

Projet

Isseïnie Calviac

Philémon Houdaille

Matthieu Rodet

23 avril 2021

Conditions portant sur les prénoms retenus

1. Sur la période allant de 1900 à 1945, le prénom a été attribué au moins 20 fois à des personnes de sexe féminin et/ou au moins 20 fois à des personnes de sexe masculin
2. Sur la période allant de 1946 à 2019, le prénom a été attribué au moins 20 fois à des personnes de sexe féminin et/ou au moins 20 fois à des personnes de sexe masculin
3. Pour une année de naissance donnée, le prénom a été attribué au moins 3 fois à des personnes de sexe féminin ou de sexe masculin

Les effectifs des prénoms ne remplissant pas les conditions 1 et 2 sont regroupés (pour chaque sexe et chaque année de naissance) dans un enregistrement dont le champ prénom (PREUSUEL) prend la valeur «*PRENOMS_RARES*». Les effectifs des prénoms remplissant la condition 2 mais pas la condition 3 sont regroupés (pour chaque sexe et chaque prénom) dans un enregistrement dont le champ année de naissance (ANNAIS) prend la valeur «XXXX».

```
[2]: # lecture du csv et affichage
prenoms = pandas.read_csv("Prénoms/nat2019.csv", header = 0, sep = ";")
prenoms
```

```
[2]:      sexe      preusuel annais  nombre
0         1  _PRENOMS_RARES   1900    1250
1         1  _PRENOMS_RARES   1901    1340
2         1  _PRENOMS_RARES   1902    1329
3         1  _PRENOMS_RARES   1903    1285
4         1  _PRENOMS_RARES   1904    1427
...      ...          ...      ...      ...
652051     2          ZYNEB   2016         6
652052     2          ZYNEB   2017         6
652053     2          ZYNEB   2018         5
652054     2          ZYNEB   2019         7
652055     2          ZYNEB   XXXX        19
```

```
[652056 rows x 4 columns]
```

Note : 1 = homme, 2 = femme

1 Un exemple de recherche de prénom selon l'année

```
[6]: def search(name, year):
    result = [0,0]
    for i in range(652056):
        if prenomns['preusuel'][i] == name:
            if prenomns['annais'][i] == year:
                result[prenomns['sexe'][i] - 1] = prenomns['nombre'][i]
    return result

research = 'KEVIN', '1986'

result = search(research[0], research[1])

print("Il y a eu " + str(result[0]) + " garçon(s) nommé(s) " + research[0] + "
→en " + research[1] + ".")
print("Il y a eu " + str(result[1]) + " fille(s) nommée(s) " + research[0] + "
→en " + research[1] + ".")
```

Il y a eu 4875 garçon(s) nommé(s) KEVIN en 1986.

Il y a eu 3 fille(s) nommée(s) KEVIN en 1986.

2 Une approche simpliste

2.1 Affichage de la courbe d'évolution des prénoms au cours du temps

```
[31]: def plotNumberByYear(dataBase, names):
    X = np.array([x for x in range(1900,2020)])
    YF = np.zeros((121))
    YM = np.zeros((121))
    for name in names:
        isName = dataBase['preusuel']==name
        allName = dataBase[isName]

        allName = allName.replace('XXXX', '2050')

        allName['annais'] = allName['annais'].astype(int)

        X_M = [x for x in allName[allName.sexe==1].annais.array]
        Y_M = [y for y in allName[allName.sexe==1].nombre.array]
        X_F = [x for x in allName[allName.sexe==2].annais.array]
        Y_F = [y for y in allName[allName.sexe==2].nombre.array]

        for i in range(1900,2020):
            if not i in X_M:
                Y_M.insert(i-1900, 0)
            if not i in X_F:
                Y_F.insert(i-1900, 0)
        X = [i for i in range(1900, 2020)]
        if not 2050 in X_M :
            Y_M.append(0)
        if not 2050 in X_F :
            Y_F.append(0)

        YF += Y_F
        YM += Y_M

    males, = plt.plot(X, YM[:-1])
    females, = plt.plot(X, YF[:-1])
    plt.legend([males, females], ['hommes', 'femmes'])
    if len(names) == 1 :
        plt.title("Nombre de bébés nommés " + names[0] + " chaque année entre_
→1900 et 2020.")
        if YM[-1] != 0 :
            print("Le prénom " + names[0] + " a été donné " + str(YM[-1]) + "_
→fois à des hommes pour des années indéfinies")
```

```

        if YF[-1] != 0 :
            print("Le prénom " + names[0] + " a été donné " + str(YF[-1]) + "
→fois à des femmes pour des années indéfinies")

        elif len(names) > 10 :
            plt.title("Nombre de bébés nommés " + names[0] + ",... chaque année
→entre 1900 et 2020.")
            if YM[-1] != 0 :
                print("Les prénoms " + names[0] + ",... ont été donnés " +
→str(YM[-1]) + " fois à des hommes pour des années indéfinies")
            if YF[-1] != 0 :
                print("Les prénoms " + names[0] + ",... ont été donnés " +
→str(YF[-1]) + " fois à des femmes pour des années indéfinies")
            else :
                plt.title("Nombre de bébés nommés " + names[0] + ",... chaque année
→entre 1900 et 2020.")
                if YM[-1] != 0 :
                    print("Les prénoms ", end = '')
                    for name in names :
                        print(name + " ", end = '')
                    print("ont été donnés " + str(YM[-1]) + " fois à des hommes pour des
→années indéfinies")
                if YF[-1] != 0 :
                    print("Les prénoms ", end = '')
                    for name in names :
                        print(name + " ", end = '')
                    print("ont été donnés " + str(YF[-1]) + " fois à des femmes pour des
→années indéfinies")

```

2.2 Quelques ensembles de prénoms

```

[33]: interesting = ["ZIDANE"], ["MICHAEL", "MICKAEL"], ["AURELIEN", "AURÉLIEN"],
→["DAVID"]
zoomers = ["MATTHIEU", "MATHIEU", "MATHIEUX"], ["MANON"], ["ENZO"],
→["QUENTIN"]
grandparents = ["JEAN"], ["JACQUES"]
fashionAgain = ["LOUIS", ["LOUISE", "LOUISON"], ["ALICE"], ["JULES"]]
lonely = ["ISSEINIE"], ["PHILEMON", "PHILÉMON"], ["LUDMILA"],
→["_PRENOMS_RARES"]
dictateurs = ["ADOLF"], ["JOSEPH"]
boomers = ["JEAN-MARIE"], ["JEAN-JACQUES"], ["JEAN-FRANÇOIS"]

databases = [interesting, zoomers, grandparents, fashionAgain, lonely,
→dictateurs, boomers]

```

```

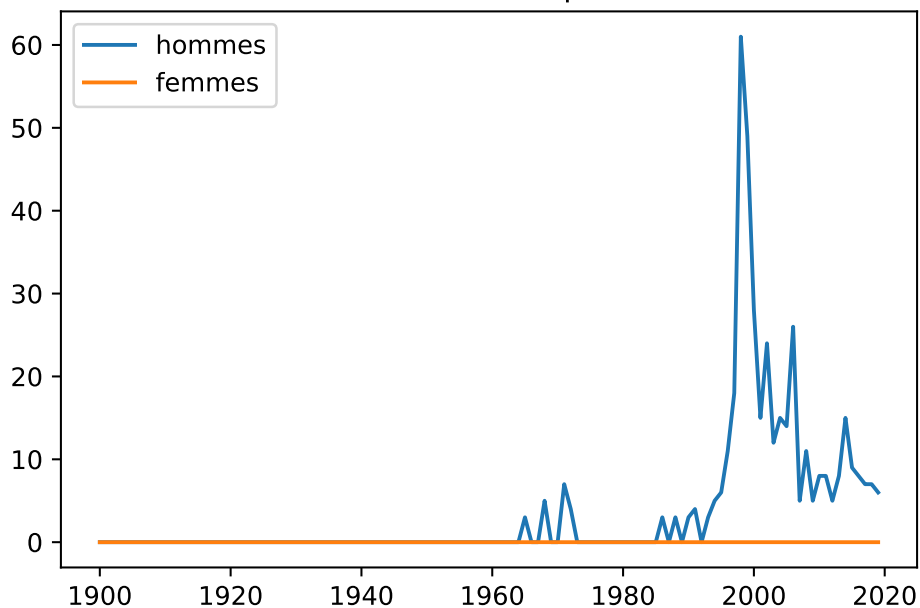
databasesNames = ["intéressants", "de zoomers", "de la génération des_
↳grands-parents", "à nouveau à la mode", "des gens uniques", "des méchants",_
↳"des boomers"]
for i in range(len(databases)) :
    print("Traitement de la base de donnée des prénoms " + databasesNames[i] + ":
↳\n\n")
    for prenom in databases[i] :
        plotNumberByYear(prenoms, prenom)
        plt.show()
        print("\n")
    print("\n\n")

```

Traitement de la base de donnée des prénoms intéressants:

Le prénom ZIDANE a été donné 31.0 fois à des hommes pour des années indéfinies

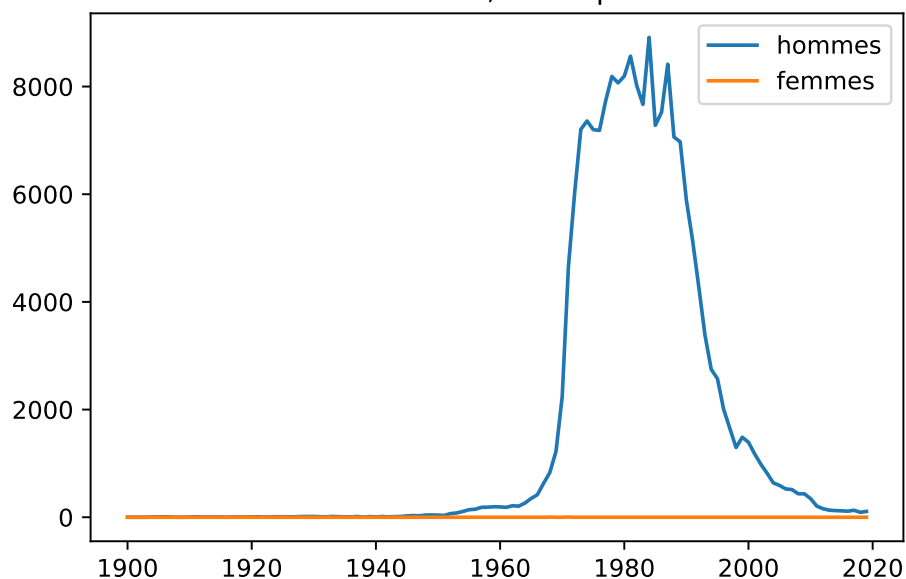
Nombre de bébés nommés ZIDANE chaque année entre 1900 et 2020.



Les prénoms MICHAEL MICKAEL ont été donnés 36.0 fois à des hommes pour des années indéfinies

Les prénoms MICHAEL MICKAEL ont été donnés 27.0 fois à des femmes pour des années indéfinies

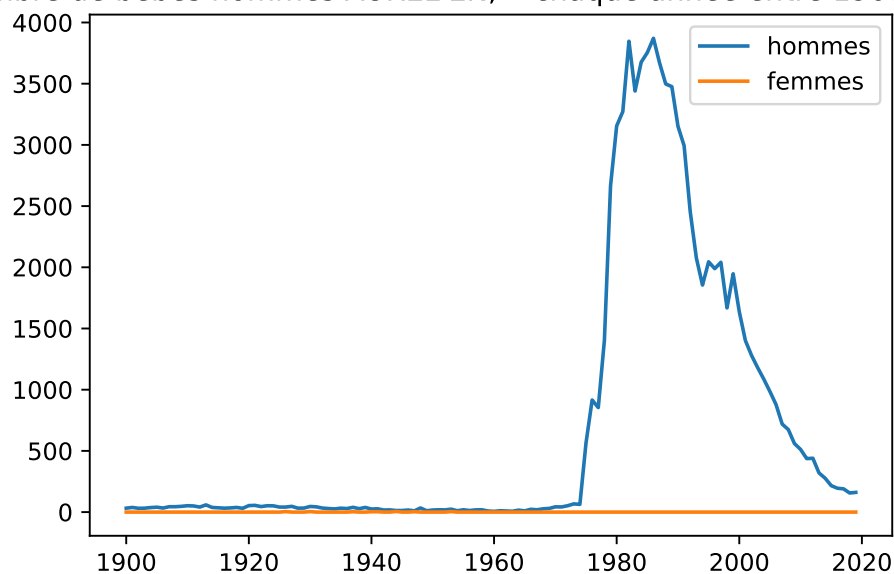
Nombre de bébés nommés MICHAEL,... chaque année entre 1900 et 2020.



Les prénoms AURELIEN AURÉLIEN ont été donnés 17.0 fois à des hommes pour des années indéfinies

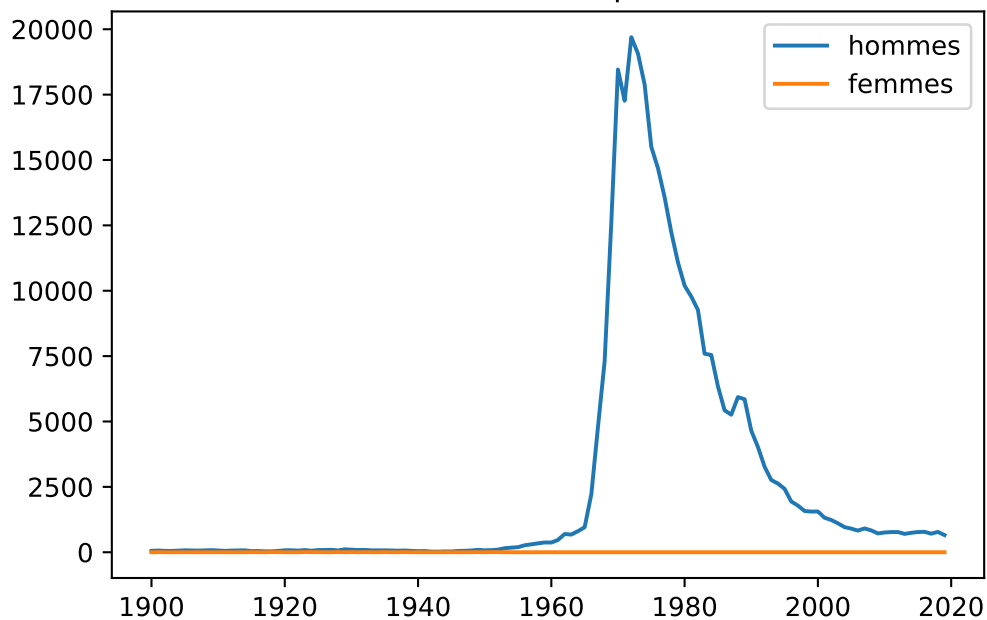
Les prénoms AURELIEN AURÉLIEN ont été donnés 52.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés AURELIEN,... chaque année entre 1900 et 2020.



Le prénom DAVID a été donné 64.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés DAVID chaque année entre 1900 et 2020.

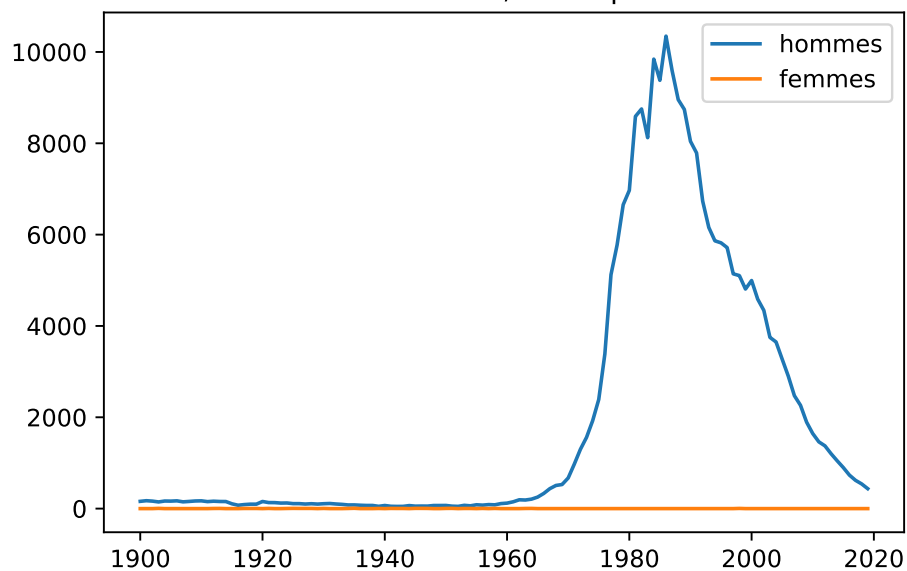


Traitement de la base de donnée des prénoms de zoomers:

Les prénoms MATTHIEU MATHIEU MATHIEUX ont été donnés 59.0 fois à des hommes pour des années indéfinies

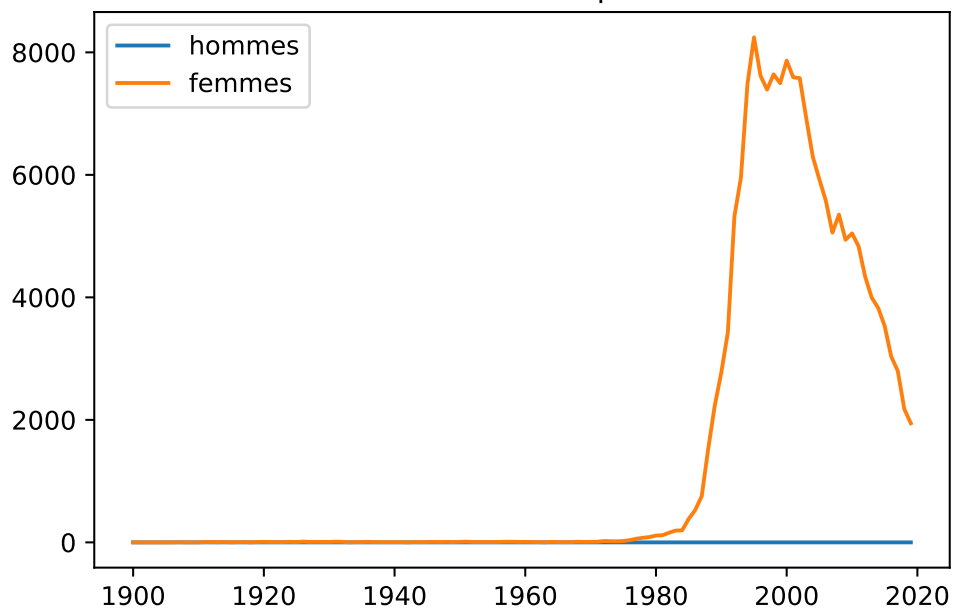
Les prénoms MATTHIEU MATHIEU MATHIEUX ont été donnés 62.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés MATTHIEU,... chaque année entre 1900 et 2020.



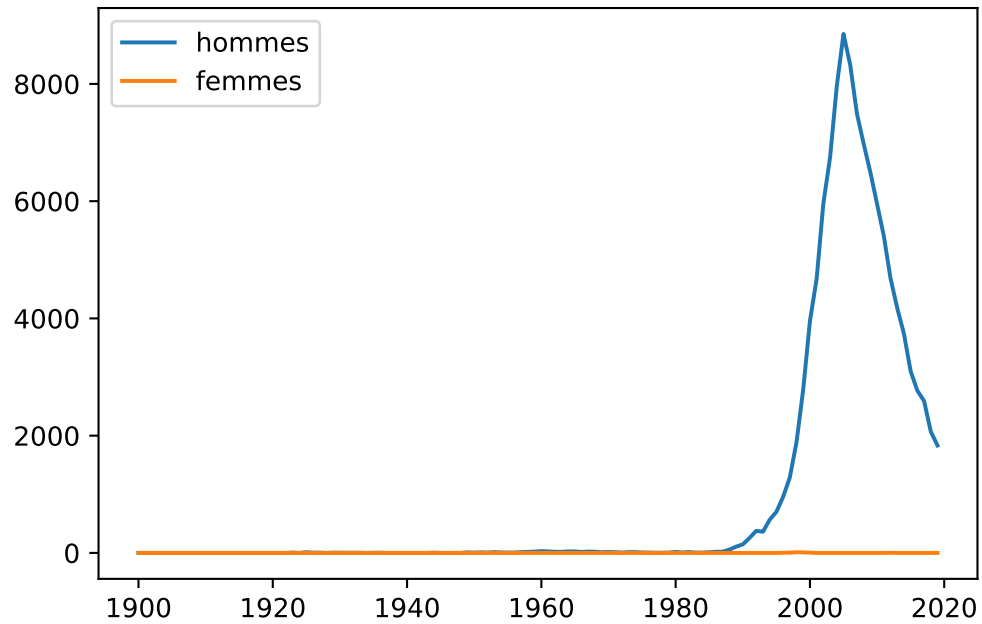
Le prénom MANON a été donné 20.0 fois à des hommes pour des années indéfinies
 Le prénom MANON a été donné 19.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés MANON chaque année entre 1900 et 2020.



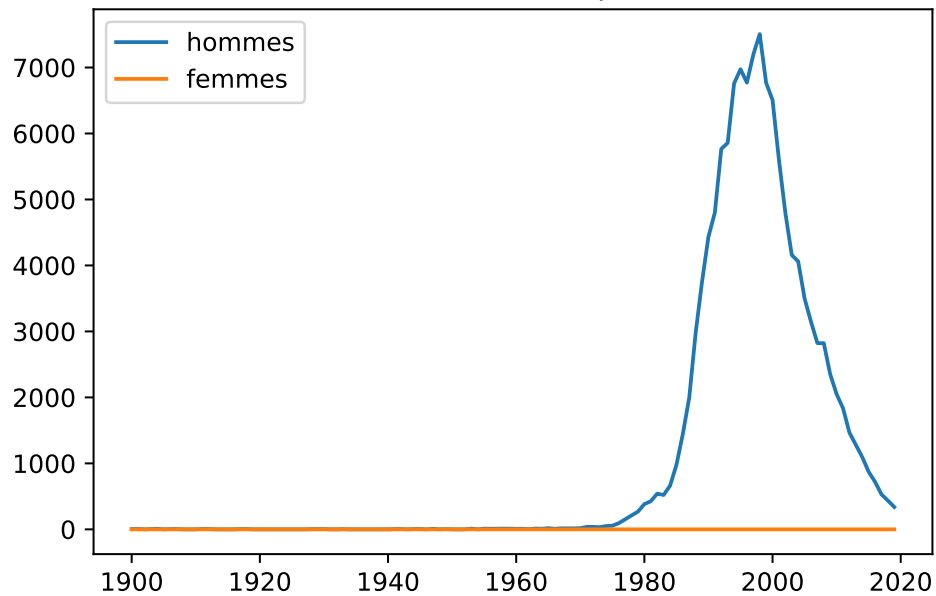
Le prénom ENZO a été donné 20.0 fois à des hommes pour des années indéfinies
Le prénom ENZO a été donné 27.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés ENZO chaque année entre 1900 et 2020.



Le prénom QUENTIN a été donné 44.0 fois à des hommes pour des années indéfinies
Le prénom QUENTIN a été donné 21.0 fois à des femmes pour des années indéfinies

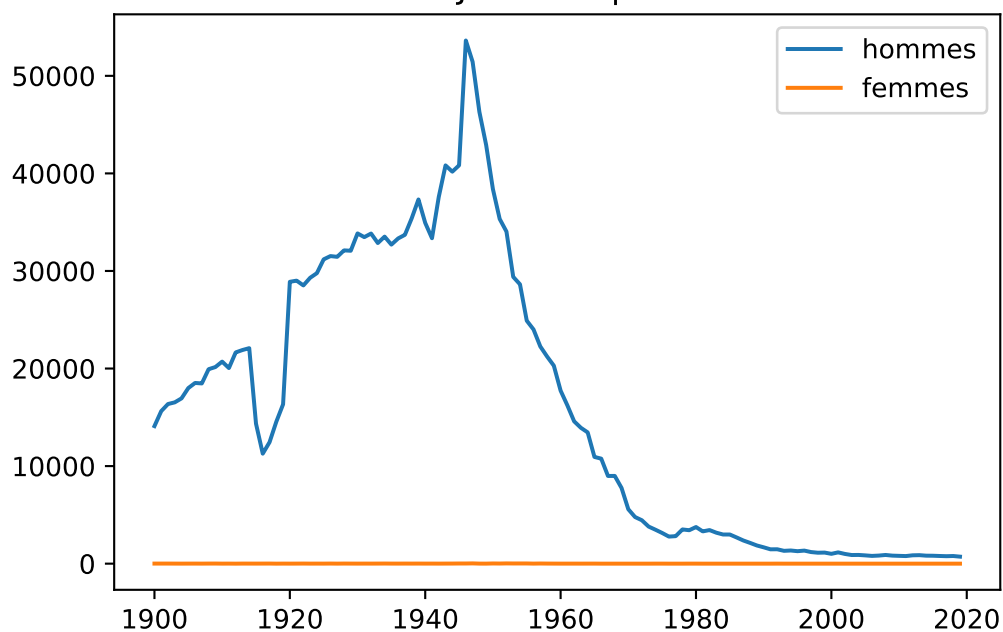
Nombre de bébés nommés QUENTIN chaque année entre 1900 et 2020.



Traitement de la base de donnée des prénoms de la génération des grands-parents:

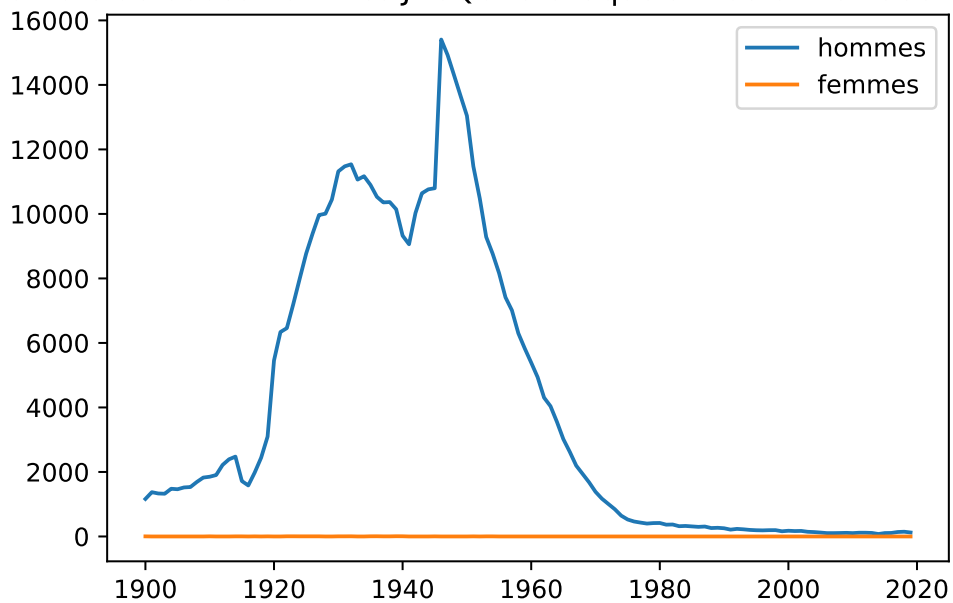
Le prénom JEAN a été donné 34.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés JEAN chaque année entre 1900 et 2020.



Le prénom JACQUES a été donné 43.0 fois à des femmes pour des années indéfinies

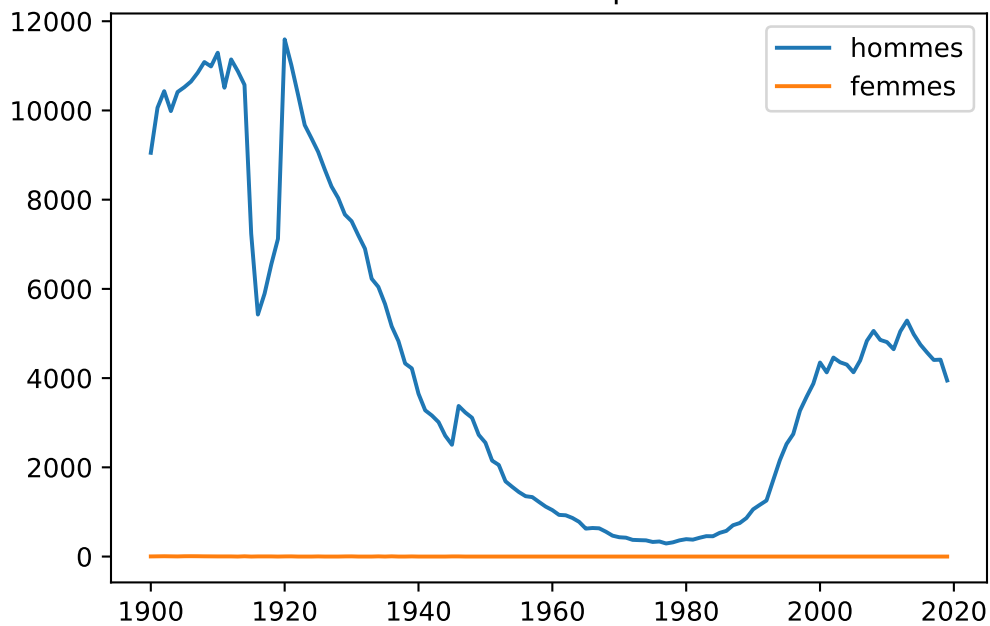
Nombre de bébés nommés JACQUES chaque année entre 1900 et 2020.



Traitement de la base de donnée des prénoms à nouveau à la mode:

Le prénom LOUIS a été donné 61.0 fois à des femmes pour des années indéfinies

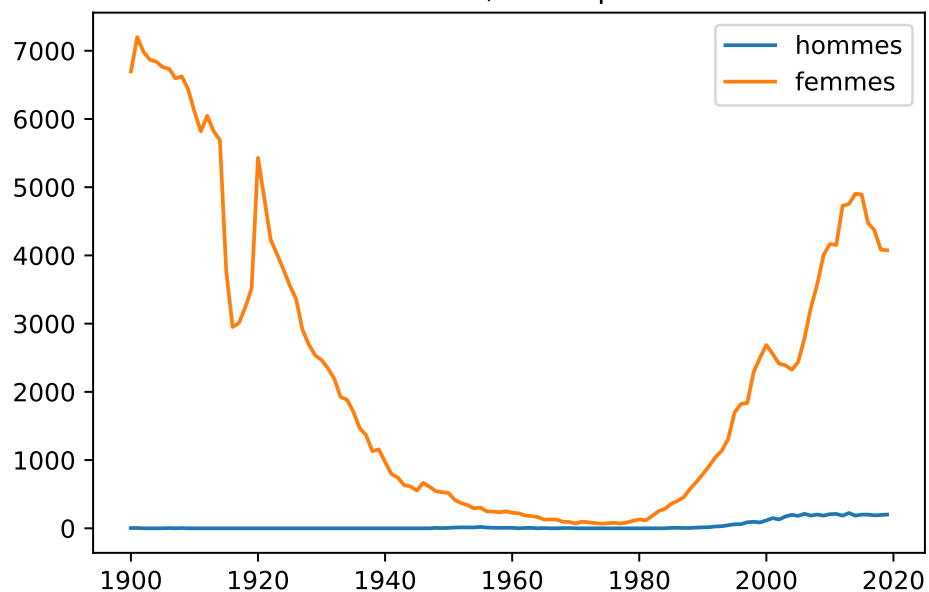
Nombre de bébés nommés LOUIS chaque année entre 1900 et 2020.



Les prénoms LOUISE LOUISON ont été donnés 50.0 fois à des hommes pour des années indéfinies

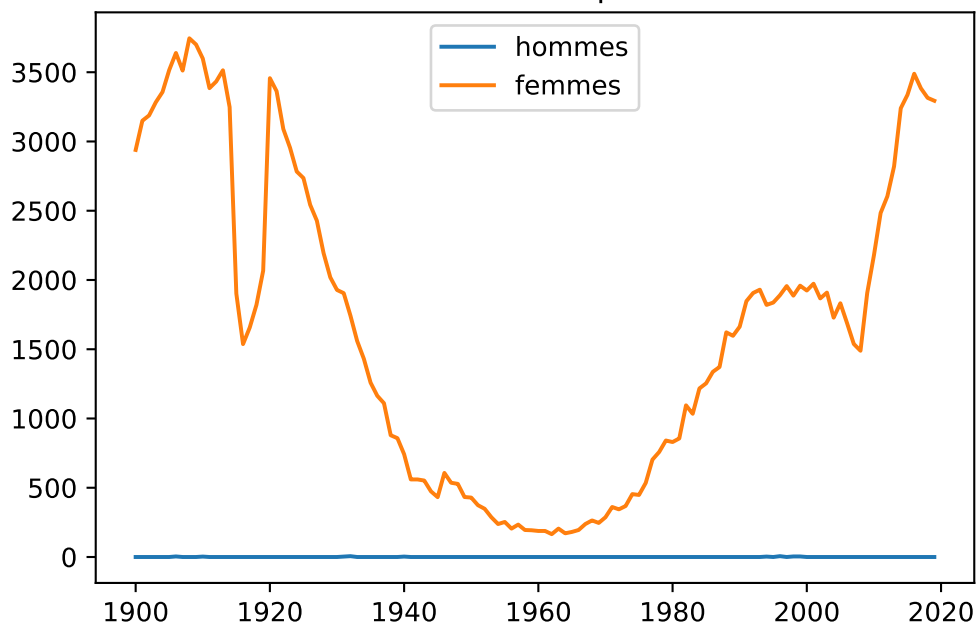
Les prénoms LOUISE LOUISON ont été donnés 52.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés LOUISE,... chaque année entre 1900 et 2020.



