Notebook - UE Prétraitement et manipulation de données

Quentin Fouché

28 avril 2022

1 Présentation des données

1.1 Première manipulation d'un jeu de données

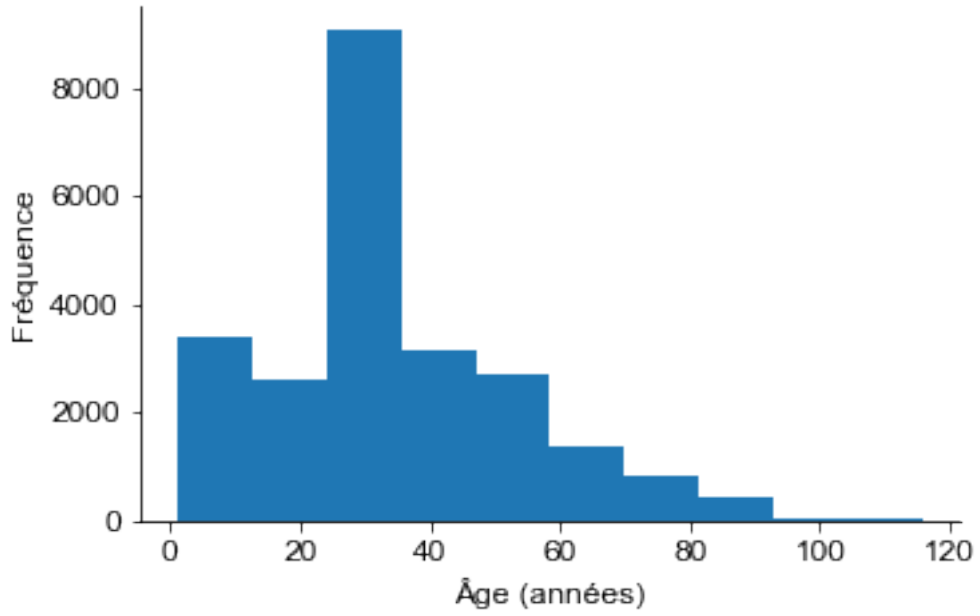
[1]:
```
# (1) Import des packages
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches
import seaborn as sns
import itertools
import sys
!{sys.executable} -m pip install missingno
import missingno as msno
!{sys.executable} -m pip install upsetplot
from upsetplot import plot, from_indicators
import warnings
warnings.filterwarnings("ignore") # permet de ne pas afficher les messages␣
↪d'erreurs
```

[2]:
```
# (2) Import du jeu de données "age_gender.csv"
age_gender = pd.read_csv(r"C:\Users\quent\Documents\DU Data Analyst 2022\UE␣
↪n°1 - Manipulation et prétraitement de données\Jeux de données\age_gender.
↪csv")
print(age_gender.head())
```

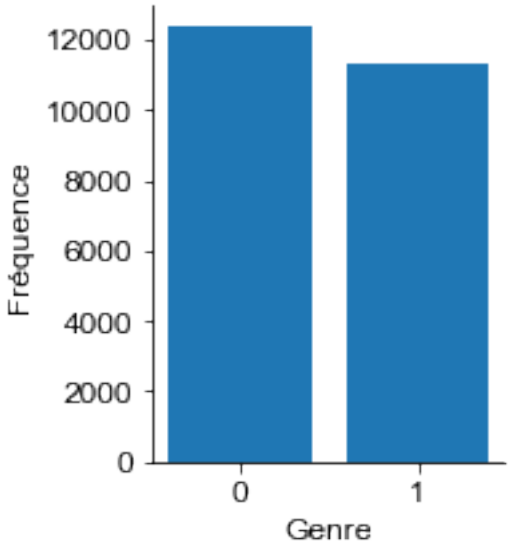
```
   age  genre
0    1      0
1    1      0
2    1      0
3    1      0
4    1      0
```

[3]:
```
# (3) Représentation de la distribution de la variable âge
plt.figure(figsize = (14/2.54, 9/2.54))
plt.rc('font', family = 'Arial', size = 12)
plt.hist(age_gender["age"], bins = 10)
plt.ylabel("Fréquence")
```

```
plt.xlabel("Âge (années)")
plt.gca().spines['right'].set_visible(False)
plt.gca().spines['top'].set_visible(False)
```



[4]:
```
# (4) Représentation de la distribution de la variable genre
plt.figure(figsize=(6/2.54,8/2.54))
plt.rc('font', family = 'Arial', size = 12)
plt.bar(x = range(2), height = age_gender["genre"].value_counts(), tick_label␣
↪= ["0","1"])
plt.ylabel("Fréquence")
plt.xlabel("Genre")
plt.gca().spines['right'].set_visible(False)
plt.gca().spines['top'].set_visible(False)
```



1.2 Analyse d'un jeu de données réelles

[5]:
```
# (1) Tentative d'import du jeu de données "PhD_v1.csv"
PhD_v1 = pd.read_csv(r"C:\Users\quent\Documents\DU Data Analyst 2022\UE n°1 -␣
↪Manipulation et prétraitement de données\Jeux de données\PhD_v1.csv")
```

```
---------------------------------------------------------------------------
[...]
ParserError: Error tokenizing data. C error: Expected 1 fields in line 3,
saw 7
```

[6]:
```
# (2) Import du jeu de données "PhD_v2.csv"
PhD_v2 = pd.read_csv(r"C:\Users\quent\Documents\DU Data Analyst 2022\UE n°1 -␣
↪Manipulation et prétraitement de données\Jeux de données\PhD_v2.csv",␣
↪low_memory = False)
PhD_v2.head()
```

[6]:
```
                Auteur Identifiant auteur  \
0       Saeed Al marri                NaN
1    Andrea Ramazzotti          174423705
2  OLIVIER BODENREIDER                NaN
3       Emmanuel Porte                NaN
4     Arthur Devriendt                NaN

                                               Titre  \
0  Le credit documentaire et l'onopposabilite des...
1  Application de la PGD a la resolution de probl...
2  Conception d'un outil informatique d'etude des...
3  Socio-histoire des politiques publiques en mat...
4  LES TECHNOLOGIES DE L'INFORMATION ET DE LA COM...

                          Directeur de these  \
0                        Philippe Delebecque
1  Jean-Claude Grandidier,Marianne Beringhier
2                             Francois Kohler
3                               Gilles Pollet
4                               Gabriel Dupuy

              Directeur de these (nom prenom) Identifiant directeur  \
0                           Delebecque Philippe              29561248
1  Grandidier Jean-Claude,Beringhier Marianne           715,441,511
2                             Kohler Francois              57030758
3                               Pollet Gilles                    na
4                               Dupuy Gabriel                    na

                       Etablissement de soutenance  \
```

```
0                                          Paris 1
1  Chasseneuil-du-Poitou, Ecole nationale superie...
2                                          Nancy 1
3                                           Lyon 2
4                                          Paris 1

  Identifiant etablissement  \
0                  27361802
1                  28024400
2                       NaN
3                  02640334X
4                  27361802

                                      Discipline    Statut  \
0                                    Driot prive   enCours
1  Mecanique des solides, des materiaux, des stru...   enCours
2                                       Medecine  soutenue
3                                Science politique   enCours
4                                       Geographie   enCours

  Date de premiere inscription en doctorat Date de soutenance    Year  \
0                                30-09-11                NaN     NaN
1                                01-10-12                NaN     NaN
2                                     NaN           01-01-93  1993.0
3                                01-06-11                NaN     NaN
4                                07-12-09                NaN     NaN

  Langue de la these Identifiant de la these Accessible en ligne  \
0                NaN                  s69480                 non
1                NaN                  s98826                 non
2                 fr            1993NAN19006                 non
3                NaN                  s88867                 non
4                NaN                  s89663                 non

  Publication dans theses.fr Mise a jour dans theses.fr
0                   26-01-12                  26-01-12
1                   22-11-13                  22-11-13
2                   24-05-13                  17-11-12
3                   12-07-13                  12-01-16
4                   13-07-13                  12-07-13
```

[7]:
```
# (3) Identification de la nature des variables
PhD_v2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 447644 entries, 0 to 447643
Data columns (total 18 columns):
 #   Column                                Non-Null Count   Dtype
---  ------                                --------------   -----
 0   Auteur                                447644 non-null  object
 1   Identifiant auteur                    317655 non-null  object
```

```
 2   Titre                                       447635 non-null  object
 3   Directeur de these                          447629 non-null  object
 4   Directeur de these (nom prenom)             447629 non-null  object
 5   Identifiant directeur                       447644 non-null  object
 6   Etablissement de soutenance                 447640 non-null  object
 7   Identifiant etablissement                   430559 non-null  object
 8   Discipline                                  447639 non-null  object
 9   Statut                                      447644 non-null  object
 10  Date de premiere inscription en doctorat    63976 non-null   object
 11  Date de soutenance                          390898 non-null  object
 12  Year                                        390898 non-null  float64
 13  Langue de la these                          383879 non-null  object
 14  Identifiant de la these                     447644 non-null  object
 15  Accessible en ligne                         447644 non-null  object
 16  Publication dans theses.fr                  447644 non-null  object
 17  Mise a jour dans theses.fr                  447467 non-null  object
dtypes: float64(1), object(17)
memory usage: 61.5+ MB
```

