

# Correction Homework

Pratiques de la Recherche en Économie

---

Florentine Oliveira-Roux

3 Février 2026

# Partie 1: Nettoyage de données (7pts)

Cette première partie vise à nettoyer et préparer les données en vue des analyses descriptives et économétriques des parties suivantes.

# Import données

Importez sur `R` la base de données que vous nommerez `df`.

# Question 1 (5pts):

Créez les variables suivantes:

- `cohorte` qui regroupe les années de naissance en cohortes de 5 ans (par exemple : "1945-1949", etc).
- `femme`, une dummy égale à 1 si l'individu est une femme, 0 sinon.

**NB: en deux bouts de code séparés pour que cela rentre sur les slides mais tout grouper en une seule fois!**

```
df = df %>%
  mutate(
    cohorte = case_when( # 0, 5
      birth_year %in% 1945:1949 ~ "1945-1949",
      birth_year %in% 1950:1954 ~ "1950-1954",
      birth_year %in% 1955:1959 ~ "1955-1959",
      birth_year %in% 1960:1964 ~ "1960-1964",
      birth_year %in% 1965:1969 ~ "1965-1969",
      birth_year %in% 1970:1974 ~ "1970-1974",
      birth_year %in% 1975:1979 ~ "1975-1979",
      birth_year %in% 1980:1984 ~ "1980-1984",
      birth_year %in% 1985:1989 ~ "1985-1989",
      TRUE ~ NA_character_
    ),
    femme = ifelse(sexe == 2, 1, 0))
```

# Question 1 (5pts):

Créez les variables suivantes:

- cohorte qui regroupe les années de naissance en cohortes de 5 ans (par exemple : "1945-1949", etc).
- femme, une dummy égale à 1 si l'individu est une femme, 0 sinon.

**NB: en deux bouts de code séparés pour que cela rentre sur les slides mais tout grouper en une seule fois!**

```
df = df %>%
  mutate(
    cohorte = case_when(
      birth_year %in% 1945:1949 ~ "1945-1949",
      birth_year %in% 1950:1954 ~ "1950-1954",
      birth_year %in% 1955:1959 ~ "1955-1959",
      birth_year %in% 1960:1964 ~ "1960-1964",
      birth_year %in% 1965:1969 ~ "1965-1969",
      birth_year %in% 1970:1974 ~ "1970-1974",
      birth_year %in% 1975:1979 ~ "1975-1979",
      birth_year %in% 1980:1984 ~ "1980-1984",
      birth_year %in% 1985:1989 ~ "1985-1989",
      TRUE ~ NA_character_
    ),
    femme = ifelse(sexe == 2, 1, 0))
```

# Question 1 (5pts):

Créez les variables suivantes:

- cohorte qui regroupe les années de naissance en cohortes de 5 ans (par exemple : "1945-1949", etc).
- femme, une dummy égale à 1 si l'individu est une femme, 0 sinon.

**NB: en deux bouts de code séparés pour que cela rentre sur les slides mais tout grouper en une seule fois!**

```
df = df %>%
  mutate(
    cohorte = case_when( # 0, 5
      birth_year %in% 1945:1949 ~ "1945-1949",
      birth_year %in% 1950:1954 ~ "1950-1954",
      birth_year %in% 1955:1959 ~ "1955-1959",
      birth_year %in% 1960:1964 ~ "1960-1964",
      birth_year %in% 1965:1969 ~ "1965-1969",
      birth_year %in% 1970:1974 ~ "1970-1974",
      birth_year %in% 1975:1979 ~ "1975-1979",
      birth_year %in% 1980:1984 ~ "1980-1984",
      birth_year %in% 1985:1989 ~ "1985-1989",
      TRUE ~ NA_character_
    ),
    femme = ifelse(sexe == 2, 1, 0))
```

# Question 1 (5pts):

- `rang_naissance` qui indique le rang de naissance de chaque individu au sein de sa fratrie (**1 pour l'aîné, 2 pour le deuxième enfant, etc.**).
- `elder`, une dummy égale à 1 si l'individu est l'aîné de sa fratrie, 0 sinon.
- `cohorte_aine` qui indique la cohorte de naissance de l'aîné pour chaque fratrie.
- `taille_fratrie` qui indique le nombre d'enfants au sein de chaque fratrie.
- `plus_de_2`, une dummy égale à 1 si la fratrie comprend 3 enfants ou plus, 0 sinon.
- `compo_genre` qui décrit la composition par sexe des deux premiers enfants de chaque fratrie. Cette variable prend quatre valeurs possibles: "FF" (deux filles), "MM" (deux garçons), "FM" (fille puis garçon), ou "MF" (garçon puis fille).

```
df = df %>%
  group_by(family_id) %>%
  arrange(birth_year) %>%
  mutate(
    rang_naissance = row_number(),
    cohorte_aine = first(cohorte), # 0,5
    taille_fratrie = n(), # 0,5
    plus_de_2 = ifelse(taille_fratrie > 2, 1, 0), # 0,5
    compo_genre = case_when( # 0,5
      sexe[rang_naissance == 1] == 1 & sexe[rang_naissance == 2] == 1 ~ "FF",
      sexe[rang_naissance == 1] == 2 & sexe[rang_naissance == 2] == 1 ~ "FM",
      sexe[rang_naissance == 1] == 1 & sexe[rang_naissance == 2] == 2 ~ "MF",
      sexe[rang_naissance == 1] == 2 & sexe[rang_naissance == 2] == 2 ~ "MM"
    )
  ) %>%
  ungroup() %>%
  mutate(elder = ifelse(rang_naissance == 1, 1, 0))
```