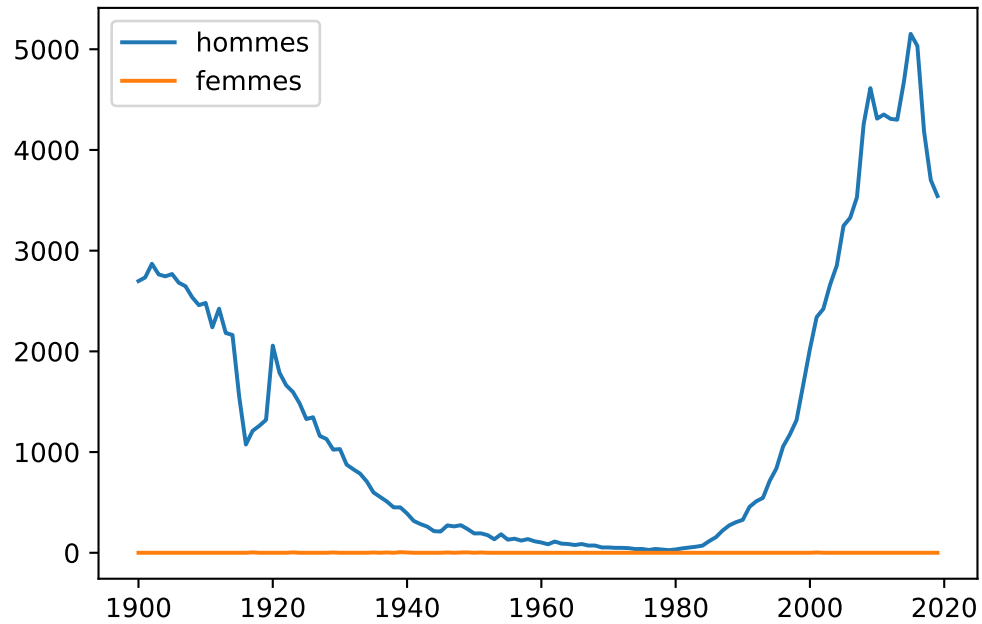
Le prénom ALICE a été donné 42.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés ALICE chaque année entre 1900 et 2020.



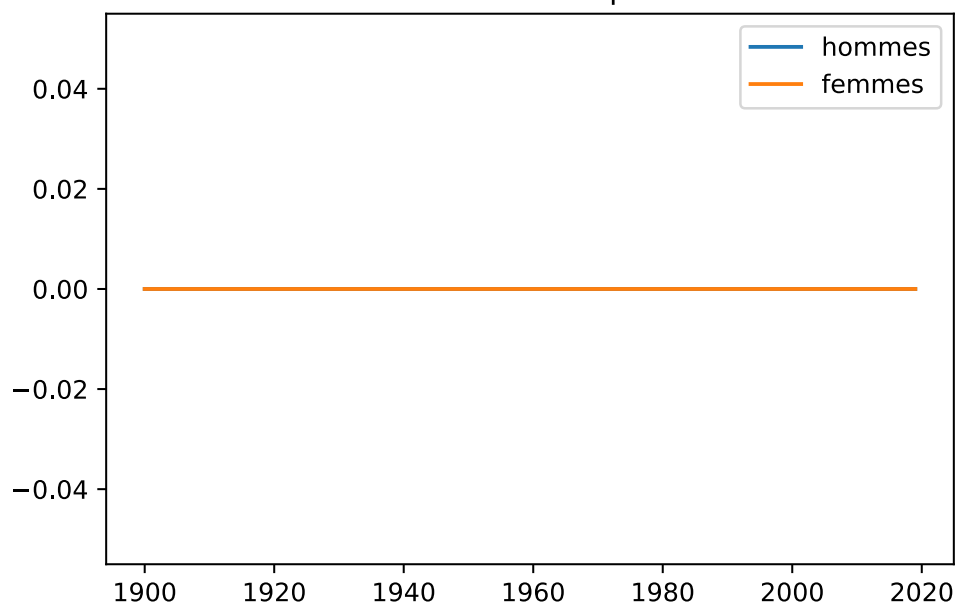
Le prénom JULES a été donné 89.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés JULES chaque année entre 1900 et 2020.



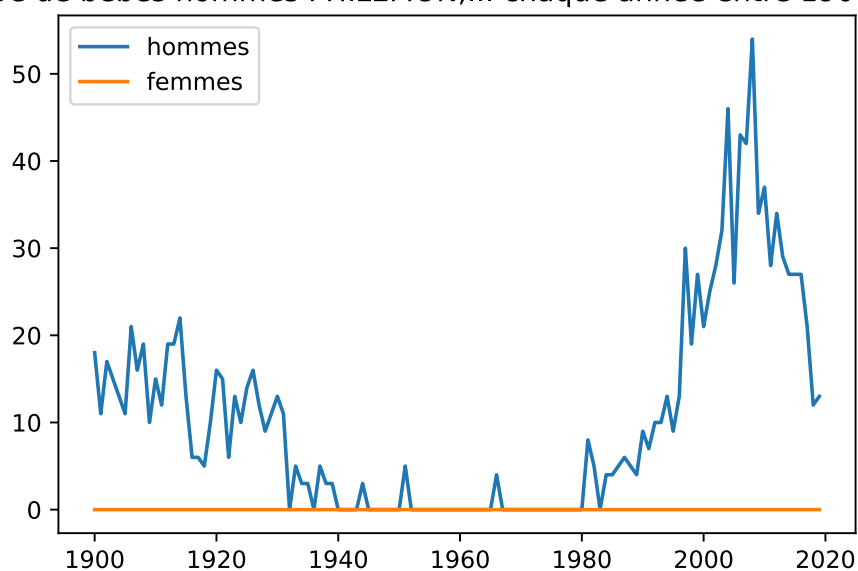
Traitement de la base de donnée des prénoms des gens uniques:

Nombre de bébés nommés ISSEINIE chaque année entre 1900 et 2020.



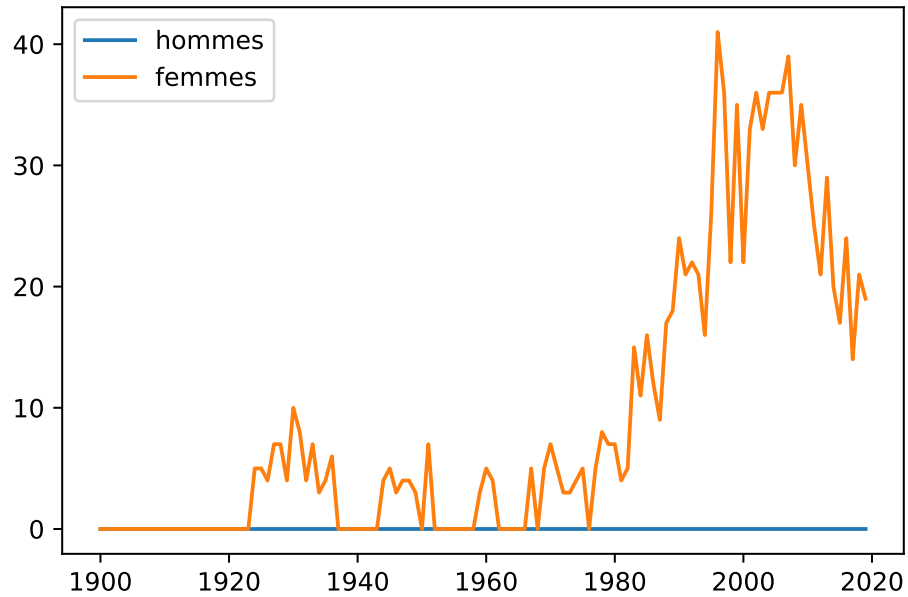
Les prénoms PHILEMON PHILÉMON ont été donnés 57.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés PHILEMON,... chaque année entre 1900 et 2020.

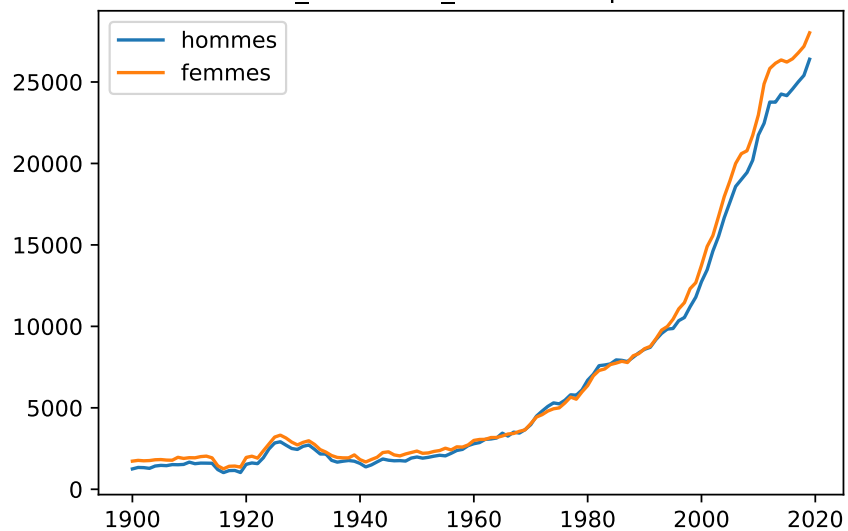


Le prénom LUDMILA a été donné 34.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés LUDMILA chaque année entre 1900 et 2020.



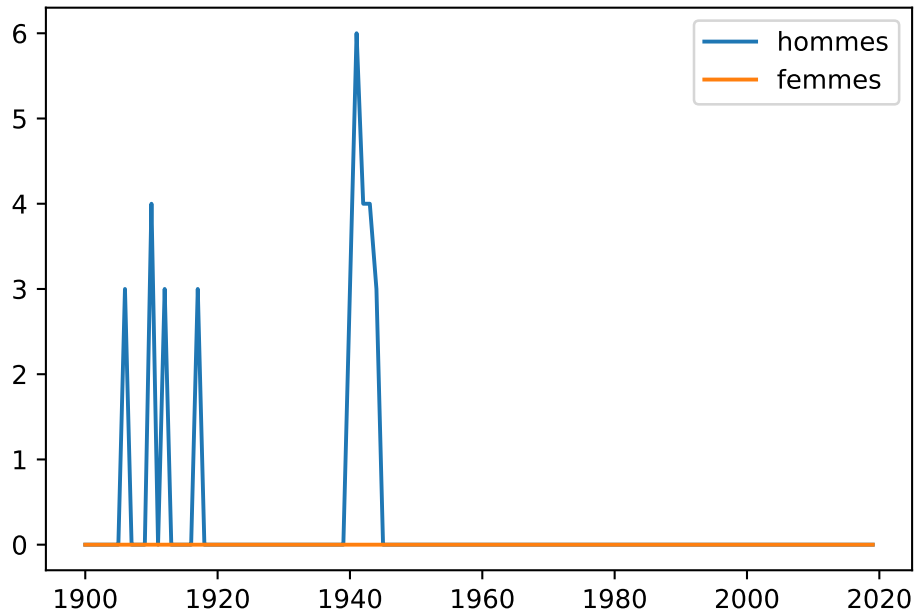
Nombre de bébés nommés _PRENOMS_RARES chaque année entre 1900 et 2020.



Traitement de la base de donnée des prénoms des méchants:

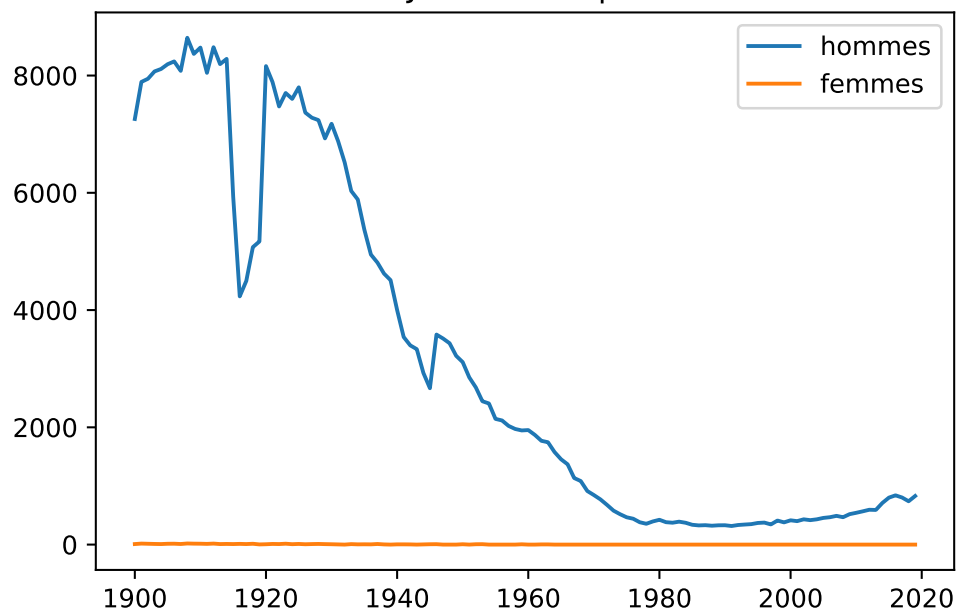
Le prénom ADOLF a été donné 23.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés ADOLF chaque année entre 1900 et 2020.



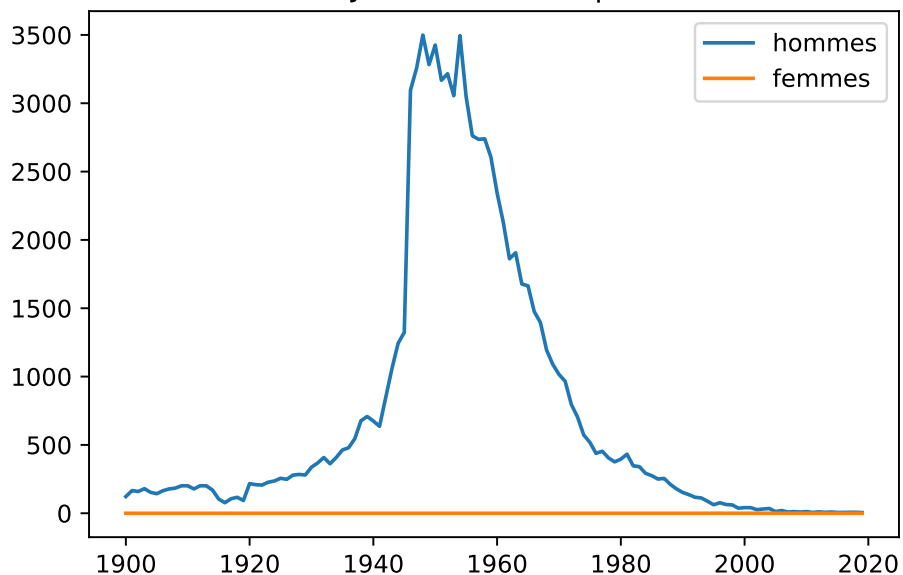
Le prénom JOSEPH a été donné 36.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés JOSEPH chaque année entre 1900 et 2020.



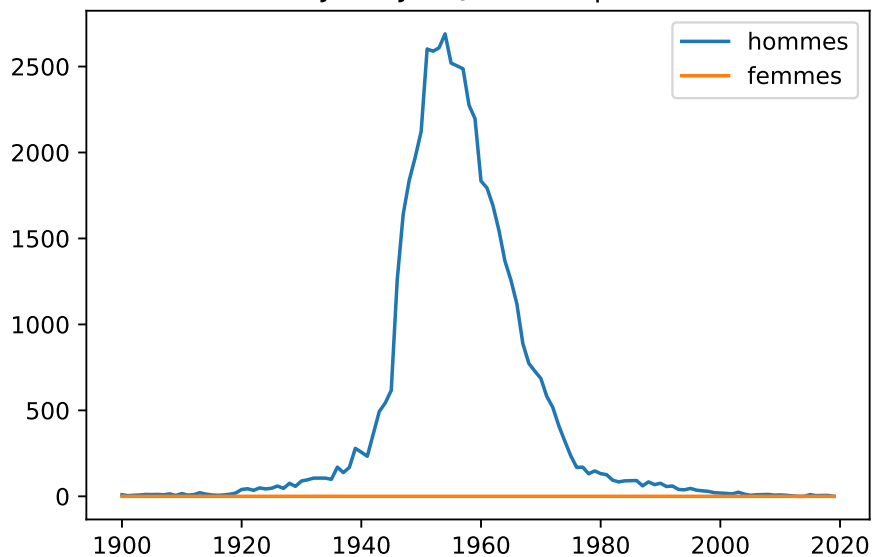
Traitement de la base de donnée des prénoms des boomers:

Nombre de bébés nommés JEAN-MARIE chaque année entre 1900 et 2020.

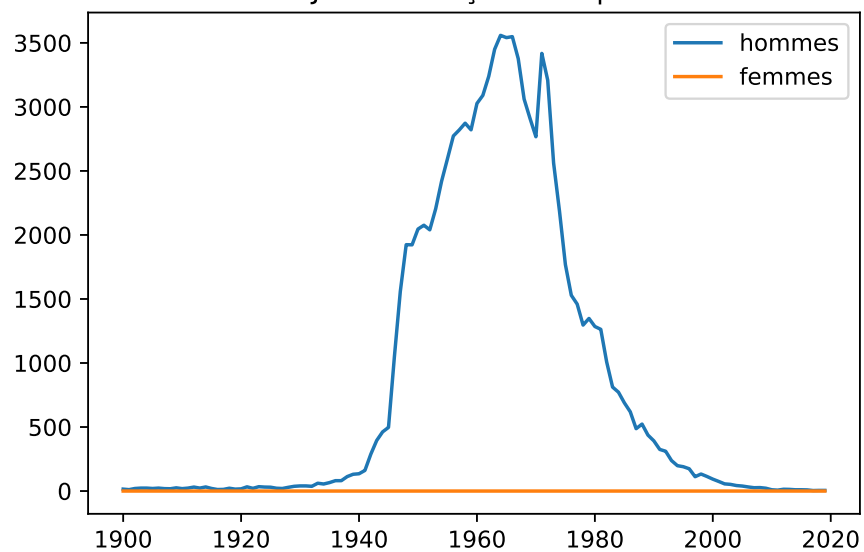


Le prénom JEAN-JACQUES a été donné 3.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés JEAN-JACQUES chaque année entre 1900 et 2020.



Nombre de bébés nommés JEAN-FRANÇOIS chaque année entre 1900 et 2020.



3 Une approche plus judicieuse

3.1 Affichage de la part des prénoms au cours du temps

```
[26]: def plotPartByYear(dataBase, name):

    isName = dataBase['preusuel']==name
    allName = dataBase[isName]

    allName = allName.replace('XXXX', '2050')

    allName['annais'] = allName['annais'].astype(int)

    X_M = allName[allName.sexe==1].annais.array
    Y_M = allName[allName.sexe==1].nombre.array
    X_F = allName[allName.sexe==2].annais.array
    Y_F = allName[allName.sexe==2].nombre.array

    for i in range(1900,2020):
        if not i in X_M:
            Y_M = np.insert(Y_M, i-1900, 0)
        if not i in X_F:
            Y_F = np.insert(Y_F, i-1900, 0)
    X = [i for i in range(1900, 2020)]
    if not 2050 in X_M :
        Y_M = np.append(Y_M, 0)
    if not 2050 in X_F :
        Y_F = np.append(Y_F, 0)

    Y_M = Y_M.astype(float)
    Y_F = Y_F.astype(float)
    for year in X :
        totalYear = dataBase[dataBase.annais==str(year)].nombre.array.sum()
        Y_M[year-1900] /= totalYear
        Y_F[year-1900] /= totalYear

    males, = plt.plot(X, Y_M[:-1], c = 'blue')
    females, = plt.plot(X, Y_F[:-1], c = 'red')
    plt.legend([males, females], ['hommes', 'femmes'])
    plt.title("Nombre de bébés nommés " + name + " chaque année entre 1900 et 2020.")
    if Y_M[-1] != 0 :
        print("Le prénom " + name + " a été donné " + str(Y_M[-1]) + " fois à des hommes pour des années indéfinies")
    if Y_F[-1] != 0 :
```

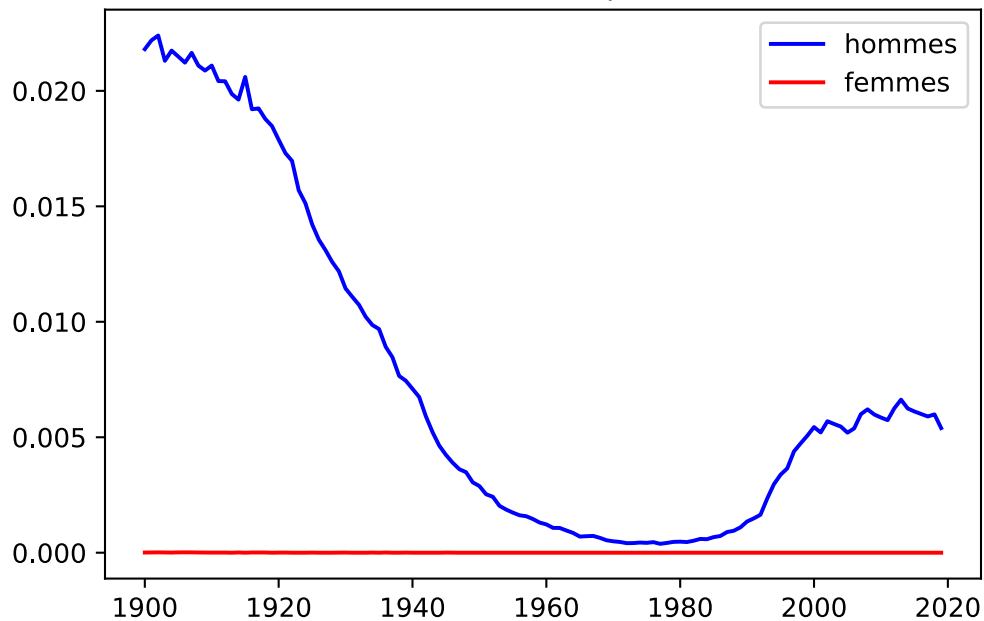
```

print("Le prénom " + name + " a été donné " + str(Y_F[-1]) + " fois à
→des femmes pour des années indéfinies")
plotPartByYear(prenoms, "LOUIS")

```

Le prénom LOUIS a été donné 61.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés LOUIS chaque année entre 1900 et 2020.



```

[8]: def plotTotalByYear(dataBase):

    X = np.array([x for x in range(1900, 2020)])
    Y_M = np.array([dataBase[dataBase.annais==str(year)].nombre.array.sum() for
→year in X])

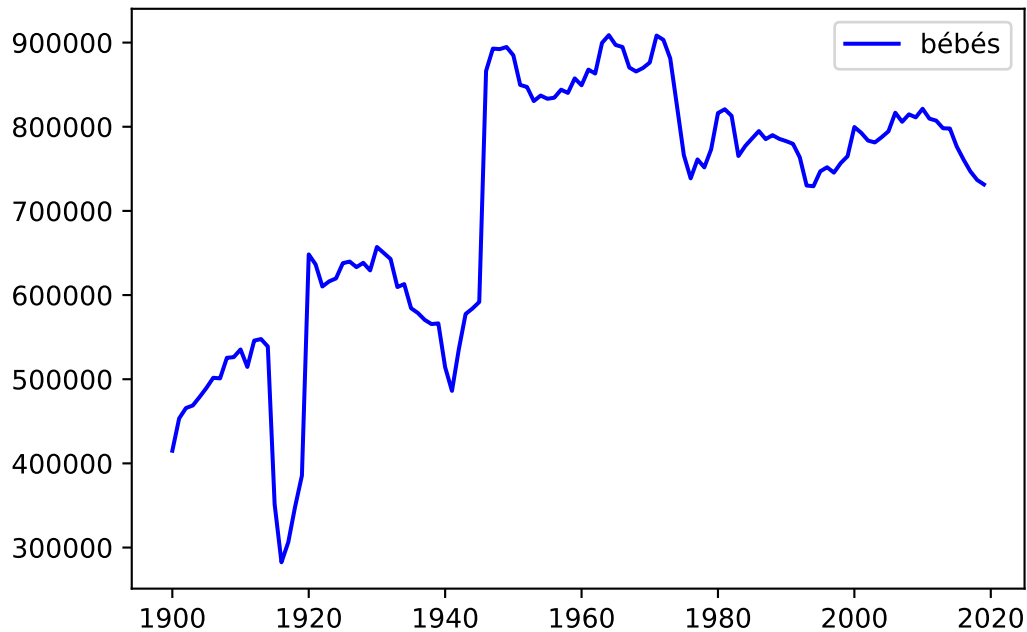
    males, = plt.plot(X, Y_M, c = 'blue')
    plt.legend([males], ['bébés'])
    plt.title("Nombre de bébés chaque année entre 1900 et 2020.")
    print("Il y a eu " + str(dataBase[dataBase.annais=="XXXX"].nombre.array.
→sum()) + " bébés lors d'années indéfinies")

plotTotalByYear(prenoms)

```

Il y a eu 821477 bébés lors d'années indéfinies

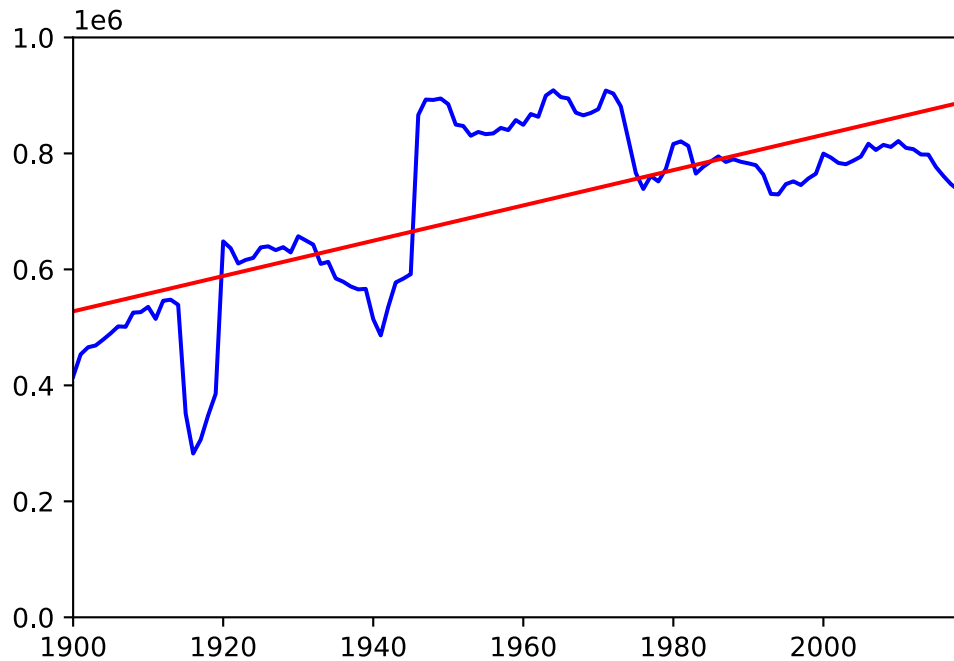
Nombre de bébés chaque année entre 1900 et 2020.



```
[33]: X = np.array([[x] for x in range(1900,2020)])
X = np.concatenate([np.ones((X.shape[0], 1)), X], axis=1)
XTX = np.dot(X.T, X)
Y = np.array([prenoms[prenoms.annais==str(x)].nombre.array.sum() for x in
→range(1900,2020)])
XTY = np.dot(X.T, Y)
A = np.linalg.solve(XTX, XTY)
quadError = np.sqrt(((X.dot(A)-Y)**2).mean())

plt.plot(X,Y,c='blue')
plt.plot(X,A[0] + A[1]*X,c='red')
plt.axis([1900,2019,0,1000000])
```

```
[33]: (1900.0, 2019.0, 0.0, 1000000.0)
```



```
[56]: def plotPartByYearList(dataBase, names):
    X = np.array([x for x in range(1900,2020)])
    YF = np.zeros((121))
    YM = np.zeros((121))
    for name in names:
        isName = dataBase['preusuel']==name
        allName = dataBase[isName]

        allName = allName.replace('XXXX', '2050')

        allName['annais'] = allName['annais'].astype(int)

        X_M = allName[allName.sexe==1].annais.array
        Y_M = allName[allName.sexe==1].nombre.array
        X_F = allName[allName.sexe==2].annais.array
        Y_F = allName[allName.sexe==2].nombre.array

        for i in range(1900,2020):
            if not i in X_M:
                Y_M = np.insert(Y_M, i-1900, 0)
            if not i in X_F:
                Y_F = np.insert(Y_F, i-1900, 0)
        if not 2050 in X_M :
```



```

        Y_M = np.append(Y_M, 0)
    if not 2050 in X_F :
        Y_F = np.append(Y_F, 0)

    Y_M = Y_M.astype(float)
    Y_F = Y_F.astype(float)
    for year in X :
        totalYear = DataBase[DataBase.annais==str(year)].nombre.array.sum()
        Y_M[year-1900] /= totalYear
        Y_F[year-1900] /= totalYear

    YF += Y_F
    YM += Y_M

    males, = plt.plot(X, YM[:-1], c = 'blue')
    females, = plt.plot(X, YF[:-1], c = 'red')
    plt.legend([males, females], ['hommes', 'femmes'])
    if len(names) == 1 :
        plt.title("Nombre de bébés nommés " + names[0] + " chaque année entre_
→1900 et 2020.")
        if YM[-1] != 0 :
            print("Le prénom " + names[0] + " a été donné " + str(YM[-1]) + "
→fois à des hommes pour des années indéfinies")
        if YF[-1] != 0 :
            print("Le prénom " + names[0] + " a été donné " + str(YF[-1]) + "
→fois à des femmes pour des années indéfinies")

    elif len(names) > 10 :
        plt.title("Nombre de bébés nommés " + names[0] + ",... chaque année_
→entre 1900 et 2020.")
        if YM[-1] != 0 :
            print("Les prénoms " + names[0] + ",... ont été donnés " +
→str(YM[-1]) + " fois à des hommes pour des années indéfinies")
        if YF[-1] != 0 :
            print("Les prénoms " + names[0] + ",... ont été donnés " +
→str(YF[-1]) + " fois à des femmes pour des années indéfinies")
        else :
            plt.title("Nombre de bébés nommés " + names[0] + ",... chaque année_
→entre 1900 et 2020.")
            if YM[-1] != 0 :
                print("Les prénoms ", end='')
                for name in names :
                    print(name + " ", end = '')
                print("ont été donnés " + str(YM[-1]) + " fois à des hommes pour des_
→années indéfinies")
            if YF[-1] != 0 :

```

```

print("Les prénoms ", end = '')
for name in names :
    print(name + " ", end = '')
print("ont été donnés " + str(YF[-1]) + " fois à des femmes pour des_
→années indéfinies")

```

3.2 Les mêmes exemples

```

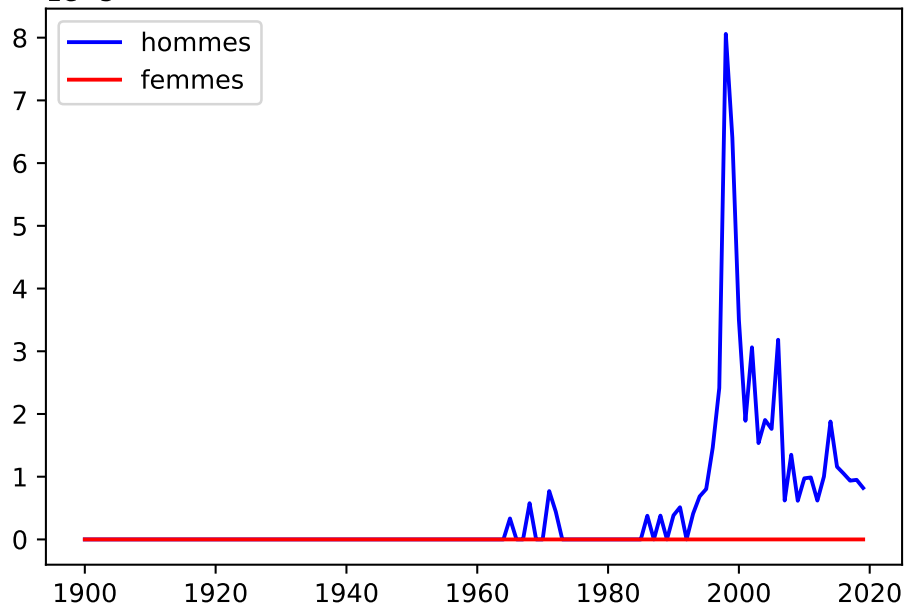
[57]: for i in range(len(databases)) :
    print("Traitement de la base de donnée des prénoms " + databasesNames[i] + ":
    →\n\n")
    for prenom in databases[i] :
        plotPartByYearList(prenoms, prenom)
        plt.show()
        print("\n")
    print("\n\n")

```

Traitement de la base de donnée des prénoms intéressants:

Le prénom ZIDANE a été donné 31.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés ZIDANE chaque année entre 1900 et 2020.