2 Données manquantes

2.1 Répartition des données manquantes

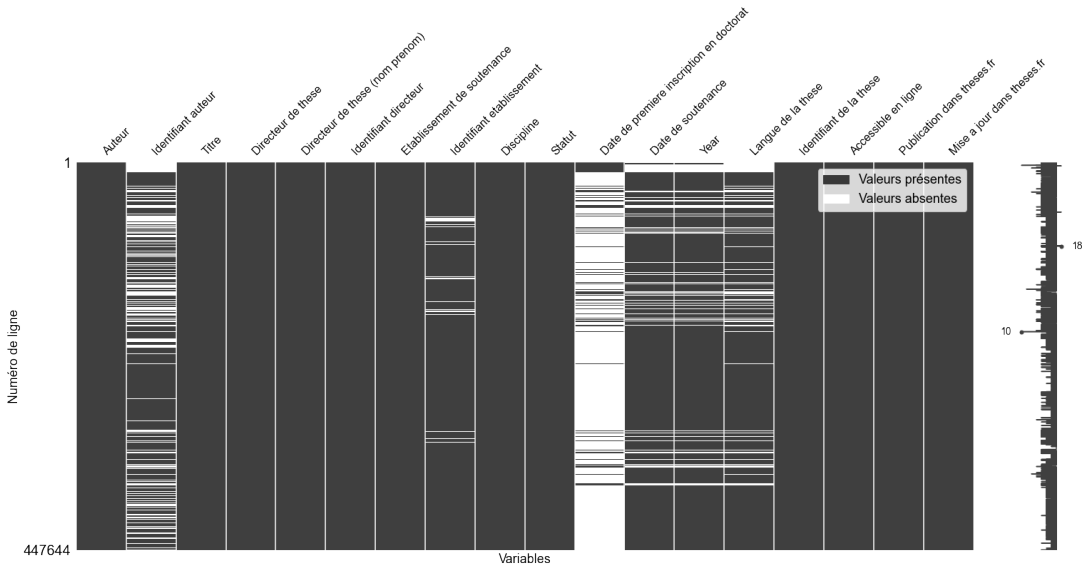
[8]:
```
# (1) Calcul du pourcentage de valeurs manquantes par variable
PhD_v2.isna().mean()*100
```

[8]:
```
Auteur                                      0.000000
Identifiant auteur                         29.038477
Titre                                       0.002011
Directeur de these                          0.003351
Directeur de these (nom prenom)             0.003351
Identifiant directeur                       0.000000
Etablissement de soutenance                 0.000894
Identifiant etablissement                   3.816649
Discipline                                  0.001117
Statut                                      0.000000
Date de premiere inscription en doctorat   85.708286
Date de soutenance                         12.676591
Year                                       12.676591
Langue de la these                         14.244578
Identifiant de la these                     0.000000
Accessible en ligne                         0.000000
Publication dans theses.fr                  0.000000
Mise a jour dans theses.fr                  0.039540
dtype: float64
```

[9]:
```
# (2) Représentation de la distribution des valeurs manquantes par variable
msno.matrix(PhD_v2)
plt.ylabel("Numéro de ligne", fontsize = 18)
plt.xlabel("Variables", fontsize = 18)
gray_patch = mpatches.Patch(color = '#383838', label = 'Valeurs présentes')
```

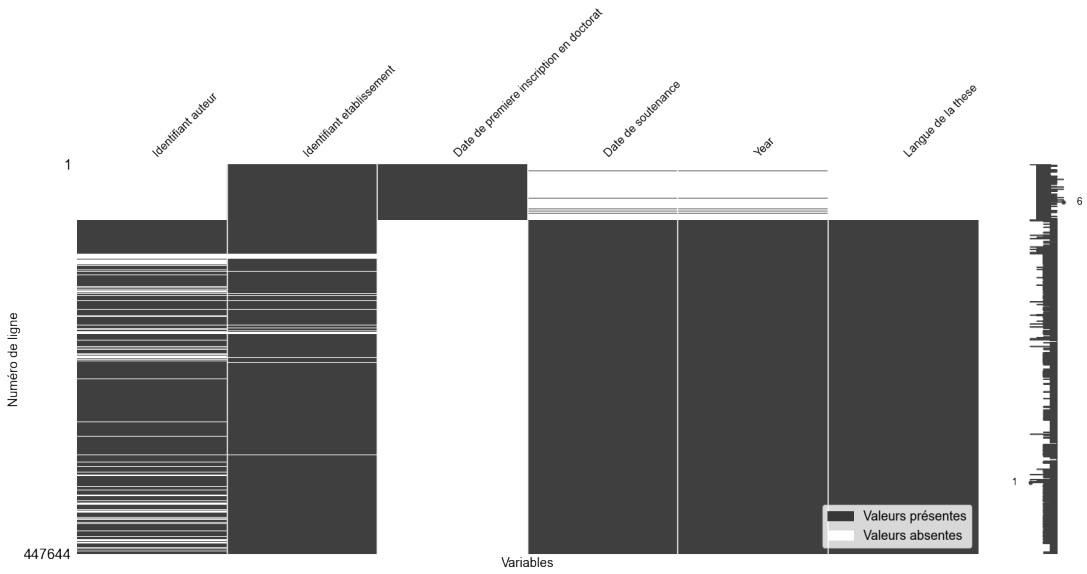
```
white_patch = mpatches.Patch(color = 'white', label = 'Valeurs absentes')
plt.legend(handles = [gray_patch, white_patch], loc = "upper right", fontsize␣
↪= 18)
```

[9]: <matplotlib.legend.Legend at 0x2169fa1bb50>



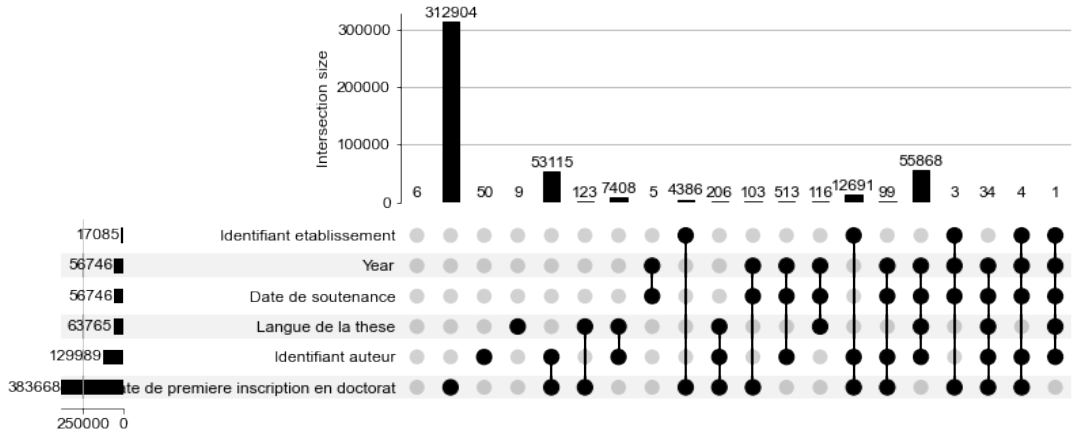
[10]:
```
# (3) Représentation de la distribution des valeurs manquantes dans les 6␣
↪variables en contenant le plus
PhD_v2_missing_data = PhD_v2.loc[:,["Identifiant auteur", "Identifiant␣
↪etablissement", "Date de premiere inscription en doctorat", "Date de␣
↪soutenance", "Year", "Langue de la these"]]
msno.matrix(PhD_v2_missing_data.sort_values("Date de premiere inscription en␣
↪doctorat"))
plt.ylabel("Numéro de ligne", fontsize = 18)
plt.xlabel("Variables", fontsize = 18)
gray_patch = mpatches.Patch(color = '#383838', label = 'Valeurs présentes')
white_patch = mpatches.Patch(color = 'white', label = 'Valeurs absentes')
plt.legend(handles = [gray_patch, white_patch], loc = "lower right", fontsize␣
↪= 18)
```

[10]: <matplotlib.legend.Legend at 0x21699eb3b80>



[11]:
```
# (4) Représentation de la distribution des valeurs manquantes dans chaque␣
 ↪combinaison de variables parmi les 6 qui en contiennent le plus
plt.rc('font', family = 'Arial', size = 12)
plot(from_indicators(indicators = pd.isna, data = PhD_v2_missing_data),␣
 ↪show_counts = True)
# remarque : pour afficher le pourcentage de thèses dans l'histogramme␣
 ↪"Intersection size", ajouter "show_percentages = True"
```

