Etude de la nocivité du tabac chez les femmes

Victor

October 14, 2021

Contents

1 Préambule

1.1 Contexte de l'exercice

Dans le cadre du *MOOC* de l'INRIA sur la recherche reproductible, je réalise un travail pratique évalué par les pairs. Ce travail pratique intervient au cours du module 3. Le sujet choisi est le numéro 6, intitulé : *Autour du Paradoxe de Simpson*.

1.2 Origine des données

Nous utilisons les données mises à disposition pour l'exercice par l'équipe pédagogique du MOOC. Ces données sont un extrait de celles d'études de l'incidence des pathologies de la thyroïde au sein de la population britannique. Furent exclus du jeu de donnée complet :

- les hommes (n = 1285) ;
- les femmes ayant arrêté de fumer (n = 162)
- les femmes dont les données n'étaient pas disponibles (n = 18).

URL des données : `https://gitlab.inria.fr/learninglab/mooc-rr/mooc-rr-ressources/-/raw/master/module3/Practical_session/Subject6_smoking.csv?inline=false`

Nous téléchargeons le jeu de donnée de l'exercice au format `.csv`. Dans ce jeu de données, chaque ligne correspond aux données d'une participante lors des études originales. Nous disposons des informations suivantes :

Nom de colonne	Description de la variable dépendante
Smoker	Si la personne fume ou non.
Status	Si la personne est vivante ou décédée.
Age	L'âge de la personne si elle est vivante, l'âge à sa mort si elle est décédée.

1.3 Bibliothèques et fonctions utilisées

Pour la suite des manipulations de données, nous faisons l'usage de plusieurs fonctions disponibles dans des bibliothèques Python.

```
from os.path import exists
from pandas import read_csv
from matplotlib import pyplot as plt, rc
```

```
from seaborn import histplot, kdeplot, lmplot
from urllib import request
from numbers import Real
from numpy import nan, arange
```

1.4 Paramètres communs des graphiques

Nous souhaitons que les graphiques soient réalisés avec TEX.

```
# Draw in LaTeX
rc('font', **{'family': 'serif', 'serif': ['Courier New']})
plt.rcParams['text.usetex'] = True
```

1.5 Import des données

Dans le cas où une copie locale n'est pas présente, nous téléchargeons le jeu de données complet depuis son adresse de dépôt.

```
data_file = "Subject6_smoking.csv"

if not exists(data_file):
    request.urlretrieve(data_url, data_file)
    print('Data set downloaded. \n')

data = read_csv(filepath_or_buffer='Subject6_smoking.csv')

data.info(verbose=True)
print('\n', data.head())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1314 entries, 0 to 1313
Data columns (total 3 columns):
Smoker    1314 non-null object
Status    1314 non-null object
Age       1314 non-null float64
dtypes: float64(1), object(2)
memory usage: 30.9+ KB

   Smoker Status   Age
0    Yes  Alive  21.0
1    Yes  Alive  19.3
```

```
2     No   Dead  57.5
3     No  Alive  47.1
4    Yes  Alive  81.4
```

1.6 Vérifications automatiques des données

Vérifions le jeu de données :

- nous ne devons avoir que les valeurs *Yes* et *No* dans la colonne *Smoker* ;
- seules les catégories *ALive* et *Dead* doivent être présentent dans la colonne *Status* ;
- l'âge doit être un réel entre 0 et 123 (le record de longévité attesté étant de moins de 123 ans).

En cas de valeur incorrecte, nous remplaçons ladite valeur par `nan` (*Not a number*).

```
for row in data.iterrows():
    if row[1][0] != 'Yes' and row[1][0] != 'No':
        print(f"Uncorrect value in 'Smoker' column: {row[1][0]} (index = {row[0]})")
        data['Smoker'].iat[row[0]] = nan
    if row[1][1] != 'Alive' and row[1][1] != 'Dead':
        print(f"Uncorrect value in 'Status' column: {row[1][1]} (index = {row[0]})")
        data['Status'].iat[row[0]] = nan
    if not isinstance(row[1][2], Real) and not 0 <= row[1][2] <= 123:
        print(f"Uncorrect value in 'Age' column: {row[1][2]} (index = {row[0]})")
        data['Age'].iat[row[0]] = nan
```

1.7 Visualisations des données brutes

1.7.1 Distribution des âges des participantes

Nous souhaitons visualiser les distributions des âges selon les habitudes de tabagisme.