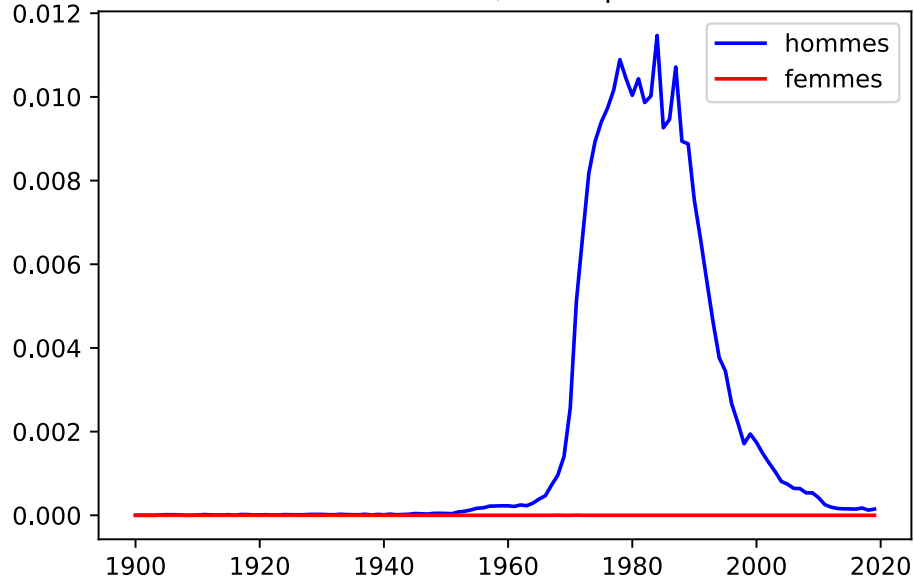


Les prénoms MICHAEL MICKAEL ont été donnés 36.0 fois à des hommes pour des

années indéfinies

Les prénoms MICHAEL MICKAEL ont été donnés 27.0 fois à des femmes pour des années indéfinies

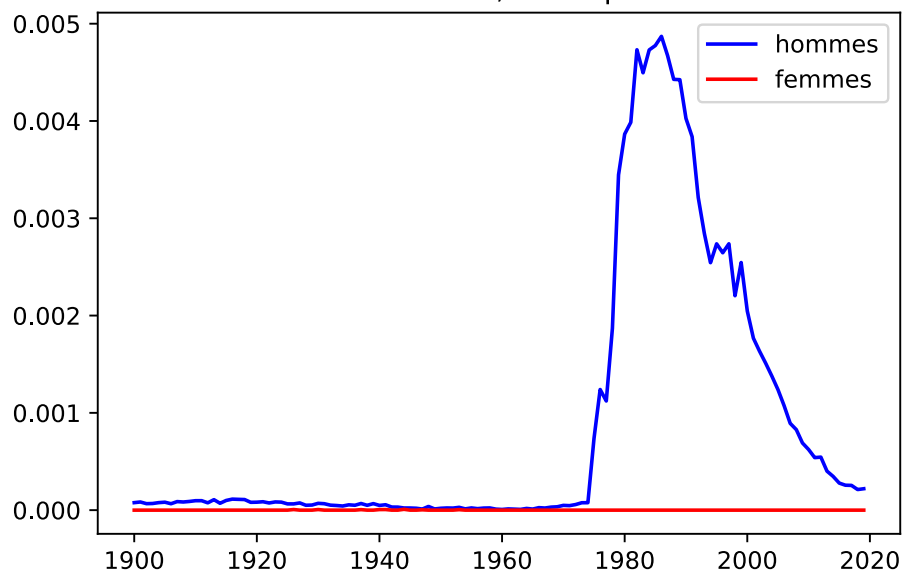
Nombre de bébés nommés MICHAEL,... chaque année entre 1900 et 2020.



Les prénoms AURELIEN AURÉLIEN ont été donnés 17.0 fois à des hommes pour des années indéfinies

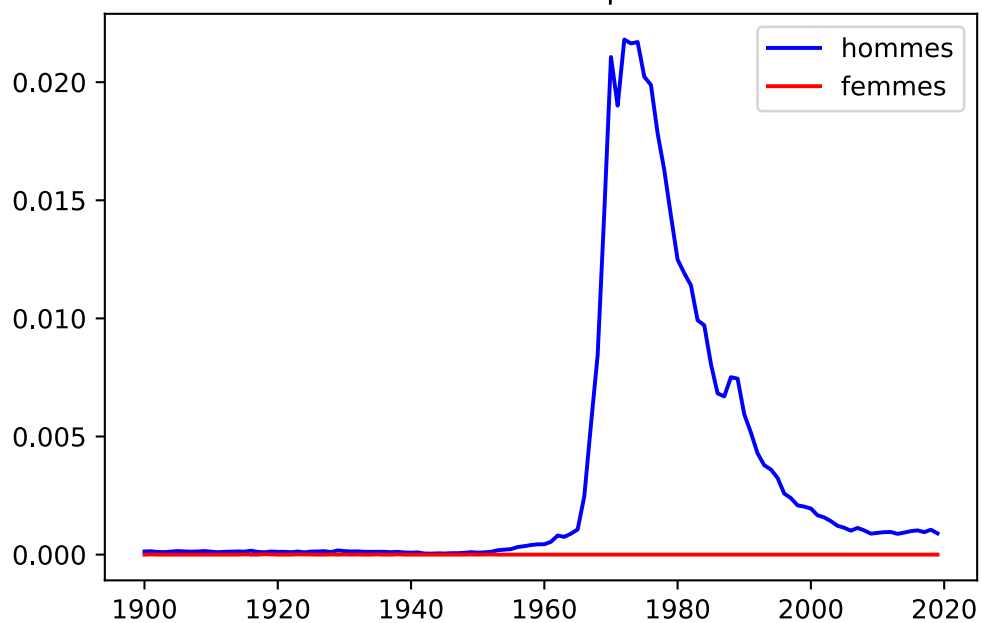
Les prénoms AURELIEN AURÉLIEN ont été donnés 52.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés AURELIEN,... chaque année entre 1900 et 2020.



Le prénom DAVID a été donné 64.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés DAVID chaque année entre 1900 et 2020.

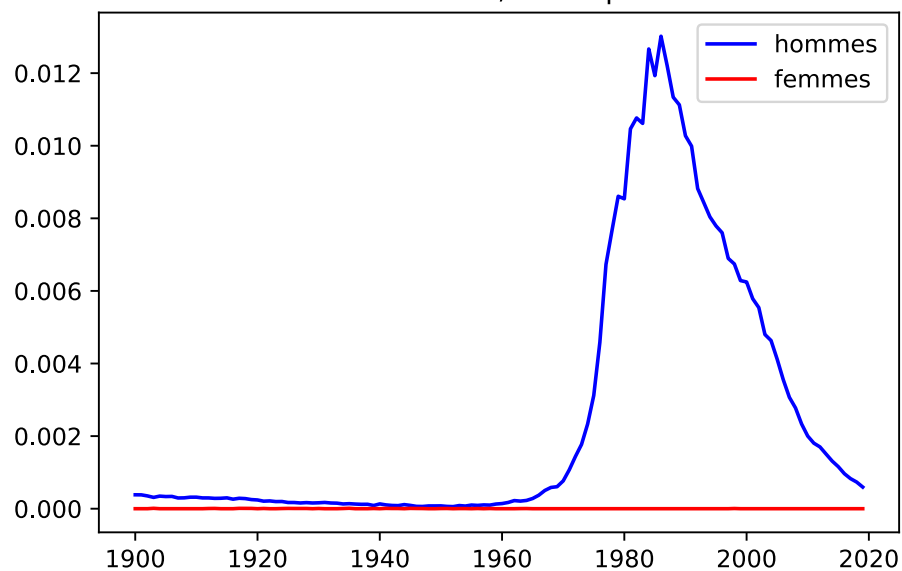


Traitement de la base de donnée des prénoms de zoomers:

Les prénoms MATTHIEU MATHIEU MATHIEUX ont été donnés 59.0 fois à des hommes pour des années indéfinies

Les prénoms MATTHIEU MATHIEU MATHIEUX ont été donnés 62.0 fois à des femmes pour des années indéfinies

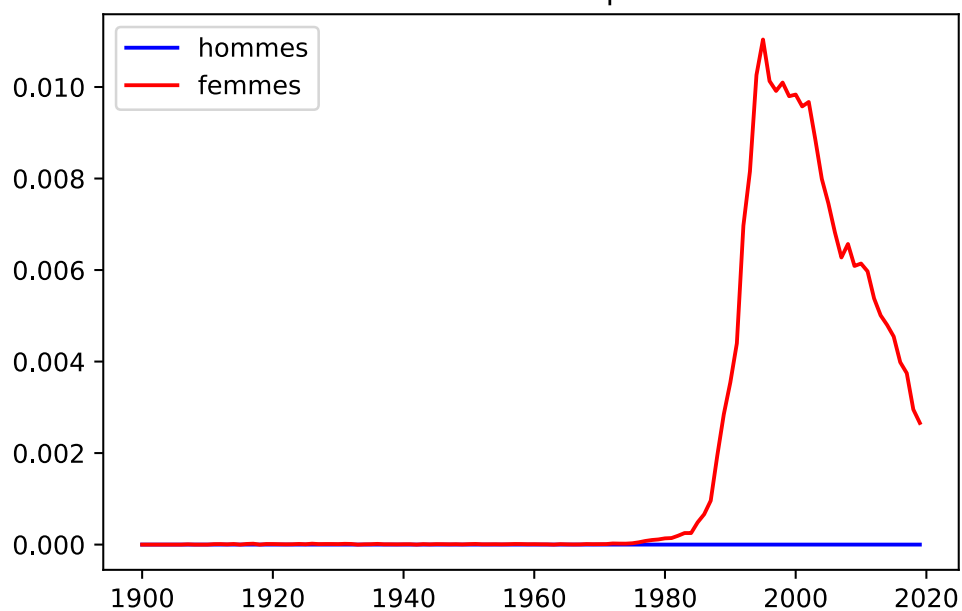
Nombre de bébés nommés MATTHIEU,... chaque année entre 1900 et 2020.



Le prénom MANON a été donné 20.0 fois à des hommes pour des années indéfinies

Le prénom MANON a été donné 19.0 fois à des femmes pour des années indéfinies

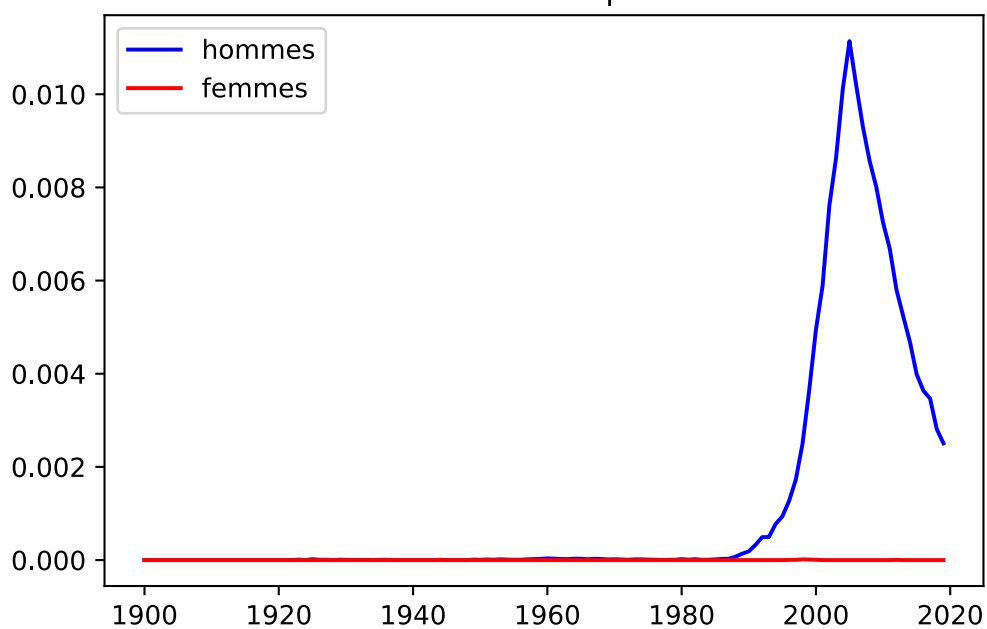
Nombre de bébés nommés MANON chaque année entre 1900 et 2020.



Le prénom ENZO a été donné 20.0 fois à des hommes pour des années indéfinies

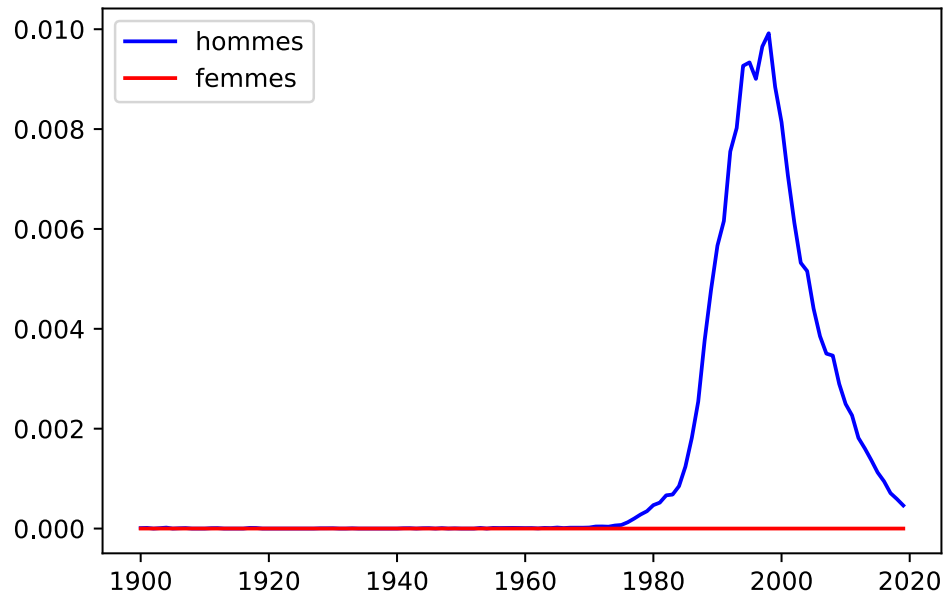
Le prénom ENZO a été donné 27.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés ENZO chaque année entre 1900 et 2020.



Le prénom QUENTIN a été donné 44.0 fois à des hommes pour des années indéfinies
Le prénom QUENTIN a été donné 21.0 fois à des femmes pour des années indéfinies

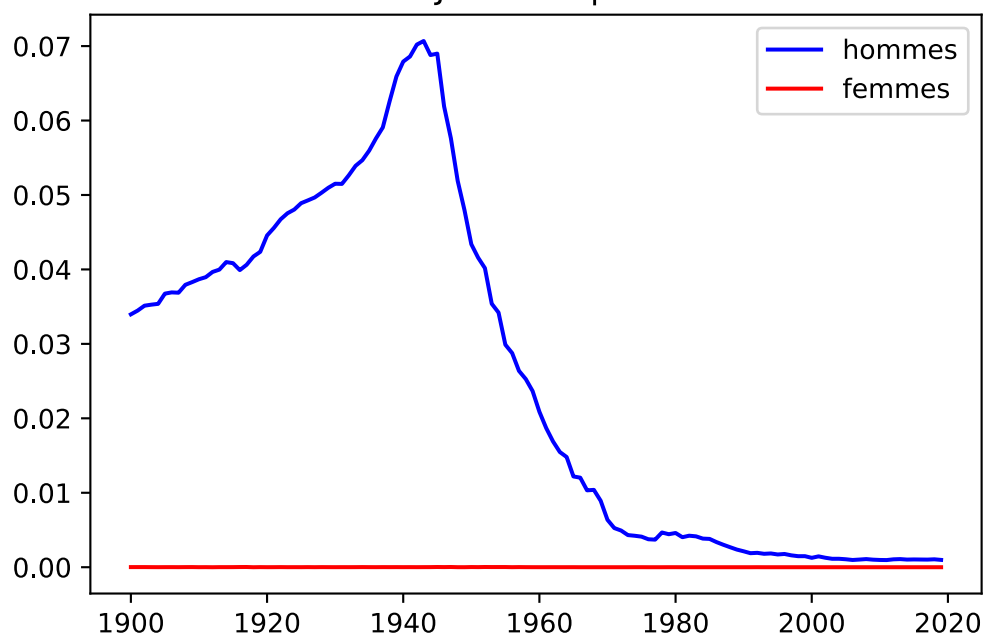
Nombre de bébés nommés QUENTIN chaque année entre 1900 et 2020.



Traitement de la base de donnée des prénoms de la génération des grands-parents:

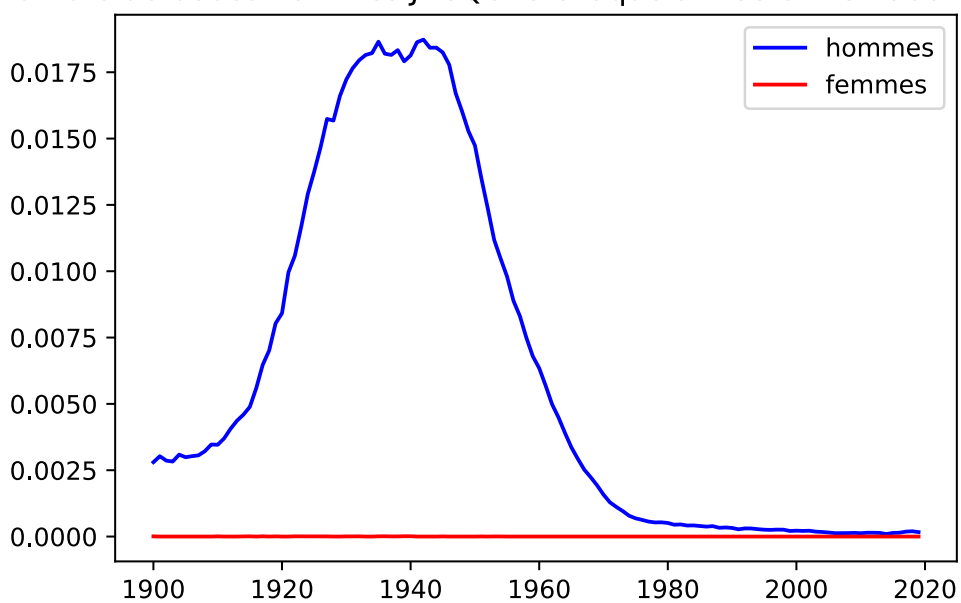
Le prénom JEAN a été donné 34.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés JEAN chaque année entre 1900 et 2020.



Le prénom JACQUES a été donné 43.0 fois à des femmes pour des années indéfinies

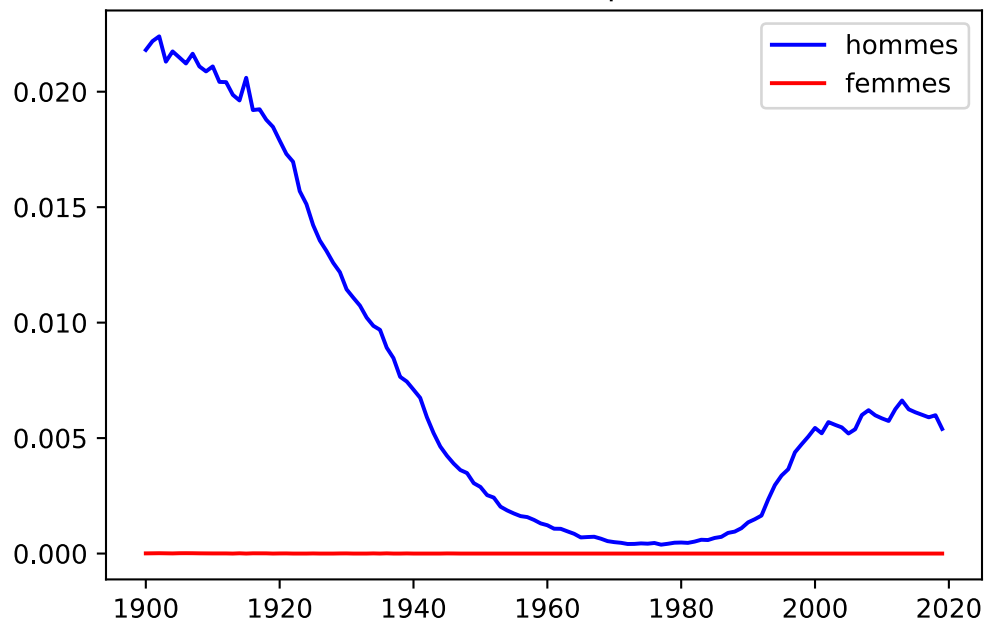
Nombre de bébés nommés JACQUES chaque année entre 1900 et 2020.



Traitement de la base de donnée des prénoms à nouveau à la mode:

Le prénom LOUIS a été donné 61.0 fois à des femmes pour des années indéfinies

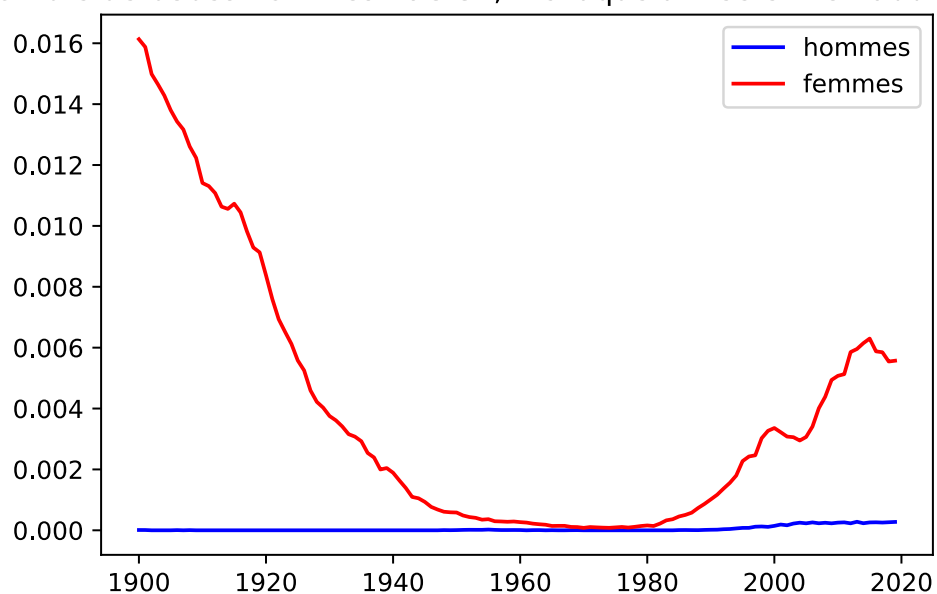
Nombre de bébés nommés LOUIS chaque année entre 1900 et 2020.



Les prénoms LOUISE LOUISON ont été donnés 50.0 fois à des hommes pour des années indéfinies

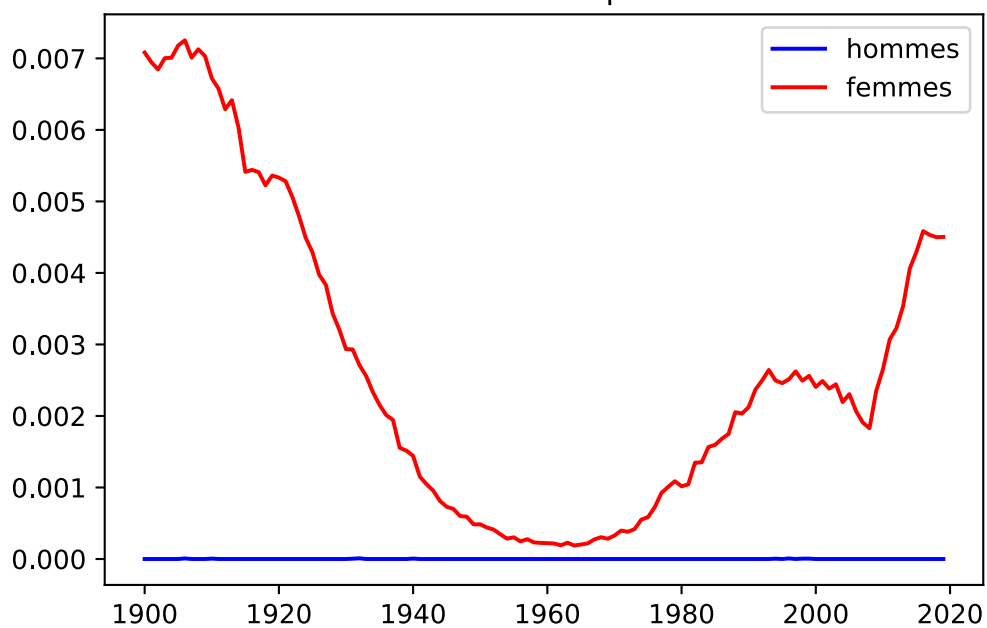
Les prénoms LOUISE LOUISON ont été donnés 52.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés LOUISE,... chaque année entre 1900 et 2020.



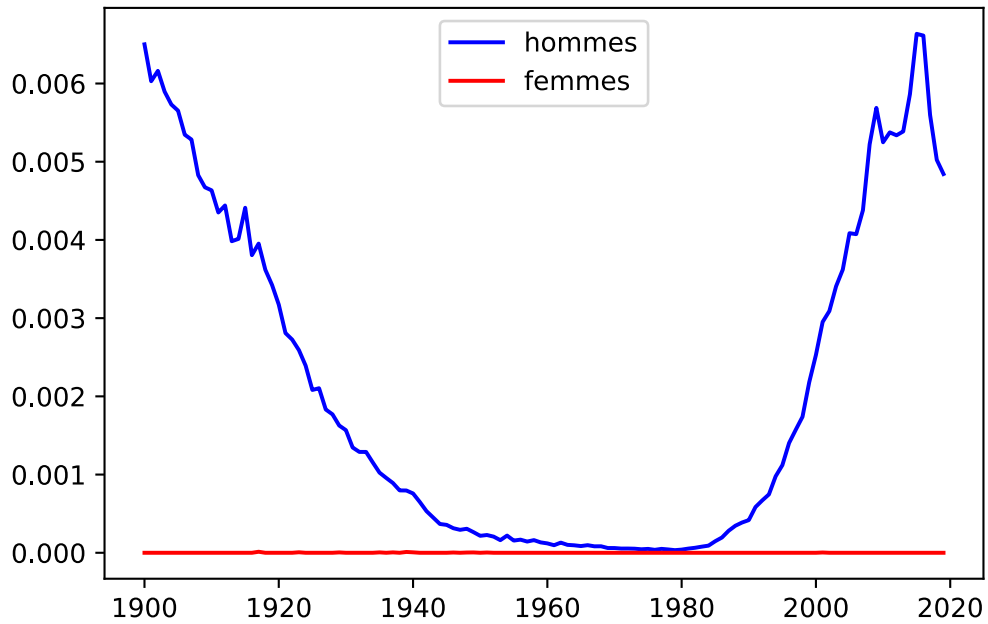
Le prénom ALICE a été donné 42.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés ALICE chaque année entre 1900 et 2020.



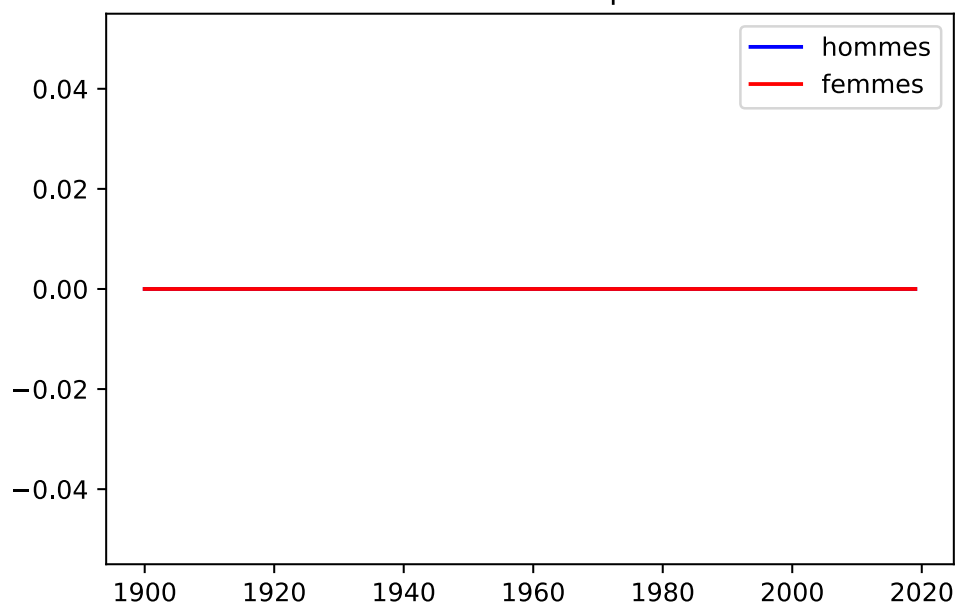
Le prénom JULES a été donné 89.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés JULES chaque année entre 1900 et 2020.



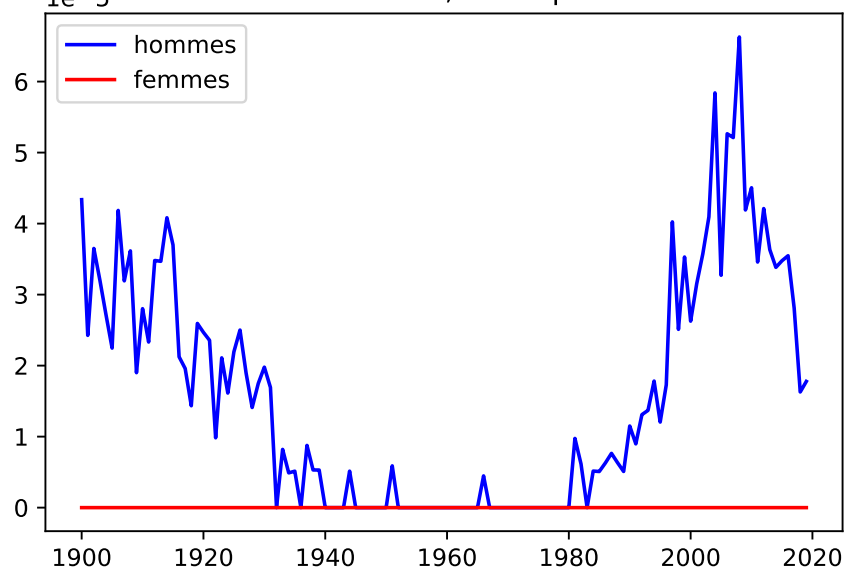
Traitement de la base de donnée des prénoms des gens uniques:

Nombre de bébés nommés ISSEINIE chaque année entre 1900 et 2020.



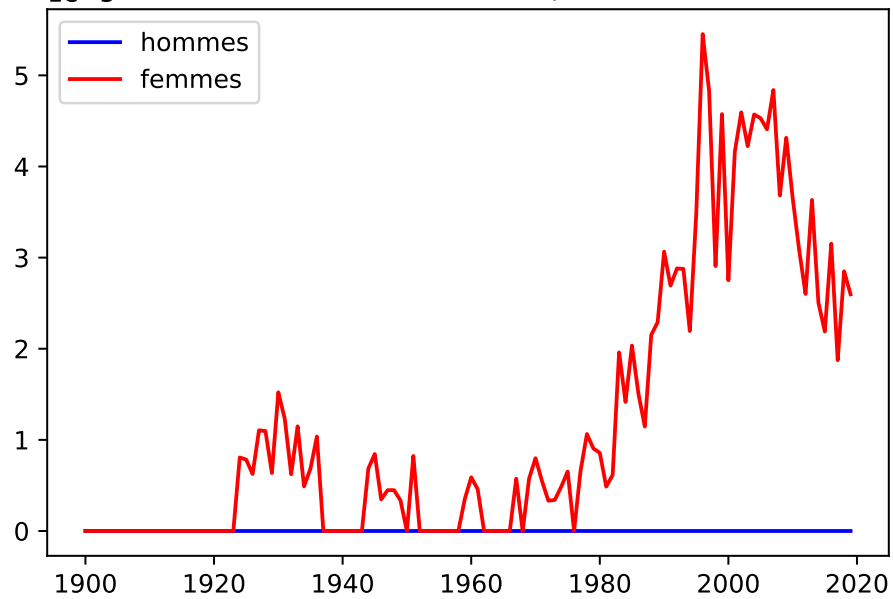
Les prénoms PHILEMON PHILÉMON ont été donnés 57.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés PHILEMON,... chaque année entre 1900 et 2020.

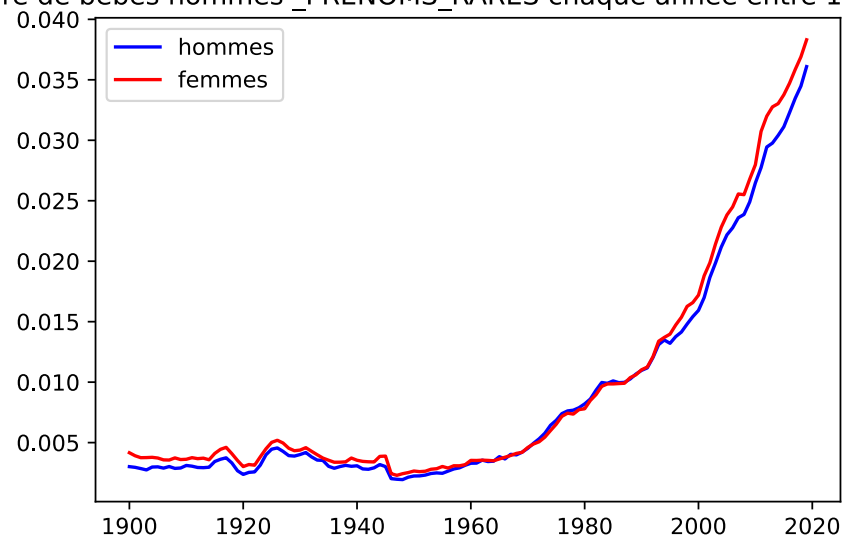


Le prénom LUDMILA a été donné 34.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés LUDMILA chaque année entre 1900 et 2020.



