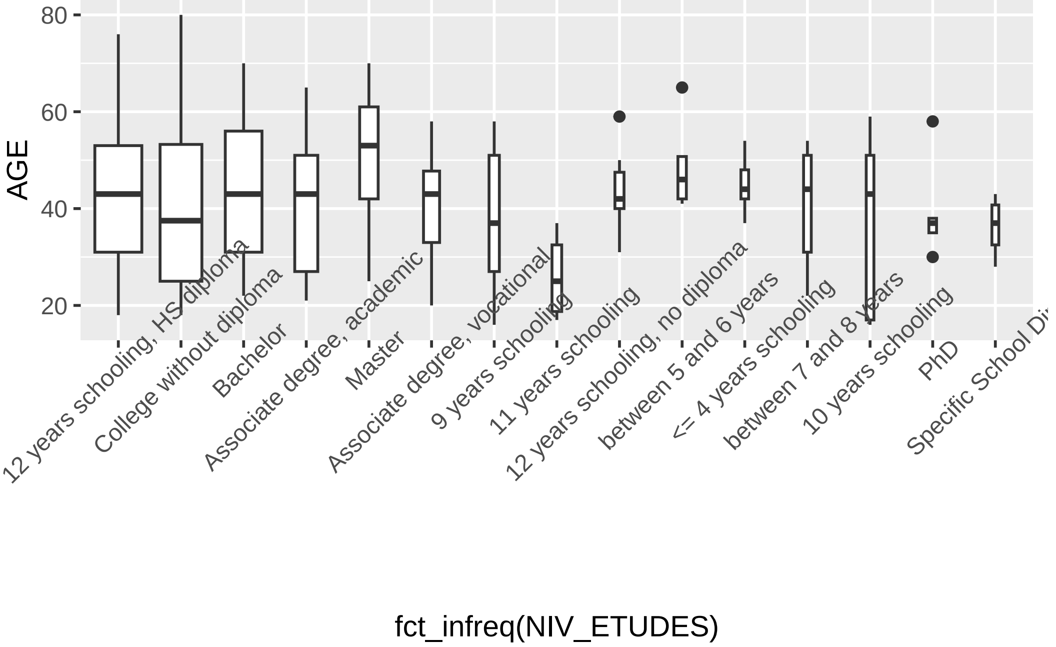
Grouped boxplots

Plot boxplots of `AGE` according to `NIV_ETUDES`

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



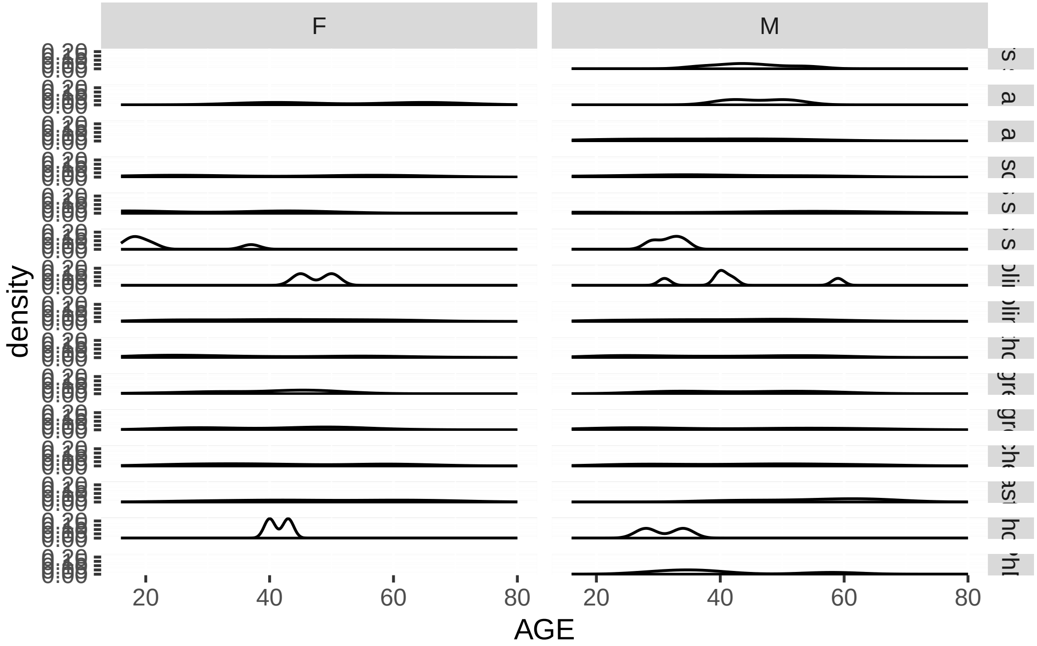
```
df |>
  ggplot() +
  aes(x=fct_infreq(NIV_ETUDES), y=AGE) +
  geom_boxplot(varwidth = T) +
  rot_x_text
```