Constatons que les distributions en âge des participantes selon les habitudes de tabagisme ne correspondent pas : les femmes de plus de 60 ans sont davantage présentes dans le groupe des non-fumeuses.

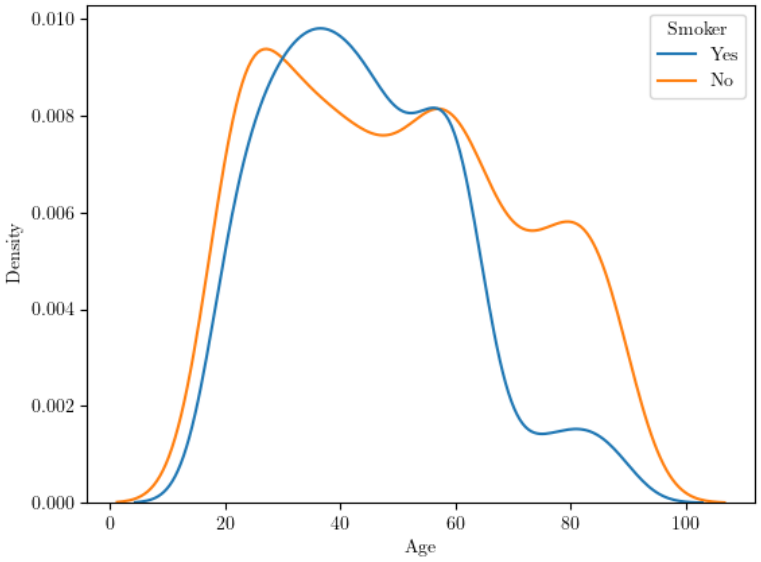

Figure 1: Distribution des âges des participantes selon les habitudes de tabagisme (estimation par noyau)

2 Analyses de la nocivité du tabagisme chez les femmes

Dans les analyses suivantes, nous souhaitons évaluer la nocivité du tabagisme chez les femmes. Dans le cadre de notre exercice, nous utilisons trois approches différentes.

Notre hypothèse est que le tabac est nocif chez les femmes.

2.1 Approche 1 | Taux de mortalité des fumeuses et non-fumeuses

Nous cherchons à déterminer les taux de mortalité des deux groupes de notre étude. Nous souhaitons associer ces taux de mortalité à des intervalles de confiance à 95%.

2.1.1 Calcul des paramètres

Commençons par séparer les deux groupes. Le nombre de participantes mortes dans un groupe ramené à l'effectif total du groupe nous donne le taux de mortalité dudit groupe.

```
rounding = 1

smokers = data[(data['Smoker'] == 'Yes')]
nonsmokers = data[(data['Smoker'] == 'No')]

dead_smokers = smokers[(smokers['Status'] == 'Dead')]
dead_nonsmokers = nonsmokers[(nonsmokers['Status'] == 'Dead')]

smokers_death_rate = round(dead_smokers.shape[0] / smokers.shape[0] * 100, rounding)
nonsmokers_death_rate = round(dead_nonsmokers.shape[0] / nonsmokers.shape[0] * 100, rou
```

Nous déterminons l'intervalle de confiance à 95% avec la formule suivante :

$\text{IC}_{95} = \frac{100}{n} d \pm 1.96\sqrt{d}$

Où `n` correspond à l'effectif du groupe considéré et `d` au nombre d'individus décédés au sein du même groupe. Voici le code que nous avons utilisé pour réaliser le calcul des intervalles de confiance :

```
def death_rate_ci95(sample_size: int,
                    deads: int,
```

```
                    per_people: int = 100) -> tuple:
    """Compute the 95% confidence interval upper and lower limits of a mortality rate i

    :param sample_size: number of people in the sample.
    :param deads: number of deads people in the sample.
    :param per_people: standardisation number: deads for per_people people.
    """
    from math import sqrt


    upper = per_people / sample_size * (deads + 1.96 * sqrt(deads))
    lower = per_people / sample_size * (deads - 1.96 * sqrt(deads))

    return lower, upper

smokers_dr_95ic = death_rate_ci95(smokers.shape[0], dead_smokers.shape[0])
nonsmokers_dr_95ic = death_rate_ci95(nonsmokers.shape[0], dead_nonsmokers.shape[0])
```

2.1.2 Résultats

Tabagisme	Effectif total	Individus décédés	Taux de mortalité (%)	IC95%
Fumeuses	582	139	23.9	[19.9;27.9]
Non fumeuses	732	230	31.4	[27.4;35.5]

Pour le jeu de donnée considéré, le taux de mortalité des fumeuses est de `23.9` contre `31.4` pour celui observé chez les non-fumeuses. Ce qui va à l'encontre de notre hypothèse initiale. Il est possible que l'hétérogénéité constatée dans la distribution des âges explique ce résultat contre intuitif.