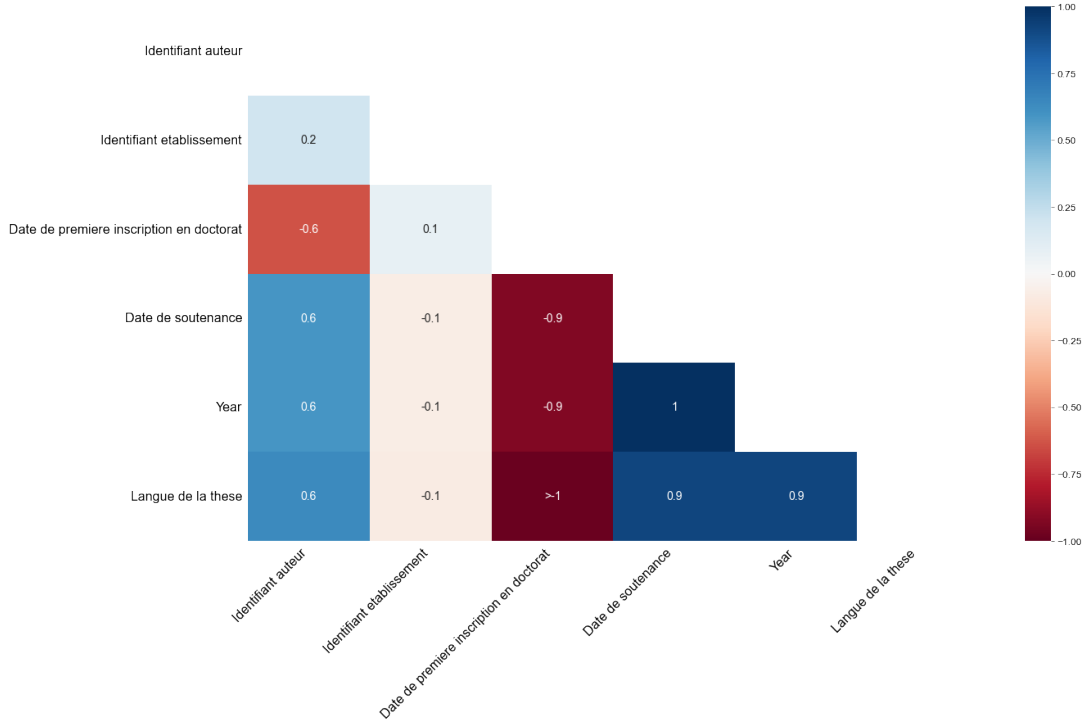
[11]:
```
{'matrix': <AxesSubplot:>,
 'shading': <AxesSubplot:>,
 'totals': <AxesSubplot:>,
 'intersections': <AxesSubplot:ylabel='Intersection size'>}
```



2.2 Corrélations entre les données manquantes

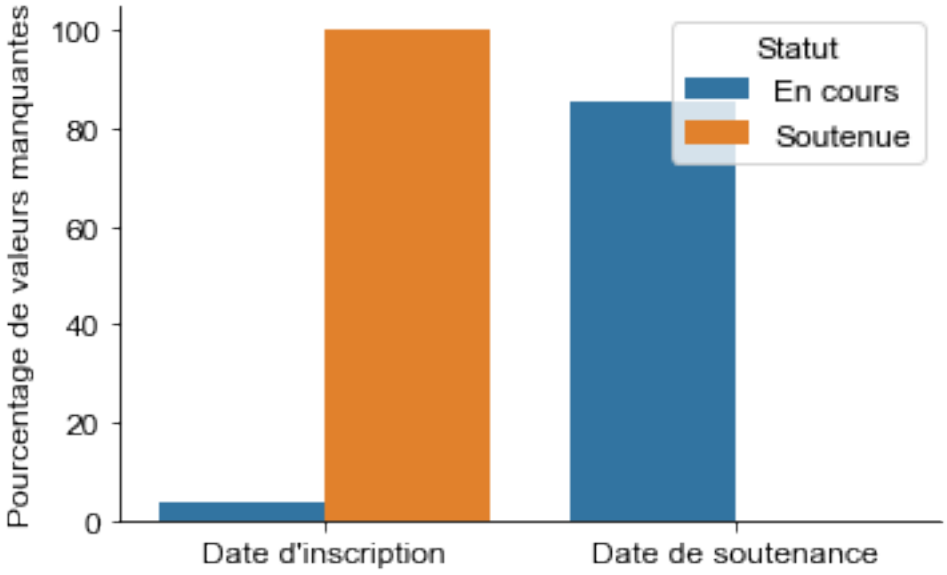
[12]:
```
# (1) Représentation de la proportion de données manquantes communes par␣
 ↪paire de variables
plt.rc('font', family = 'Arial', size = 12)
msno.heatmap(PhD_v2_missing_data)
```

[12]:
```
<AxesSubplot:>
```



[13]:
```
# (2) Création d'un tableau affichant le pourcentage de valeurs manquantes en␣
 ↪fonction du statut de la thèse pour les variables "Date de premiere␣
 ↪inscription en doctorat" et "Date de soutenance"
PhD_v2_enCours_missing_data = PhD_v2[PhD_v2["Statut"] == "enCours"].isna().
 ↪mean() * 100
PhD_v2_soutenue_missing_data = PhD_v2[PhD_v2["Statut"] == "soutenue"].isna().
 ↪mean() * 100
Date_inscr_enCours = PhD_v2_enCours_missing_data["Date de premiere␣
 ↪inscription en doctorat"]
Date_inscr_soutenue = PhD_v2_soutenue_missing_data["Date de premiere␣
 ↪inscription en doctorat"]
Date_soutenance_enCours = PhD_v2_enCours_missing_data["Date de soutenance"]
Date_soutenance_soutenue = PhD_v2_soutenue_missing_data["Date de soutenance"]
```

```
Prop_missing_data = pd.DataFrame([["En cours", "Date d'inscription",␣
 ↪Date_inscr_enCours], ["En cours", "Date de soutenance",␣
 ↪Date_soutenance_enCours], ["Soutenue", "Date d'inscription",␣
 ↪Date_inscr_soutenue], ["Soutenue", "Date de soutenance",␣
 ↪Date_soutenance_soutenue]], columns = ["Statut", "Date", "Pourcentage de␣
 ↪valeurs manquantes"])

# (3) Représentation du pourcentage de valeurs manquantes en fonction du␣
 ↪statut de la thèse
plt.figure(figsize = (14/2.54, 9/2.54))
plt.rc('font', family = 'Arial', size = 12)
sns.barplot(data = Prop_missing_data, x = "Date", y = "Pourcentage de valeurs␣
 ↪manquantes", hue = "Statut")
plt.xlabel("")
plt.gca().spines['right'].set_visible(False)
plt.gca().spines['top'].set_visible(False)
```