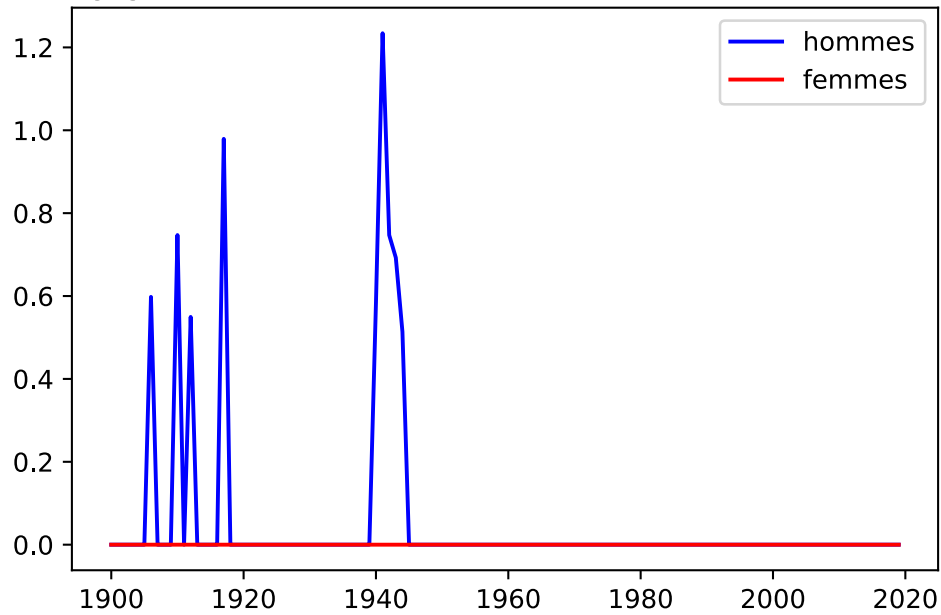
Nombre de bébés nommés _PRENOMS_RARES chaque année entre 1900 et 2020.



Traitement de la base de donnée des prénoms des méchants:

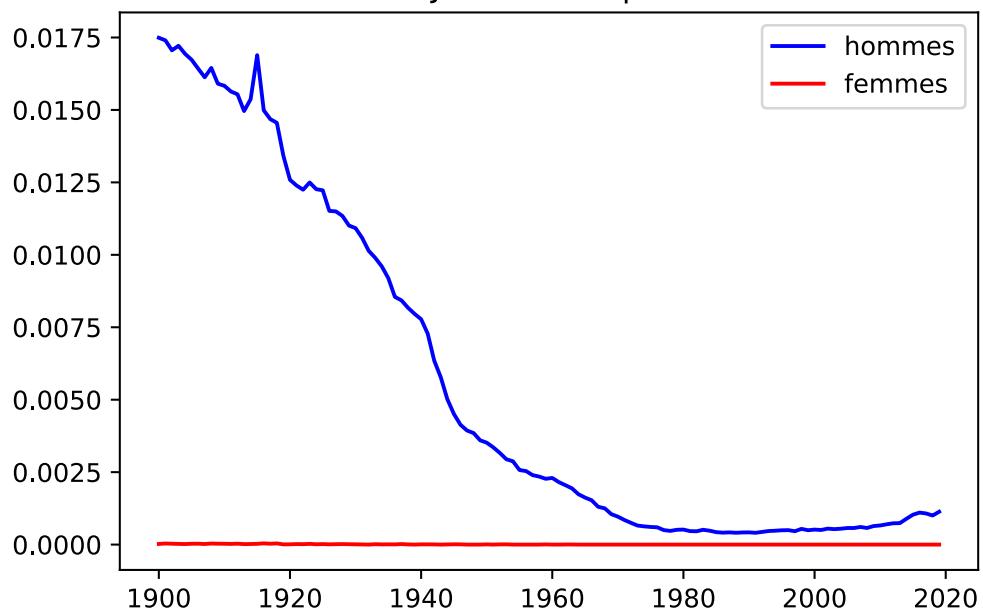
Le prénom ADOLF a été donné 23.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés ADOLF chaque année entre 1900 et 2020.



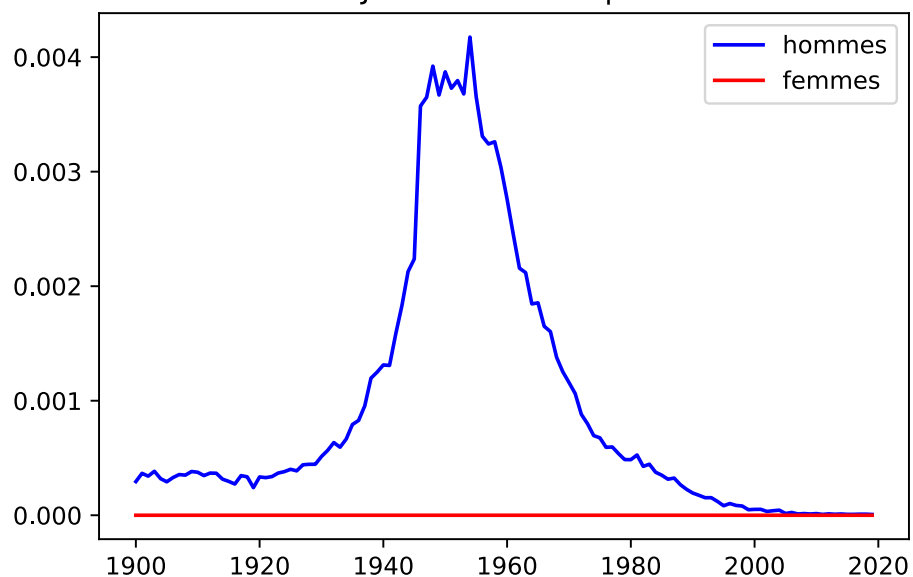
Le prénom JOSEPH a été donné 36.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés JOSEPH chaque année entre 1900 et 2020.



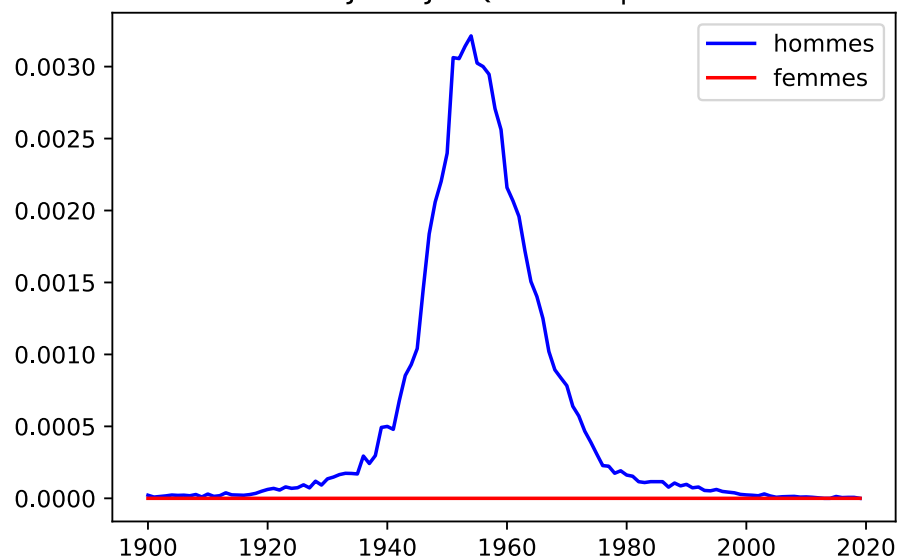
Traitement de la base de donnée des prénoms des boomers:

Nombre de bébés nommés JEAN-MARIE chaque année entre 1900 et 2020.

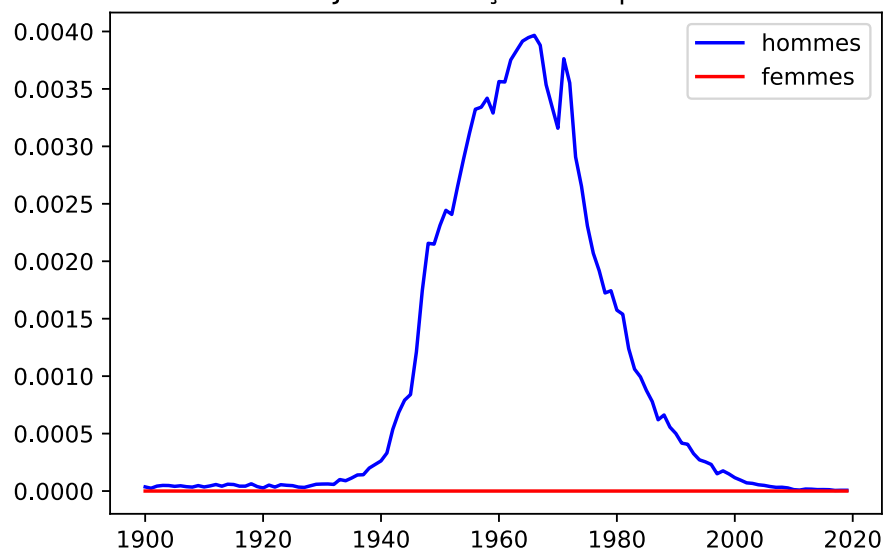


Le prénom JEAN-JACQUES a été donné 3.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés JEAN-JACQUES chaque année entre 1900 et 2020.



Nombre de bébés nommés JEAN-FRANÇOIS chaque année entre 1900 et 2020.



4 Une approximation des courbes

4.1 Matching-pursuit

```
[21]: def prod_scal(f, D):  
    '''  
    f : The function to match.  
    D : The dictionary of gaussian functions to match f.  
  
    return : The absolute distance between the function f and each gaussian in  
    →the dictionary D.  
    '''  
    # ret is the array of each absolute distance  
    ret = np.zeros(D.shape[0])  
    for i in range(D.shape[0]):  
        # i_m_n is the index of the maximum of each gaussian  
        i_m_n = np.argmax(D[i])  
        # a_n is the coefficient for each gaussian that permit to minimize its  
    →distance with f  
        a_n = f[i_m_n]/D[i][i_m_n]  
        # The scalar product is finally the sum of the difference between the  
    →function f and the gaussian  
        ret[i] = np.abs(f - a_n*D[i]).sum()  
    return ret  
  
def matching_pursuit(f, dictionary, epsilon=1e-4, it_limit=10):  
    '''  
    Implements the matching pursuit algorithm applied to functions, with a  
    →dictionary of gaussians.  
    f : The function to match.  
    dictionary : The dictionary of gaussian functions to match f.  
  
    return : The list of the index of the selected gaussian in the dictionary,  
    →the list of the corresponding coefficients to ponderate each selected  
    →gaussian, and the mean of f to match it perfectly.  
    '''  
    f_appro = f  
    n = 0  
    res_index = np.array([], dtype=int)  
    res_coeff = np.array([])  
    while np.max(np.abs(f_appro)) > epsilon and n < it_limit :  
        i_max = np.argmin(prod_scal(f_appro, dictionary))  
        plt.plot(f_appro)  
        plt.show()  
        i_m_n = np.argmax(dictionary[i_max])  
        a_n = f_appro[i_m_n]/dictionary[i_max][i_m_n]  
        f_appro = f_appro - a_n*dictionary[i_max]
```

```

        res_index = np.append(res_index, i_max)
        res_coeff = np.append(res_coeff, a_n)
        n += 1
    return res_index, res_coeff

def loi_nor_cent_red(x, center, ecart_type):
    """
    Basically return the value of a gaussian centered in `center` with a standard_
    →deviation of `ecart_type` at the point x.
    """
    return np.exp(-((x-center)/ecart_type)**2 /2)/(ecart_type*np.sqrt(2*np.pi))

def init_dictionary():
    """
    Initialize the gaussian dictionary.
    In the final dictionary, there are 120 gaussians, centered in the 120_
    →different years, for each standard deviation (here : 1, 5, 10 and 20, which_
    →permits to match correctly every kind of function we are studying).

    return : an array of gaussians (represented as an array of size 120).
    """
    res_dic = []
    n = 0
    for i in range(120):
        n += 1
        for x in range(120):
            res_dic.append(loi_nor_cent_red(x, i, 1))
    for i in range(120):
        n += 1
        for x in range(120):
            res_dic.append(loi_nor_cent_red(x, i, 5))
    for i in range(120):
        n += 1
        for x in range(120):
            res_dic.append(loi_nor_cent_red(x, i, 10))
    for i in range(120):
        n += 1
        for x in range(120):
            res_dic.append(loi_nor_cent_red(x, i, 20))
    return np.array(res_dic).reshape((n,120))

def extractPartByYear(dataBase, names):
    """
    This function is exactly identical with the function `plotPartByYear`,_
    →except that it return the lists instead of plotting them.
    """
    X = np.array([x for x in range(1900,2020)])

```

```

YF = np.zeros((121))
YM = np.zeros((121))
for name in names:
    isName = DataBase['preusuel']==name
    allName = DataBase[isName]

    allName = allName.replace('XXXX', '2050')

    allName['annais'] = allName['annais'].astype(int)

    X_M = allName[allName.sexe==1].annais.array
    Y_M = allName[allName.sexe==1].nombre.array
    X_F = allName[allName.sexe==2].annais.array
    Y_F = allName[allName.sexe==2].nombre.array

    for i in range(1900,2020):
        if not i in X_M:
            Y_M = np.insert(Y_M, i-1900, 0)
        if not i in X_F:
            Y_F = np.insert(Y_F, i-1900, 0)
    if not 2050 in X_M :
        Y_M = np.append(Y_M, 0)
    if not 2050 in X_F :
        Y_F = np.append(Y_F, 0)

    Y_M = Y_M.astype(float)
    Y_F = Y_F.astype(float)
    for year in X :
        totalYear = DataBase[DataBase.annais==str(year)].nombre.array.sum()
        Y_M[year-1900] /= totalYear
        Y_F[year-1900] /= totalYear

    YF += Y_F
    YM += Y_M

    return YM[: -1], YF[: -1]

f = extractPartByYear(prenoms, ['LOUIS'])[0]
dico = init_dictionary()

MP_i, MP_c = matching_pursuit(f, dico, it_limit=20)

# We now reconstruct the initial function with the result of our
→`matching_pursuit` algorithm
approx = np.zeros(120)

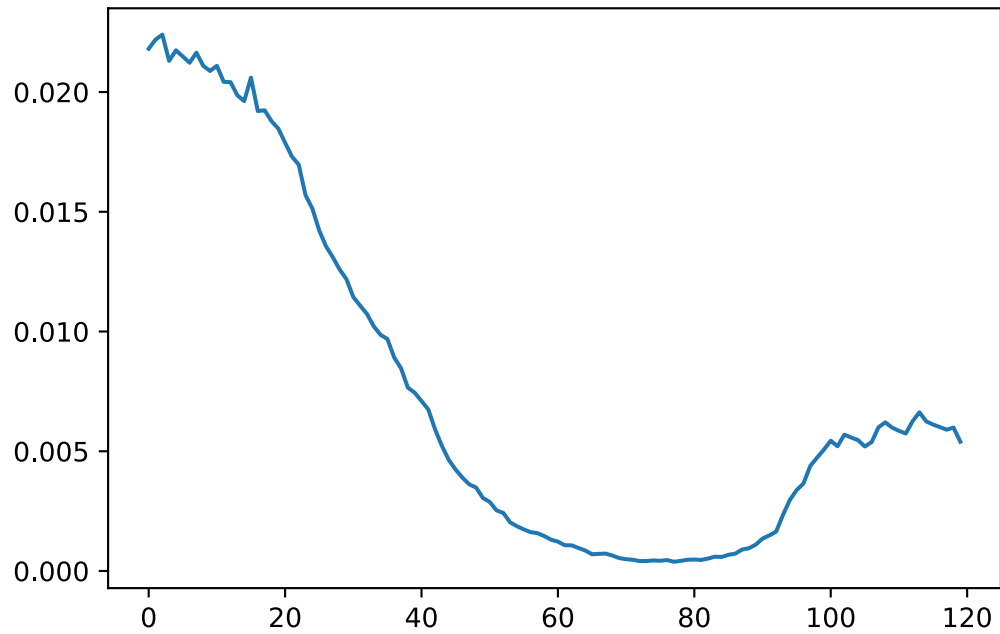
```

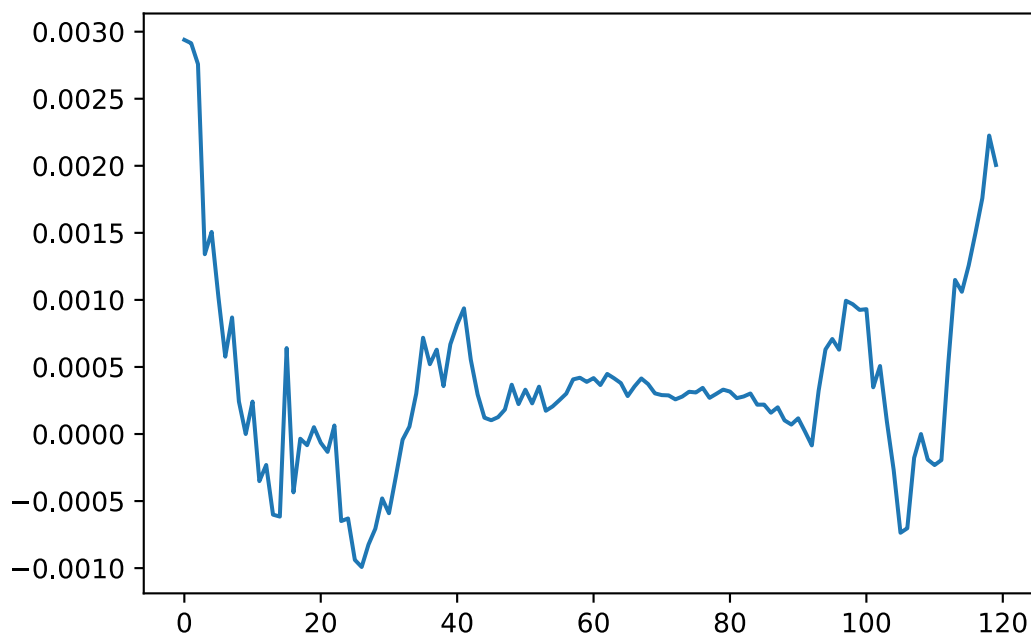
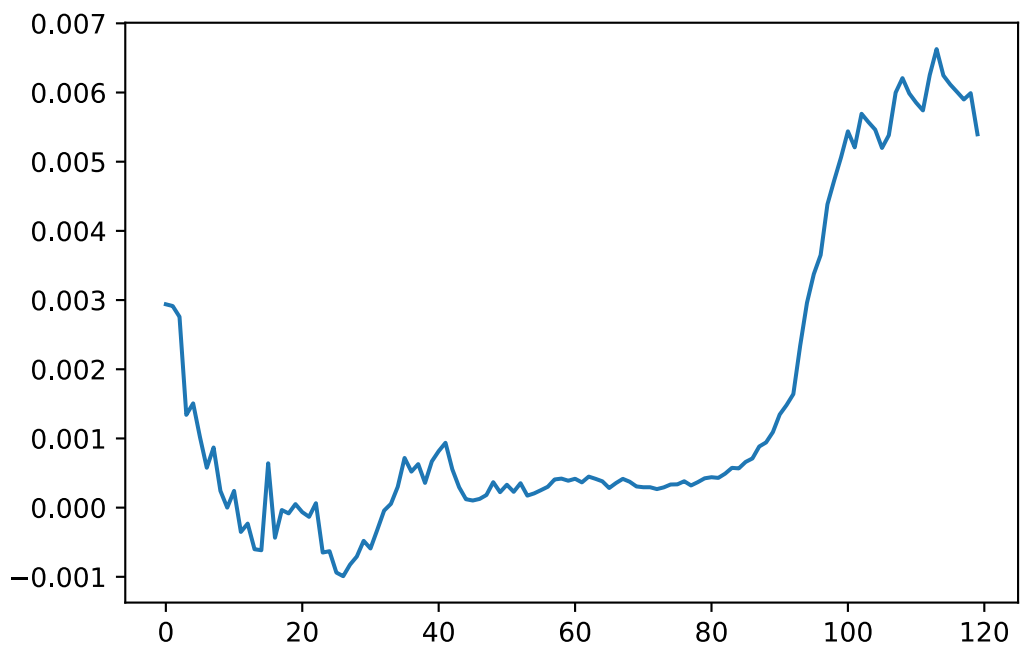
```

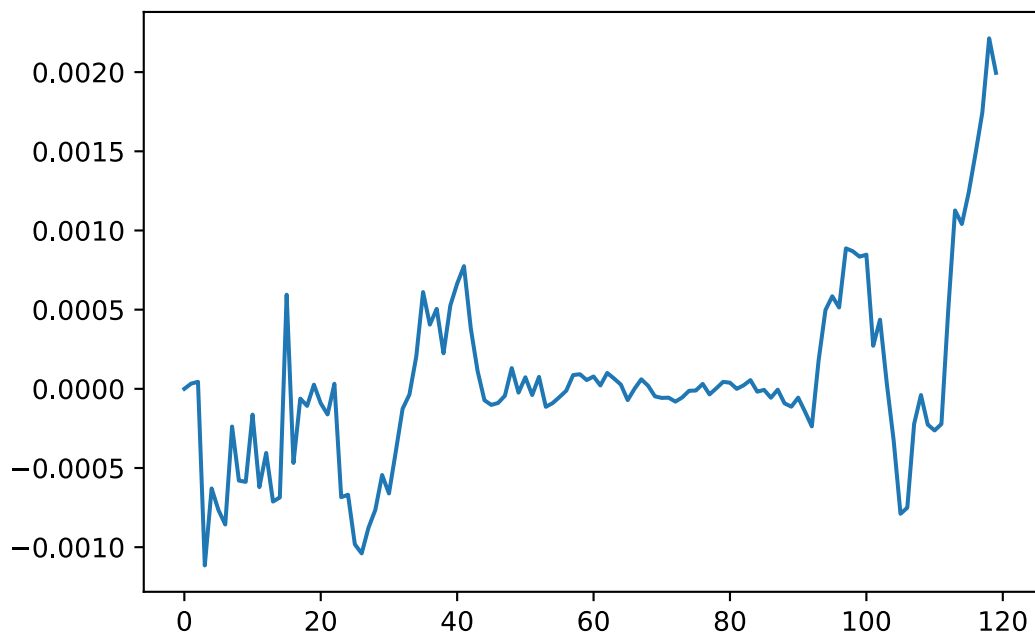
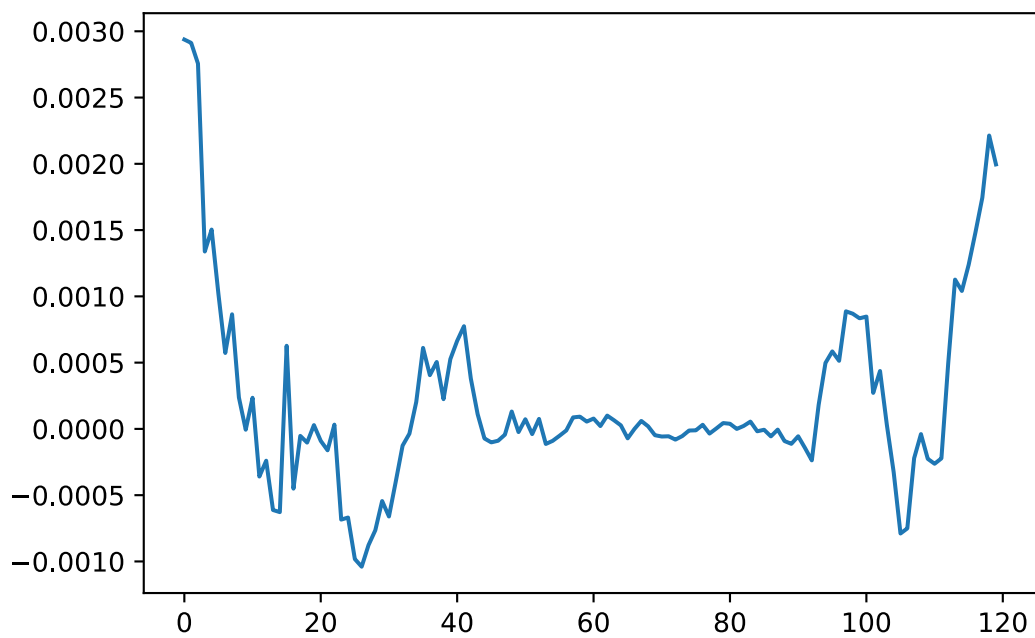
for i in range(len(MP_i)):
    approx += dico[MP_i[i]]*MP_c[i]

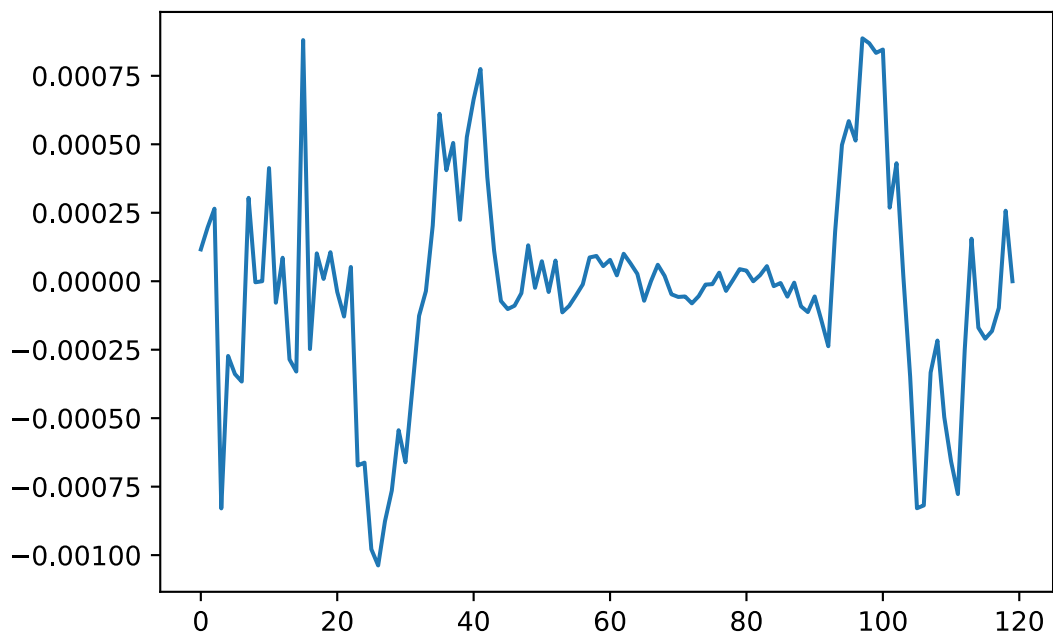
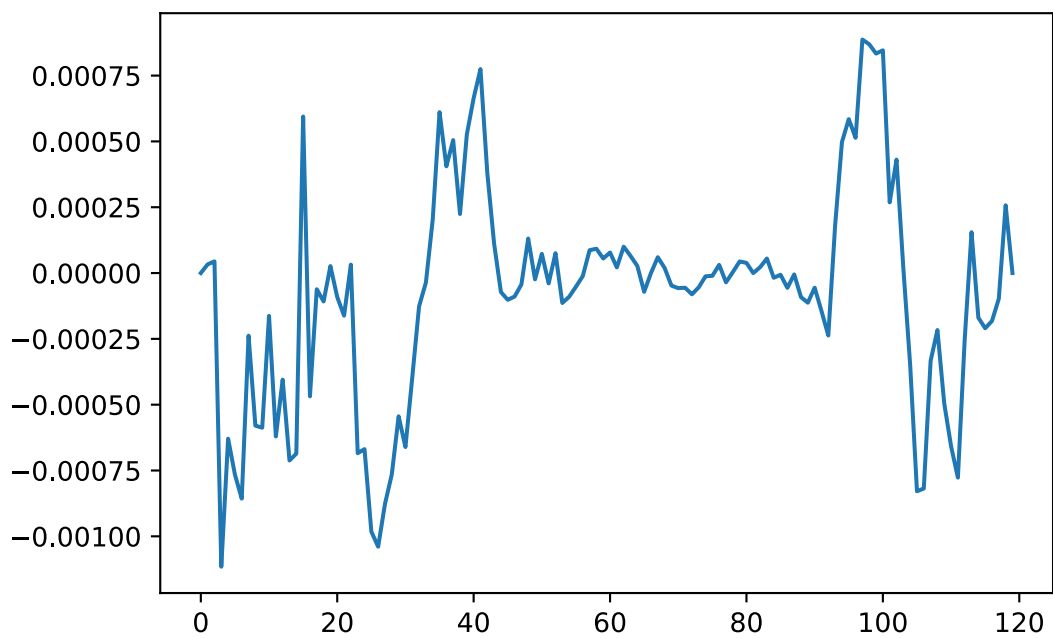
# And we plot the result
X = [x for x in range(1900,2020)]
plt.plot(X, f)
plt.plot(X, approx)
MP_i, MP_c, "Absolute error : " + str(np.abs(f-approx).sum()), "Max local error : 
→ " + str(np.abs(f-approx).max())

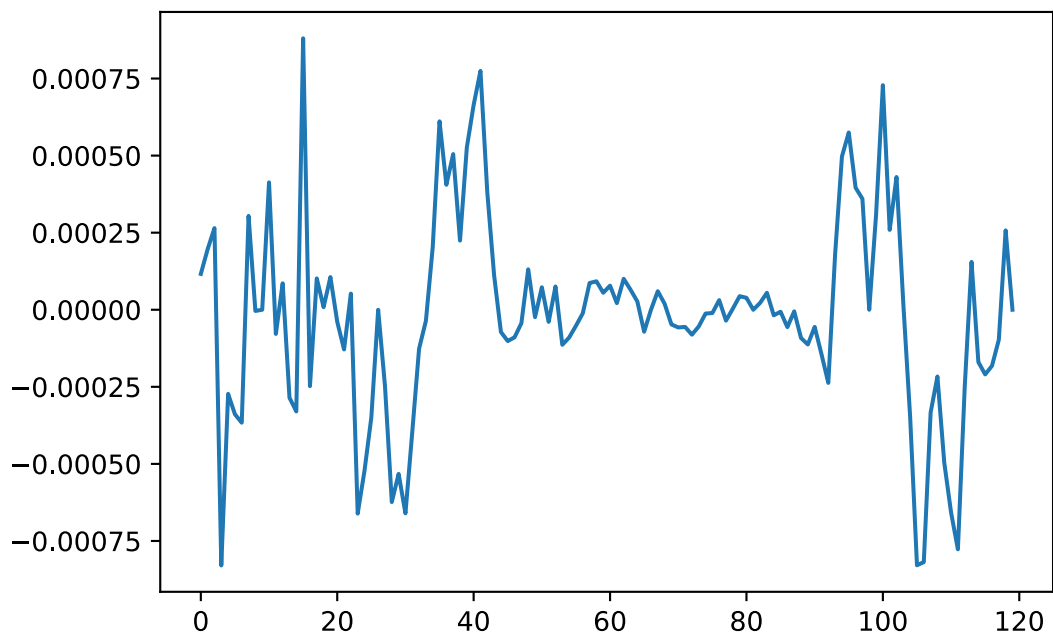
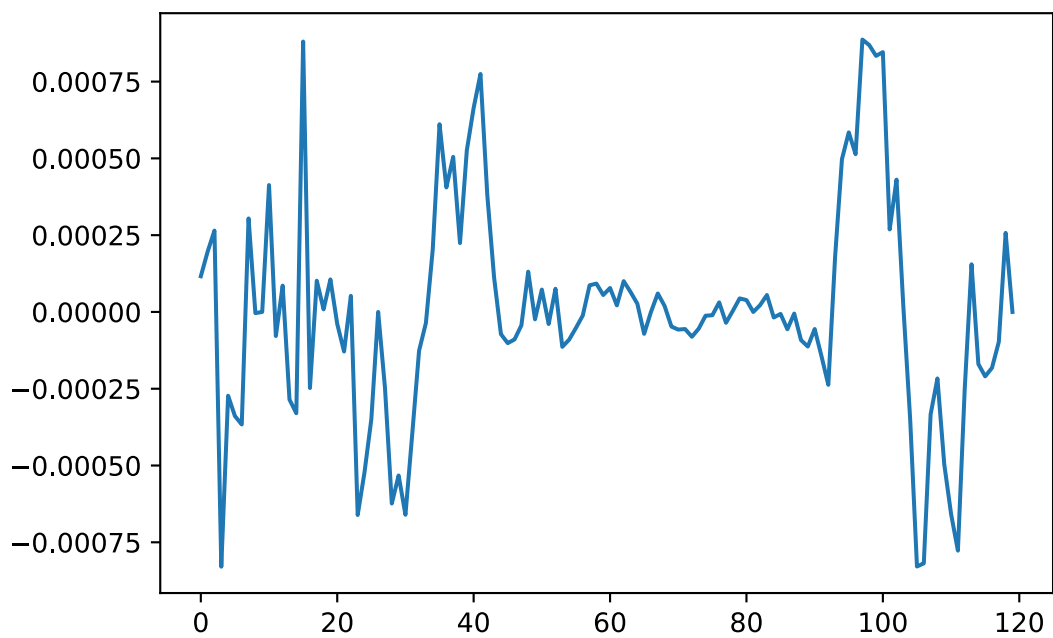
```