# Question 1 (5pts):

- `rang_naissance` qui indique le rang de naissance de chaque individu au sein de sa fratrie (1 pour l'aîné, 2 pour le deuxième enfant, etc.).
- `elder`, **une dummy égale à 1 si l'individu est l'aîné de sa fratrie, 0 sinon.**
- `cohorte_aine` qui indique la cohorte de naissance de l'aîné pour chaque fratrie.
- `taille_fratrie` qui indique le nombre d'enfants au sein de chaque fratrie.
- `plus_de_2`, une dummy égale à 1 si la fratrie comprend 3 enfants ou plus, 0 sinon.
- `compo_genre` qui décrit la composition par sexe des deux premiers enfants de chaque fratrie. Cette variable prend quatre valeurs possibles: "FF" (deux filles), "MM" (deux garçons), "FM" (fille puis garçon), ou "MF" (garçon puis fille).

```
df = df %>%
  group_by(family_id) %>%
  arrange(birth_year) %>%
  mutate(
    rang_naissance = row_number(), # 0,5 group_by, 0,5 c
    cohorte_aine = first(cohorte), # 0,5
    taille_fratrie = n(), # 0,5
    plus_de_2 = ifelse(taille_fratrie > 2, 1, 0), # 0,5
    compo_genre = case_when( # 0,5
      sexe[rang_naissance == 1] == 1 & sexe[rang_naissar
      sexe[rang_naissance == 1] == 2 & sexe[rang_naissar
      sexe[rang_naissance == 1] == 1 & sexe[rang_naissar
      sexe[rang_naissance == 1] == 2 & sexe[rang_naissar
    )
  ) %>%
  ungroup() %>%
  mutate(elder = ifelse(rang_naissance == 1, 1, 0))
```

# Question 1 (5pts):

- `rang_naissance` qui indique le rang de naissance de chaque individu au sein de sa fratrie (1 pour l'aîné, 2 pour le deuxième enfant, etc.).
- `elder`, une dummy égale à 1 si l'individu est l'aîné de sa fratrie, 0 sinon.
- `cohorte_aine` **qui indique la cohorte de naissance de l'aîné pour chaque fratrie.**
- `taille_fratrie` qui indique le nombre d'enfants au sein de chaque fratrie.
- `plus_de_2`, une dummy égale à 1 si la fratrie comprend 3 enfants ou plus, 0 sinon.
- `compo_genre` qui décrit la composition par sexe des deux premiers enfants de chaque fratrie. Cette variable prend quatre valeurs possibles: "FF" (deux filles), "MM" (deux garçons), "FM" (fille puis garçon), ou "MF" (garçon puis fille).

```
df = df %>%
  group_by(family_id) %>%
  arrange(birth_year) %>%
  mutate(
    rang_naissance = row_number(), # 0,5 group_by, 0,5 c
    cohorte_aine = first(cohorte),
    taille_fratrie = n(), # 0,5
    plus_de_2 = ifelse(taille_fratrie > 2, 1, 0), # 0,5
    compo_genre = case_when( # 0,5
      sexe[rang_naissance == 1] == 1 & sexe[rang_naissar
      sexe[rang_naissance == 1] == 2 & sexe[rang_naissar
      sexe[rang_naissance == 1] == 1 & sexe[rang_naissar
      sexe[rang_naissance == 1] == 2 & sexe[rang_naissar
    )
  ) %>%
  ungroup() %>%
  mutate(elder = ifelse(rang_naissance == 1, 1, 0))
```

# Question 1 (5pts):

- `rang_naissance` qui indique le rang de naissance de chaque individu au sein de sa fratrie (1 pour l'aîné, 2 pour le deuxième enfant, etc.).
- `elder`, une dummy égale à 1 si l'individu est l'aîné de sa fratrie, 0 sinon.
- `cohorte_aine` qui indique la cohorte de naissance de l'aîné pour chaque fratrie.
- `taille_fratrie` **qui indique le nombre d'enfants au sein de chaque fratrie.**
- `plus_de_2`, une dummy égale à 1 si la fratrie comprend 3 enfants ou plus, 0 sinon.
- `compo_genre` qui décrit la composition par sexe des deux premiers enfants de chaque fratrie. Cette variable prend quatre valeurs possibles: "FF" (deux filles), "MM" (deux garçons), "FM" (fille puis garçon), ou "MF" (garçon puis fille).

```
df = df %>%
  group_by(family_id) %>%
  arrange(birth_year) %>%
  mutate(
    rang_naissance = row_number(), # 0,5 group_by, 0,5 c
    cohorte_aine = first(cohorte), # 0,5
    taille_fratrie = n(),
    plus_de_2 = ifelse(taille_fratrie > 2, 1, 0), # 0,5
    compo_genre = case_when( # 0,5
      sexe[rang_naissance = 1] = 1 & sexe[rang_naissar
      sexe[rang_naissance = 1] = 2 & sexe[rang_naissar
      sexe[rang_naissance = 1] = 1 & sexe[rang_naissar
      sexe[rang_naissance = 1] = 2 & sexe[rang_naissar
    )
  ) %>%
  ungroup() %>%
  mutate(elder = ifelse(rang_naissance = 1, 1, 0))
```

# Question 1 (5pts):

- `rang_naissance` qui indique le rang de naissance de chaque individu au sein de sa fratrie (1 pour l'aîné, 2 pour le deuxième enfant, etc.).
- `elder`, une dummy égale à 1 si l'individu est l'aîné de sa fratrie, 0 sinon.
- `cohorte_aine` qui indique la cohorte de naissance de l'aîné pour chaque fratrie.
- `taille_fratrie` qui indique le nombre d'enfants au sein de chaque fratrie.
- `plus_de_2`, **une dummy égale à 1 si la fratrie comprend 3 enfants ou plus, 0 sinon.**
- `compo_genre` qui décrit la composition par sexe des deux premiers enfants de chaque fratrie. Cette variable prend quatre valeurs possibles: "FF" (deux filles), "MM" (deux garçons), "FM" (fille puis garçon), ou "MF" (garçon puis fille).

```
df = df %>%
  group_by(family_id) %>%
  arrange(birth_year) %>%
  mutate(
    rang_naissance = row_number(), # 0,5 group_by, 0,5 c
    cohorte_aine = first(cohorte), # 0,5
    taille_fratrie = n(), # 0,5
    plus_de_2 = ifelse(taille_fratrie > 2, 1, 0),
    compon.genre = case_when( # 0,5
      sexe[rang_naissance == 1] == 1 & sexe[rang_naissar
      sexe[rang_naissance == 1] == 2 & sexe[rang_naissar
      sexe[rang_naissance == 1] == 1 & sexe[rang_naissar
      sexe[rang_naissance == 1] == 2 & sexe[rang_naissar
    )
  ) %>%
  ungroup() %>%
  mutate(elder = ifelse(rang_naissance == 1, 1, 0))
```