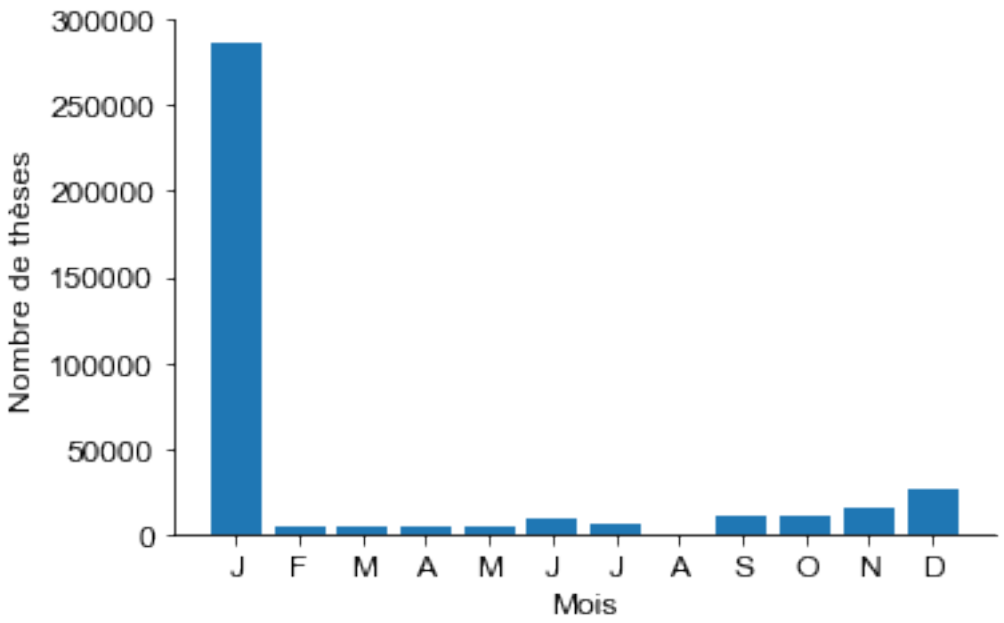
3 Principaux problèmes détectés

3.1 Anomalie dans le mois de soutenance

3.1.1 Distribution du mois de soutenance entre 1984 et 2018

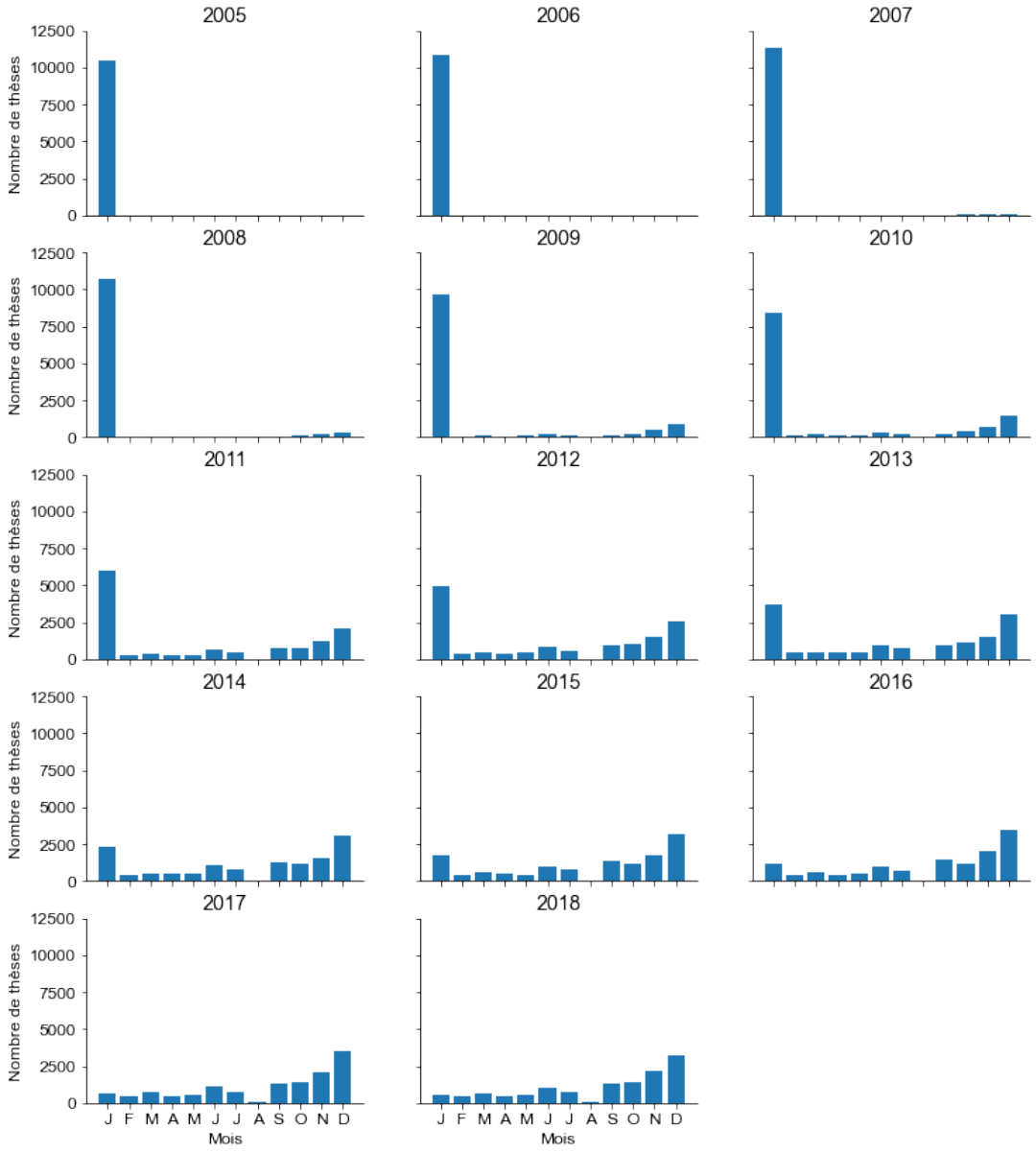
[14]:
```
# (1) Conversion de la variable 'Date de soutenance' en type 'datetime'
PhD_v2["Date de soutenance"] = pd.to_datetime(PhD_v2["Date de soutenance"],␣
 ↪format = "%d-%m-%y")

# (2) Suppression des thèses n'ayant pas de date de soutenance
PhD_v2.dropna(subset=["Date de soutenance"], how = "all", inplace = True)
```

```
# (3) Création des variables "Année de soutenance", "Mois de soutenance" et "Jour de soutenance"
PhD_v2["Année de soutenance"] = PhD_v2["Date de soutenance"].dt.year
PhD_v2["Mois de soutenance"] = PhD_v2["Date de soutenance"].dt.month
PhD_v2["Jour de soutenance"] = PhD_v2["Date de soutenance"].dt.day

# (4) Sélection des données sur la période 1984-2018
PhD_v2_1984_2018 = PhD_v2[np.logical_and(PhD_v2["Date de soutenance"] >="1984-01-01", PhD_v2["Date de soutenance"] < "2019-01-01")]

# (5) Calcul du nombre de thèses par mois de soutenance sur la période 1984-2018
PhD_v2_1984_2018_distr_mois = PhD_v2_1984_2018.groupby("Mois de soutenance").size()

# (6) Représentation de la distribution du mois de soutenance sur la période 1984-2018
plt.figure(figsize = (14/2.54, 9/2.54))
plt.rc('font', family = 'Arial', size = 12)
plt.bar(x = range(12), height = PhD_v2_1984_2018_distr_mois, tick_label =["J","F","M","A","M","J","J","A","S","O","N","D"])
plt.ylabel("Nombre de thèses")
plt.xlabel("Mois")
plt.gca().spines['right'].set_visible(False)
plt.gca().spines['top'].set_visible(False)
```



3.1.2 Distribution du mois de soutenance pour chaque année de 2005 à 2018

[15]:
```
# (1) Sélection des données sur la période 2005-2018
PhD_v2_2005_2018 = PhD_v2[np.logical_and(PhD_v2['Date de soutenance'] >=␣
 ↪"2005-01-01", PhD_v2['Date de soutenance'] < "2019-01-01")]

# (2) Calcul du nombre de thèses soutenues par mois et par année
PhD_v2_2005_2018_distr_mois_annee = PhD_v2_2005_2018.pivot_table(values =␣
 ↪'Date de soutenance', index = 'Mois de soutenance', columns = 'Année de␣
 ↪soutenance', aggfunc = 'count')

# (3) Représentation de la distribution du mois de soutenance pour chaque␣
 ↪année de 2005 à 2018
plt.rc('font', family = 'Arial', size = 12)
fig, axs = plt.subplots(5, 3, figsize=(12,14))
plt.setp(axs, ylim=(0,12500))

for i in range(5):
    for j in range(3):
        if 2005+i*3+j < 2019:
            axs[i, j].bar(x = range(12), height =␣
 ↪PhD_v2_2005_2018_distr_mois_annee[(2005+i*3+j)], tick_label =␣
 ↪["J","F","M","A","M","J","J","A","S","O","N","D"])
            axs[i, j].set_title(str(2005+i*3+j))
            axs[i, j].spines['top'].set_visible(False)
            axs[i, j].spines['right'].set_visible(False)
        else:
            fig.delaxes(axs[4][2])

for ax in axs.flat:
    ax.set(ylabel='Nombre de thèses', xlabel="Mois")

for ax in axs.flat:
    ax.label_outer()
```



3.1.3 Proportion de thèses soutenues par mois et par année entre 2005 et 2018

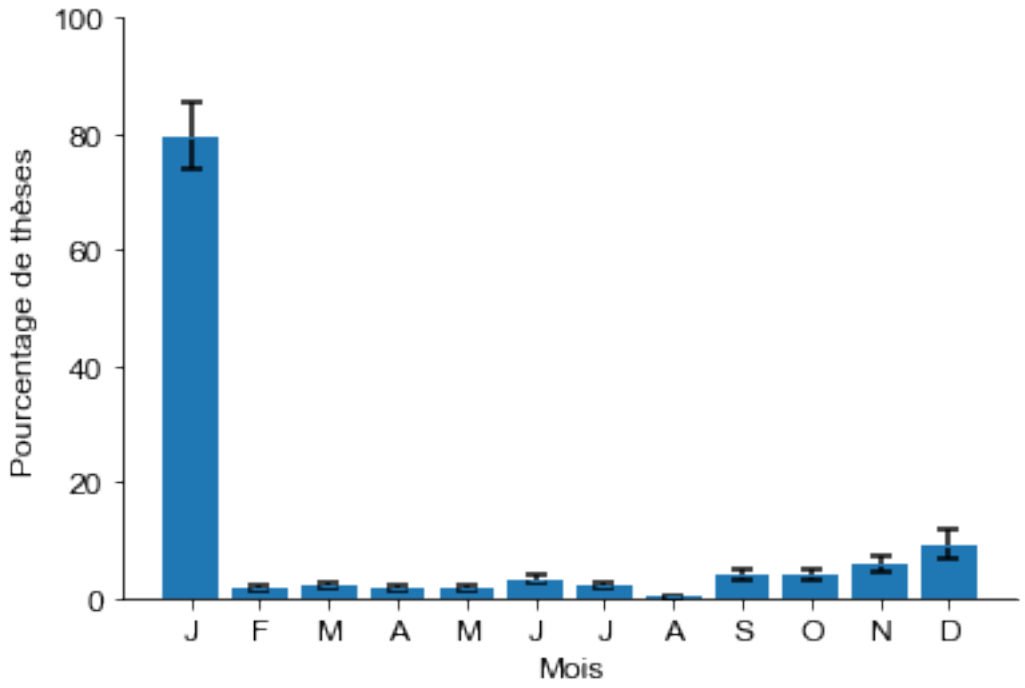
[16]:
```
# (1) Calcul de la proportion de thèses soutenues par mois et par année sur␣
 ↪la période 2005-2018
PhD_v2_2005_2018_prop_mois_annee = PhD_v2_2005_2018_distr_mois_annee.
 ↪copy(deep = True)

for i in range(14):
    PhD_v2_2005_2018_prop_mois_annee[(2005+i)] =␣
 ↪PhD_v2_2005_2018_prop_mois_annee[(2005+i)] /␣
 ↪PhD_v2_2005_2018_prop_mois_annee[2005+i].sum()
```

```
# (2) Représentation de la proportion de thèses soutenues par mois pour␣
 ↪chaque année de 2005 à 2018
plt.rc('font', family = 'Arial', size = 12)
fig, axs = plt.subplots(5, 3, figsize=(12,14))
plt.setp(axs, ylim=(0,1))

for i in range(5):
    for j in range(3):
        if 2005+i*3+j < 2019:
            axs[i, j].bar(x = range(12), height =␣
 ↪PhD_v2_2005_2018_prop_mois_annee[(2005+i*3+j)], tick_label =␣
 ↪["J","F","M","A","M","J","J","A","S","O","N","D"])
            axs[i, j].set_title(str(2005+i*3+j))
            axs[i, j].spines['top'].set_visible(False)
            axs[i, j].spines['right'].set_visible(False)
        else:
            fig.delaxes(axs[4][2])

for ax in axs.flat:
    ax.set(ylabel='Proportion de thèses', xlabel="Mois")

for ax in axs.flat:
    ax.label_outer()
```