2.2 Approche 2 | Taux de mortalité des fumeuses et non-fumeuses selon la classe d'âge

Nous étudions désormais le taux de mortalité selon les habitudes de tabagisme et selon la classe d'âge. Nous considérons quatre classes :

- [18 ; 34[
- [34 ; 54[
- [54 ; 64[
- [64 ; 123[

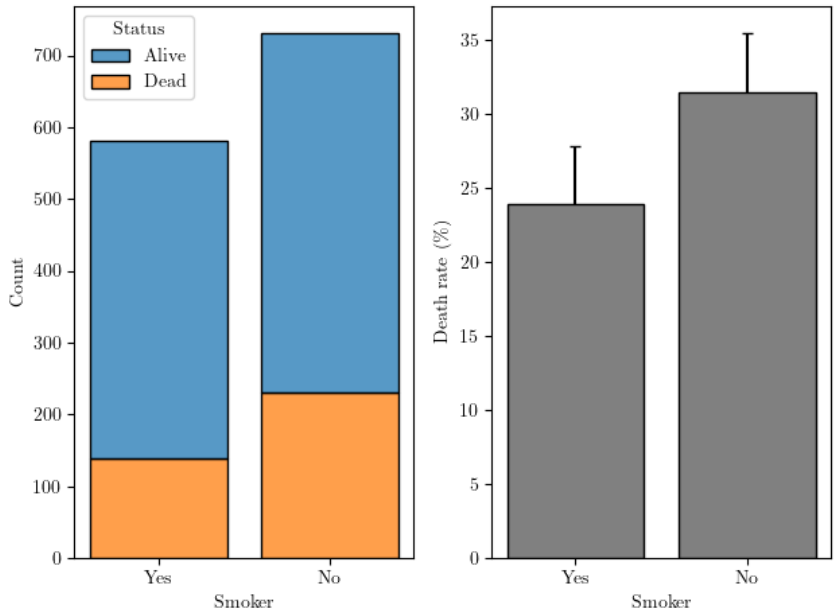

Figure 2: Effectifs et taux de mortalité parmi les fumeuses et les non fumeuses.

2.2.1 Calcul des paramètres et résultats

Premièrement nous devons séparer le jeu de donnée selon les classes d'âge. Puis nous procédons de la même manière que précédemment : nous identifions les effectifs de chaque groupe, le nombre de décès au sein des sous-groupes et nous calculons le taux de mortalité associé à son intervalle de confiance à 95%.

```
categories = [(18, 34), (34, 54), (54, 64), (64, 123)]

death_rates = []  # plotting purpose
dr_uncertainties = []  # plotting purpose

print(f"|Classe d'âge|Tabagisme|Effectif total|Individus décédés|\
        Taux de mortalité (%)|IC95%|\n"
      f"|-+-+-+-+-+-|")

for category in categories:
    for smoker in data['Smoker'].unique():

        alive, dead = nan, nan

        if smoker == 'Yes':
            smoke = 'Fumeuses'
        else:
            smoke = 'Non-fumeuses'

        for status in data['Status'].unique():

            subset = data[(data['Age'] >= category[0])
                          & (data['Age'] < category[1])
                          & (data['Smoker'] == smoker)
                          & (data['Status'] == status)]

            if status == 'Alive':
                alive = subset.shape[0]
            else:
                dead = subset.shape[0]

        death_rate = dead / (alive + dead) * 100
        death_rates.append(death_rate)
```

```
        dr_ci95 = death_rate_ci95(alive + dead, dead)
        dr_uncertainties.append(dr_ci95[1] - death_rate)

        rounded_dr_ci95 = list(dr_ci95)
        rounded_dr_ci95[0] = round(rounded_dr_ci95[0], rounding)
        rounded_dr_ci95[1] = round(rounded_dr_ci95[1], rounding)
        r_dr_ci95 = tuple(rounded_dr_ci95)

        print(f"|[{category[0]};{category[1]}[|{smoke}|{alive + dead}|\
                {dead}|{round(death_rate, rounding)}|{r_dr_ci95}|")
```