# Question 1 (5pts):

- `rang_naissance` qui indique le rang de naissance de chaque individu au sein de sa fratrie (1 pour l'aîné, 2 pour le deuxième enfant, etc.).
- `elder`, une dummy égale à 1 si l'individu est l'aîné de sa fratrie, 0 sinon.
- `cohorte_aine` qui indique la cohorte de naissance de l'aîné pour chaque fratrie.
- `taille_fratrie` qui indique le nombre d'enfants au sein de chaque fratrie.
- `plus_de_2`, une dummy égale à 1 si la fratrie comprend 3 enfants ou plus, 0 sinon.
- `compo_genre` **qui décrit la composition par sexe des deux premiers enfants de chaque fratrie. Cette variable prend quatre valeurs possibles: "FF" (deux filles), "MM" (deux garçons), "FM" (fille puis garçon), ou "MF" (garçon puis fille).**

```
df = df %>%
  group_by(family_id) %>%
  arrange(birth_year) %>%
  mutate(
    rang_naissance = row_number(), # 0,5 group_by, 0,5 c
    cohorte_aine = first(cohorte), # 0,5
    taille_fratrie = n(), # 0,5
    plus_de_2 = ifelse(taille_fratrie > 2, 1, 0), # 0,5
    compo_genre = case_when( # 0,5
      sexe[rang_naissance == 1] == 1 & sexe[rang_naissar
      sexe[rang_naissance == 1] == 2 & sexe[rang_naissar
      sexe[rang_naissance == 1] == 1 & sexe[rang_naissar
      sexe[rang_naissance == 1] == 2 & sexe[rang_naissar
    )
  ) %>%
  ungroup() %>%
  mutate(elder = ifelse(rang_naissance == 1, 1, 0))
```

## Question 2 (0,5pt):

Recodez les variables `mother_cs` et `father_cs` en remplaçant les codes numériques (1 à 6) par les labels des catégories socioprofessionnelles correspondantes.

```
df = df %>%
  mutate(
    father_cs = recode(as.character(father_cs), # 0,25
      "1" = "Agriculteur",
      "2" = "Artisan",
      "3" = "Cadre",
      "4" = "Profession intermédiaire",
      "5" = "Employé",
      "6" = "Ouvrier"),
    mother_cs = recode(as.character(mother_cs), # 0,25
      "1" = "Agricultrice",
      "2" = "Artisan",
      "3" = "Cadre",
      "4" = "Profession intermédiaire",
      "5" = "Employée",
      "6" = "Ouvrière"))
```

## Question 3 (1,5pt):

- Représentez la distribution de la taille de la fratrie. Commentez.
- Supprimer les fratries dont la taille fait parti des 1% plus élevées.

# Question 3 (1,5pt):

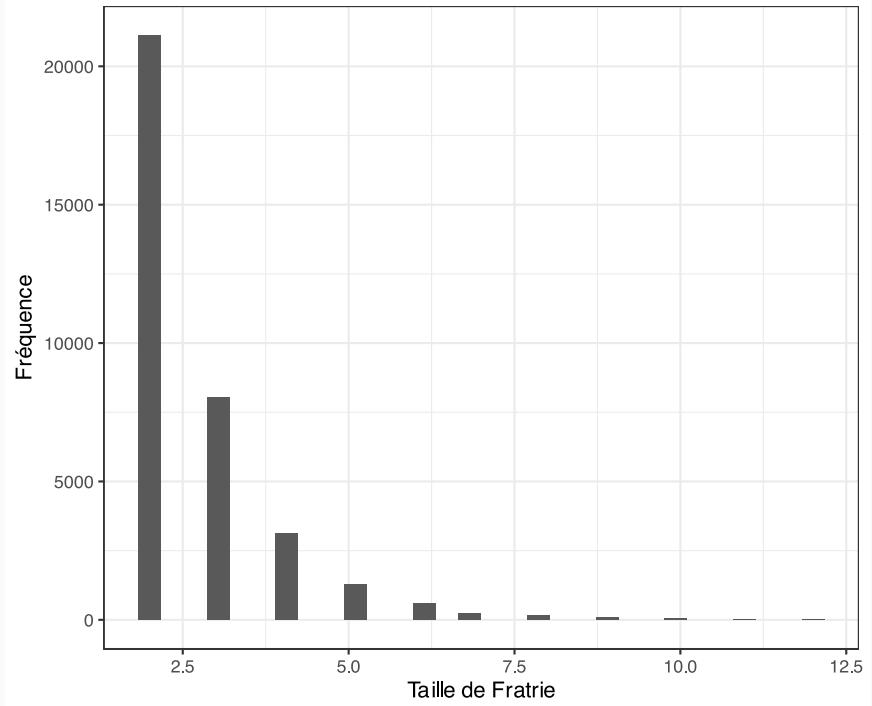
- **Représentez la distribution de la taille de la fratrie. Commentez.**
- Supprimer les fratries dont la taille fait parti des 1% plus élevées.

```
df_fratries = df %>%
  select(family_id, taille_fratrie) %>%
  distinct()
```

```
ggplot(df_fratries, aes(x = taille_fratrie)) +
  geom_histogram() +
  labs(x = "Taille de Fratrie", y = "Fréquence") +
  theme_bw()
```

```
top1 = quantile(df_fratries$taille_fratrie, 0.99)
```

```
df = df %>%
  filter(taille_fratrie < top1)
```



# Question 3 (1,5pt):

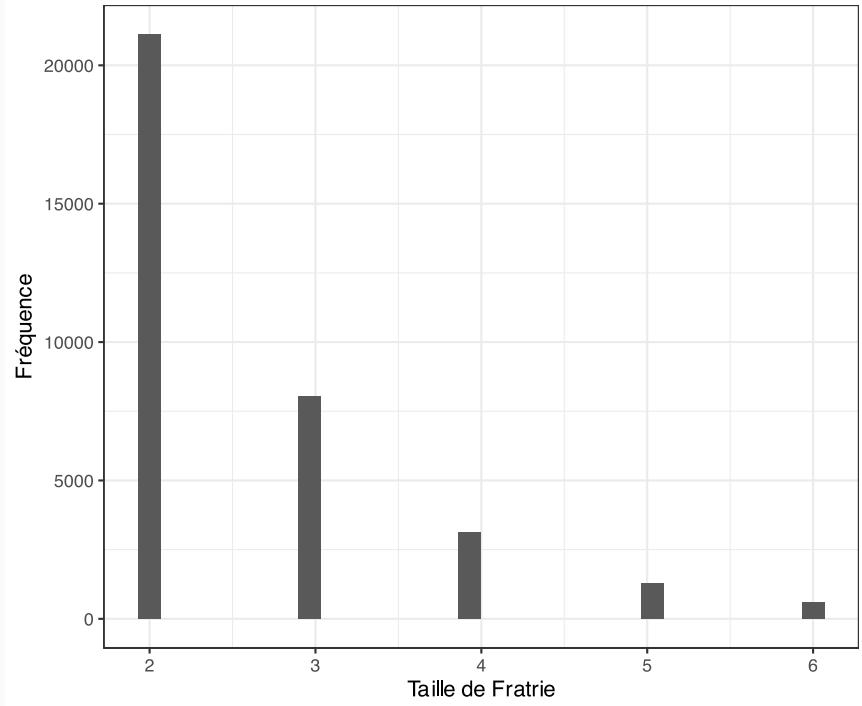
- Représentez la distribution de la taille de la fratrie. Commentez.
- **Supprimer les fratries dont la taille fait parti des 1% plus élevées.**

```
df_fratries = df %>%
  select(family_id, taille_fratrie) %>%
  distinct()

ggplot(df_fratries, aes(x = taille_fratrie)) +
  geom_histogram() +
  labs(x = "Taille de Fratrie", y = "Fréquence") +
  theme_bw()

top1 = quantile(df_fratries$taille_fratrie, 0.99)

df = df %>%
  filter(taille_fratrie < top1)
```