3.1.4 Pourcentage moyen de thèses soutenues par mois entre 1984 et 2018

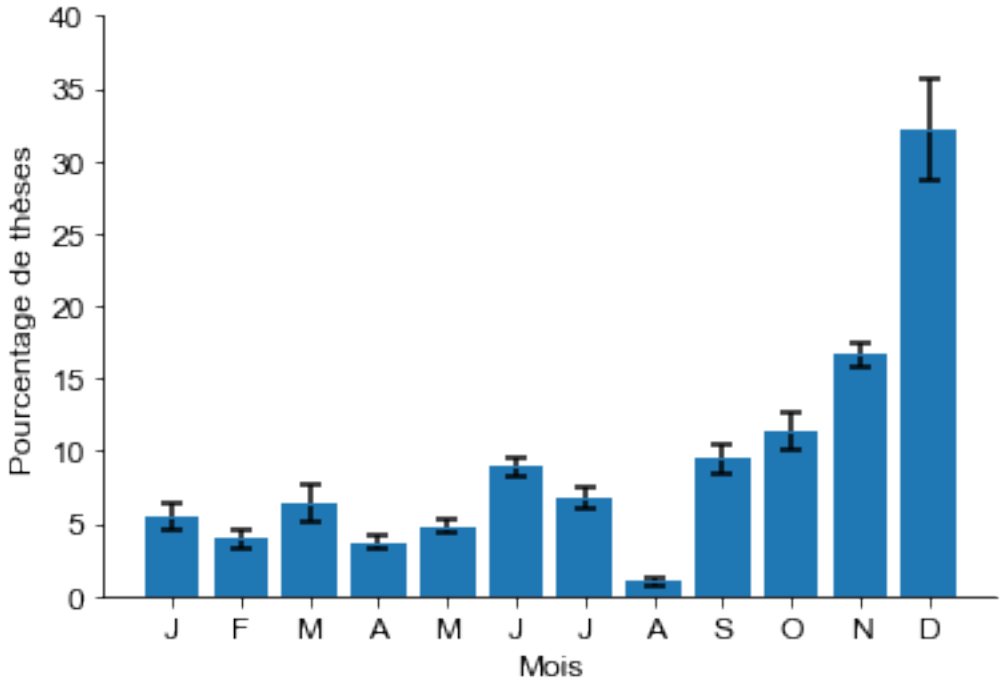
[17]:
```
# (1) Calcul du nombre de thèses soutenues par mois et par année entre 1984␣
 ↪et 2018
PhD_v2_distr_mois_annee_1984_2018 = PhD_v2_1984_2018.pivot_table(values =␣
 ↪'Date de soutenance', index = 'Mois de soutenance', columns = 'Année de␣
 ↪soutenance', aggfunc = 'count')

# (2) Calcul de la proportion de thèses soutenues par mois et par année
PhD_v2_prop_mois_annee_1984_2018 = PhD_v2_distr_mois_annee_1984_2018.
 ↪copy(deep = True)

for i in PhD_v2_1984_2018['Année de soutenance'].unique():
```

```
    PhD_v2_prop_mois_annee_1984_2018[i] = PhD_v2_prop_mois_annee_1984_2018[i]␣
 ↪/ PhD_v2_prop_mois_annee_1984_2018[i].sum()

# (3) Calcul du pourcentage moyen de thèses soutenues par mois
PhD_v2_prop_mois_mean_1984_2018 = []
for i in range(1,13):
    PhD_v2_prop_mois_mean_1984_2018.append(PhD_v2_prop_mois_annee_1984_2018.
 ↪loc[i,].mean()*100)

# (4) Calcul de l'erreur standard pour chaque pourcentage moyen de thèses
PhD_v2_prop_mois_esm_1984_2018 = []
for i in range(1,13):
    PhD_v2_prop_mois_esm_1984_2018.append(PhD_v2_prop_mois_annee_1984_2018.
 ↪loc[i,].sem()*100)

# (5) Représentation du pourcentage moyen de thèses soutenues par mois
plt.rc('font', family = 'Arial', size = 12)
plt.bar(x = range(12), height = PhD_v2_prop_mois_mean_1984_2018, tick_label =␣
 ↪["J","F","M","A","M","J","J","A","S","O","N","D"])
plt.errorbar(x = range(12), y = PhD_v2_prop_mois_mean_1984_2018, yerr =␣
 ↪PhD_v2_prop_mois_esm_1984_2018, fmt='_', capsize = 4, capthick = 1.5,␣
 ↪ecolor = "black")
plt.ylabel("Pourcentage de thèses")
plt.xlabel("Mois")
plt.ylim(0,100)
plt.gca().spines['right'].set_visible(False)
plt.gca().spines['top'].set_visible(False)
```



3.1.5 Pourcentage moyen de thèses soutenues par mois sans le 1er janvier (1984-2018)

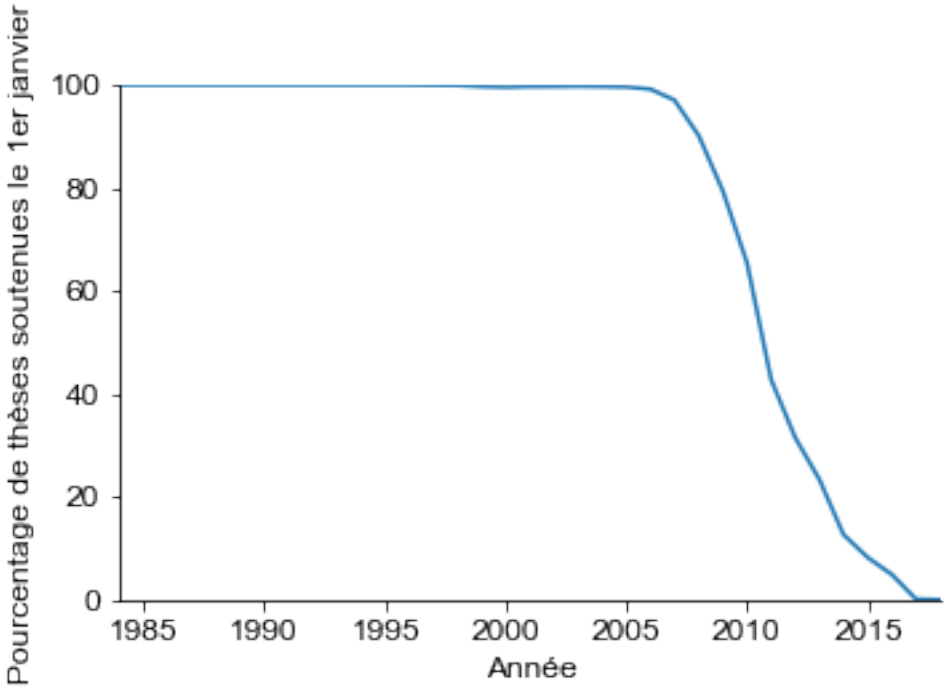
[18]:
```
# (1) Exclusion des thèses soutenues le 1er janvier
PhD_v2_sans_janv_1984_2018 = PhD_v2_1984_2018[PhD_v2['Mois de soutenance'] !=␣
 ↪1]
PhD_v2_janv_sans_1er_1984_2018 = PhD_v2_1984_2018[np.
 ↪logical_and(PhD_v2_1984_2018['Mois de soutenance'] == 1,␣
 ↪PhD_v2_1984_2018['Jour de soutenance'] != 1)]
PhD_v2_sans_1er_janv_1984_2018 = PhD_v2_sans_janv_1984_2018.
 ↪append(PhD_v2_janv_sans_1er_1984_2018)

# (2) Calcul du nombre de thèses soutenues par mois et par année
PhD_v2_distr_mois_annee_1984_2018 = PhD_v2_sans_1er_janv_1984_2018.
 ↪pivot_table(values = 'Date de soutenance', index = 'Mois de soutenance',␣
 ↪columns = 'Année de soutenance', aggfunc = 'count')

# (3) Calcul de la proportion de thèses soutenues par mois et par année
PhD_v2_prop_mois_annee_1984_2018 = PhD_v2_distr_mois_annee_1984_2018.
 ↪copy(deep = True)

for i in PhD_v2_sans_1er_janv_1984_2018['Année de soutenance'].unique():
    PhD_v2_prop_mois_annee_1984_2018[i] = PhD_v2_prop_mois_annee_1984_2018[i]␣
 ↪/ PhD_v2_prop_mois_annee_1984_2018[i].sum()

# (4) Calcul du pourcentage moyen de thèses soutenues par mois
PhD_v2_prop_mois_mean_1984_2018 = []
for i in range(1,13):
    PhD_v2_prop_mois_mean_1984_2018.append(PhD_v2_prop_mois_annee_1984_2018.
 ↪loc[i,].mean()*100)

# (5) Calcul de l'erreur standard pour chaque pourcentage moyen de thèses
PhD_v2_prop_mois_esm_1984_2018 = []
for i in range(1,13):
    PhD_v2_prop_mois_esm_1984_2018.append(PhD_v2_prop_mois_annee_1984_2018.
 ↪loc[i,].sem()*100)

# (6) Représentation graphique du pourcentage moyen de thèses soutenues par␣
 ↪mois sur la période 1984-2018
plt.rc('font', family = 'Arial', size = 12)
plt.bar(x = range(12), height = PhD_v2_prop_mois_mean_1984_2018, tick_label =␣
 ↪["J","F","M","A","M","J","J","A","S","O","N","D"])
plt.errorbar(x = range(12), y = PhD_v2_prop_mois_mean_1984_2018, yerr =␣
 ↪PhD_v2_prop_mois_esm_1984_2018, fmt='_', capsize = 4, capthick = 1.5,␣
 ↪ecolor = "black")
plt.ylabel("Pourcentage de thèses")
plt.xlabel("Mois")
plt.ylim(0,40)
plt.gca().spines['right'].set_visible(False)
plt.gca().spines['top'].set_visible(False)
```