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

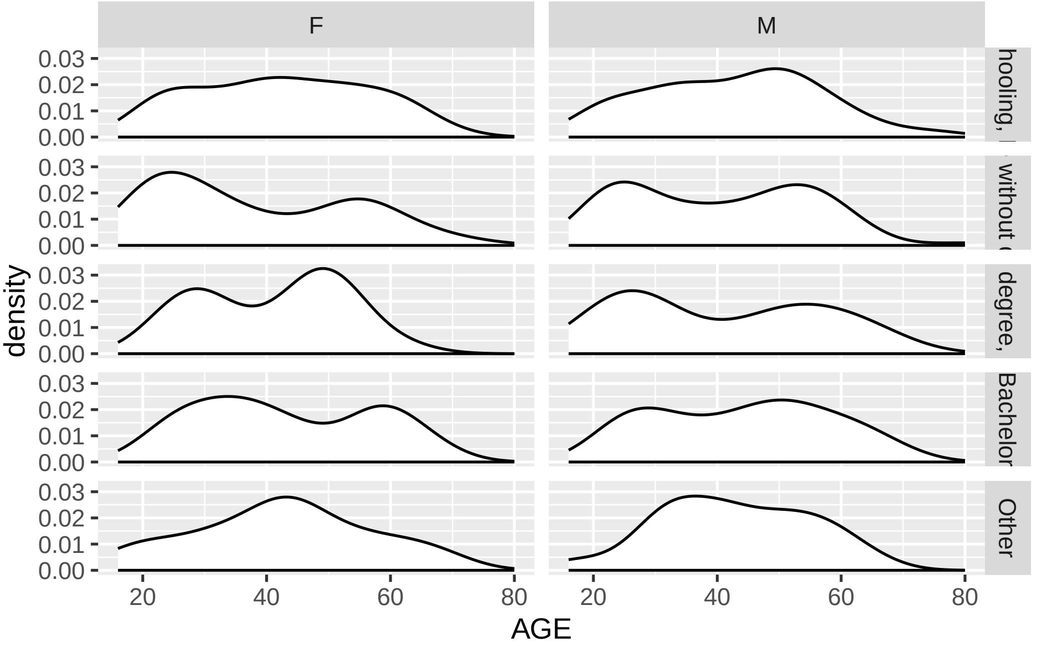
```r
p <- df |>
  ggplot() +
  aes(x=AGE) +
  stat_density(fill="white", color="black") +
  facet_grid(rows=vars(NIV_ETUDES),
             cols=vars(SEXE))

p
```

```
Warning: Groups with fewer than two data points have been dropped.
Groups with fewer than two data points have been dropped.
Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
-Inf
Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
-Inf
```



Collapse rare levels of `NIV_ETUDES` and replay.