Classe d'âge	Tabagisme	Effectif total	Individus décédés	Taux de mortalité (%)	IC95%
[18;34[	Fumeuses	179	5	2.8	(0.3, 5.2)

```
fig, ax = plt.subplots()
x_pos = arange(len(categories))
width = 0.35
xlabels = list(map(str, categories))
for x_l, xlabel in enumerate(xlabels):
    head, _, tail = xlabel.partition(',')
    xlabels[x_l] = '[' + head[1:] +  ';' + tail[:-1] + '['
plt.bar(x=x_pos - width / 2,
        height=death_rates[0:len(death_rates):2],
        yerr=dr_uncertainties[0:len(death_rates):2],
        label='Smoker', width=width, edgecolor='k',
        error_kw=dict(lolims=True))
plt.bar(x=x_pos + width / 2,
        height=death_rates[1:len(death_rates):2],
        yerr=dr_uncertainties[1:len(death_rates):2],
        label='Nonsmoker', width=width, edgecolor='k',
        error_kw=dict(lolims=True))
# In order to modify the default lolims error bars shape
for ch in ax.get_children():
    if str(ch).startswith('Line2D'):
        ch.set_marker('_')
        ch.set_markersize(6)
plt.ylabel('Death rate (%)')
plt.xlabel('Age category')
plt.xticks(x_pos)
ax.set_xticklabels(xlabels)
plt.legend()
plt.tight_layout()
plt.savefig("age_categories_death_rate.png")
```

Pour chacune de nos classes d'âge, les résultats indiquent que le taux de mortalité des fumeuses est supérieur à celui des non-fumeuses ; soit une interprétation des données en contradiction avec la première analyse sans prise en compte de l'âge.

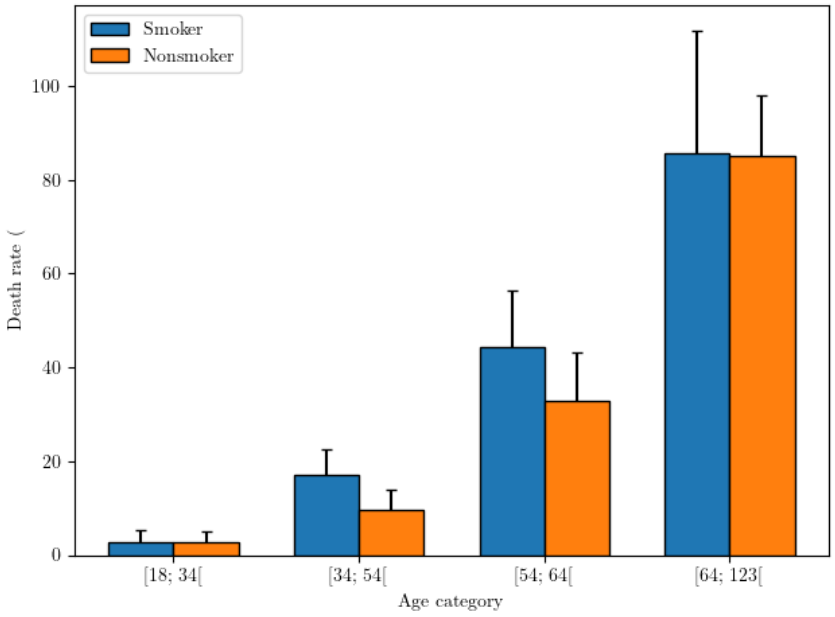

Figure 3: Taux de mortalité et IC95% selon les habitudes de tabagisme et la classe d'âge

Outre un biais potentiel lié aux distributions des âges hétérogènes, d'autres biais sont susceptibles d'exister dans les analyses précédentes :

- un biais lié à la non prise en compte de l'âge dans le calcul des taux de mortalité ;
- un biais lié aux choix de catégorisation.

En somme au moins trois biais peuvent se combiner dans nos études précédentes et fausser nos interprétations.
Enfin, notons que l'âge semble avoir un effet supérieur au tabac sur le taux de mortalité puisque l'amplitude des différences entre les classes d'âge est supérieure à celle entre les habitudes de tabagisme. En somme : le tabac tue, mais vieillir tue davantage.