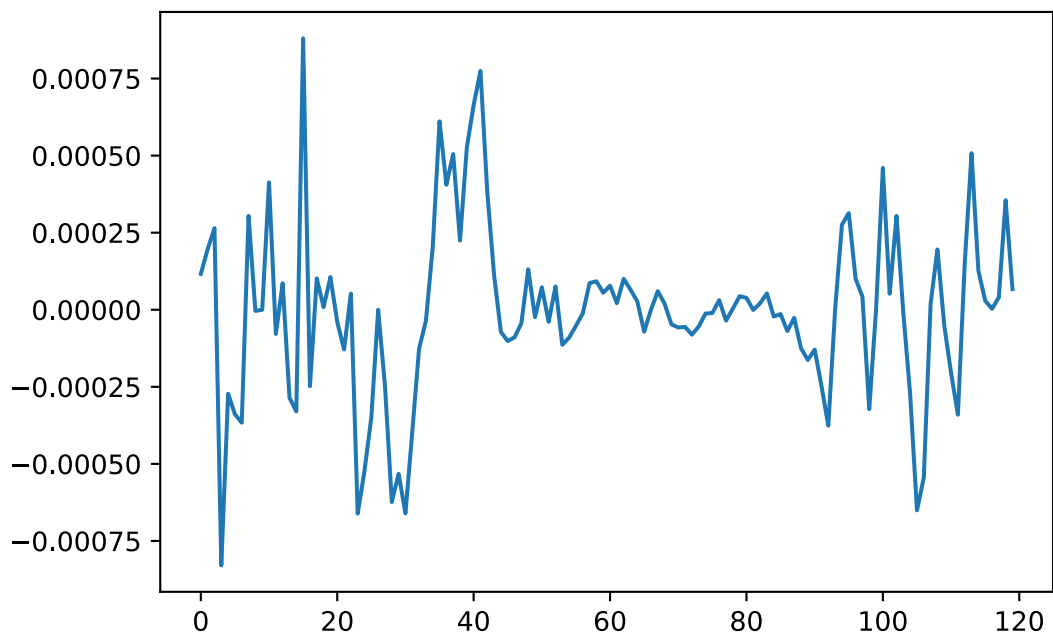
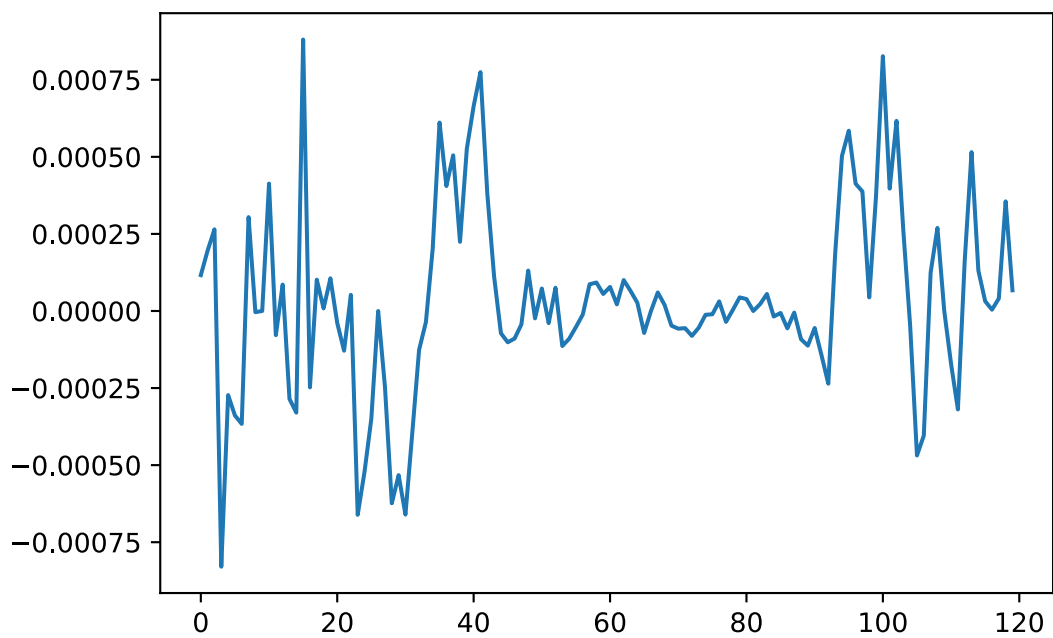


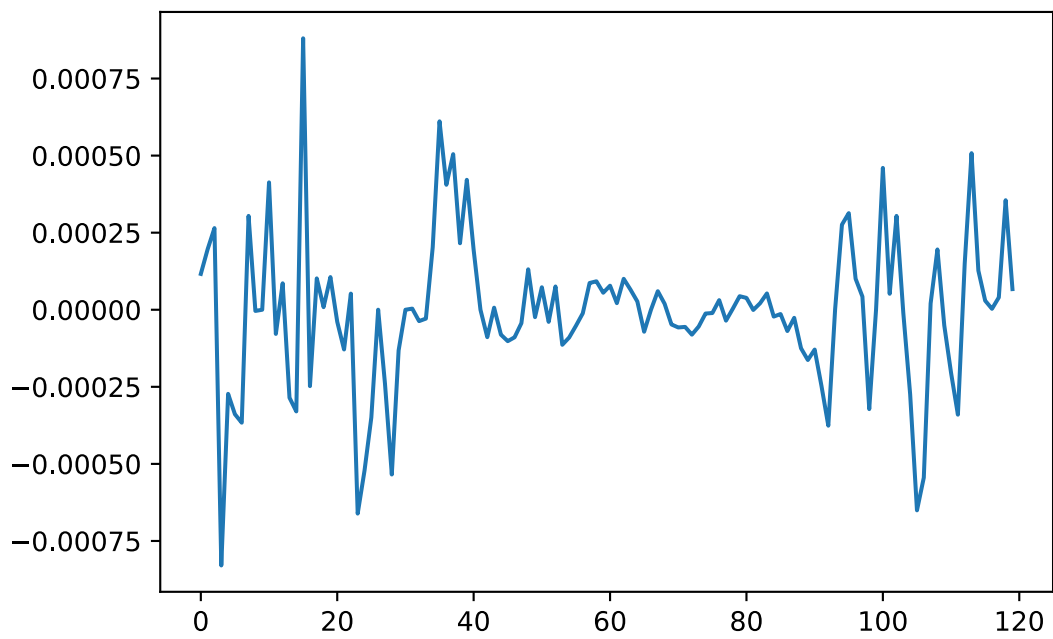
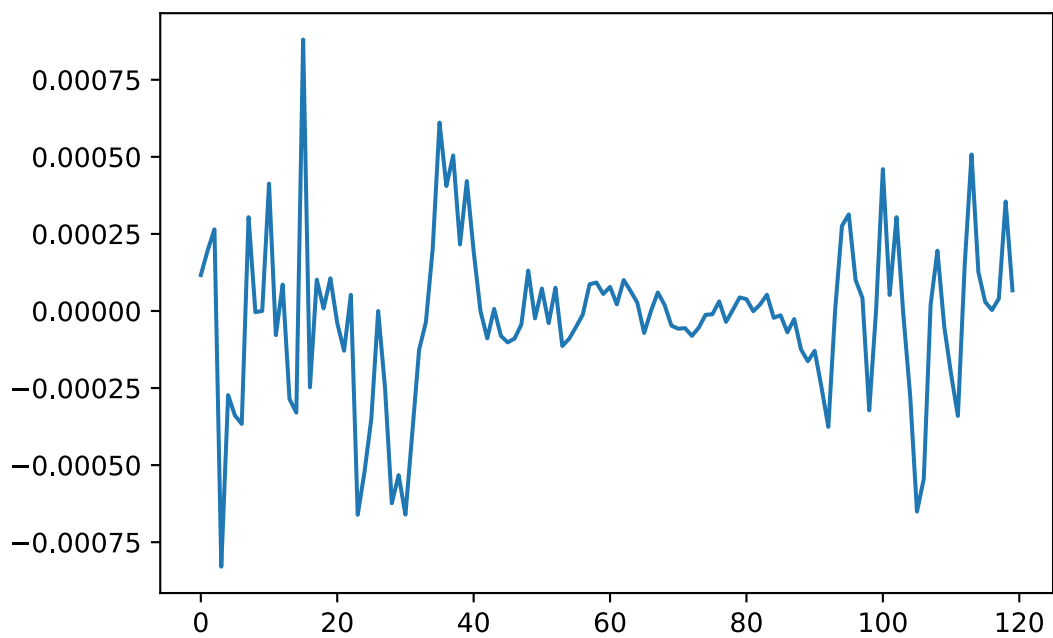


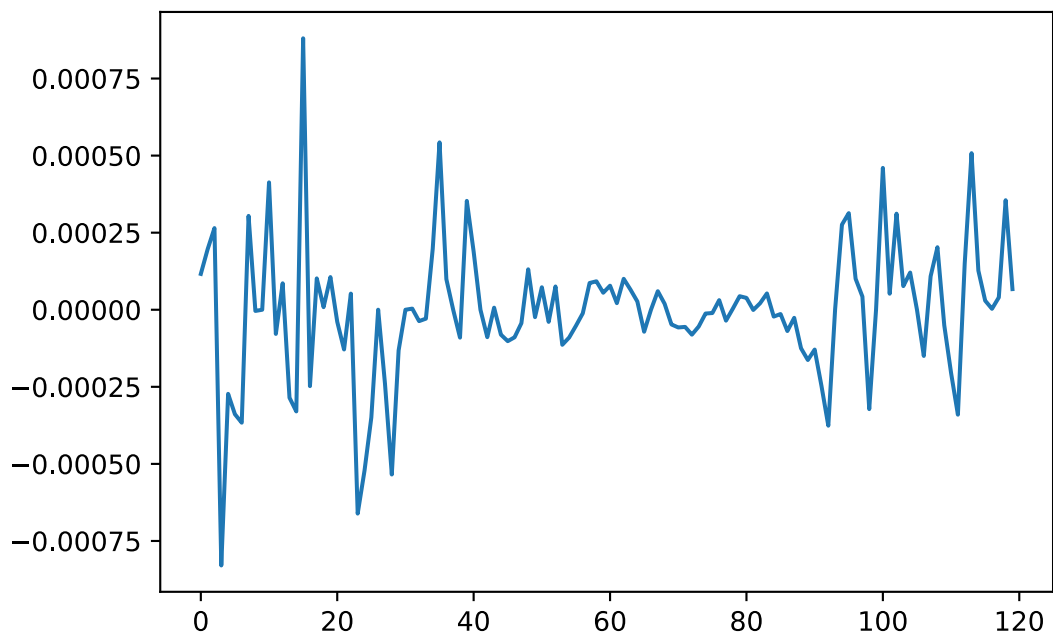
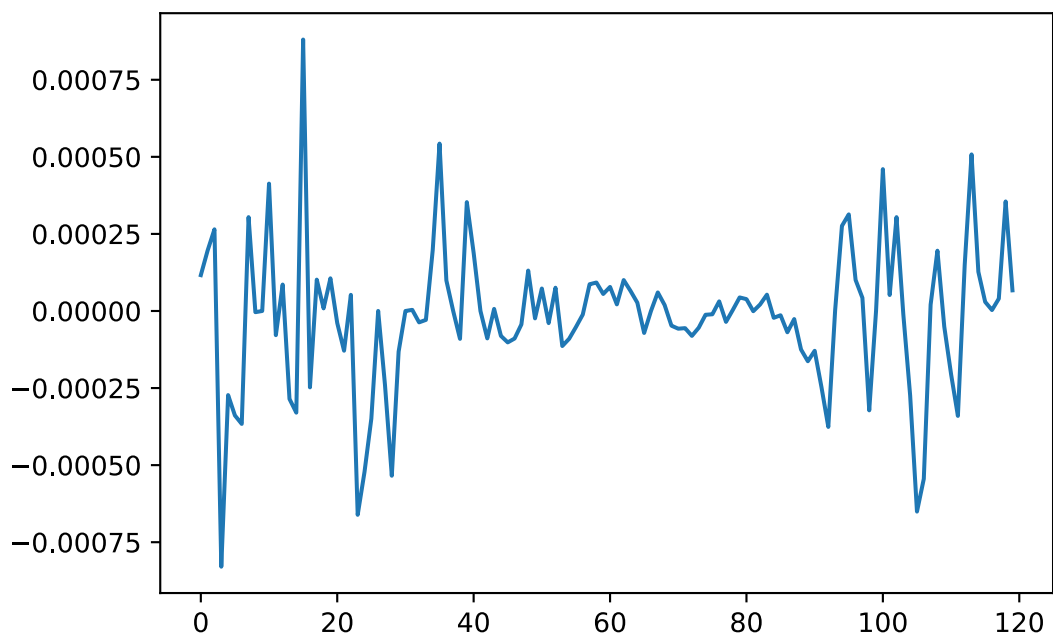


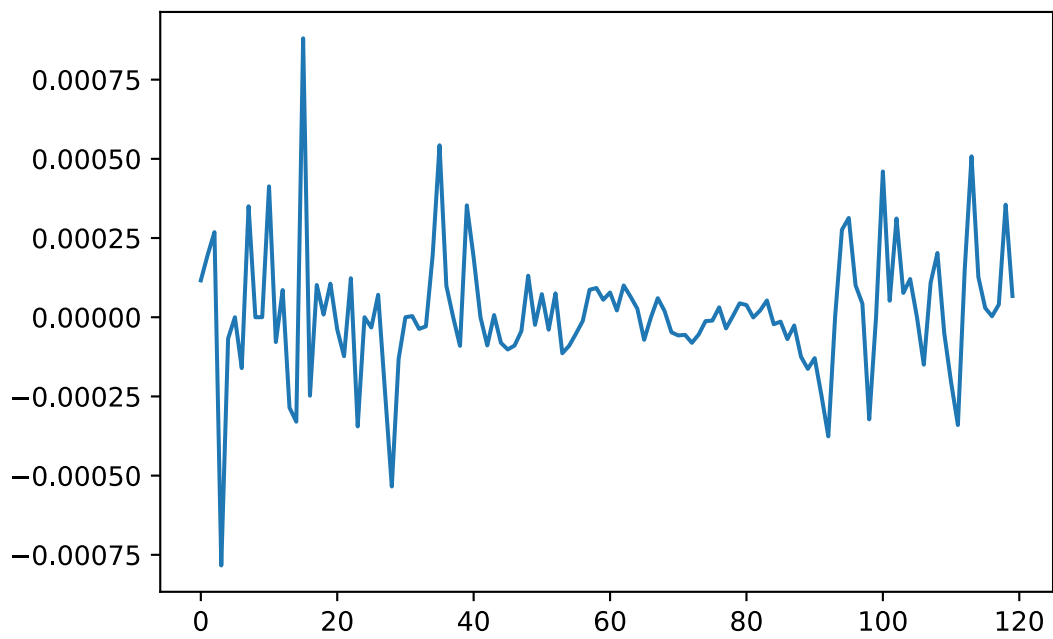
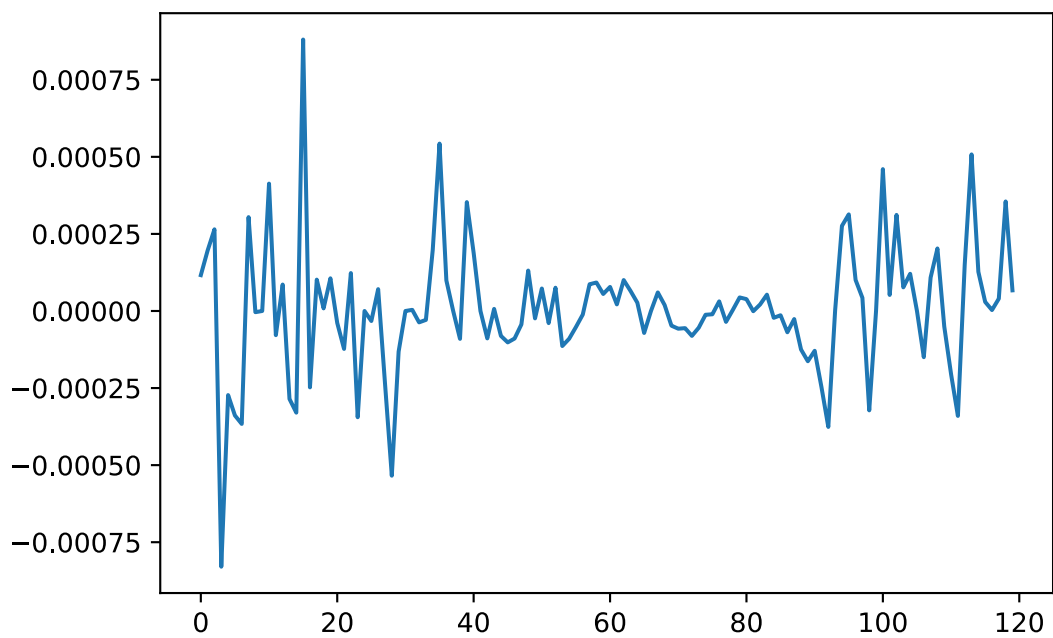


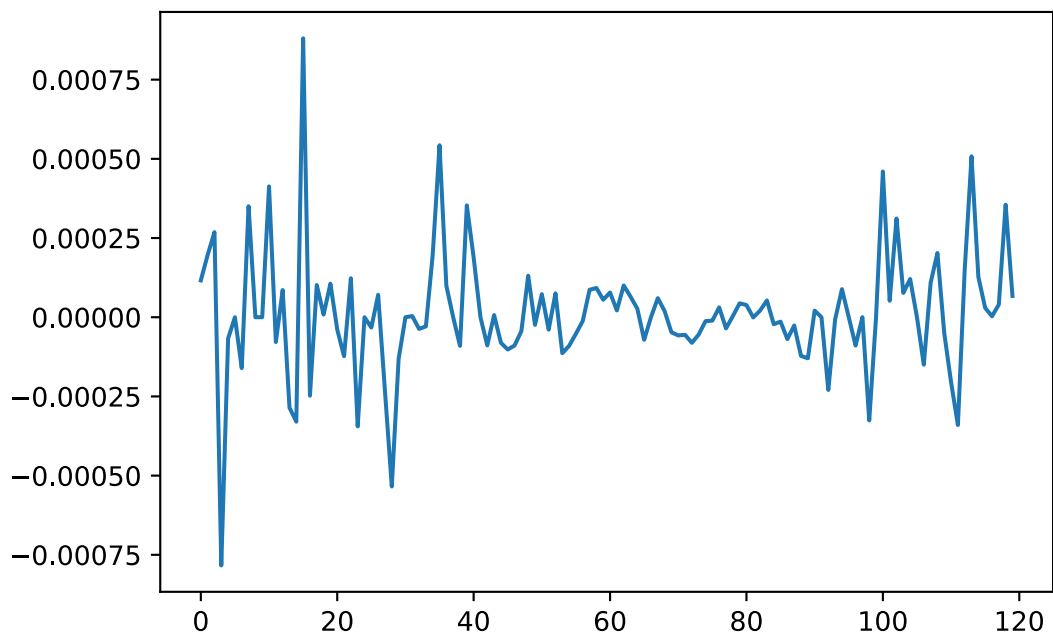
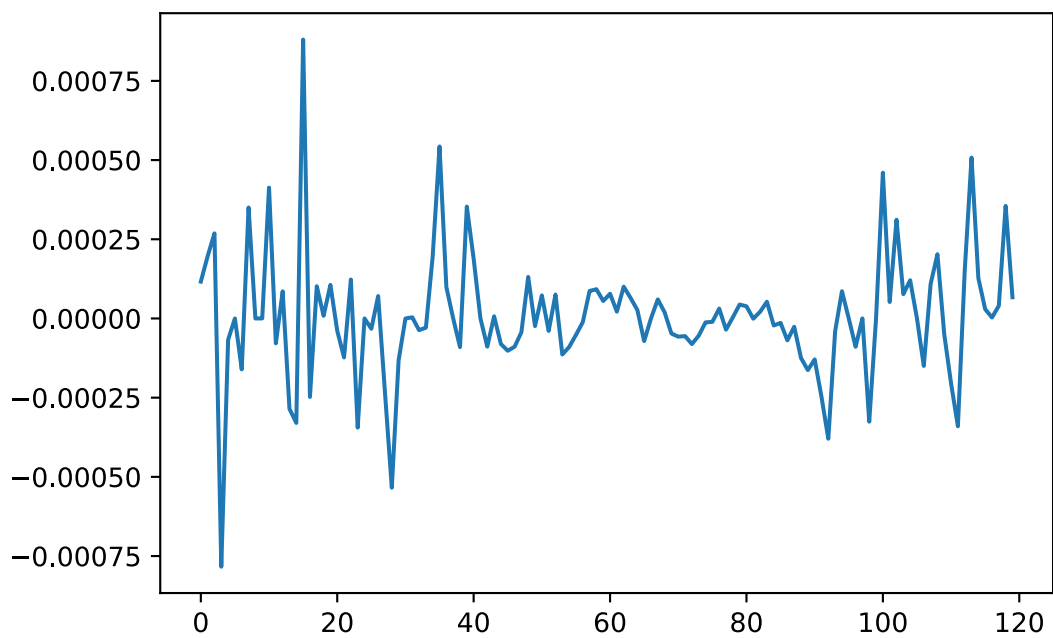


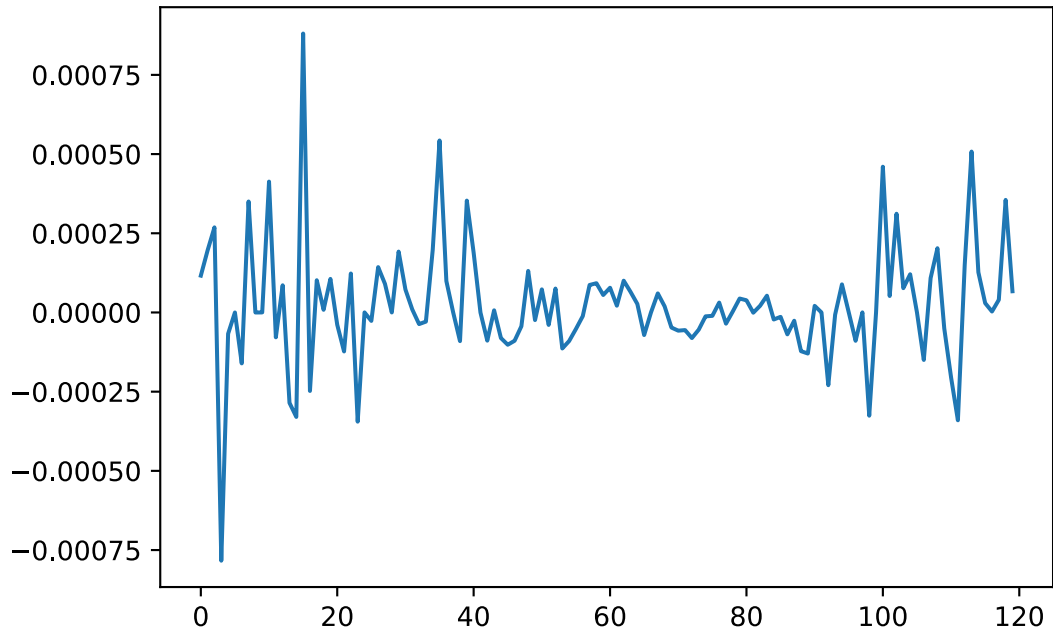




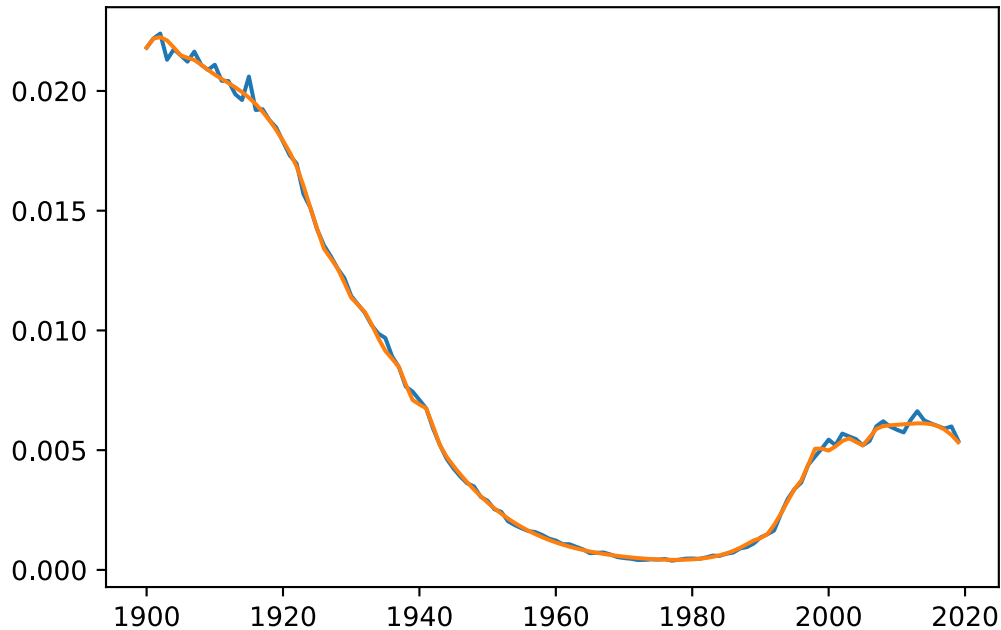








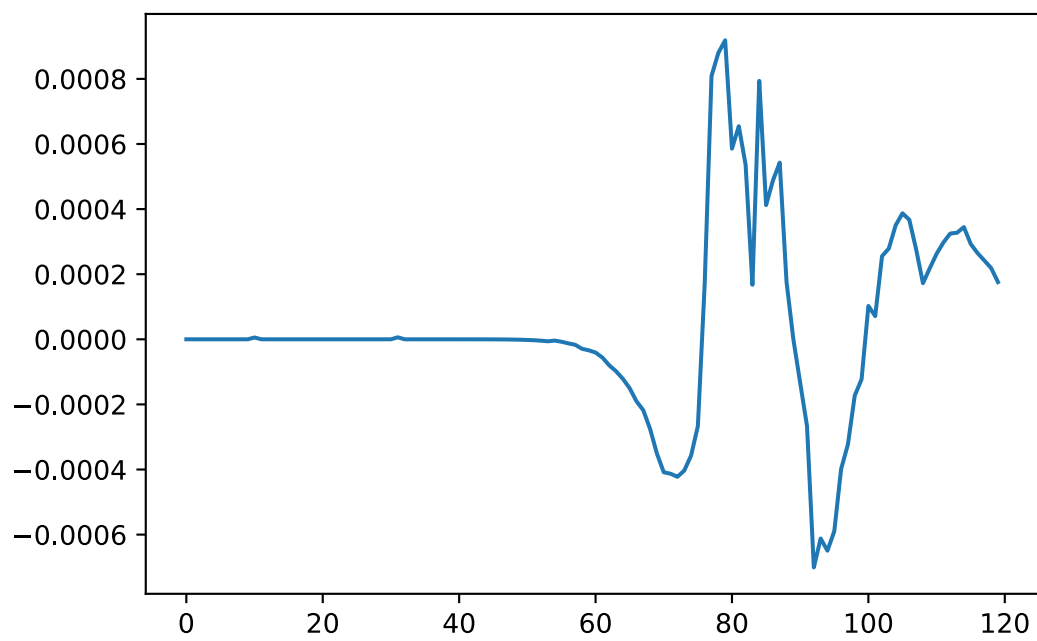
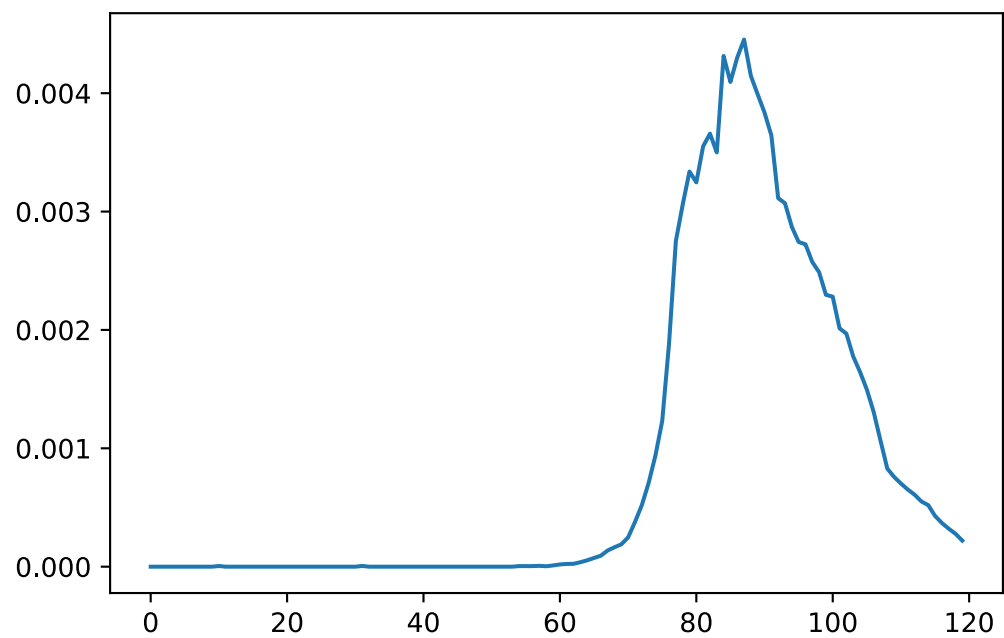
```
[21]: (array([369, 348, 426, 120, 239, 129, 26, 98, 229, 219, 41, 30, 37,
            105, 24, 5, 95, 91, 28, 1]),
      array([ 1.04673677e+00,  1.55612259e-01,  1.77788684e-02,  3.68165003e-02,
            2.50113188e-02, -7.36250875e-03, -2.60106014e-03,  2.17795476e-03,
           -6.21545919e-03,  4.68531675e-03,  1.94214497e-03, -1.65544767e-03,
            1.26511175e-03, -1.63101968e-03, -1.30806538e-03, -8.49519163e-04,
            7.85477555e-04, -6.20033148e-04, -1.33916083e-03,  4.92404545e-04]),
      'Absolute error : 0.013527427483245688',
      'Max local error : 0.0008804507433731248')
```

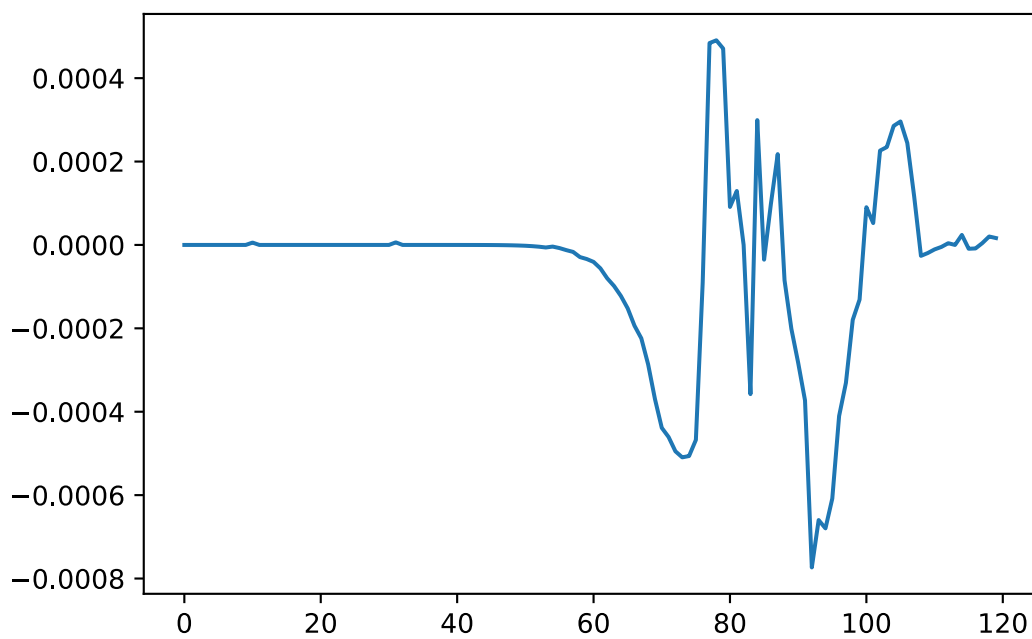
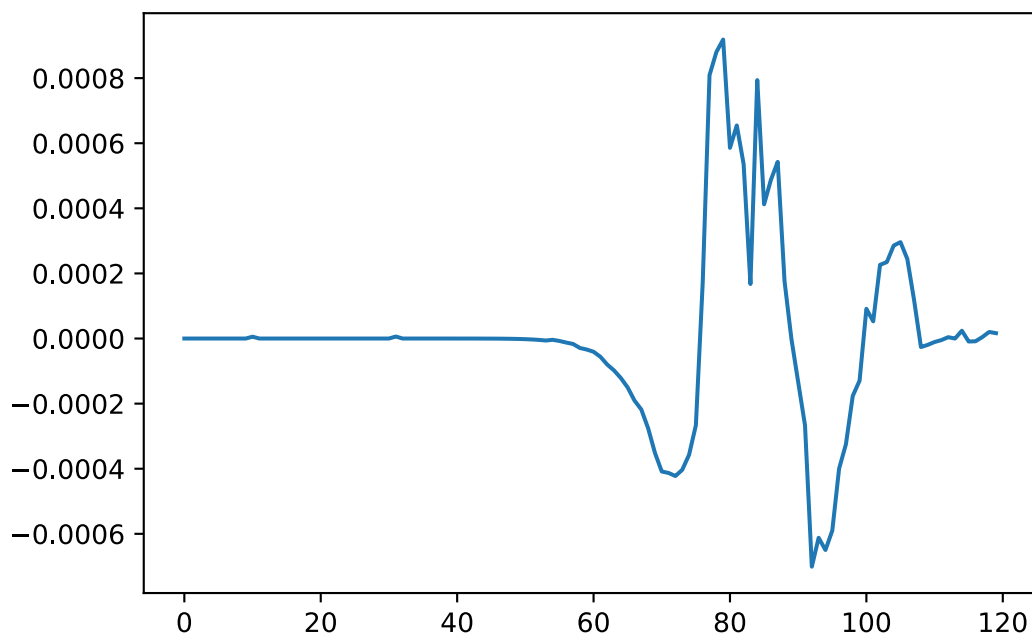