## Partie 2: Statistiques Descriptives (8pts)

Cette partie a pour objectif d'explorer les données et comprendre les déterminants du niveau de diplôme.

## Partie A: Taille de la fratrie

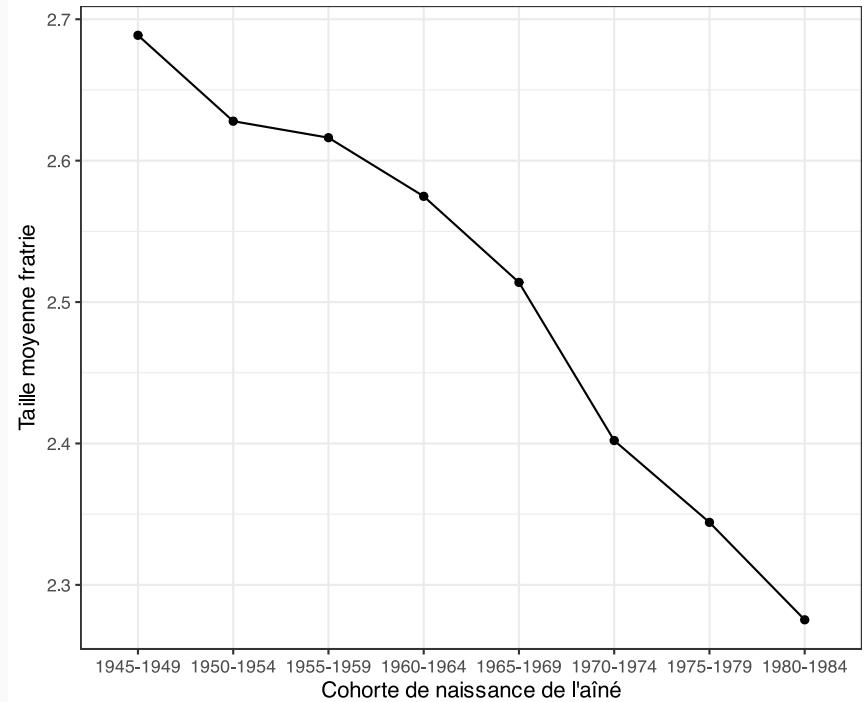
# Question 4 (0,5pt):

Représentez graphiquement l'évolution de la taille de la fratrie selon la cohorte de naissance de l'aîné. Commentez.

```
df %>%
  filter(rang_naissance == 1) %>%
  group_by(cohorte_aine) %>%
  summarise(mean_family_size = mean(taille_fratrie, na.rm = TRUE))
  ggplot(aes(x = cohorte_aine, y = mean_family_size)) +
  geom_point() +
  geom_line(group = 1) +
  labs(x = "Cohorte de naissance de l'aîné", y = "Taille moyenne fratrie")
  theme_bw()
```



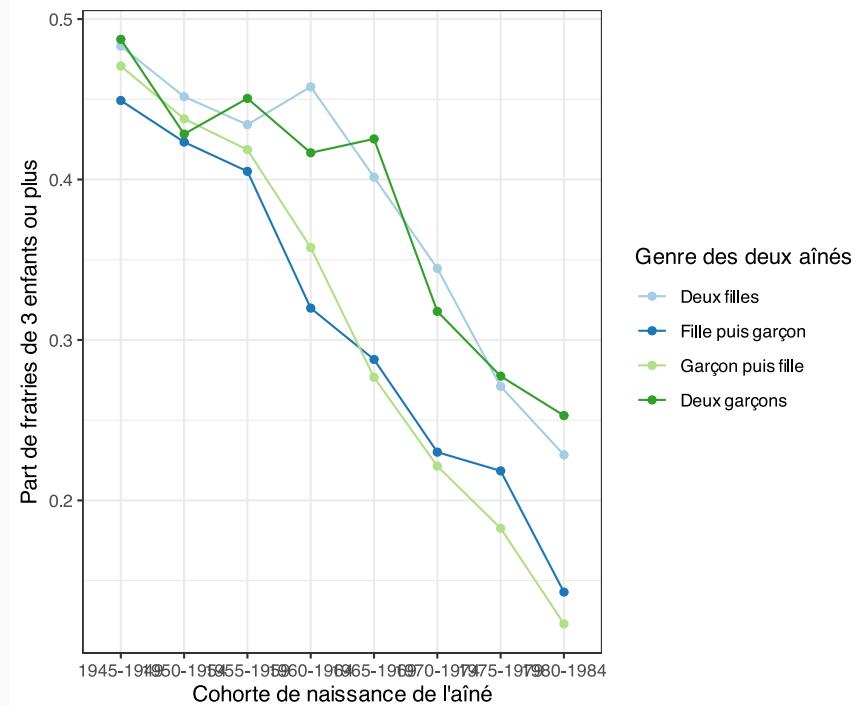
Attention à ne garder qu'une observation par fratrie!



# Question 5 (1pt):

Représentez graphiquement l'évolution de la part de fratries de 3 enfants ou plus selon la cohorte de naissance de l'aîné en distinguant les quatre configurations possibles du genre des deux aînés. Commentez.