3.1.6 Pourcentage moyen de thèses soutenues le 1er janvier en fonction de l'année

[19]:
```
# (1) Sélection des thèses soutenues le 1er janvier
PhD_v2_1984_2018_1er_janv = PhD_v2_1984_2018[np.
 ↪logical_and(PhD_v2_1984_2018['Mois de soutenance'] == 1,␣
 ↪PhD_v2_1984_2018['Jour de soutenance'] == 1)]

# (2) Calcul du nombre de thèses soutenues le 1er janvier pour chaque année
PhD_v2_1984_2018_total_1er_janv = PhD_v2_1984_2018_1er_janv.groupby("Année de␣
 ↪soutenance").size()

# (3) Calcul du nombre de thèses soutenues par année
PhD_v2_1984_2018_total_annee = PhD_v2_1984_2018.groupby("Année de␣
 ↪soutenance").size()

# (4) Calcul du pourcentage de thèses soutenues le 1er janvier pour chaque␣
 ↪année
PhD_v2_1984_2018_prop_1er_janv = (PhD_v2_1984_2018_total_1er_janv /␣
 ↪PhD_v2_1984_2018_total_annee) * 100

# (5) Représentation du pourcentage de thèses soutenues le 1er janvier en␣
 ↪fonction de l'année
plt.figure(figsize = (14/2.54, 9/2.54))
plt.rc('font', family = 'Arial', size = 12)
plt.plot(list(PhD_v2_1984_2018_prop_1er_janv.index),␣
 ↪list(PhD_v2_1984_2018_prop_1er_janv))
plt.ylabel("Pourcentage de thèses soutenues le 1er janvier")
```

```
plt.xlabel("Année")
plt.ylim(0,100)
plt.xlim(1984,2018)
plt.gca().spines['right'].set_visible(False)
plt.gca().spines['top'].set_visible(False)
```



3.2 Erreurs liées aux homonymies : le cas de Cécile Martin

[20]:
```
# (1) Sélection des thèses soutenues sous le nom de Cécile Martin
PhD_v2 = pd.read_csv(r"C:\Users\quent\Documents\DU Data Analyst 2022\UE n°1 -␣
↪Manipulation et prétraitement de données\Jeux de données\PhD_v2.csv",␣
↪low_memory = False)
PhD_v2_Cecile_Martin = PhD_v2[PhD_v2["Auteur"] == "Cecile Martin"]

# (2) Calcul du nombre de valeurs identiques pour chaque identifiant auteur
PhD_v2_Cecile_Martin["Identifiant auteur"].value_counts()
```

[20]:
```
81323557     4
203208145    1
179423568    1
182118703    1
Name: Identifiant auteur, dtype: int64
```

[21]:
```
# (3) Affichage du titre des thèses de Cécile Martin dont l'identifiant est␣
↪81323557
PhD_v2_Cecile_Martin[PhD_v2_Cecile_Martin["Identifiant auteur"] ==␣
↪"81323557"]["Titre"]
```

[21]:
```
166820    Systeme laitier et filiere lait au mexique : c...
410228    Modelisation et criteres de combustibilite en ...
414771    Caracterisation electrophysiologique et pharma...
426351    Influence du ph ruminal sur la digestion des p...
Name: Titre, dtype: object
```

[22]:
```
# (4) Affichage du nom de l'établissement de soutenance des thèses de Cécile␣
 ↪Martin dont l'identifiant est 81323557
print(PhD_v2_Cecile_Martin[PhD_v2_Cecile_Martin["Identifiant auteur"] ==␣
 ↪"81323557"]["Etablissement de soutenance"])
```

```
166820    Institut national agronomique Paris-Grignon
410228                                       Compiegne
414771                                     Bordeaux 2
426351                               Clermont-Ferrand 2
Name: Etablissement de soutenance, dtype: object
```

[23]:
```
# (5) Affichage du nom du directeur de thèse
print(PhD_v2_Cecile_Martin[PhD_v2_Cecile_Martin["Identifiant auteur"] ==␣
 ↪"81323557"]["Directeur de these"])
```

```
166820     JEAN LOSSOUARN
410228    Gerard Antonini
414771     Jean Mironneau
426351        Yves Briand
Name: Directeur de these, dtype: object
```

[24]:
```
# (6) Affichage de la discipline de chaque thèse
print(PhD_v2_Cecile_Martin[PhD_v2_Cecile_Martin["Identifiant auteur"] ==␣
 ↪"81323557"]["Discipline"])
```

```
166820    Sciences biologiques fondamentales et applique...
410228                        Genie des procedes industriels
414771                                          Neurosciences
426351    Sciences biologiques et fondamentales applique...
Name: Discipline, dtype: object
```

[25]:
```
# (7) Affichage de la date de soutenance de chaque thèse
print(PhD_v2_Cecile_Martin[PhD_v2_Cecile_Martin["Identifiant auteur"] ==␣
 ↪"81323557"]["Date de soutenance"])
```

```
166820    01-01-00
410228    01-01-01
414771    01-01-91
426351    01-01-94
Name: Date de soutenance, dtype: object
```

[26]:
```
# (8) Affichage de la date de mise à jour dans theses.fr pour chaque thèse
print(PhD_v2_Cecile_Martin[PhD_v2_Cecile_Martin["Identifiant auteur"] ==␣
 ↪"81323557"]["Mise a jour dans theses.fr"])
```

```
166820    10-12-19
410228    08-07-20
414771    07-07-20
```

```
426351    07-07-20
Name: Mise a jour dans theses.fr, dtype: object
```