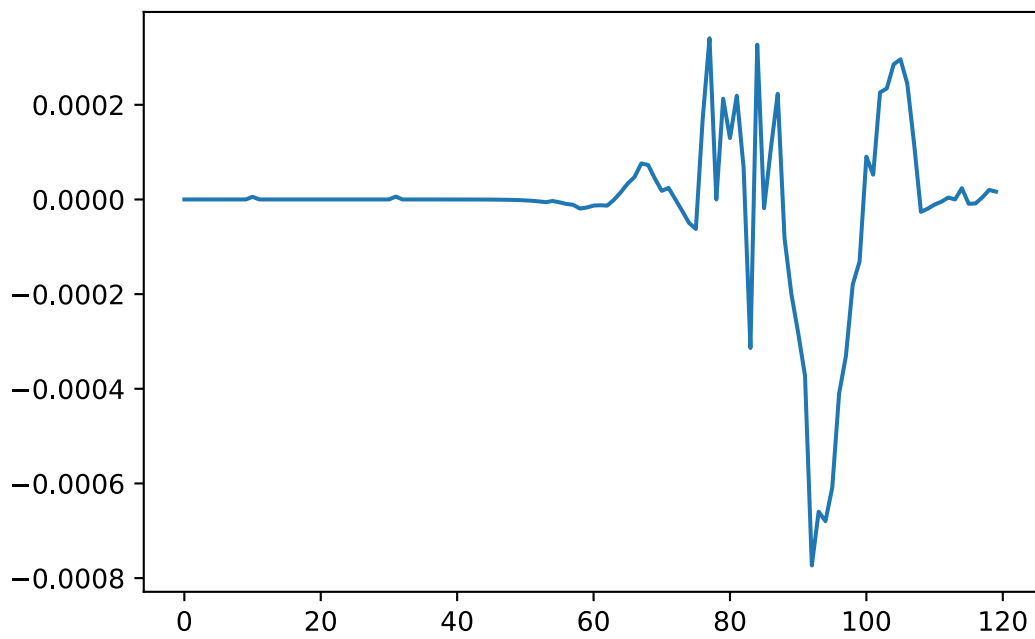
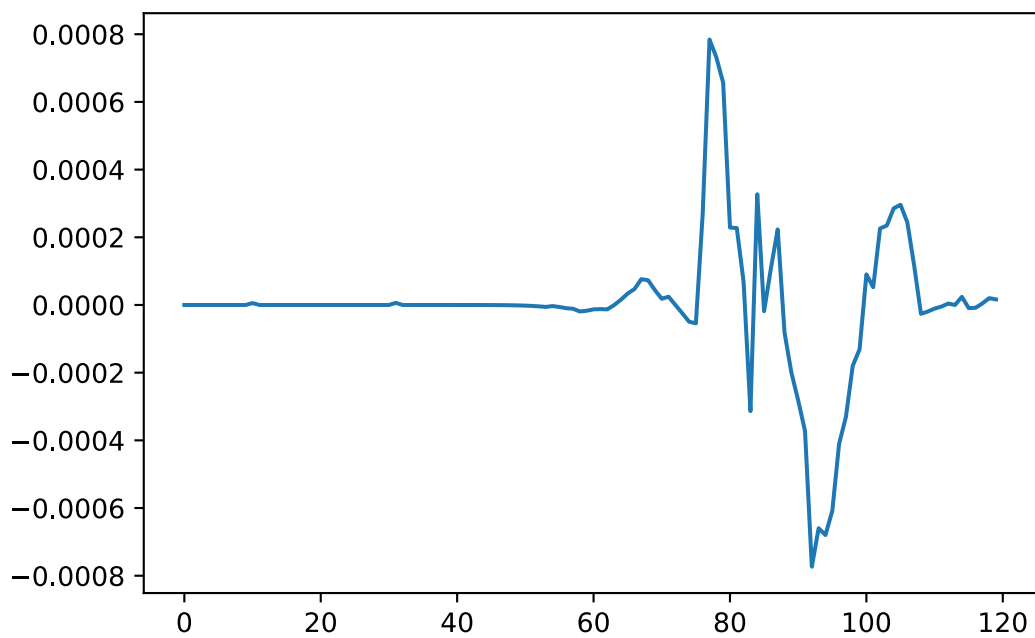
On remarque que Les indices des gaussiennes sélectionnées sont, globalement, dans [360,480] puis dans [240, 360] puis dans [120,240] en enfin dans [0,120]. On peut l'expliquer par les fonctions que ces intervalles représentent : les premiers concernent les gaussiennes avec un grand écart type tandis que les dernières représentent plus des pics ponctuels. Or il est avantageux de d'abord approcher globalement la fonction initiale (donc par des gaussiennes plus larges), puis ensuite de corriger les différents pics locaux créés.

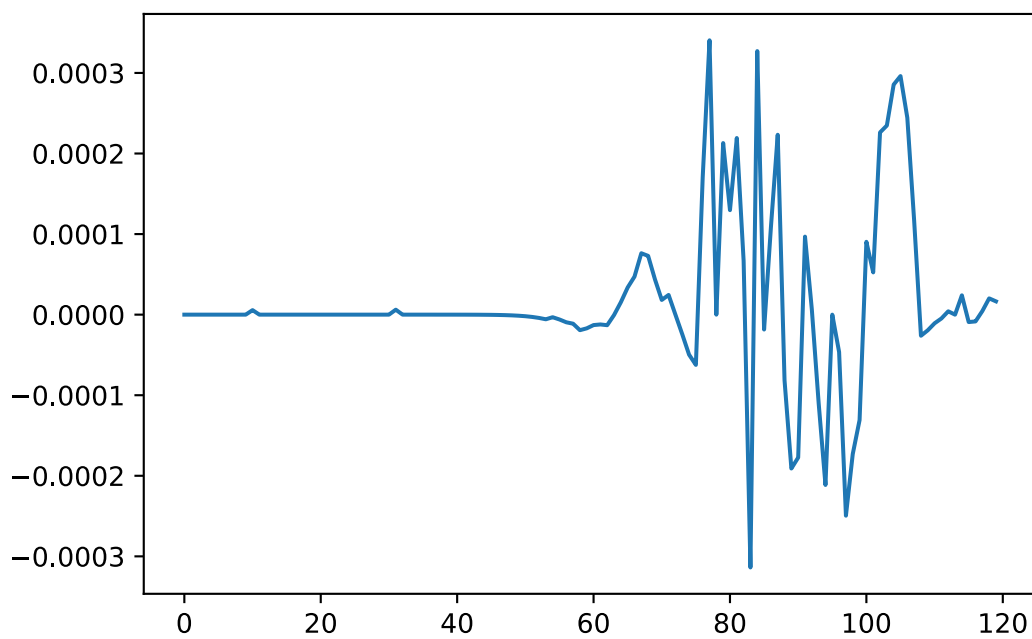
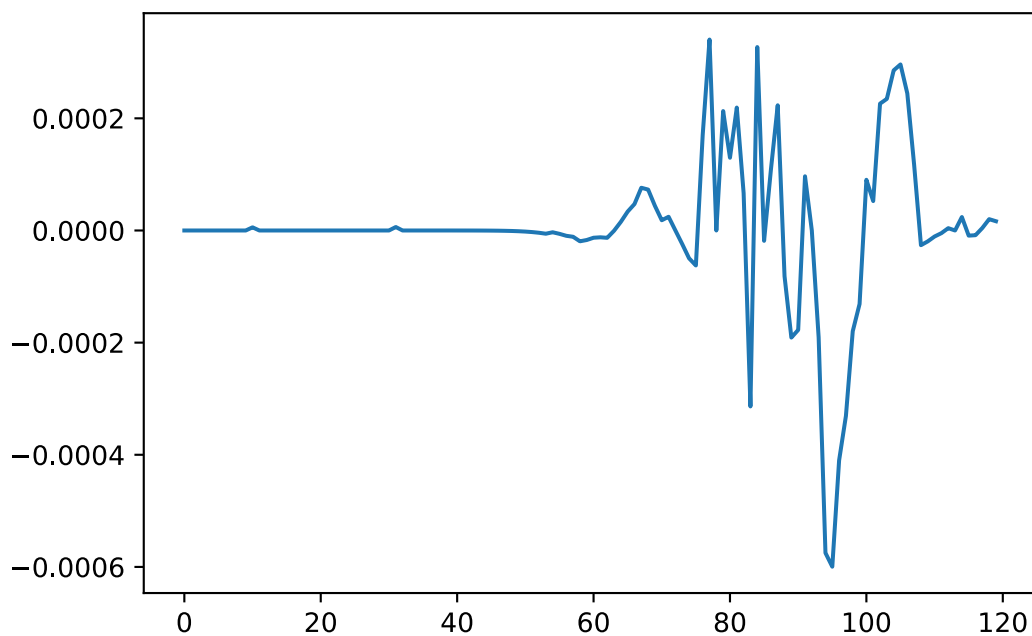
```
[18]: f = extractPartByYear(prenoms, ['MATTHIEU'])[0]
      dico = init_dictionary()

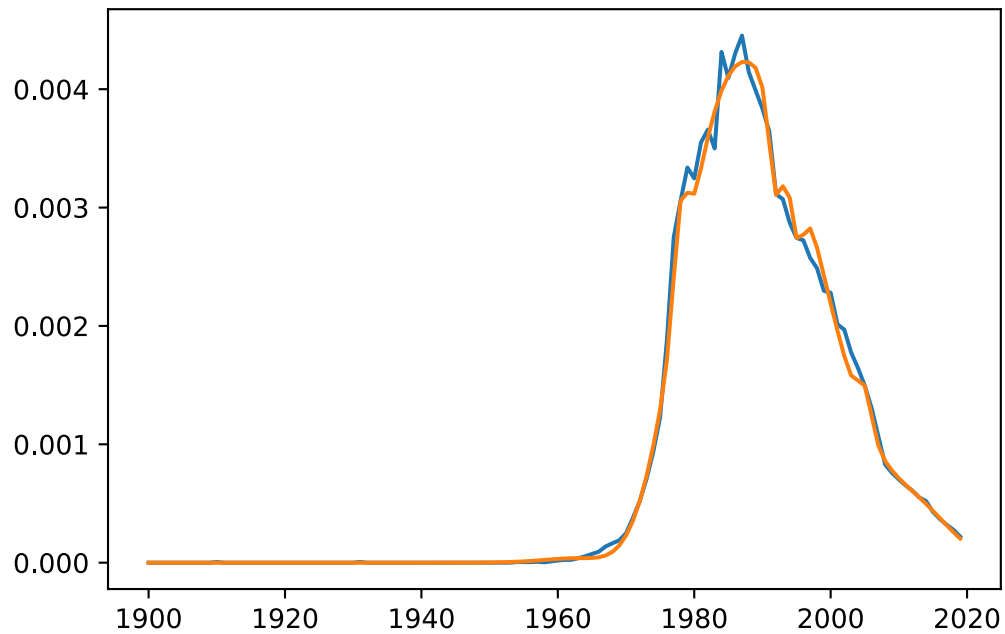
      MP_i, MP_c = matching_pursuit(f, dico, it_limit=8)
      approx = np.zeros(120)
      for i in range(len(MP_i)):
          approx += dico[MP_i[i]]*MP_c[i]
      X = [x for x in range(1900,2020)]
      plt.plot(X, f)
      plt.plot(X,approx)
      plt.show()
      MP_i, MP_c, "Absolute error : " + str(np.abs(f-approx).sum()), "Max local error :
      → " + str(np.abs(f-approx).max())
```









```
[18]: (array([329, 233, 202, 192, 78, 92, 95, 105]),
      array([ 0.09999978,  0.0041022 ,  0.00671496, -0.00620262,  0.00183386,
             -0.00193827, -0.00150288,  0.00074226]),
      'Absolute error : 0.005167055979372113',
      'Max local error : 0.0003404958079664066')
```

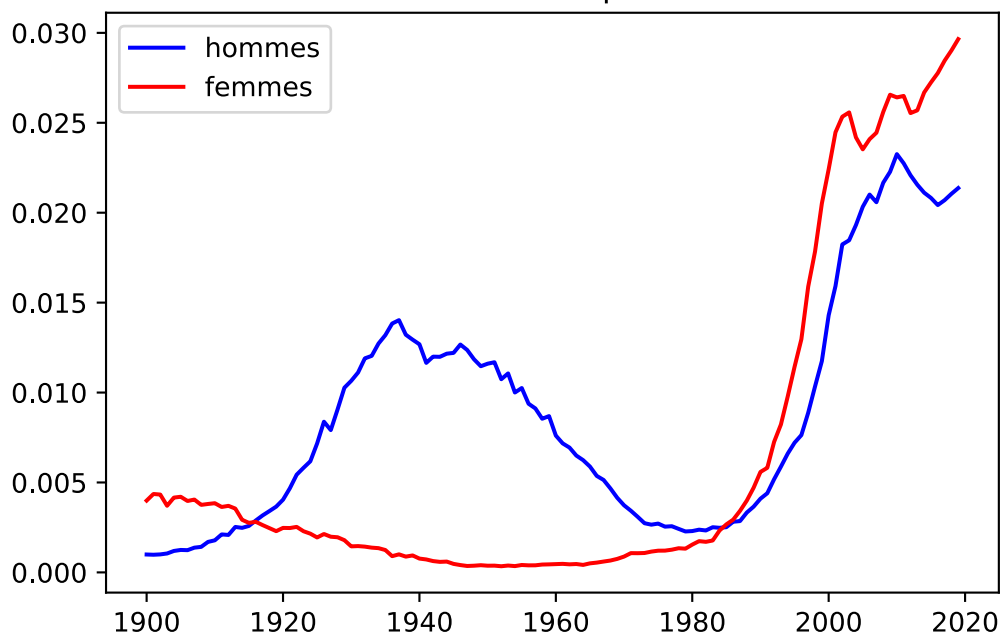
5 D'autres évolutions intéressantes

5.1 La longueur des prénoms

```
[11]: def pickNamesByLength(database, n) :  
    L = []  
    for name in database['preusuel']:  
        if len(str(name))==n and not(name in L):  
            L.append(name)  
    return L  
  
pitits = pickNamesByLength(prenoms, 3)  
#plotPartByYearList(prenoms, pitits)
```

Le prénom ABA a été donné 7214.0 fois à des hommes pour des années indéfinies
Le prénom ABA a été donné 5976.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés ABA chaque année entre 1900 et 2020.



```
[12]: prenoms['longueur']= [len(str(prenoms.iloc[i][1])) for i in range(652056)]  
prenoms
```

```
[12]:
```

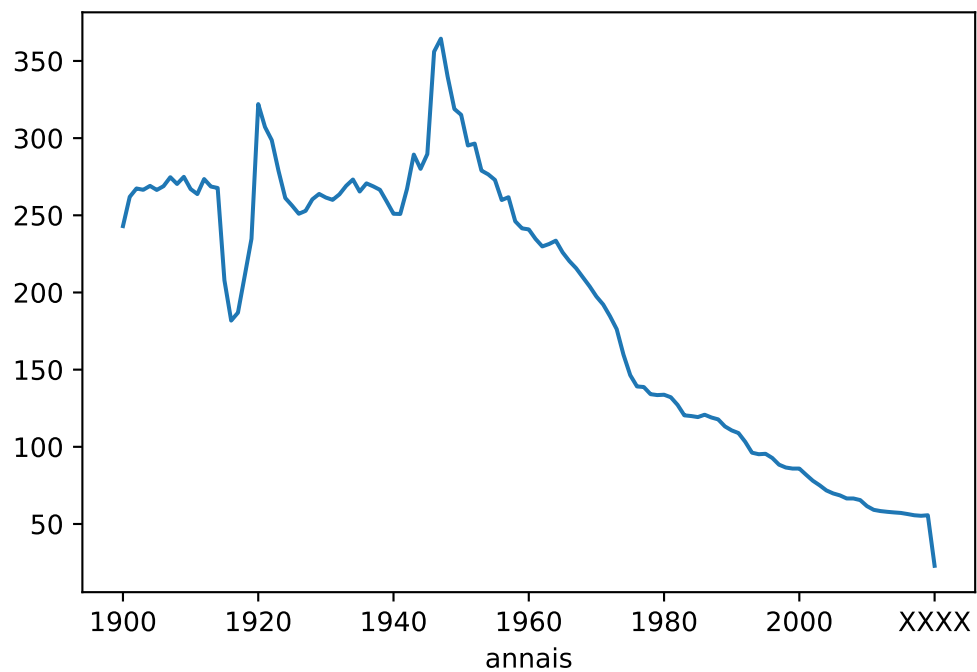
	sexe	preusuel	annais	nombre	longueur
0	1	_PRENOMS_RARES	1900	1250	14
1	1	_PRENOMS_RARES	1901	1340	14
2	1	_PRENOMS_RARES	1902	1329	14
3	1	_PRENOMS_RARES	1903	1285	14

4	1	_PRENOMS_RARES	1904	1427	14
...
652051	2	ZYNEB	2016	6	5
652052	2	ZYNEB	2017	6	5
652053	2	ZYNEB	2018	5	5
652054	2	ZYNEB	2019	7	5
652055	2	ZYNEB	XXXX	19	5

[652056 rows x 5 columns]

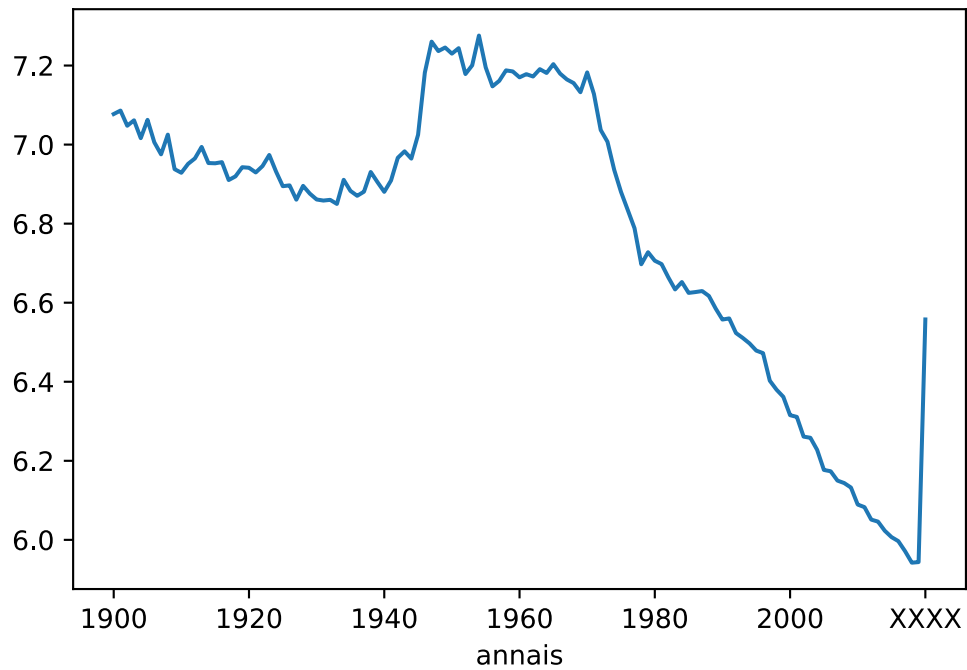
```
[23]: prenoms.groupby('annais').nombre.mean().plot()
```

```
[23]: <AxesSubplot:xlabel='annais'>
```



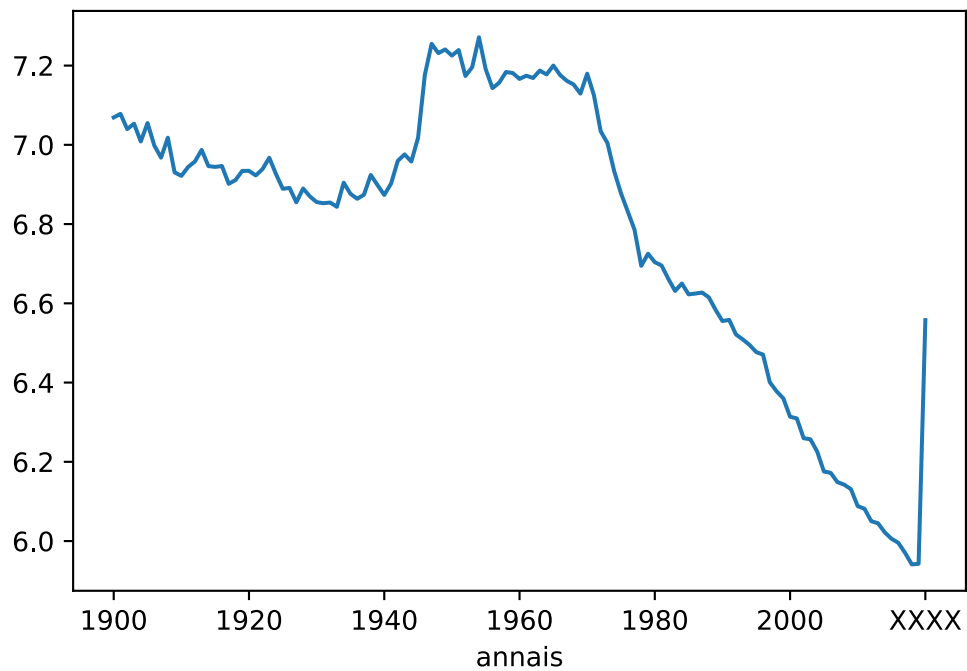
```
[42]: prenoms.groupby('annais').longueur.mean().plot()
```

```
[42]: <AxesSubplot:xlabel='annais'>
```



```
[46]: prenomless = prenommes[prenommes['preusuel'] != "_PRENOMS_RARES"]
      prenomless.groupby('annais').longueur.mean().plot()
```

```
[46]: <AxesSubplot:xlabel='annais'>
```



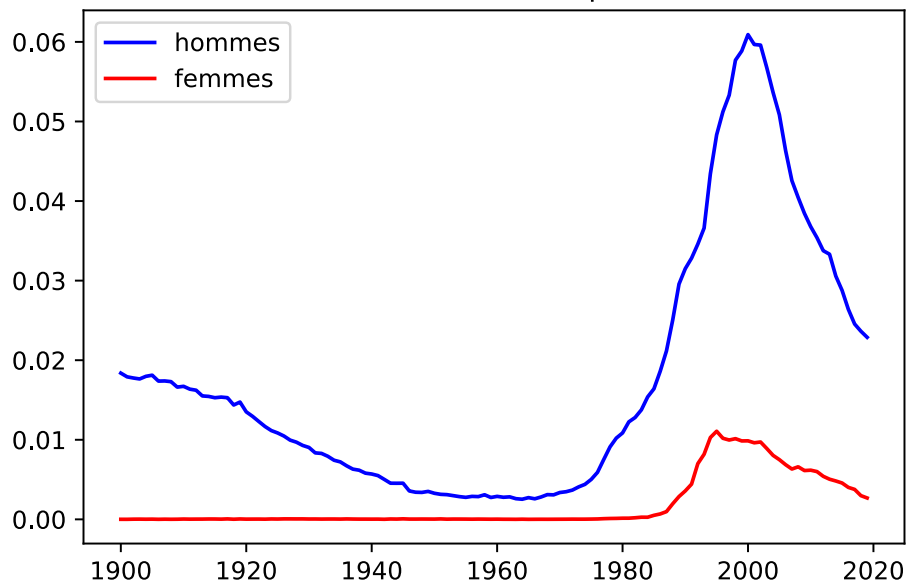
5.2 D'autres exemples

```
[58]: classe = ["ARTHUR", "BAPTISTE", "CHARLES", "CLÉMENT", "CLEMENT", "DYLAN",  
→ "ENZO", "ÉTIENNE", "ETIENNE", "FÉLIX", "FLORENT", "HUGO", "IGOR", "ISSEÏNIE",  
→ "LUDMILA", "MANON", "MATTHIEU", "PHILÉMON", "PHILEMON", "QUENTIN", "THÉO",  
→ "THEO"]  
plotPartByYearList(prenoms, classe)
```

Les prénoms ARTHUR,... ont été donnés 269.0 fois à des hommes pour des années indéfinies

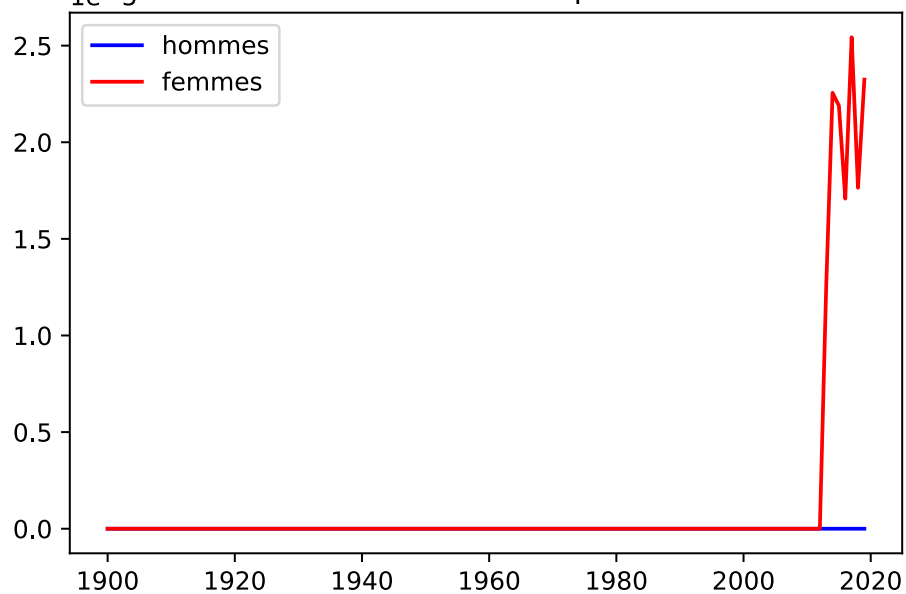
Les prénoms ARTHUR,... ont été donnés 339.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés ARTHUR,... chaque année entre 1900 et 2020.



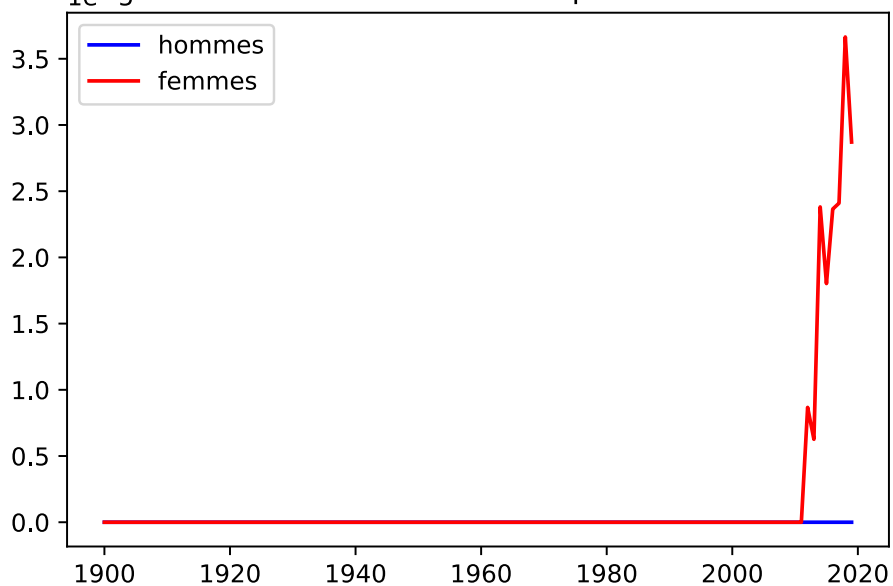
```
[64]: got = ["KHALEESI", "DAENERYS", "JON", "SANSA", "ARYA"]  
for prenom in got :  
    plotPartByYearList(prenoms, prenom)  
plt.show()
```

Nombre de bébés nommés KHALEESI chaque année entre 1900 et 2020.



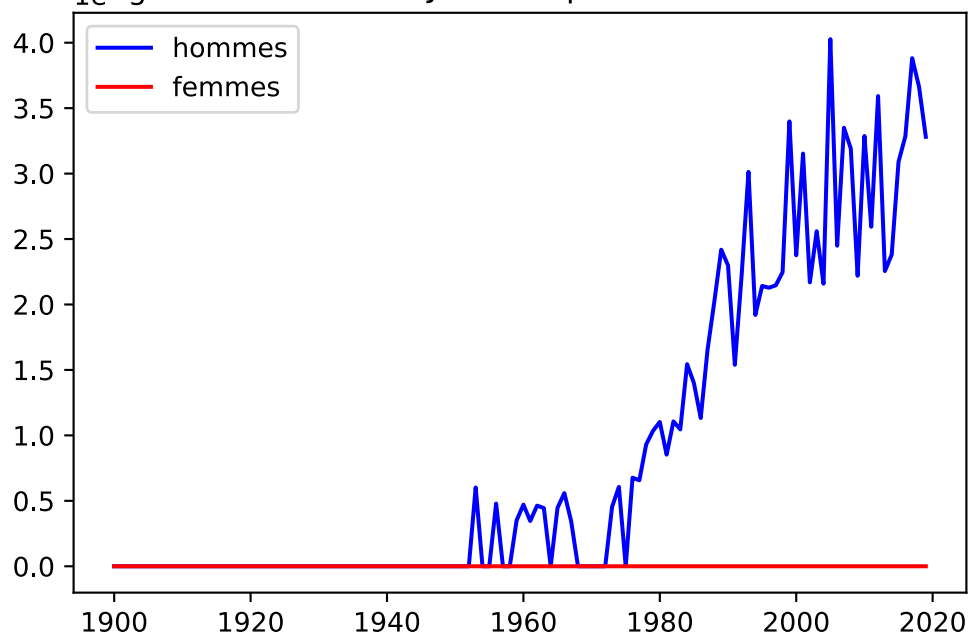
Le prénom DAENERYS a été donné 3.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés DAENERYS chaque année entre 1900 et 2020.



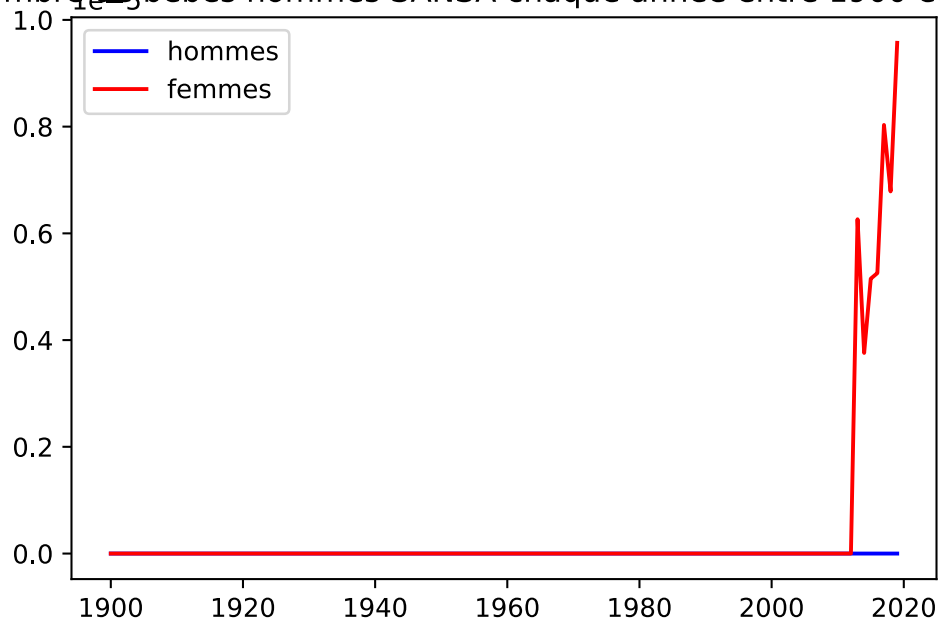
Le prénom JON a été donné 10.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés JON chaque année entre 1900 et 2020.