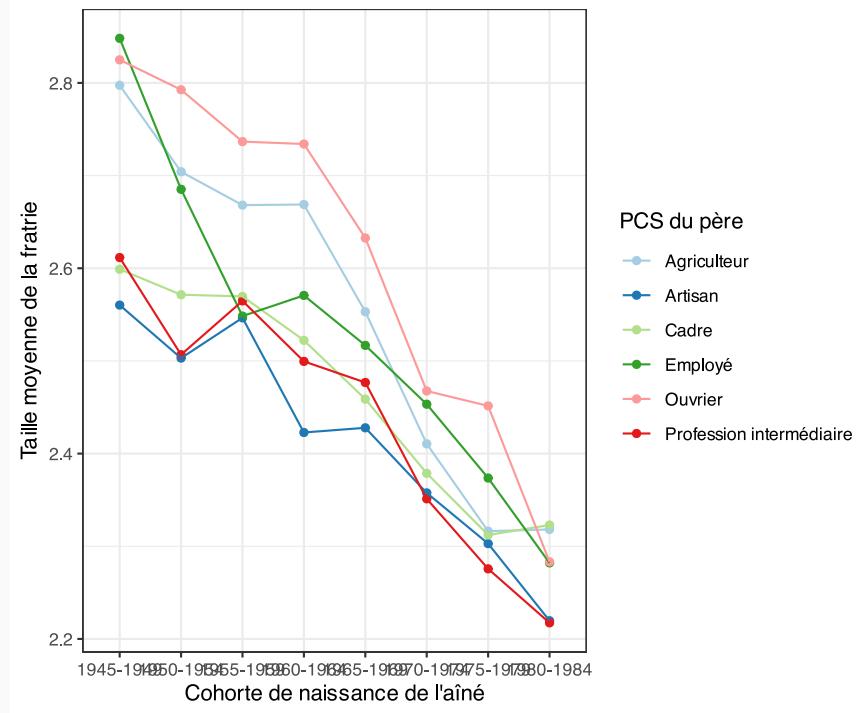
```
df %>%
  filter(rang_naissance = 1) %>% # Attention à bien pré
  group_by(cohorte_aine, compo_genre) %>%
  summarise(mean_family_size = mean(plus_de_2, na.rm = T
  ggplot(aes(x = cohorte_aine, y = mean_family_size,
             color = compo_genre, group = compo_genre))
  geom_point() +
  geom_line() +
  scale_color_brewer(
    palette = "Paired",
    name = "Genre des deux aînés",
    labels = c(
      "FF" = "Deux filles",
      "FM" = "Fille puis garçon",
      "MF" = "Garçon puis fille",
      "MM" = "Deux garçons")) +
  labs(x = "Cohorte de naissance de l'aîné",
       y = "Part de fratries de 3 enfants ou plus") +
  theme_bw()
```



# Question 6 (1,5pt):

Représentez l'évolution de la taille moyenne des fratries selon la PCS du père. Selon vous, quelles raisons peuvent expliquer la différence de fécondité entre les différents groupes sociaux et son évolution?

```
df %>%
  filter(rang_naissance = 1) %>%
  group_by(cohorte_aine, father_cs) %>%
  summarise(mean_family_size = mean(taille_fratrie, na.rm = TRUE))
  ggplot(aes(x = cohorte_aine, y = mean_family_size,
             color = father_cs, group = father_cs)) +
  geom_point() +
  geom_line() +
  scale_color_brewer(
    palette = "Paired",
    name = "PCS du père" ) +
  labs(x = "Cohorte de naissance de l'aîné",
       y = "Taille moyenne de la fratrie"
  ) +
  theme_bw()
```

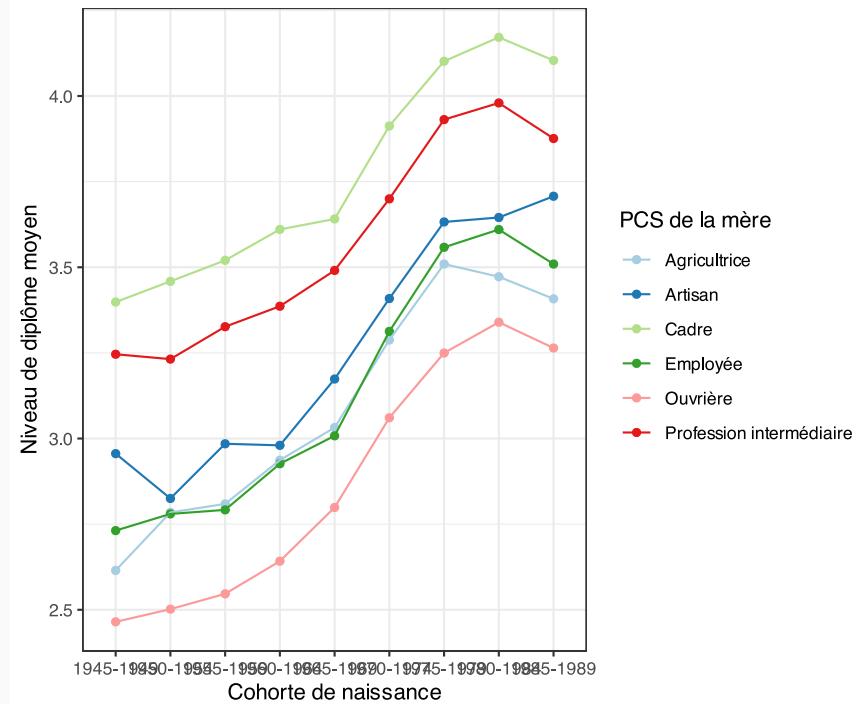


## Partie B: Origine sociale

# Question 7 (2pts):

Représentez l'évolution du niveau de diplôme moyen selon la CSP de la mère, en ne conservant que les individus ayant un niveau de diplôme connu. Commentez.

```
df %>%
  filter(!is.na(dipl)) %>%
  group_by(cohorte, mother_cs) %>%
  summarise(mean_dipl = mean(dipl, na.rm = TRUE)) %>%
  ggplot(aes(x = cohorte, y = mean_dipl,
             color = mother_cs, group = mother_cs)) +
  geom_point() +
  geom_line() +
  scale_color_brewer(
    palette = "Paired",
    name = "PCS de la mère" ) +
  labs(x = "Cohorte de naissance",
       y = "Niveau de diplôme moyen") +
  theme_bw()
```