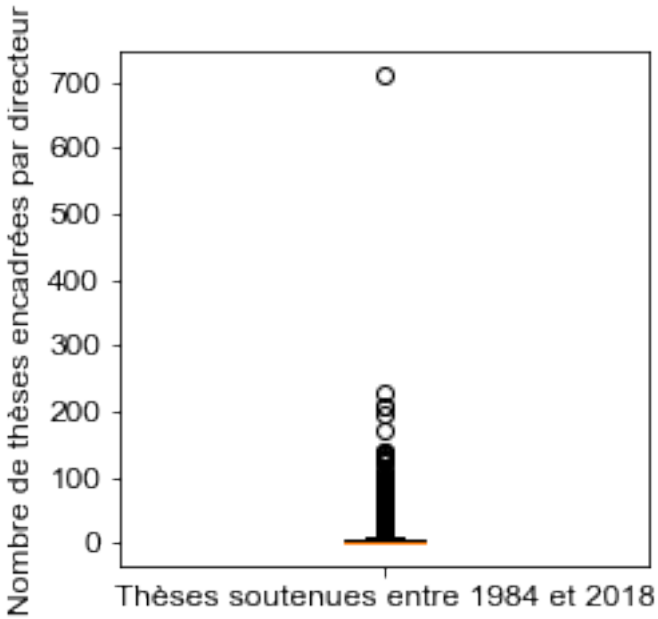
4 Outliers

4.1 Directeurs ayant encadré un nombre relativement anormal de thèses

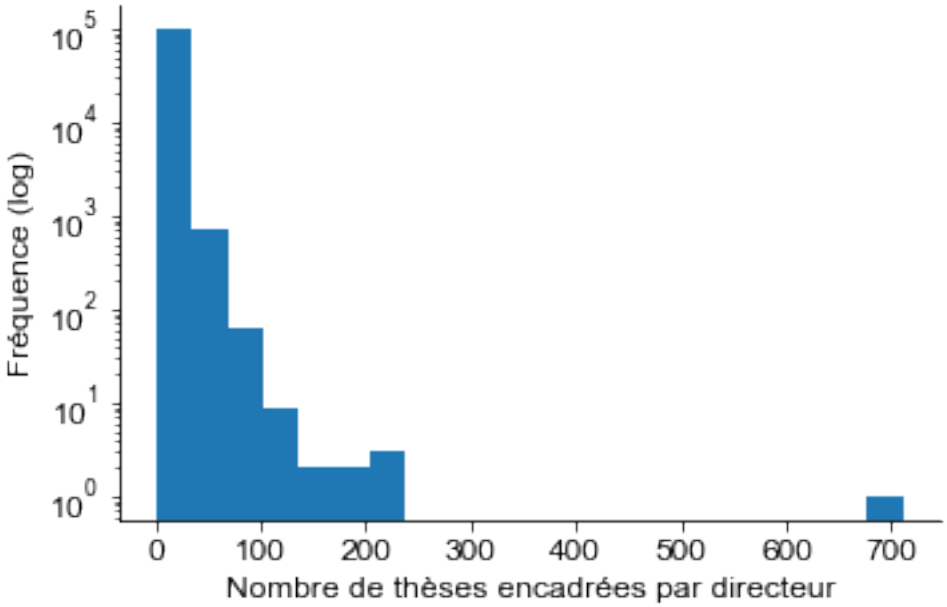
[27]:
```
# (1) Ré-import du jeu de données et suppression des valeurs manquantes de la␣
 ↪colonne "Directeur de these"
PhD_v2 = pd.read_csv(r"C:\Users\quent\Documents\DU Data Analyst 2022\UE n°1 -␣
 ↪Manipulation et prétraitement de données\Jeux de données\PhD_v2.csv",␣
 ↪low_memory = False)
PhD_v2.dropna(subset=["Directeur de these"], how = "all", inplace = True)

# (2) Sélection des thèses soutenues entre 1984 et 2018
PhD_v2["Date de soutenance"] = pd.to_datetime(PhD_v2["Date de soutenance"],␣
 ↪format = "%d-%m-%y")
PhD_v2_1984_2018 = PhD_v2[np.logical_and(PhD_v2["Date de soutenance"] >=␣
 ↪"1984-01-01", PhD_v2["Date de soutenance"] < "2019-01-01")]

# (3) Création d'une liste "Directeurs" regroupant toutes les occurences de␣
 ↪nom de directeur et directrice de thèse
j = list(PhD_v2_1984_2018["Directeur de these"])
k = []

for i in range(0,len(j)):
    if "," in str(j[i]):
        k.append(j[i].split(','))
    else:
        k.append([j[i]])

Directeurs = list(itertools.chain.from_iterable(k))

# (4) Création d'une liste "Directeurs_noms_prenoms" regroupant toutes les␣
 ↪occurences des noms et prénoms de directeur et directrice de thèse
j = list(PhD_v2_1984_2018["Directeur de these (nom prenom)"])
k = []

for i in range(0,len(j)):
    if "," in str(j[i]):
        k.append(j[i].split(','))
    else:
        k.append([j[i]])

Directeurs_noms_prenoms = list(itertools.chain.from_iterable(k))

# (5) Création d'un tableau "Nombre_theses_encadrees" indiquant le nombre de␣
 ↪thèses encadrées par directeur et directrice de thèse, ainsi que leurs nom␣
 ↪et prénom
```

```
my_df = pd.DataFrame({"Directeurs":Directeurs,"Directeurs (nom prenom)":
 ↪Directeurs_noms_prenoms})
my_df = pd.DataFrame({"Nombre de theses encadrees" : my_df.
 ↪groupby(["Directeurs","Directeurs (nom prenom)"]).size()}).reset_index()
my_df = my_df.drop(my_df[my_df["Directeurs"] == ""].index)
my_df = my_df.drop(my_df[my_df["Directeurs"] == " "].index)
my_df = my_df.drop(my_df[my_df["Directeurs"] == "  "].index)
Nombre_theses_encadrees = my_df

# (6) Représentation par boxplot du nombre de thèses encadrées
plt.figure(figsize = (9/2.54, 9/2.54))
plt.rc('font', family = 'Arial', size = 12)
plt.boxplot(Nombre_theses_encadrees["Nombre de theses encadrees"])
plt.ylabel("Nombre de thèses encadrées par directeur")
plt.xticks([1],["Thèses soutenues entre 1984 et 2018"])
```

[27]:
```
([<matplotlib.axis.XTick at 0x216af9d0eb0>],
 [Text(1, 0, 'Thèses soutenues entre 1984 et 2018')])
```



[28]:
```
# (7) Représentation de la distribution du nombre de thèses encadrées par␣
 ↪directeur
plt.figure(figsize = (14/2.54, 9/2.54))
plt.rc('font', family = 'Arial', size = 12)
plt.hist(Nombre_theses_encadrees["Nombre de theses encadrees"], bins = 21)
plt.xlabel("Nombre de thèses encadrées par directeur")
plt.ylabel("Fréquence (log)")
plt.gca().spines['right'].set_visible(False)
plt.gca().spines['top'].set_visible(False)
plt.yscale('log')
```



[29]:
```
# (8) Tri des valeurs de la colonne "Nombre de theses encadrees" par ordre␣
 ↪décroissant
Nombre_theses_encadrees_tri = Nombre_theses_encadrees.sort_values("Nombre de␣
 ↪theses encadrees", ascending = False)

# (9) Affichage des 10 directeurs et directrices ayant encadré le plus de␣
 ↪thèses
Nombre_theses_encadrees_tri[0:10]
```

[29]:

	Directeurs	Directeurs (nom prenom)	Nombre de theses encadrees
24233	Directeur de these inconnu	Directeur de these inconnu	711
32837	Francois-Paul Blanc	Blanc Francois-Paul	227
53175	Jean-Michel Scherrmann	Scherrmann Jean-Michel	209
83336	Pierre Brunel	Brunel Pierre	206
39394	Guy Pujolle	Pujolle Guy	196
69652	Michel Bertucat	Bertucat Michel	173
11501	Bernard Teyssie	Teyssie Bernard	140
41280	Henry de Lumley	Lumley Henry de	139
70680	Michel Maffesoli	Maffesoli Michel	
12734	Bruno Foucart	Foucart Bruno	

```
70680                          136
12734                          135
```

4.2 Enquête sur le directeur le plus prolifique : François-Paul Blanc

[30]:
```
# (1) Sélection des thèses encadrées par François-Paul Blanc
PhD_v2_FP_Blanc = PhD_v2_1984_2018[PhD_v2_1984_2018["Directeur de these"].str.
 ↪contains("Francois-Paul Blanc")]

# (2) Calcul du nombre de valeurs identiques pour chaque identifiant directeur
# (2a) Focus sur les thèses dont François-Paul Blanc était le seul directeur
PhD_v2_FP_Blanc_unique = PhD_v2_FP_Blanc[PhD_v2_FP_Blanc["Directeur de␣
 ↪these"] == "Francois-Paul Blanc"]
print(PhD_v2_FP_Blanc_unique["Identifiant directeur"].value_counts(sort =␣
 ↪True))
```

```
26730774    201
Name: Identifiant directeur, dtype: int64
```

[31]:
```
# (2b) Focus sur les thèses dont François-Paul Blanc était co-directeur
PhD_v2_FP_Blanc_codir = PhD_v2_FP_Blanc[PhD_v2_FP_Blanc["Directeur de these"]␣
 ↪!= "Francois-Paul Blanc"]
print(PhD_v2_FP_Blanc_codir["Identifiant directeur"].value_counts(sort =␣
 ↪True))
```

```
267,307,740    11
267,307,741     7
26730774        4
112,501,095     1
112,299,172     1
3               1
973,903,640     1
Name: Identifiant directeur, dtype: int64
```

[32]:
```
# (3) Calcul du nombre de valeurs identiques pour chaque établissement de␣
 ↪soutenance
PhD_v2_FP_Blanc["Etablissement de soutenance"].value_counts(sort = True)
```

[32]:
```
Perpignan    227
Name: Etablissement de soutenance, dtype: int64
```

[33]:
```
# (4) Calcul du nombre d'auteurs différents
len(PhD_v2_FP_Blanc["Auteur"].value_counts())
```

[33]:
```
227
```

[34]:
```
# (5) Calcul du nombre de valeurs identiques pour chaque langue
PhD_v2_FP_Blanc["Langue de la these"].value_counts(sort = True)
```