Le prénom SANSA a été donné 1.0 fois à des femmes pour des années indéfinies

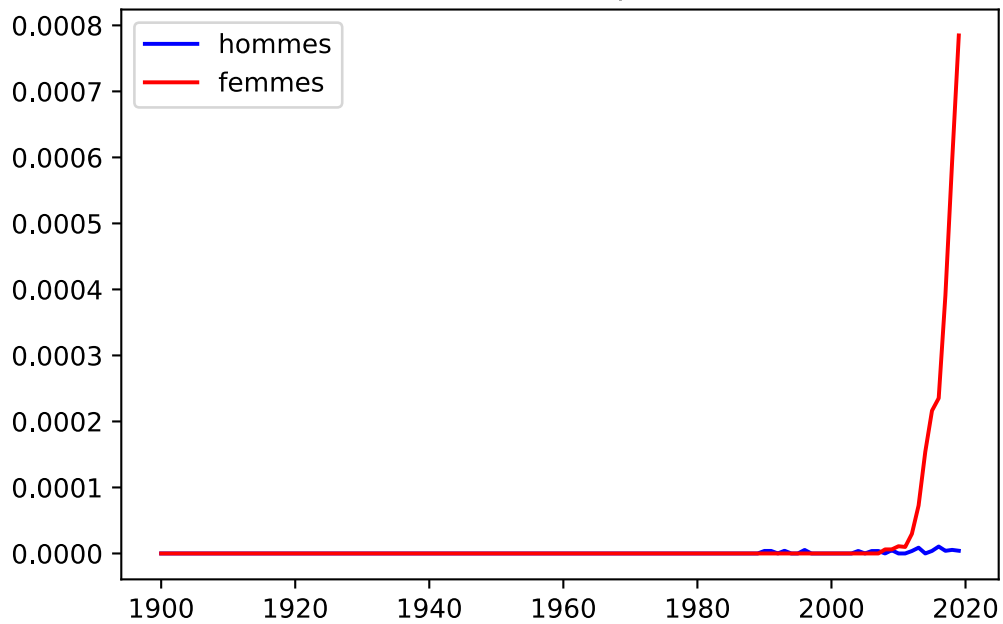
Nombre de bébés nommés SANSA chaque année entre 1900 et 2020.



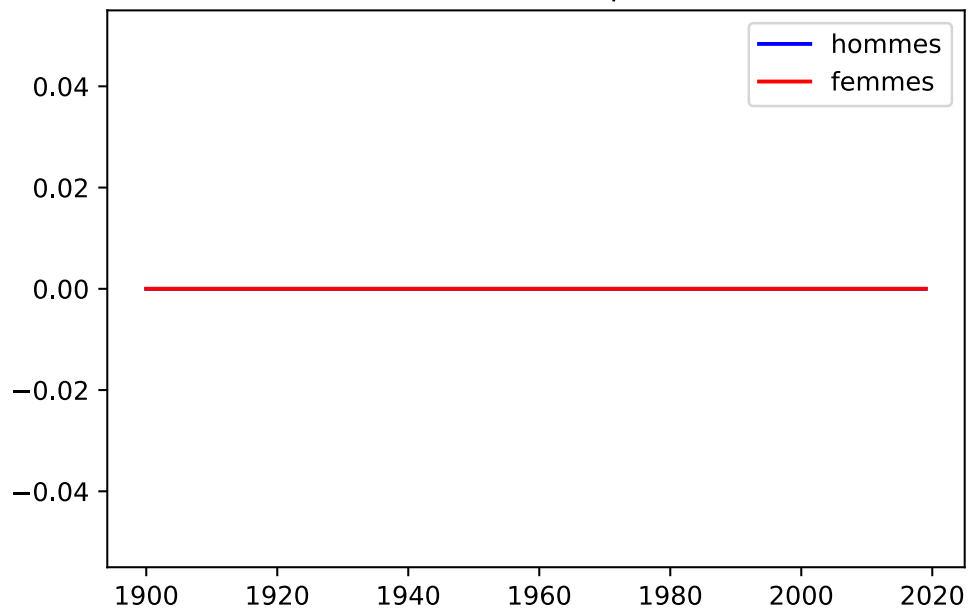
Le prénom ARYA a été donné 26.0 fois à des hommes pour des années indéfinies

Le prénom ARYA a été donné 22.0 fois à des femmes pour des années indéfinies

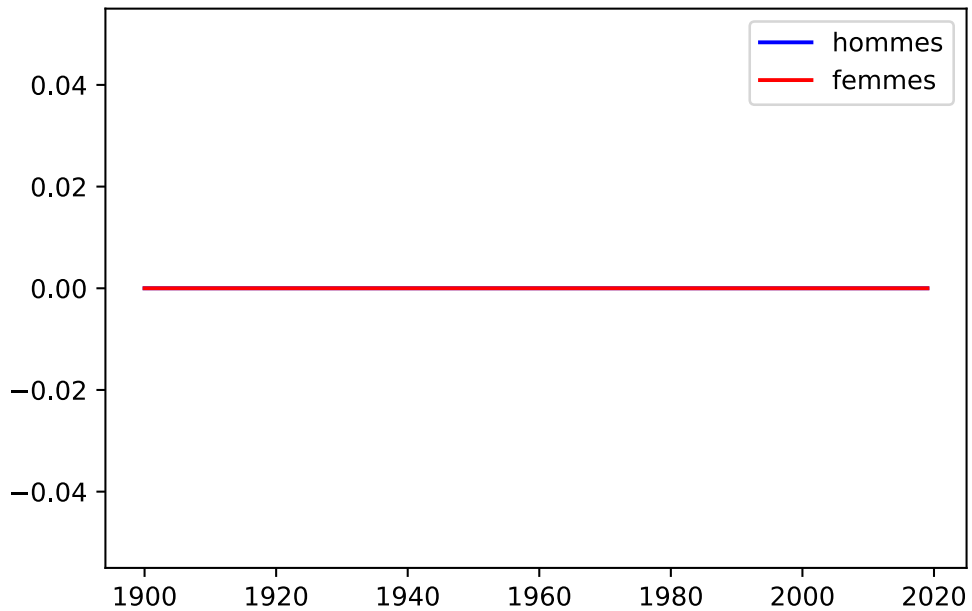
Nombre de bébés nommés ARYA chaque année entre 1900 et 2020.



Nombre de bébés nommés TYRION chaque année entre 1900 et 2020.



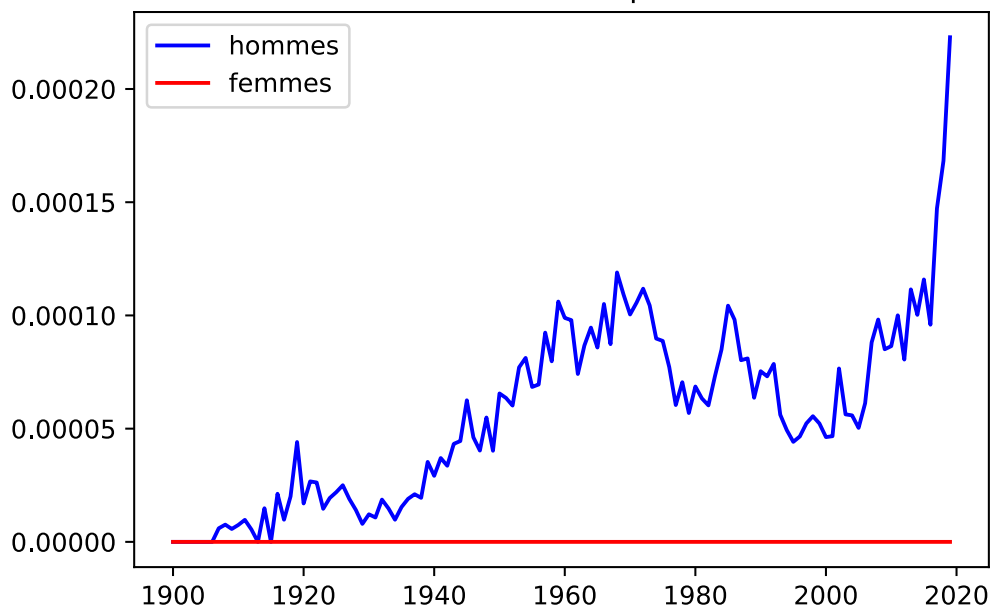
Nombre de bébés nommés CERSEI chaque année entre 1900 et 2020.



```
[65]: hp = ["HARRY", "HERMIONE", "RON", "LUNA", "GINNY"]
for prenom in hp :
    plotPartByYearList(prenoms, prenom)
plt.show()
```

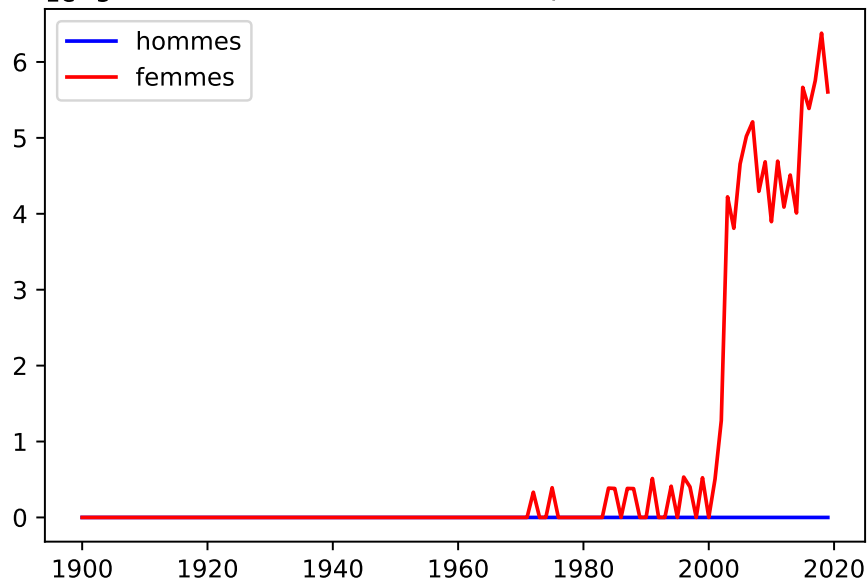
Le prénom HARRY a été donné 13.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés HARRY chaque année entre 1900 et 2020.



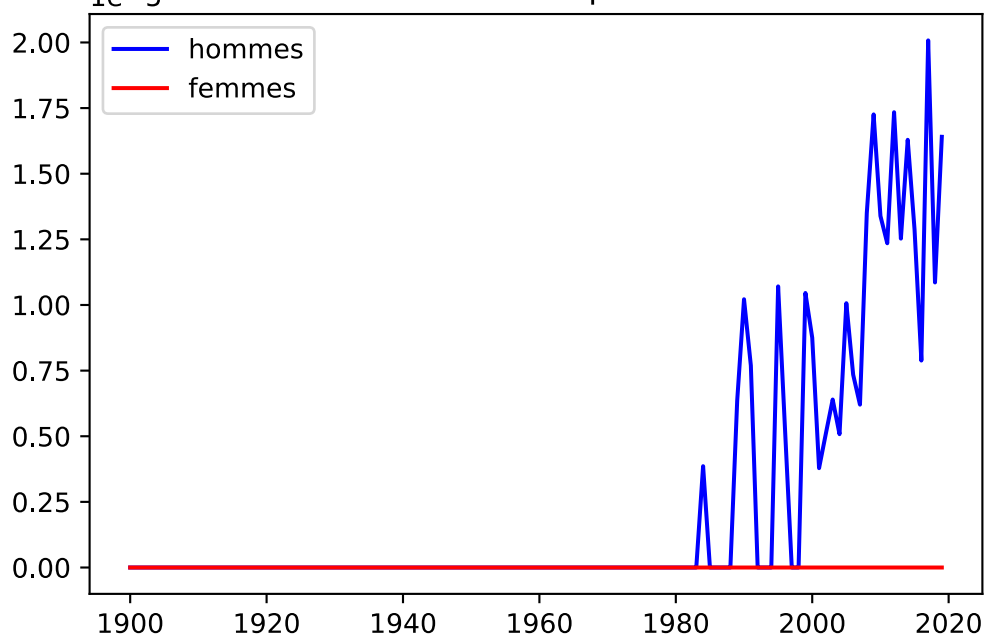
Le prénom HERMIONE a été donné 26.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés HERMIONE chaque année entre 1900 et 2020.



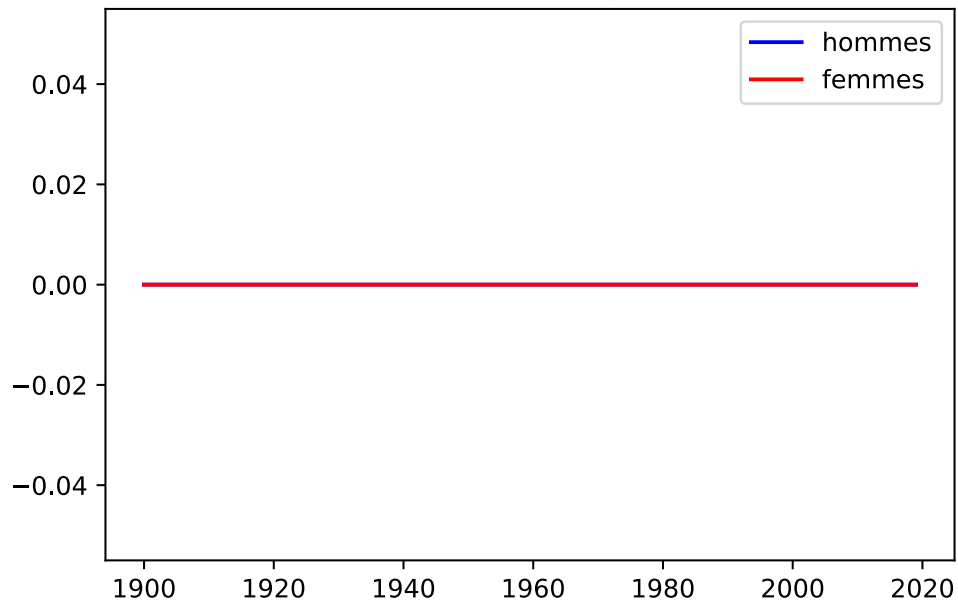
Le prénom RON a été donné 25.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés RON chaque année entre 1900 et 2020.



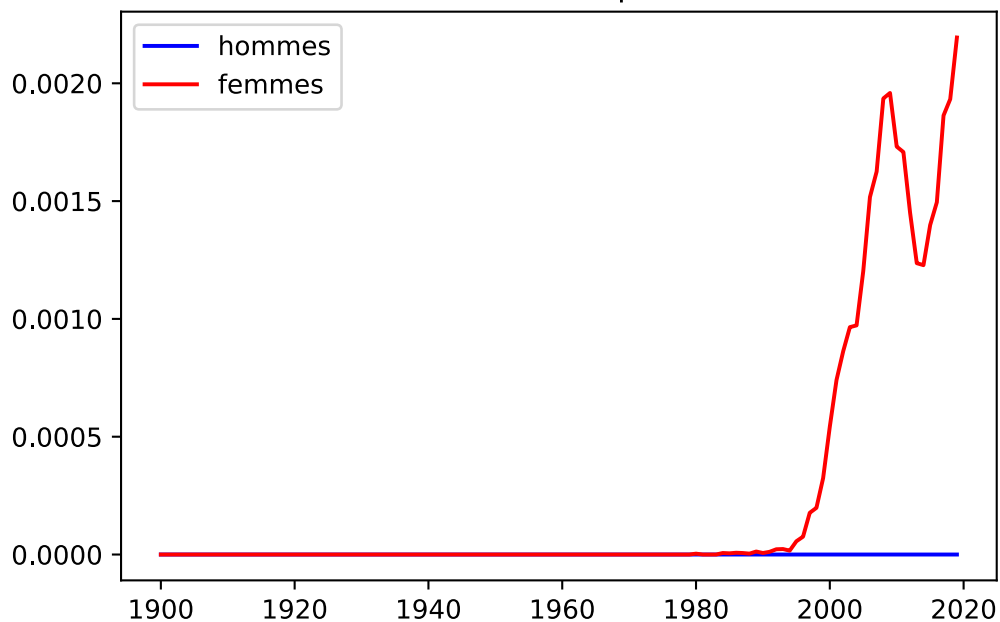
Le prénom NEVILLE a été donné 33.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés NEVILLE chaque année entre 1900 et 2020.



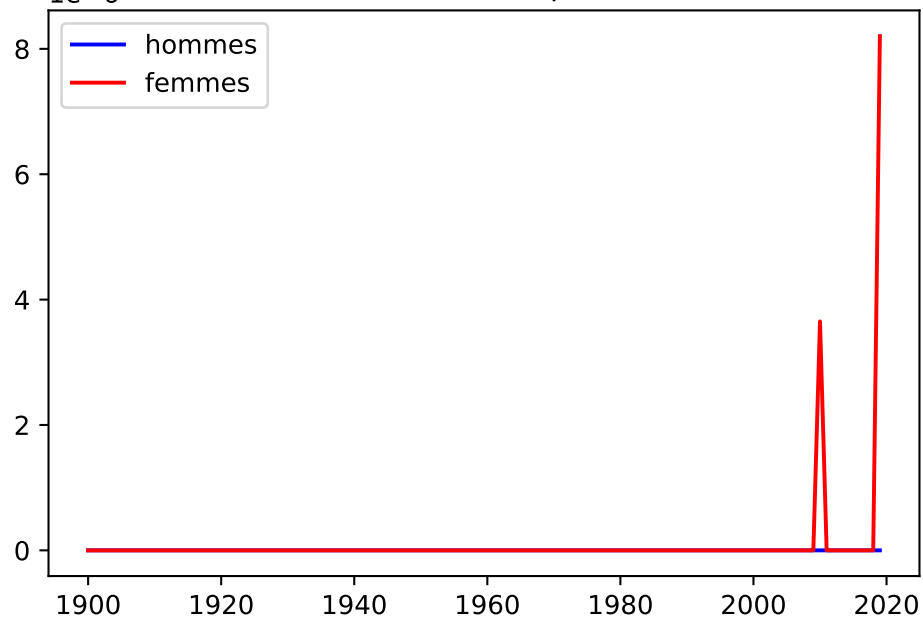
Le prénom LUNA a été donné 23.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés LUNA chaque année entre 1900 et 2020.

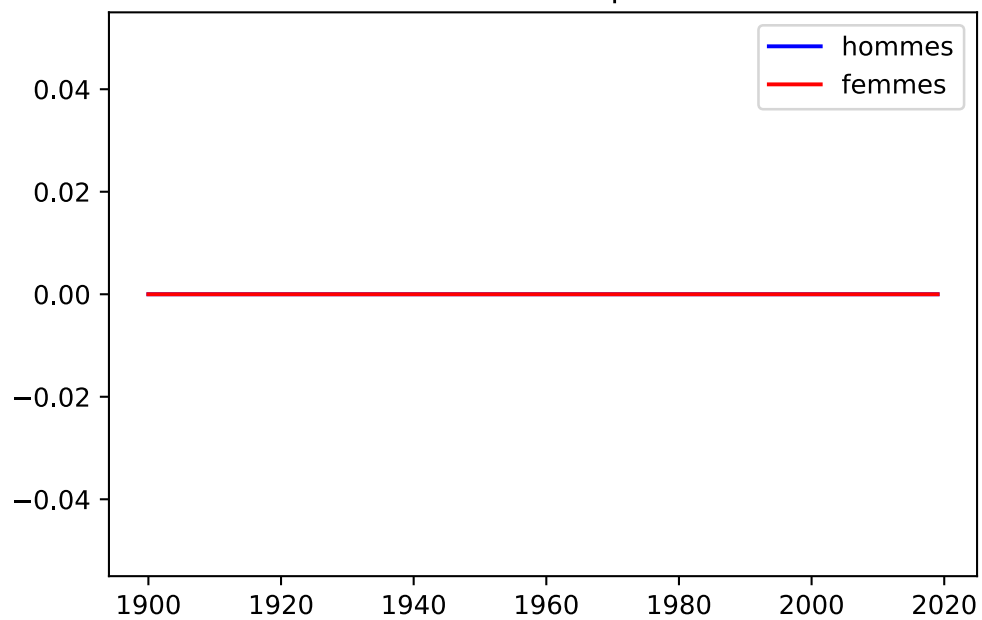


Le prénom GINNY a été donné 24.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés GINNY chaque année entre 1900 et 2020.



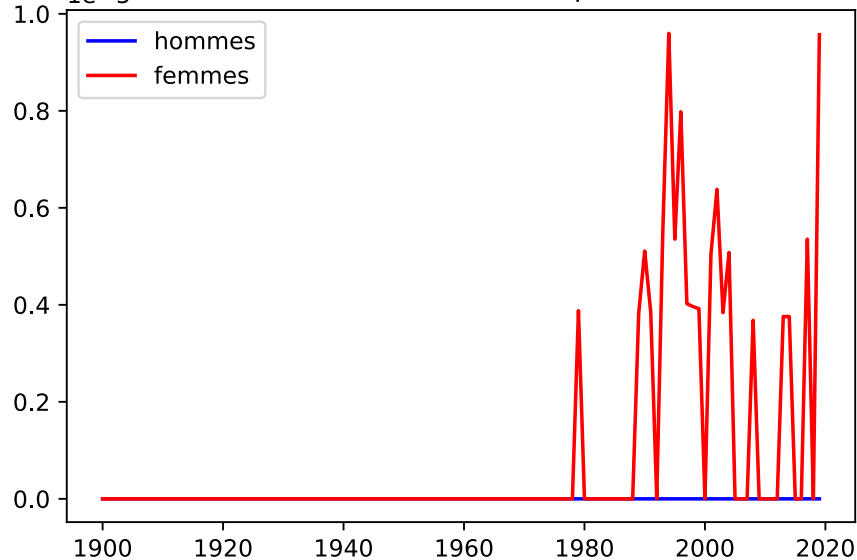
Nombre de bébés nommés ALBUS chaque année entre 1900 et 2020.




```
[67]: lotr = [["GALADRIEL", "GALADRIELLE"], ["ARWEN", "ARWENN"], ["EOWYN", "ÉOWYN"], ["BEREN"]]
for prenom in lotr :
    plotPartByYearList(prenoms, prenom)
plt.show()
```

Les prénoms GALADRIEL GALADRIELLE ont été donnés 58.0 fois à des femmes pour des années indéfinies

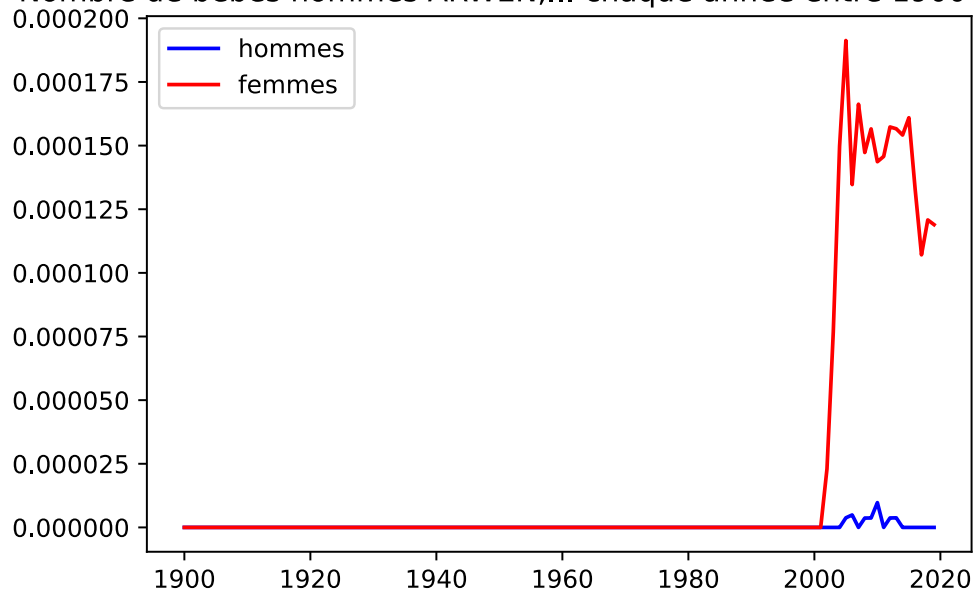
Nombre de bébés nommés GALADRIEL,... chaque année entre 1900 et 2020.



Les prénoms ARWEN ARWENN ont été donnés 18.0 fois à des hommes pour des années indéfinies

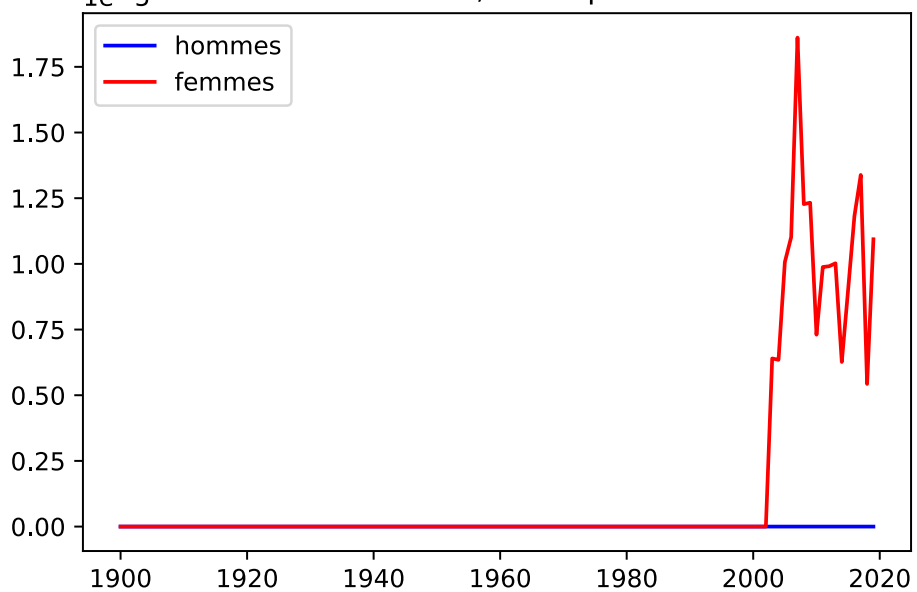
Les prénoms ARWEN ARWENN ont été donnés 22.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés ARWEN,... chaque année entre 1900 et 2020.



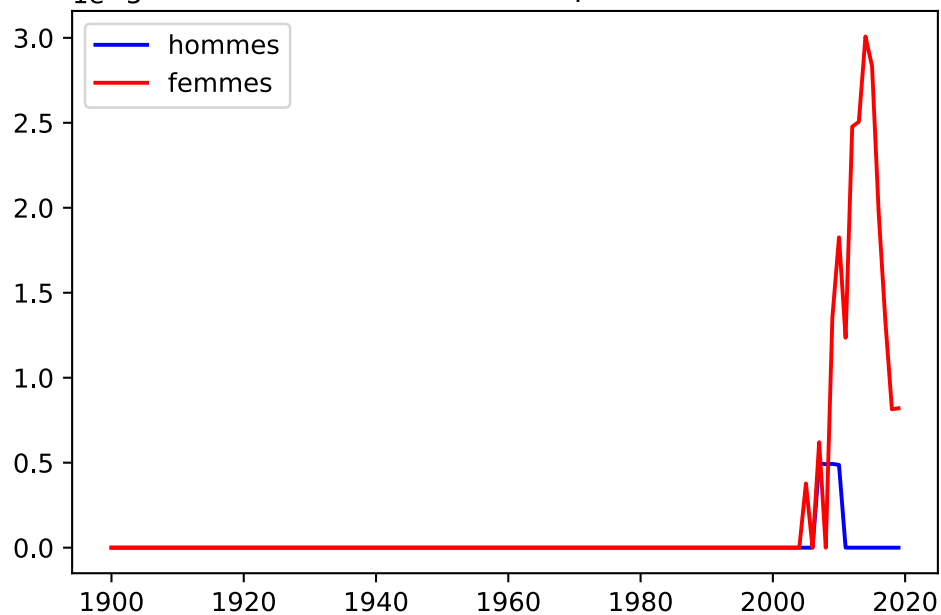
Les prénoms EOWYN ÉOWYN ont été donnés 12.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés EOWYN,... chaque année entre 1900 et 2020.



Le prénom BEREN a été donné 7.0 fois à des hommes pour des années indéfinies
 Le prénom BEREN a été donné 7.0 fois à des femmes pour des années indéfinies

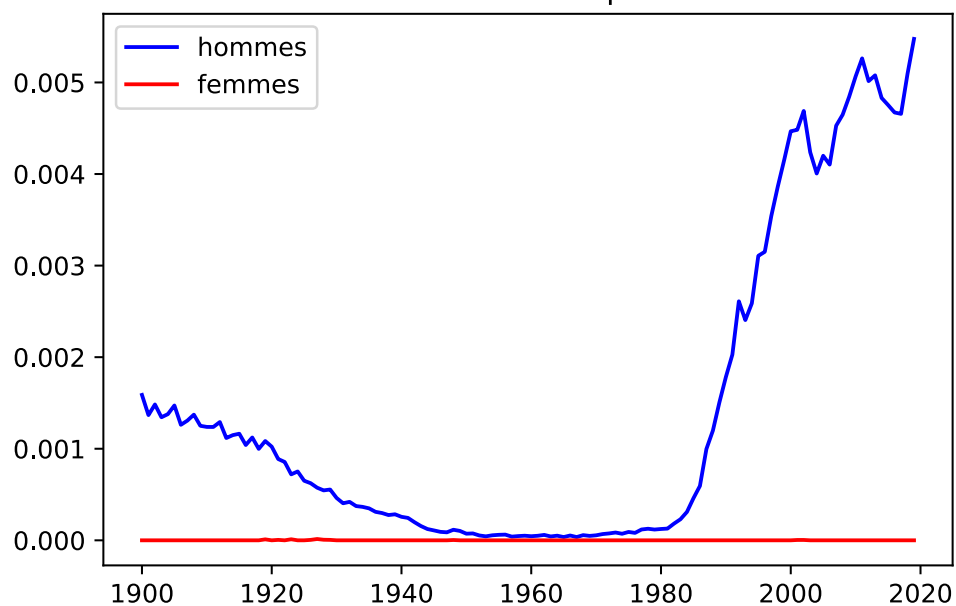
Nombre de bébés nommés BEREN chaque année entre 1900 et 2020.



```
[68]: legendeearthurienne = [{"ARTHUR"}, {"GUENIEVRE", "GUENIÈVRE"}, {"TRISTAN"},  
    ↳ {"ISEULT", "YSEULT"}, {"GALAAD"}, {"PERCEVAL"}, {"GAUVIN"}, {"MERLIN"},  
    ↳ {"MORGANE", "MORGAN"}, {"VIVIANE", "VIVIANNE"}, {"YVAIN", "IVAIN"}]  
for prenom in legendeearthurienne :  
    plotPartByYearList(prenoms, prenom)  
    plt.show()
```

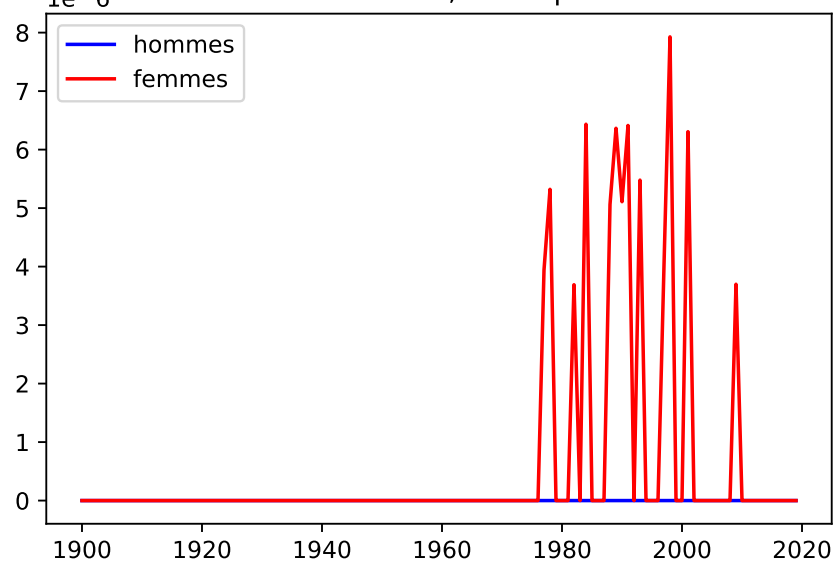
Le prénom ARTHUR a été donné 35.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés ARTHUR chaque année entre 1900 et 2020.



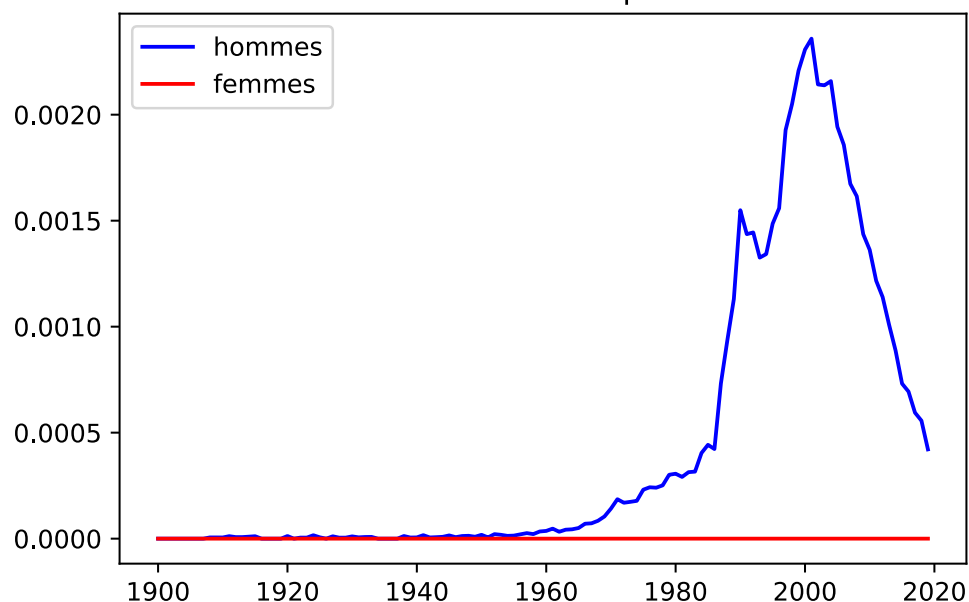
Les prénoms GUENIEVRE GUENIÈVRE ont été donnés 50.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés GUENIEVRE,... chaque année entre 1900 et 2020.