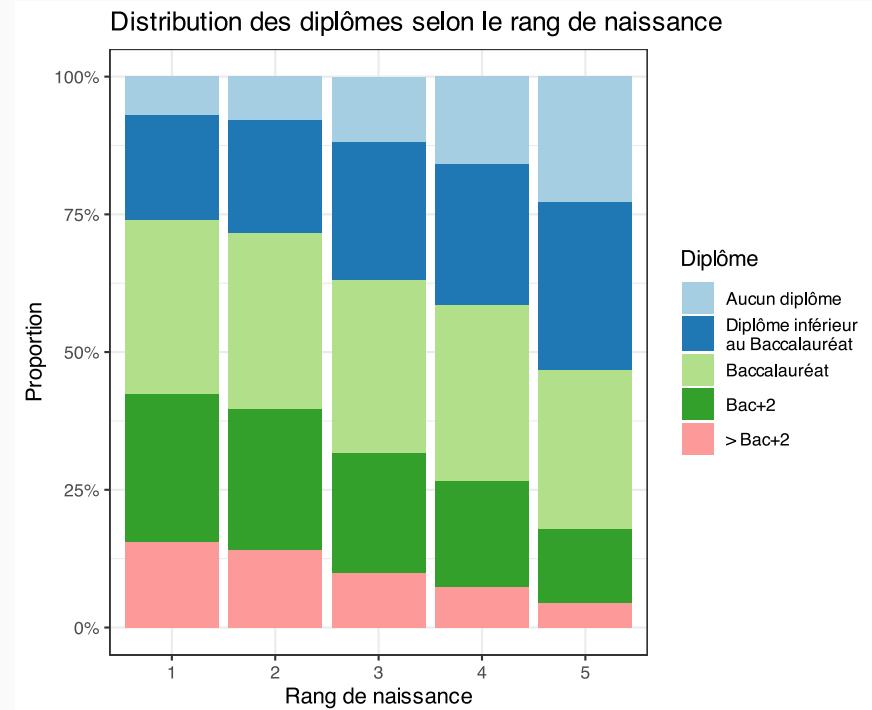


## Partie C: Caractéristiques individuelles

# Question 8 (2pts):

Représentez graphiquement le niveau de diplôme moyen selon le rang de naissance, en ne conservant que les rangs jusqu'à 5 maximum et les individus ayant un niveau de diplôme connu. Commentez. Selon vous, quels mécanismes peuvent expliquer cela?

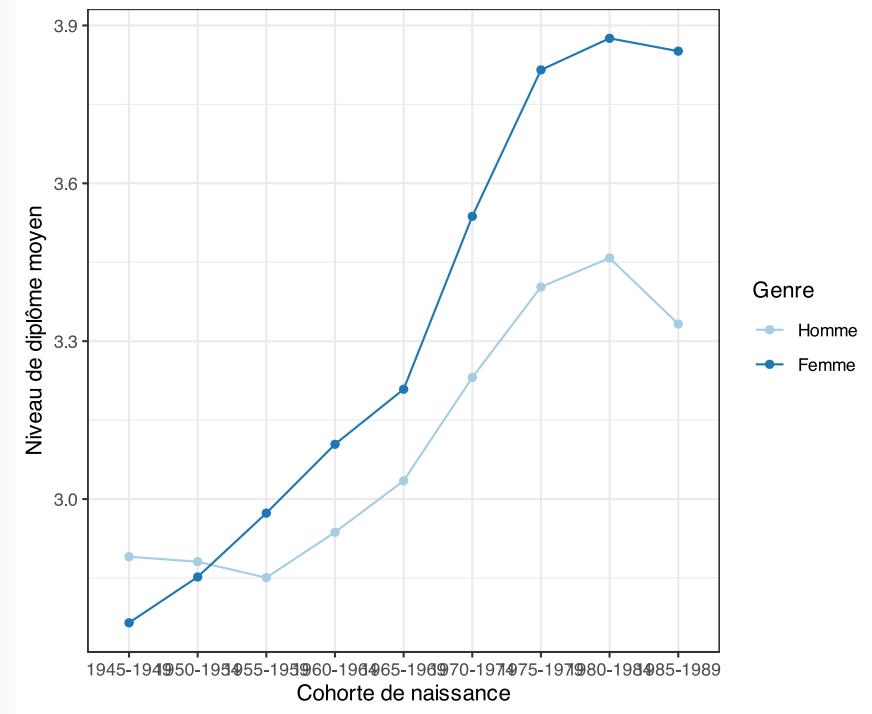
```
df %>%
  filter(rang_naissance <= 5,
         !is.na(dipl)) %>%
  group_by(rang_naissance, dipl) %>%
  summarise(n = n()) %>%
  group_by(rang_naissance) %>%
  mutate(proportion = n / sum(n)) %>%
  ggplot(aes(x = factor(rang_naissance), y = proportion,
  geom_col(position = "stack") +
  scale_fill_brewer(palette = "Paired", name = "Diplôme"
                    labels = c("1" = "Aucun diplôme",
                               "2" = "Diplôme inférieur
                               au Baccalauréat",
                               "3" = "Baccalauréat",
                               "4" = "Bac+2",
                               "5" = "> Bac+2")) +
  labs(x = "Rang de naissance",
       y = "Proportion",
       fill = "Niveau de diplôme")
```



# Question 9 (1pt):

Représentez graphiquement l'évolution du niveau de diplôme moyen par cohorte de naissance selon le sexe. Commentez.

```
df %>%  
  filter(!is.na(dipl)) %>%  
  group_by(cohorte, sexe) %>%  
  summarise(mean_dipl = mean(dipl, na.rm = TRUE)) %>%  
  ggplot(aes(x = cohorte, y = mean_dipl,  
             color = sexe, group = sexe)) +  
  geom_point() +  
  geom_line() +  
  scale_color_brewer(  
    palette = "Paired",  
    name = "Genre",  
    labels = c("1"= "Homme", "2"= "Femme")) +  
  labs(x = "Cohorte de naissance",  
       y = "Niveau de diplôme moyen") +  
  theme_bw()
```



## Partie 3: Analyse Économétrique (11pts)

Cette analyse économétrique explore les déterminants du niveau de diplôme.

## Question 10 (1pt):

Selon vous, la taille de la fratrie a t-elle un effet positif, négatif, ou nul sur le niveau de diplôme? Pourquoi?