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

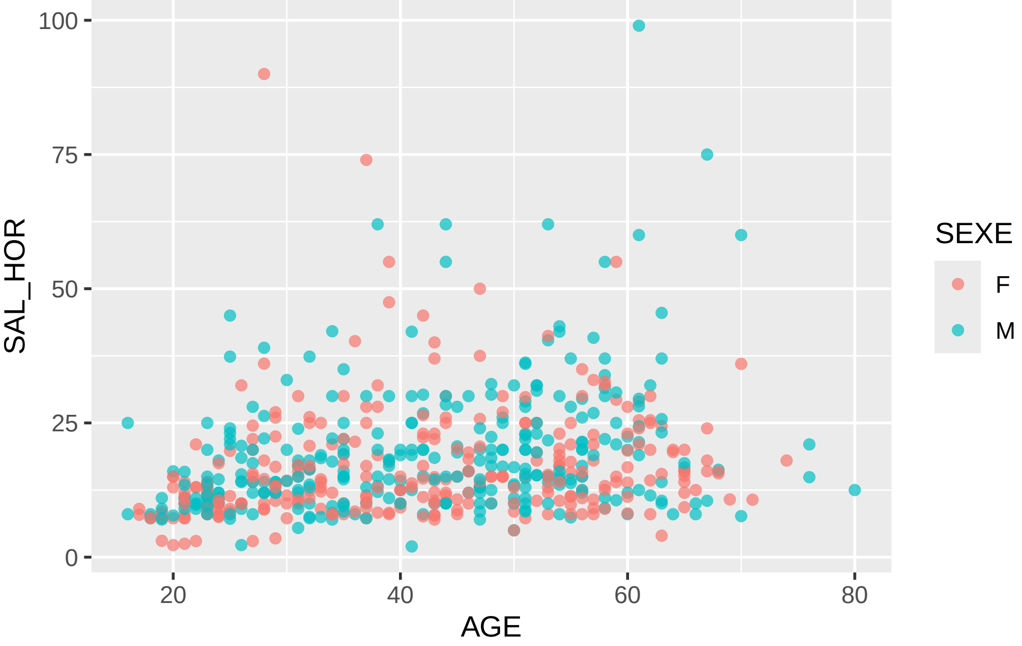


Numerical/Numerical pairs

Scatterplots

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

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



Correlations

- Linear correlation coefficient (Pearson ρ)
- Linear rank correlation coefficient (Spearman, Kendall)
- ξ rank correlation coefficient (Chatterjee)

`pairs` from base `R`

`ggpairs()`

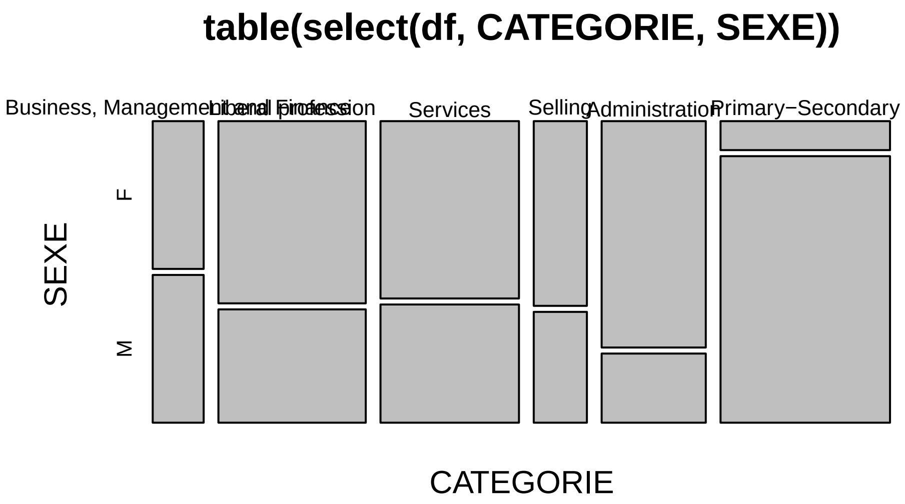
Useful links

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

```r
expand.grid(levels(df$CATEGORIE), levels(df$SEXE))
```

```
                               Var1 Var2
1  Business, Management and Finance    F
2                Liberal profession    F
3                          Services    F
4                           Selling    F
5                    Administration    F
6                 Primary-Secondary    F
7  Business, Management and Finance    M
8                Liberal profession    M
9                          Services    M
10                          Selling    M
11                   Administration    M
12                Primary-Secondary    M
```

```r
df |>
  select(CATEGORIE, SEXE) |>
  table() |>
  mosaicplot()
```



```r
pchisq(140, df=5, lower.tail = F)
```

```
[1] 1.789245e-28
```