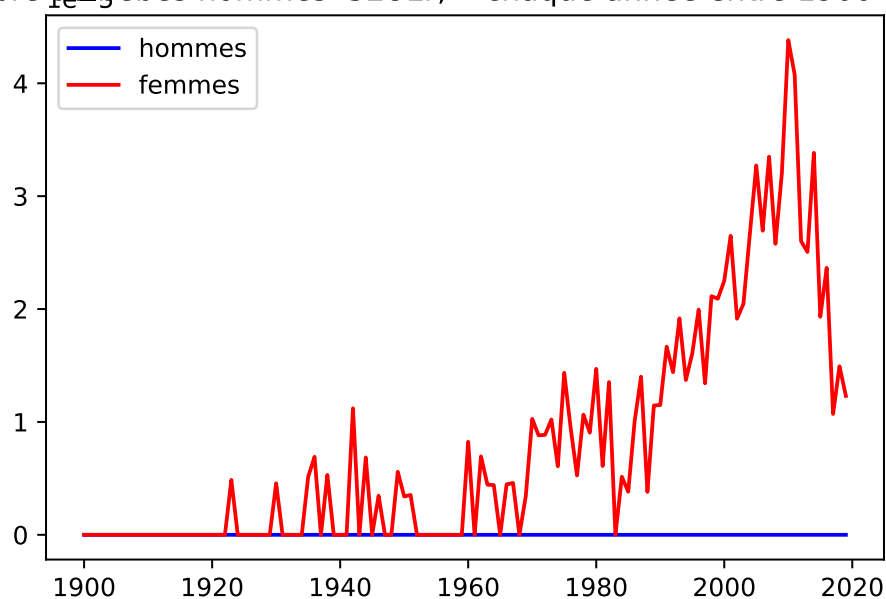
Le prénom TRISTAN a été donné 25.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés TRISTAN chaque année entre 1900 et 2020.



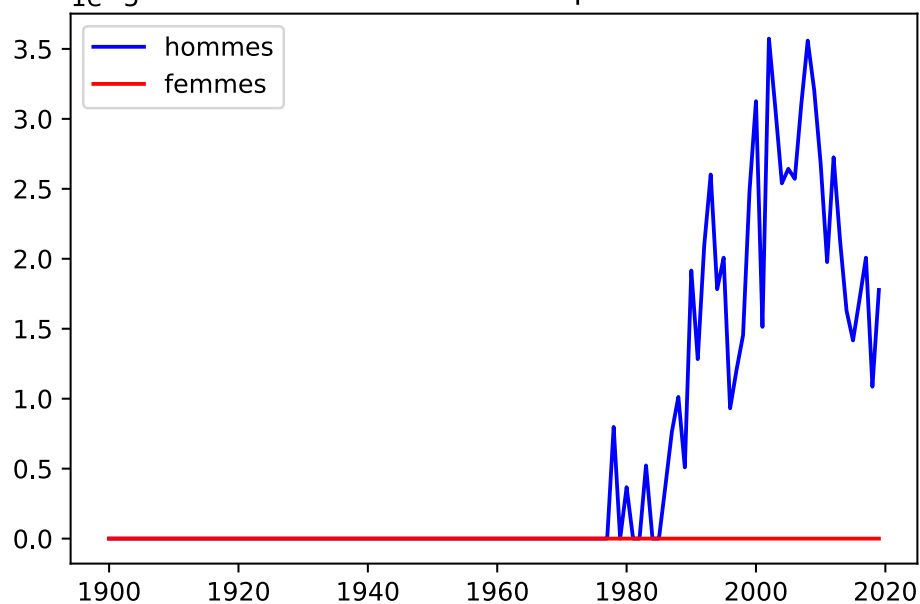
Les prénoms ISEULT YSEULT ont été donnés 91.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés ISEULT,... chaque année entre 1900 et 2020.



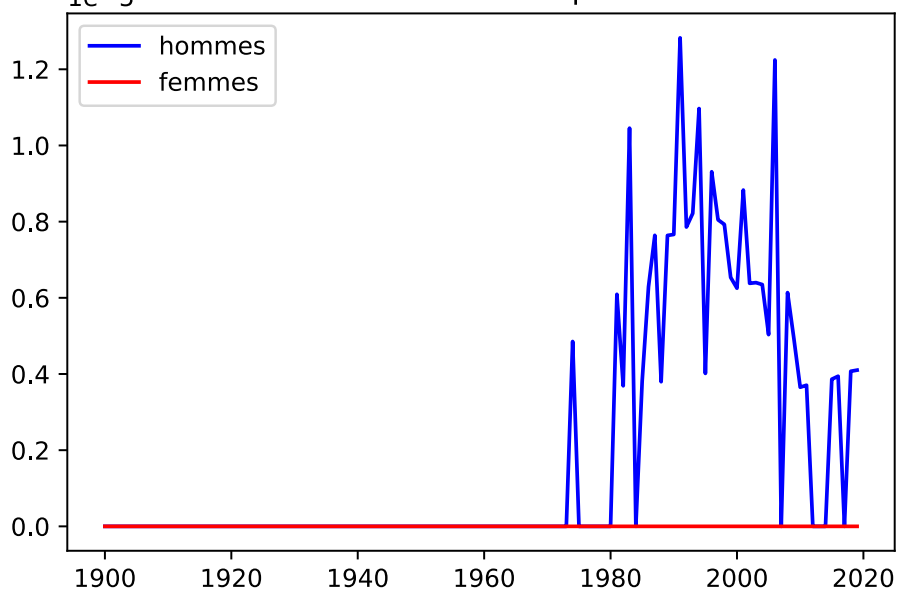
Le prénom GALAAD a été donné 11.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés GALAAD chaque année entre 1900 et 2020.



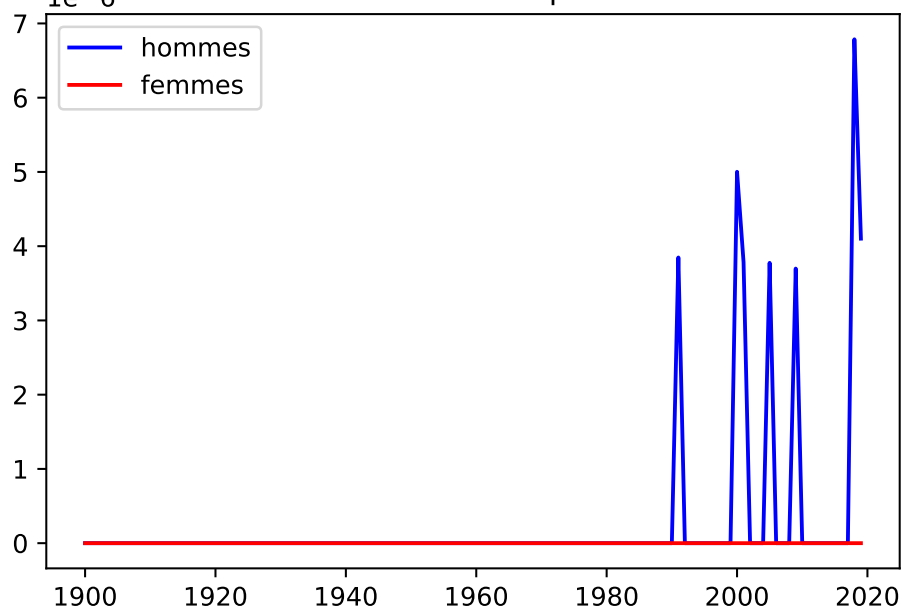
Le prénom PERCEVAL a été donné 25.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés PERCEVAL chaque année entre 1900 et 2020.



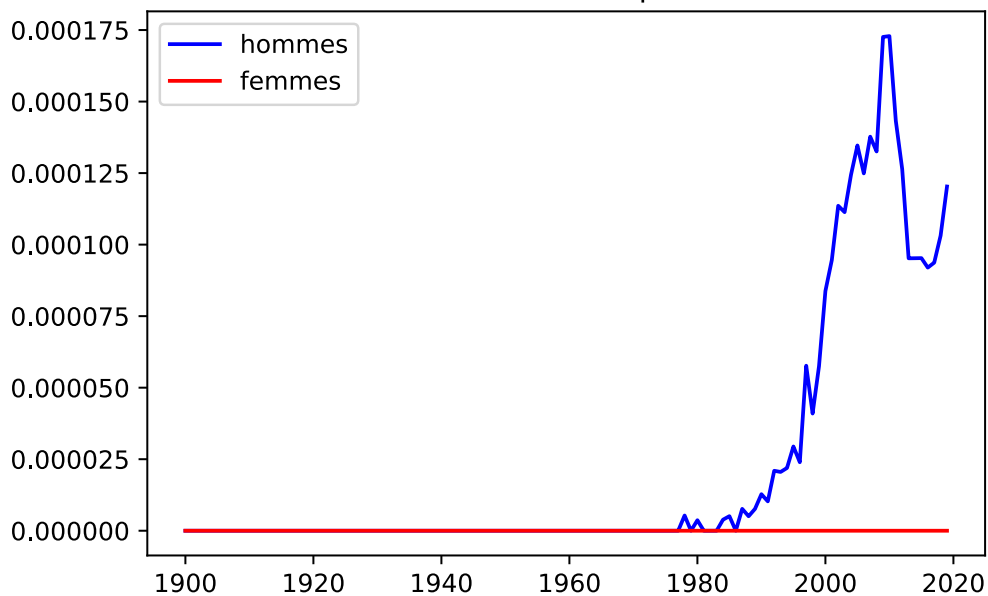
Le prénom GAUVIN a été donné 22.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés GAUVIN chaque année entre 1900 et 2020.



Le prénom MERLIN a été donné 22.0 fois à des hommes pour des années indéfinies

Nombre de bébés nommés MERLIN chaque année entre 1900 et 2020.

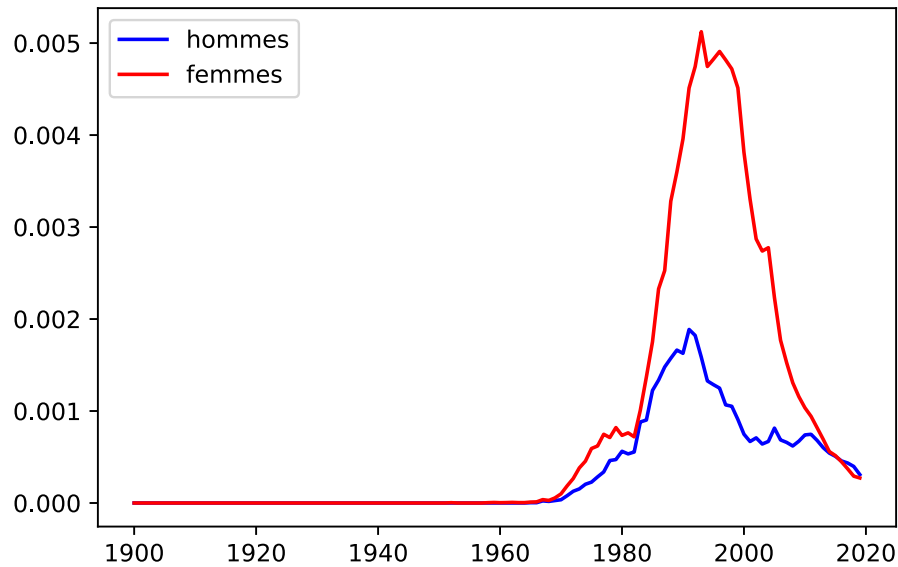


Les prénoms MORGANE MORGAN ont été donnés 33.0 fois à des hommes pour des années indéfinies

Les prénoms MORGANE MORGAN ont été donnés 20.0 fois à des femmes pour des années

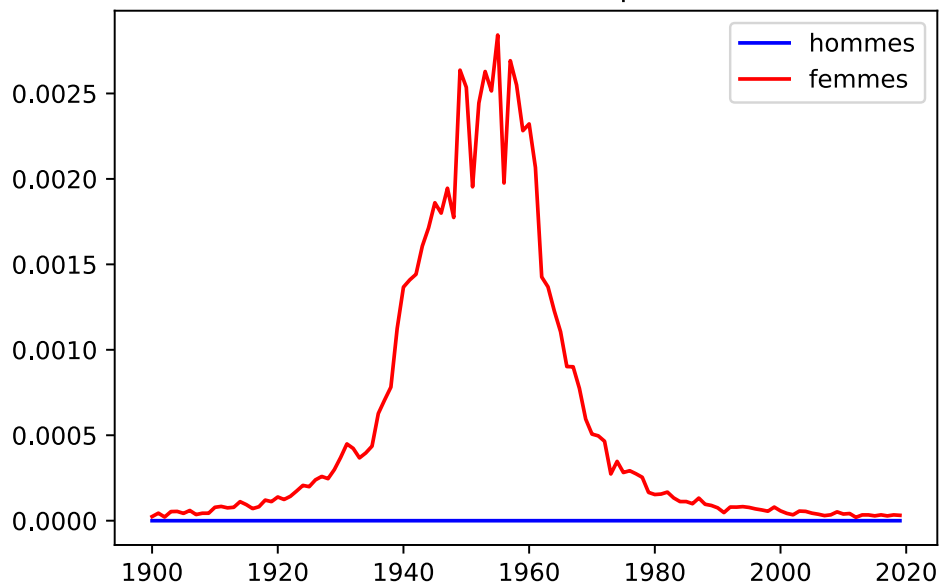
indéfinies

Nombre de bébés nommés MORGANE,... chaque année entre 1900 et 2020.



Les prénoms VIVIANE VIVIANNE ont été donnés 39.0 fois à des femmes pour des années indéfinies

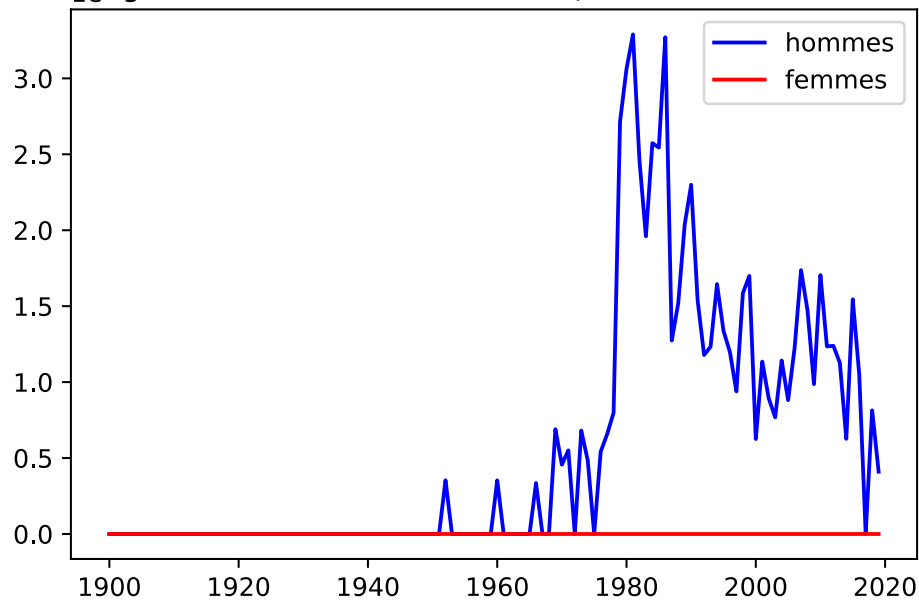
Nombre de bébés nommés VIVIANE,... chaque année entre 1900 et 2020.



Les prénoms YVAIN IVAIN ont été donnés 42.0 fois à des hommes pour des années

indéfinies

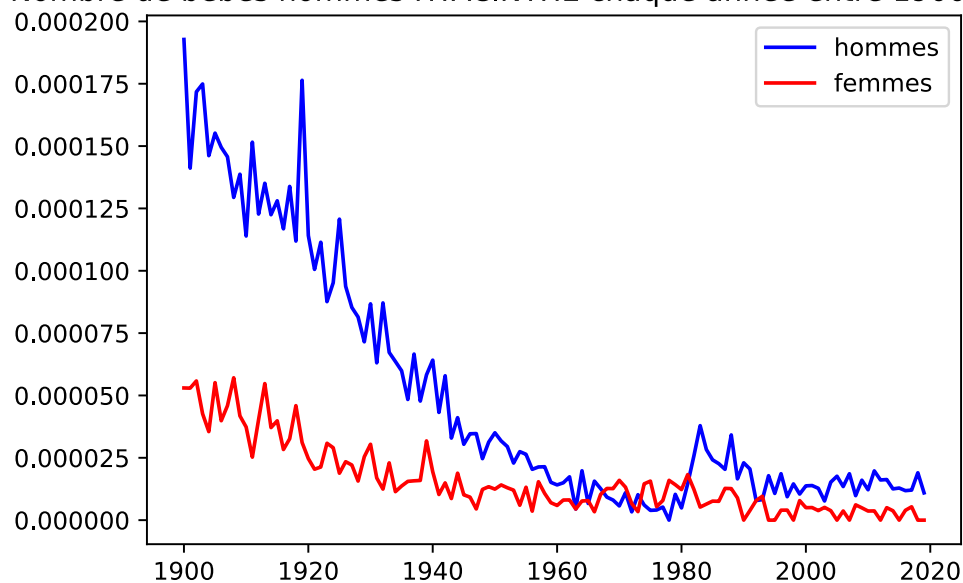
Nombre de bébés nommés YVAIN,... chaque année entre 1900 et 2020.



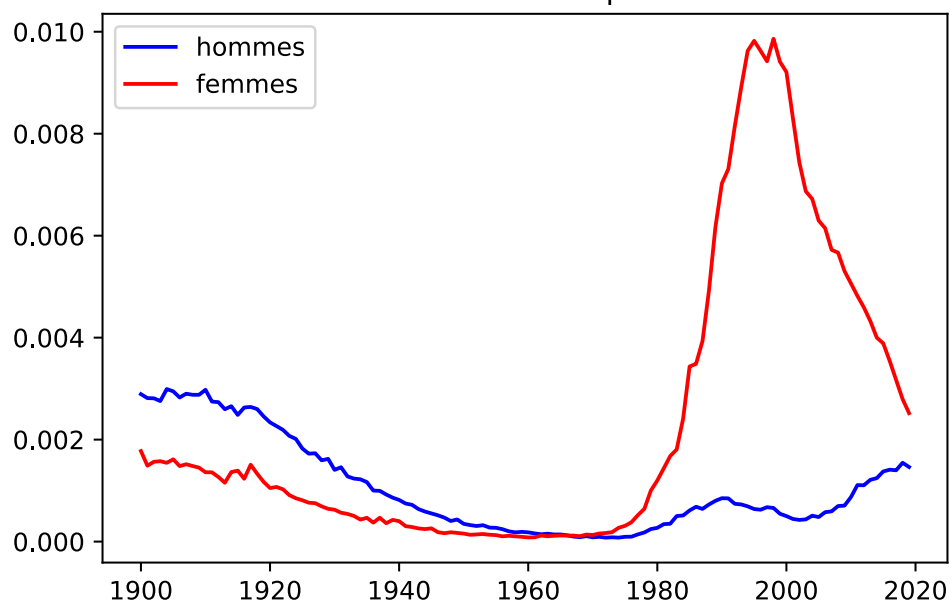
```
[70]: mixtes = [{"HYACINTHE"}, {"CAMILLE"}, {"SASHA", "SACHA"}, {"NOA", "NOAH"},  
→ {"LOU"}, {"EDEN"}, {"CHARLIE"}, {"ALIX"}, {"SWAN", "SWANN"}]  
for prenom in mixtes :  
    plotPartByYearList(prenoms, prenom)  
    plt.show()
```

Le prénom HYACINTHE a été donné 2.0 fois à des hommes pour des années indéfinies
Le prénom HYACINTHE a été donné 15.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés HYACINTHE chaque année entre 1900 et 2020.



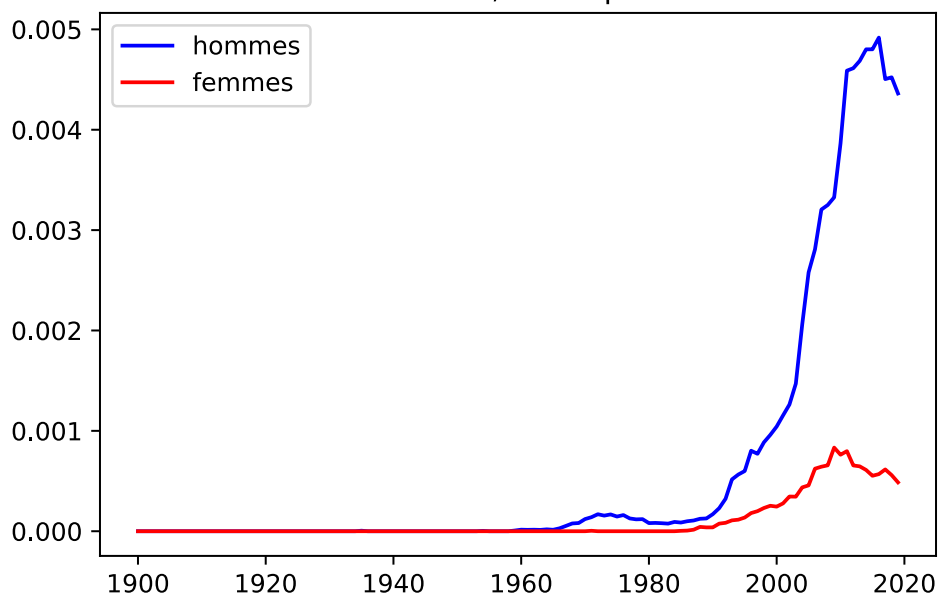
Nombre de bébés nommés CAMILLE chaque année entre 1900 et 2020.



Les prénoms SASHA SACHA ont été donnés 51.0 fois à des hommes pour des années indéfinies

Les prénoms SASHA SACHA ont été donnés 31.0 fois à des femmes pour des années indéfinies

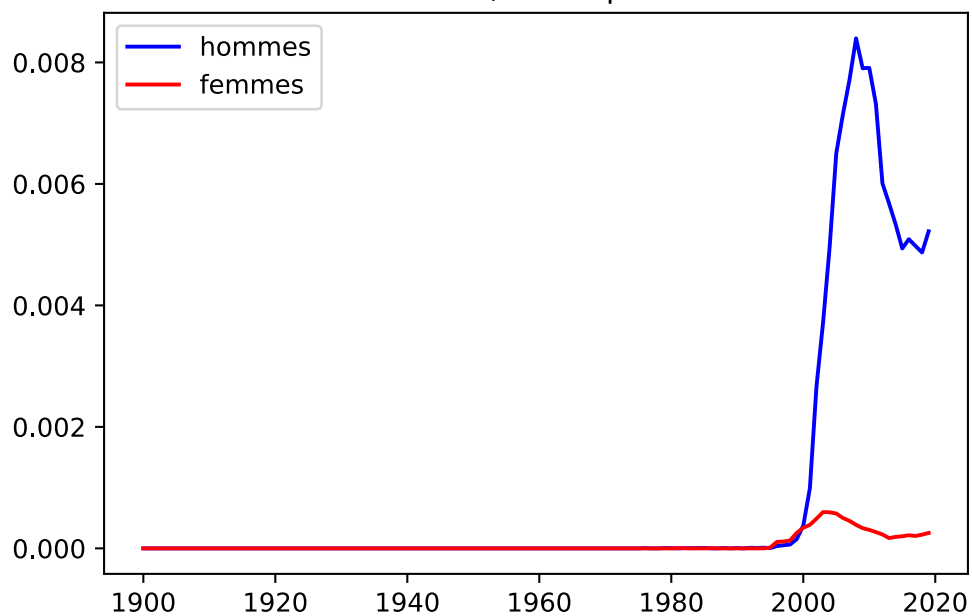
Nombre de bébés nommés SASHA,... chaque année entre 1900 et 2020.



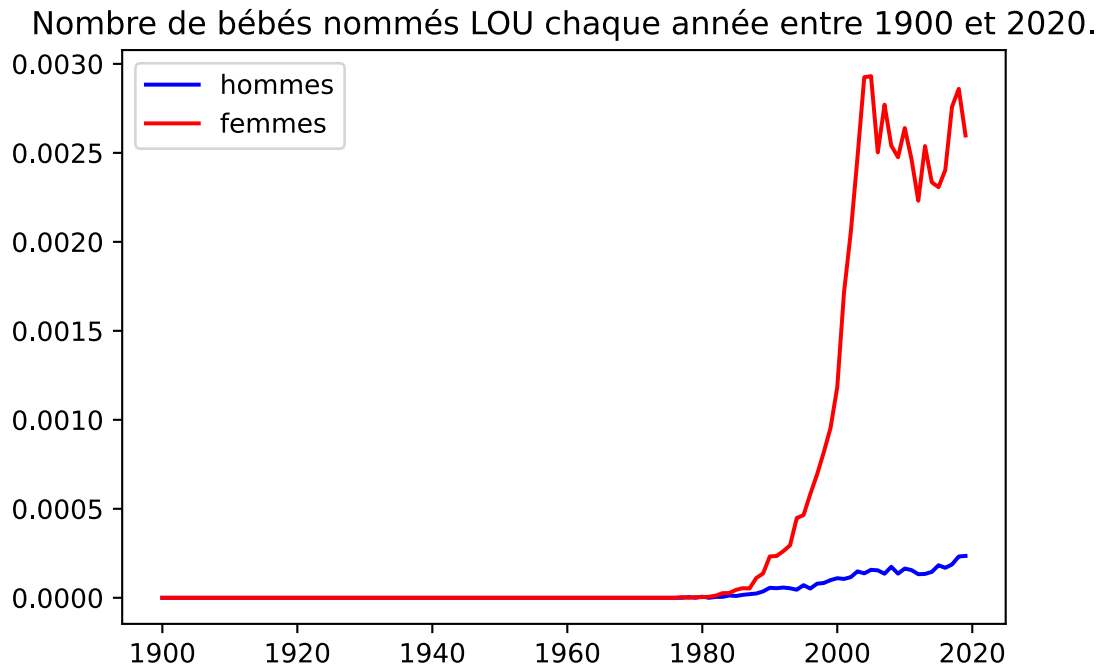
Les prénoms NOA NOAH ont été donnés 30.0 fois à des hommes pour des années indéfinies

Les prénoms NOA NOAH ont été donnés 37.0 fois à des femmes pour des années indéfinies

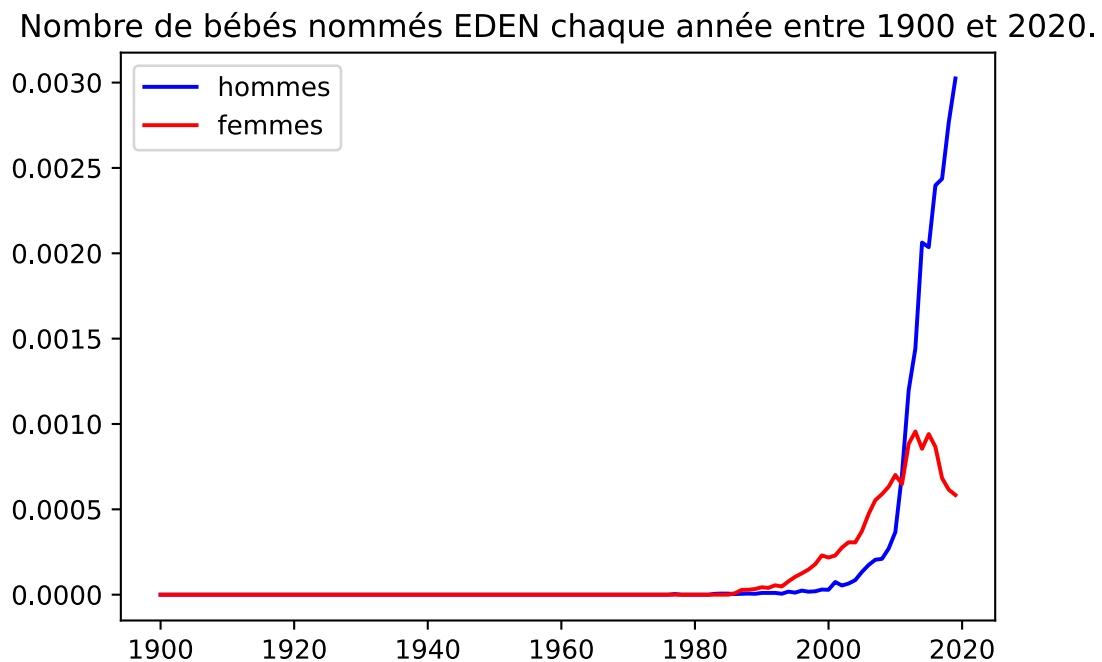
Nombre de bébés nommés NOA,... chaque année entre 1900 et 2020.



Le prénom LOU a été donné 6.0 fois à des hommes pour des années indéfinies
Le prénom LOU a été donné 23.0 fois à des femmes pour des années indéfinies

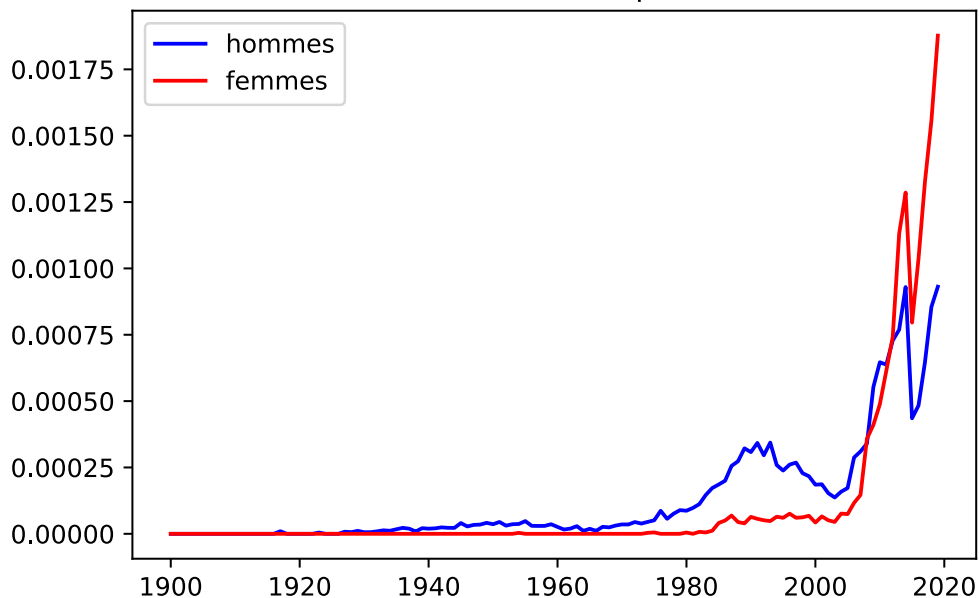


Le prénom EDEN a été donné 19.0 fois à des hommes pour des années indéfinies
Le prénom EDEN a été donné 11.0 fois à des femmes pour des années indéfinies

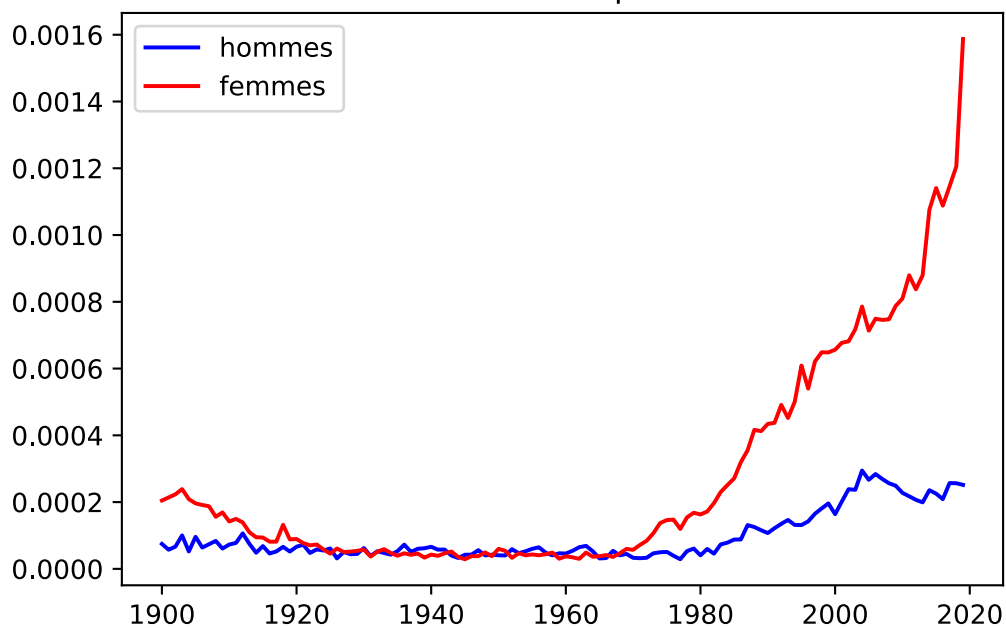


Le prénom CHARLIE a été donné 14.0 fois à des hommes pour des années indéfinies
Le prénom CHARLIE a été donné 27.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés CHARLIE chaque année entre 1900 et 2020.



Nombre de bébés nommés ALIX chaque année entre 1900 et 2020.



Les prénoms SWAN SWANN ont été donnés 17.0 fois à des hommes pour des années indéfinies

Les prénoms SWAN SWANN ont été donnés 32.0 fois à des femmes pour des années indéfinies

Nombre de bébés nommés SWAN,... chaque année entre 1900 et 2020.