# Question 11 (1,5pt):

- Régressez le niveau de diplôme sur la taille de fratrie. Vous nommerez ce modèle `reg1`.
- Régressez le niveau de diplôme sur la dummy `plus_de_2`. Vous nommerez ce modèle `reg2`.
- Dans les deux cas, que représentent  $\hat{\alpha}$  et  $\hat{\beta}$ ? Peut-on dire que  $\hat{\beta}$  représente l'effet causal de la taille de la fratrie sur le niveau de diplôme? Pourquoi?

```
reg1 = lm(dipl ~ taille_fratrie, data = df)
summary(reg1)
```

```
##  
## Call:  
## lm(formula = dipl ~ taille_fratrie, data = df)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max  
## -2.31243 -0.87189 -0.09216  0.68757  2.34839  
##  
## Coefficients:  
##                 Estimate Std. Error t value Pr(>|t|)  
## (Intercept)  3.752979  0.012134 309.29 <2e-16 ***  
## taille_fratrie -0.220273  0.004105 -53.67 <2e-16 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# Question 11 (1,5pt):

- Régressez le niveau de diplôme sur la taille de fratrie. Vous nommerez ce modèle `reg1`.
- Régressez le niveau de diplôme sur la dummy `plus_de_2`. Vous nommerez ce modèle `reg2`.
- Dans les deux cas, que représentent  $\hat{\alpha}$  et  $\hat{\beta}$ ? Peut-on dire que  $\hat{\beta}$  représente l'effet causal de la taille de la fratrie sur le niveau de diplôme? Pourquoi?

```
reg1 = lm(dipl ~ taille_fratrie, data = df)
summary(reg1)
```

```
##  
## Call:  
## lm(formula = dipl ~ taille_fratrie, data = df)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max  
## -2.31243 -0.87189 -0.09216  0.68757  2.34839  
##  
## Coefficients:  
##                 Estimate Std. Error t value Pr(>|t|)  
## (Intercept)  3.752979  0.012134 309.29 <2e-16 ***  
## taille_fratrie -0.220273  0.004105 -53.67 <2e-16 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# Question 11 (1,5pt):

- Régressez le niveau de diplôme sur la taille de fratrie. Vous nommerez ce modèle `reg1`.
- **Régressez le niveau de diplôme sur la dummy `plus_de_2`. Vous nommerez ce modèle `reg2`.**
- Dans les deux cas, que représentent  $\hat{\alpha}$  et  $\hat{\beta}$ ? Peut-on dire que  $\hat{\beta}$  représente l'effet causal de la taille de la fratrie sur le niveau de diplôme? Pourquoi?

```
reg2 = lm(dipl ~ plus_de_2, data = df)
summary(reg2)
```

```
##
## Call:
## lm(formula = dipl ~ plus_de_2, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.31785 -0.95755  0.04245  0.68215  2.04245
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.317850  0.005555 597.33 <2e-16 ***
## plus_de_2   -0.360297  0.007824 -46.05 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## Question 11 (1,5pt):

- Régressez le niveau de diplôme sur la taille de fratrie. Vous nommerez ce modèle `reg1`.
- Régressez le niveau de diplôme sur la dummy `plus_de_2`. Vous nommerez ce modèle `reg2`.
- **Dans les deux cas, que représentent  $\hat{\alpha}$  et  $\hat{\beta}$ ? Peut-on dire que  $\hat{\beta}$  représente l'effet causal de la taille de la fratrie sur le niveau de diplôme? Pourquoi?**

```
##           Estimate Std. Error t value Pr(>|t|) ##           Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 3.7529785 0.012134077 309.29245      0 ## (Intercept) 3.3178504 0.005554502 597.32637      0  
## taille_fratrie -0.2202729 0.004104537 -53.66571      0 ## plus_de_2    -0.3602965 0.007823850 -46.05105      0
```

# Question 12 (1,5pt):

Ajoutez le fait d'être une femme, l'aîné ainsi que la cohorte de naissance dans le modèle précédemment estimé (`reg1`), que vous nommerez `reg3`. Commentez.

```
reg3 = lm(dipl ~ taille_fratrie + femme + elder + cohorte, data = df)
summary(reg3)$coefficients
```

```
##                      Estimate Std. Error t value Pr(>|t| )
## (Intercept)      3.15069381 0.019786255 159.236487 0.000000e+00
## taille_fratrie -0.19262139 0.004075047 -47.268506 0.000000e+00
## femme            0.17376695 0.007530153  23.076152 1.847406e-117
## elder            0.17772660 0.008028901  22.135856 2.889803e-108
## cohorte1950-1954 0.09620064 0.017811983   5.400894 6.648653e-08
## cohorte1955-1959 0.15039841 0.017217896   8.735005 2.479539e-18
## cohorte1960-1964 0.25878125 0.016979848  15.240493 2.233512e-52
## cohorte1965-1969 0.35762224 0.017100432  20.913052 7.146752e-97
## cohorte1970-1974 0.61404121 0.017486228  35.115704 3.254311e-268
## cohorte1975-1979 0.84382368 0.018821059  44.834018 0.000000e+00
## cohorte1980-1984 0.90656076 0.022549126  40.203810 0.000000e+00
## cohorte1985-1989 0.92565969 0.037983631  24.369963 1.001889e-130
```

# Question 13 (1pt):

Ajoutez la catégorie socio-professionnelle du père et de la mère dans le modèle précédent que vous nommerez `reg4`. Que constatez-vous?

```
reg4 = lm(dipl ~ taille_fratrie + femme + elder + cohorte + father_cs + mother_cs, data = df)
summary(reg4)$coefficients
```

```
##                                     Estimate Std. Error t value
## (Intercept)                   2.685226425 0.023328476 115.1050929
## taille_fratrie                -0.145921630 0.003626647 -40.2359620
## femme                          0.172687306 0.006649850  25.9686020
## elder                          0.180400937 0.007090294  25.4433659
## cohorte1950-1954              0.094565202 0.015730883   6.0114364
## cohorte1955-1959              0.153926456 0.015206225  10.1225953
## cohorte1960-1964              0.255318376 0.014996212  17.0255248
## cohorte1965-1969              0.362324427 0.015102776  23.9905847
## cohorte1970-1974              0.624452874 0.015443584  40.4344530
## cohorte1975-1979              0.844202869 0.016623121  50.7848597
## cohorte1980-1984              0.902384714 0.019917041  45.3071668
## cohorte1985-1989              0.956064488 0.033546963  28.4992857
## father_csArtisan               0.265097159 0.014627416  18.1233081
## father_csCadre                 0.834919389 0.013021732  64.1173831
## father_csEmployé                -0.008228766 0.017322788 -0.4750255
## father_csOuvrier                0.306764689 0.012062572  22.4824494
```

# Question 14 (1pt):

Régressez le niveau de diplôme sur l'interaction entre `femme` et `elder` ainsi que la taille de la fratrie et la CSP des parents. Nommez ce modèle `reg5`. Commentez.