2.3 Approche 3 | Probabilité de décès des fumeuses et non-fumeuses selon l'âge

Evitons les deux derniers biais en passant du calcul du taux de mortalité au calcul de la probabilité de décès selon l'âge et les habitudes de tabagisme. Pour ce faire nous pouvons utiliser la régression logistique.

2.3.1 Manipulation des données

Plutôt que de disposer des status `Alive` et `Dead`, nous voulons qu'ils soient respectivement codés avec `0` et `1`. Nous souhaitons à nouveau disposer de jeux de données spécifiques à chaque groupe : fumeuses et non fumeuses.

```
deaths = []

for status in data['Status']:
    if status == 'Alive':
        deaths.append(0)
    else:
        deaths.append(1)

data['Death'] =  deaths

smokers = data[(data['Smoker'] == 'Yes')]
nonsmokers = data[(data['Smoker'] == 'No')]
print(smokers.head(), '\n', '\n', nonsmokers.head())
```

```
  Smoker Status   Age  Death
0    Yes  Alive  21.0      0
1    Yes  Alive  19.3      0
4    Yes  Alive  81.4      0
7    Yes   Dead  57.5      1
8    Yes  Alive  24.8      0

    Smoker Status   Age  Death
2       No   Dead  57.5      1
3       No  Alive  47.1      0
5       No  Alive  36.8      0
6       No  Alive  23.8      0
11      No   Dead  66.0      1
```

2.3.2 Régressions logistiques

1. Réglages et code de la modélisation Pour effectuer la régression logistique, nous utilisons la bibliothèque `statsmodels`. Nous laissons les paramètres par défaut : le modèle est le logit, la méthode est celle du maximum de vraisemblance (*Maximum Likelyhood Estimation*).

```
def logistic_regression(independent_variable: list,
                        dependent_variable: list) -> object:
    """Logisitic regression with Logit model and MLE method.

    :param independent_variable: the independent variable
    :param dependent_variable: the dependent variable

    :return fitted_model: the logisitc regression fitted model.
    """
    import statsmodels.api as sm

    model = sm.Logit(dependent_variable, sm.add_constant(independent_variable))
    fitted_model = model.fit()

    print('\n', fitted_model.summary())

    return fitted_model
```

2. Cas des fumeuses

```
proba_smokers, model_smokers = logistic_regression(smokers['Age'],
                                                   smokers['Death'])
```

```
/usr/lib/python3/dist-packages/numpy/core/fromnumeric.py:2495: FutureWarning: Meth
  return ptp(axis=axis, out=out, **kwargs)
Optimization terminated successfully.
         Current function value: 0.412727
         Iterations 7
```

```
                           Logit Regression Results
==============================================================================
Dep. Variable:                  Death   No. Observations:                  582
Model:                          Logit   Df Residuals:                      580
Method:                           MLE   Df Model:                            1
Date:               jeu., 14 oct. 2021   Pseudo R-squ.:                  0.2492
Time:                         10:06:32   Log-Likelihood:                -240.21
converged:                       True   LL-Null:                       -319.94
Covariance Type:            nonrobust   LLR p-value:                 1.477e-36
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const         -5.5081      0.466    -11.814      0.000      -6.422      -4.594
Age            0.0890      0.009     10.203      0.000       0.072       0.106
==============================================================================
```

3. Cas des non-fumeuses

```
proba_nonsmokers, model_nonsmokers = logistic_regression(nonsmokers['Age'],
                                                         nonsmokers['Death'])
```

```
/usr/lib/python3/dist-packages/numpy/core/fromnumeric.py:2495: FutureWarning: Meth
  return ptp(axis=axis, out=out, **kwargs)
Optimization terminated successfully.
         Current function value: 0.354560
         Iterations 7
```

```
                           Logit Regression Results
==============================================================================
Dep. Variable:                  Death   No. Observations:                  732
```

```
Model:                          Logit   Df Residuals:                      730
Method:                           MLE   Df Model:                            1
Date:              jeu., 14 oct. 2021   Pseudo R-squ.:                  0.4304
Time:                          10:06:32   Log-Likelihood:                -259.54
converged:                         True   LL-Null:                       -455.62
Covariance Type:            nonrobust   LLR p-value:                 2.808e-87
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const         -6.7955      0.479    -14.174      0.000      -7.735      -5.856
Age            0.1073      0.008     13.742      0.000       0.092       0.123
==============================================================================
```