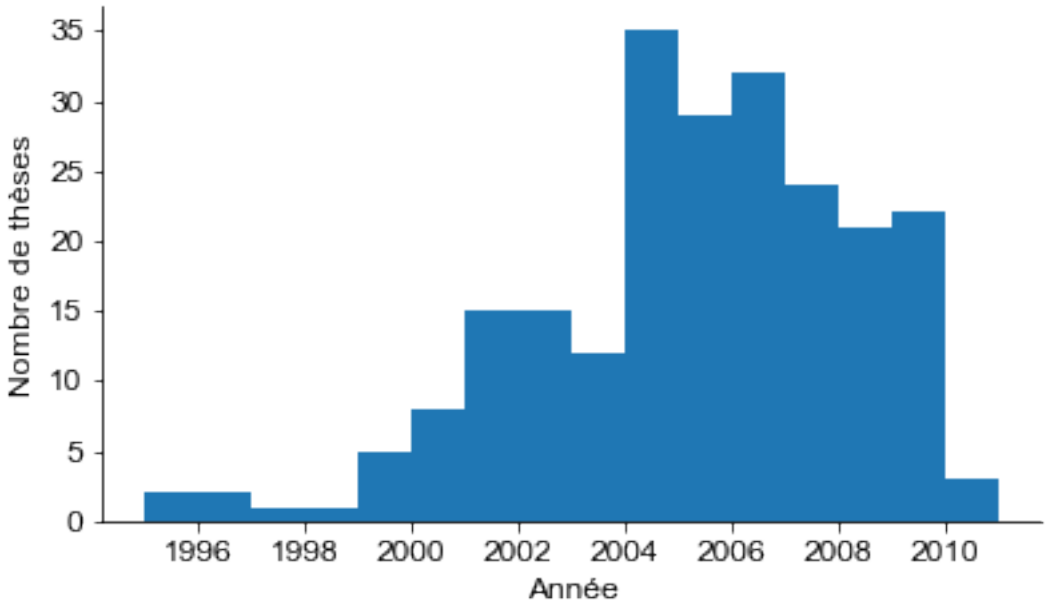
[34]:
```
fr      220
arfr      5
Name: Langue de la these, dtype: int64
```

[35]:
```
# (6) Calcul du nombre de valeurs identiques pour chaque discipline
PhD_v2_FP_Blanc["Discipline"].value_counts(sort = True)
```

[35]:
```
Droit prive                                                    86
Droit                                                          37
Droit prive et sciences criminelles                            35
Droit public                                                   27
Histoire du droit et des institutions                          19
Histoire du droit                                              10
Droit compare                                                   3
Droit prive et sciences criminelles Droit prive et sciences criminelles     2
Science politique                                               2
Droit priveDroit prive                                          1
Science politique. Relations internationales                    1
Droit et sciences politiques                                    1
Droit prive. Droit musulman compare                             1
DroitDroit                                                      1
Droit maritime prive                                            1
Name: Discipline, dtype: int64
```

[36]:
```
# (7) Représentation du nombre de thèses encadrées en fonction de l'année de
 ↪soutenance
plt.figure(figsize = (16/2.54, 9/2.54))
plt.rc('font', family = 'Arial', size = 12)
plt.hist(PhD_v2_FP_Blanc["Year"], bins = 16)
plt.xlabel("Année")
plt.gca().spines['right'].set_visible(False)
plt.gca().spines['top'].set_visible(False)
plt.ylabel("Nombre de thèses")
```

[36]:
```
Text(0, 0.5, 'Nombre de thèses')
```



5 Résultats préliminaires

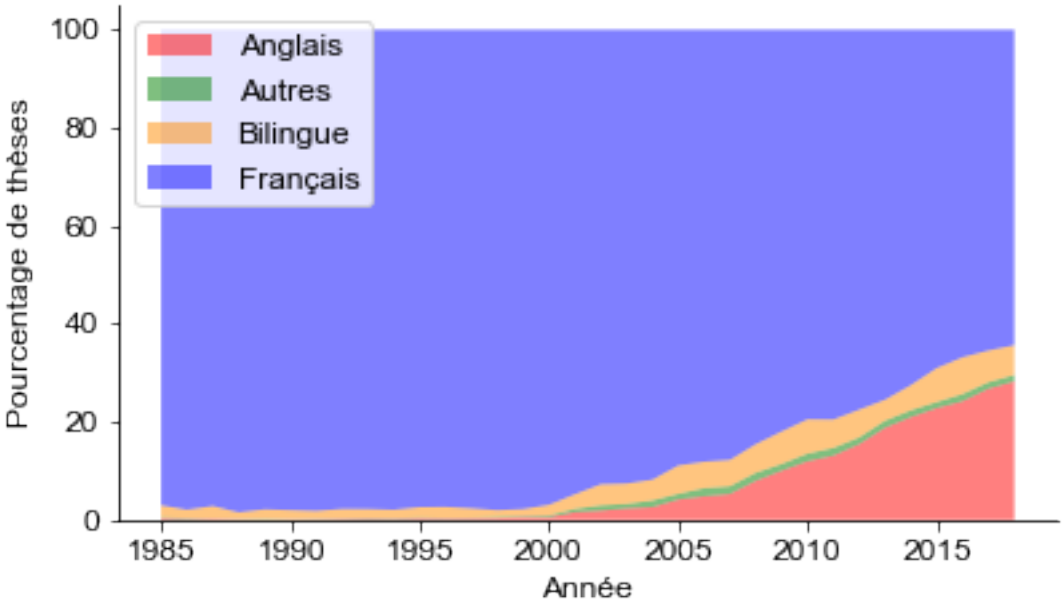
5.1 Modification de la variable “Langue de la these”

[37]:
```
# (1) Ré-import du jeu de données et suppression des valeurs manquantes dans␣
 ↪la langue de thèse
PhD_v2 = pd.read_csv(r"C:\Users\quent\Documents\DU Data Analyst 2022\UE n°1 -␣
 ↪Manipulation et prétraitement de données\Jeux de données\PhD_v2.csv",␣
 ↪low_memory = False)
PhD_v2.dropna(subset=["Langue de la these"], how = "all", inplace = True)

# (2) Calcul du nombre de valeurs identiques pour chaque langue
PhD_v2["Langue de la these"].value_counts(sort = True)

# (3) Sélection des langues autres que le français, l'anglais et les thèses␣
 ↪bilingues fr-ang
Langues = list(PhD_v2["Langue de la these"].value_counts(sort = True).keys())
Langues_autres = Langues[4:]

# (4) Conversion de ces langues en "Autres"
PhD_v2["Langue de la these"] = PhD_v2["Langue de la these"].
 ↪replace(Langues_autres,"Autres")

# (5) Conversion des langues "fr" et "en" en "Français" et "Anglais",␣
 ↪respectivement
PhD_v2["Langue de la these"] = PhD_v2["Langue de la these"].
 ↪replace(["fr"],"Français")
PhD_v2["Langue de la these"] = PhD_v2["Langue de la these"].
 ↪replace(["en"],"Anglais")

# (6) Conversion des langues "enfr" et "fren" en "Bilingue"
PhD_v2["Langue de la these"] = PhD_v2["Langue de la these"].replace(["enfr",␣
 ↪"fren"],"Bilingue")

# (7) Conversion du titre de la variable "Langue de la these" en "Langue"
PhD_v2["Langue"] = PhD_v2["Langue de la these"]
PhD_v2["Langue"].value_counts(sort = True)
```

[37]:
```
Français    334404
Anglais      30942
Bilingue     15369
Autres        3164
Name: Langue, dtype: int64
```

5.2 Évolution temporelle du choix de la langue de thèse

[38]:
```
# (1) Sélection des données sur la période 1984-2018
PhD_v2["Date de soutenance"] = pd.to_datetime(PhD_v2["Date de soutenance"],␣
 ↪format = "%d-%m-%y")
PhD_v2_1984_2018 = PhD_v2[np.logical_and(PhD_v2['Date de soutenance'] >=␣
 ↪"1984-01-01", PhD_v2['Date de soutenance'] < "2019-01-01")]

# (2) Calcul du nombre de thèses soutenues par langue et par année
PhD_v2_langue_annee = PhD_v2_1984_2018.pivot_table(values = 'Date de␣
 ↪soutenance', index = 'Langue', columns = 'Year', aggfunc = 'count')

# (3) Conversion en pourcentage
PhD_v2_prop_langue_annee = PhD_v2_langue_annee.copy(deep = True)
PhD_v2_prop_langue_annee.reset_index(inplace = True)

for i in list(PhD_v2_langue_annee.columns)[1:]:
    PhD_v2_prop_langue_annee[i] = (PhD_v2_prop_langue_annee[i] /␣
 ↪PhD_v2_prop_langue_annee[i].sum()) * 100

# (4) Représentation du pourcentage de thèses en fonction de l'année et de la␣
 ↪langue
Dict_PhD_v2_prop_langue_annee = {}
for i in list(PhD_v2_prop_langue_annee["Langue"]):
    Dict_PhD_v2_prop_langue_annee[i] =␣
 ↪list(PhD_v2_prop_langue_annee[PhD_v2_prop_langue_annee["Langue"] == i].
 ↪values[0][1:])

plt.rc('font', family = 'Arial', size = 12)
fig, ax = plt.subplots(figsize=(16/2.54, 9/2.54))
ax.stackplot(list(PhD_v2_prop_langue_annee.columns)[1:],␣
 ↪Dict_PhD_v2_prop_langue_annee.values(), labels =␣
 ↪list(PhD_v2_prop_langue_annee["Langue"]), alpha = 0.5, colors = ["red",␣
 ↪"green", "darkorange", "blue"])
ax.legend(loc = "upper left")
ax.set_xlabel("Année")
ax.set_ylabel("Pourcentage de thèses")
plt.gca().spines['right'].set_visible(False)
plt.gca().spines['top'].set_visible(False)
```


Anglais
Autres
Bilingue
Français
Pourcentage de thèses
100
80
60
40
20
0
1985
1990
1995
2000
2005
2010
2015
Année