```
reg5 = lm(dipl ~ taille_fratrie + femme*elder + cohorte + father_cs + mother_cs, data = df)
summary(reg5)$coefficients
```

	Estimate	Std. Error	t value
## (Intercept)	2.672444518	0.023482672	113.8049602
## taille_fratrie	-0.145867973	0.003626213	-40.2259796
## femme	0.197959736	0.008541508	23.1762056
## elder	0.212864568	0.009884103	21.5360531
## cohorte1950-1954	0.094552712	0.015728924	6.0113911
## cohorte1955-1959	0.153884729	0.015204333	10.1211103
## cohorte1960-1964	0.255425890	0.014994361	17.0347966
## cohorte1965-1969	0.362209167	0.015100914	23.9859095
## cohorte1970-1974	0.624421225	0.015441662	40.4374374
## cohorte1975-1979	0.844255782	0.016621054	50.7943588
## cohorte1980-1984	0.902712220	0.019914681	45.3289813
## cohorte1985-1989	0.956632803	0.033543001	28.5195951
## father_csArtisan	0.265095072	0.014625593	18.1254234
## father_csCadre	0.834817992	0.013020128	64.1174961
## father_csEmployé	-0.008248555	0.017320631	-0.4762272
## father_csOuvrier	0.306000941	0.012061977	22.4957488

# Question 15 (1pt):

Reprenez le modèle de la question 13 en remplaçant elder par le rang de naissance et nommez-le reg6. Commentez.

```
reg6 = lm(dipl ~ taille_fratrie + femme + as.factor(rang_naissance) + cohorte + father_cs + mother_cs, data = df)
summary(reg6)$coefficients
```

	Estimate	Std. Error	t value
## (Intercept)	2.735928852	0.023379414	117.0229880
## taille_fratrie	-0.101560758	0.004333799	-23.4345813
## femme	0.172327233	0.006636269	25.9674858
## as.factor(rang_naissance)2	-0.141402283	0.007543564	-18.7447573
## as.factor(rang_naissance)3	-0.268904951	0.011042638	-24.3515144
## as.factor(rang_naissance)4	-0.372482501	0.017498372	-21.2866940
## as.factor(rang_naissance)5	-0.577816070	0.030139688	-19.1712693
## cohorte1950-1954	0.104350488	0.015717076	6.6393068
## cohorte1955-1959	0.169204456	0.015202156	11.1302935
## cohorte1960-1964	0.273257451	0.015000539	18.2165091
## cohorte1965-1969	0.382143792	0.015113031	25.2857148
## cohorte1970-1974	0.649030032	0.015470566	41.9525708
## cohorte1975-1979	0.872904223	0.016659840	52.3957149
## cohorte1980-1984	0.933886842	0.019947779	46.8165824
## cohorte1985-1989	1.002899194	0.033571661	29.8733859
## father_csArtisan	0.265357271	0.014597385	18.1784120
## father csCadre	0.834493278	0.012995140	64.2157959

# Question 16 (2pts):

- Régressez la taille de la fratrie sur la variable `compo_genre`, le fait d'être une femme, l'aîné, la cohorte de naissance et la PCS des parents. Nommez ce modèle `reg7`. Commentez uniquement les effets observés de la variable `compo_genre`.
- En utilisant la fonction `ivreg` du package du même nom, estimez l'effet de la taille de la fratrie en utilisant comme instrument `compo_genre` et nommez ce modèle `reg8`. Commentez.

```
reg7 = lm(taille_fratrie ~ compo_genre + femme + elder + cohorte + mother_cs + father_cs, data = df)
summary(reg7)$coefficients
```

```
##                                     Estimate Std. Error t value
## (Intercept)                   3.170076447 0.020826957 152.2102582
## compo_genreFM                 -0.126057932 0.009280820 -13.5826286
## compo_genreMF                 -0.127317803 0.009313872 -13.6696962
## compo_genreMM                 -0.022558162 0.010531356 -2.1419997
## femme                          -0.013442102 0.007522922 -1.7868192
## elder                          -0.429746679 0.006522572 -65.8860784
## cohorte1950-1954              0.015178480 0.014835647  1.0231087
## cohorte1955-1959              0.041988830 0.014340201  2.9280503
## cohorte1960-1964              0.010926505 0.014142713  0.7725891
## cohorte1965-1969              -0.018782468 0.014243214 -1.3186960
## cohorte1970-1974              -0.075318340 0.014562776 -5.1719769
## cohorte1975-1979              -0.121553097 0.015671864 -7.7561354
## cohorte1980-1984              -0.152664056 0.018777488 -8.1301638
```

# Question 16 (2pts):

- Régressez la taille de la fratrie sur la variable `compo_genre`, le fait d'être une femme, l'aîné, la cohorte de naissance et la PCS des parents. Nommez ce modèle `reg7`. Commentez uniquement les effets observés de la variable `compo_genre`.
- **En utilisant la fonction `ivreg` du package du même nom, estimez l'effet de la taille de la fratrie en utilisant comme instrument `compo_genre` et nommez ce modèle `reg8`. Commentez.**

```
reg8 = ivreg(dipl ~ taille_fratrie + femme + elder + cohorte + father_cs + mother_cs | compo_genre + femme + elder +  
summary(reg8)$coefficients
```

	Estimate	Std. Error	t value
## (Intercept)	2.954547930	0.178887350	16.5162485
## taille_fratrie	-0.232900482	0.057391416	-4.0581065
## femme	0.172396767	0.006675017	25.8271669
## elder	0.142869553	0.025718127	5.5552083
## cohorte1950-1954	0.095999140	0.015812149	6.0712266
## cohorte1955-1959	0.157505263	0.015438433	10.2021535
## cohorte1960-1964	0.256220970	0.015058516	17.0150217
## cohorte1965-1969	0.360792197	0.015187258	23.7562432
## cohorte1970-1974	0.618058503	0.016057575	38.4901525
## cohorte1975-1979	0.833754579	0.018042524	46.2105290
## cohorte1980-1984	0.889335284	0.021753388	40.8826110
## cohorte1985-1989	0.956239158	0.033660285	28.4085283
## father_csArtisan	0.248827649	0.018171040	13.6936384

## Question 17 (2pts):

Selon-vous, le modèle précédent permet-il d'estimer l'effet causal de la taille de la fratrie sur le niveau de diplôme?