2.3.3 Visualisation des résultats

```
fig, ax = plt.subplots()
ax.set_xlim(15, 100)
ax.set_ylabel('Death probability')
ax.set_xlabel('Age')

lmplot(data=data, x='Age', y='Death', hue='Smoker',
       logistic=True, y_jitter=0.01, truncate=False,
       markers=['o', 'x'], scatter_kws=dict(alpha=0.3))

plt.tight_layout()

plt.savefig('death_probabilities.png')
```

L'analyse graphique des résultats semble indiquer une plus forte probabilité de décès des femmes fumeuses avant 55 ans. Après cet âge, les probabilités de décès ne paraissent plus influencées par les habitudes de tabagisme. Le tabac chez les femmes est donc nocif, surtout avant 55 ans.

3 Paradoxe de Simpson

Nous sommes face à un paradoxe de Simpson quand la tendance observée au sein de sous-groupes disparaît ou est inversée quand les sous-groupes sont combinés.

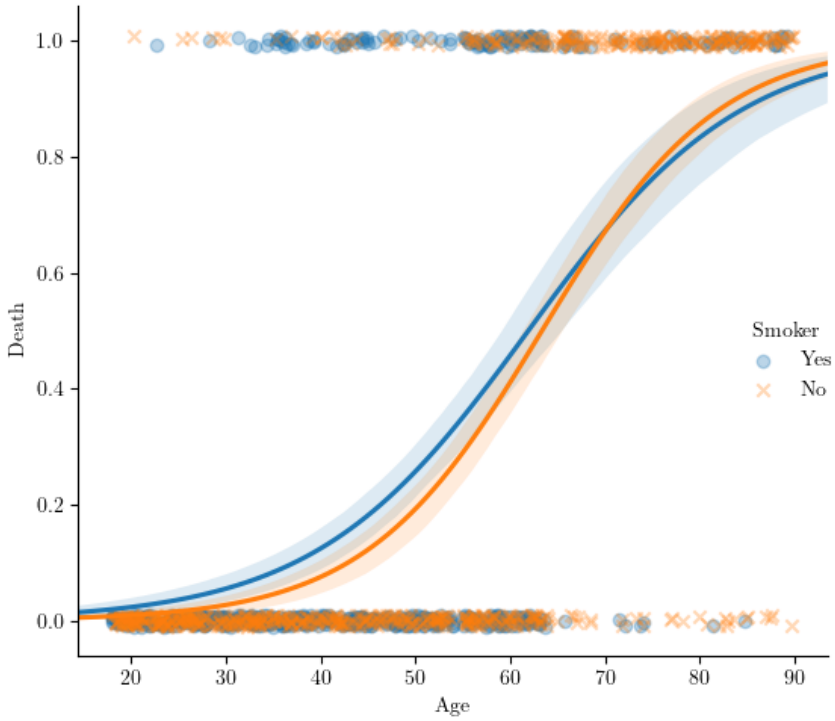

Figure 4: Estimation des probabilités de décès selon l'âge et les habitudes de tabagisme.

3.1 Qu'en est-il du paradoxe de Simpson dans notre cas ?

Lors de notre première approche, les résultats indiquent que le taux de mortalité est supérieur chez les non-fumeuses (`31.4` contre `23.9`). Lors de la deuxième approche, lors de la prise en compte de l'âge, les résultats indiquent l'inverse : le taux de mortalité est supérieur chez les fumeuses. Nous sommes face à un paradoxe de Simpson : la prise en compte ou non de la variable `Age` inverse les interprétations que nous pouvons faire des effets du tabagisme chez les femmes.

3.2 Quelles sont les explications dans notre cas ?

Il est probable que le paradoxe observé dans le cadre de cet exercice s'explique en partie par les faits suivants :

- il existe une différence de `150` individus entre les effectifs des fumeuses et des non fumeuses, soit environ `11.0` % de l'effectif total, ce qui n'est pas négligeable ;
- les distributions en âge au sein des groupes (selon les habitudes de tabagisme) ne coïncident pas ;
- l'âge a un effet supérieur au tabac sur le taux de mortalité.