

# 12. 人口ピラミッド

プログラミング・データサイエンス I

2023/7/6

## 1 今日の目的

今日の目的

- Excel ファイルのデータを作図する
- 人口ピラミッド
- 佐賀県の将来人口予測

全国的に少子高齢化が進行しています。前回の授業では、佐賀県でも、人口全体が減少するなかで、高齢化が進んでいることを見ました。今日は、厚生労働省の国立社会保障・人口問題研究所が公開している将来人口予測のデータを使って、将来の人口ピラミッドを描いてみましょう。人口ピラミッドは、例えば男性を左、女性を右に、それぞれ年代別にその数を横向きの棒グラフで描くものです。かつては、若年層ほど人口が多いために、ピラミッド型をしていました。

<https://www.ipss.go.jp>

今日のサンプルプログラムは以下から取得してください。

<https://github.com/first-programming-saga/SagaFuturePopulation>

## 2 ピラミッドを描く

ピラミッドを描く

- 横向きグラフ
- 左向き棒グラフ

人口ピラミッドを描くには、左右に横向き棒グラフを描く必要があります。前々回に使用した九州各県の人口のデータを使って例を作りましょう。

残念ながら、標準では人口ピラミッドのような図を直接的に描く機能は、`matplotlib`

には無いようです。そこで、matplotlibにある `barh()` という横向き棒グラフを描くメソッドを活用することにします。もちろん、`Series` にも `barh()` というメソッドがありますから、Excel から行や列のデータを `Series` として取り出せば、すぐに横向き棒グラフを描くことができます。

人口ピラミッドの右側は、`barh()` メソッドを使うことができるため、特に工夫は不要でしょう。しかし、左側はどうでしょうか。左側に描きたいデータの符号を変えればよさそうです。左側に描きたいデータが `Series` ならば、それに `-1` を掛ければ、左向きの棒グラフになりそうです。しかし、それでは、横軸のラベルもマイナスの値になってしまいます。

それならば、横軸のラベルを、左側も正の値になるように変更すればよいでしょう。matplotlibの座標軸は、各目盛りのラベルを書き換えることができます。

#### ソースコード 2.1 九州各県の人口推移

```
1  ser = createData(2015)
2  ser2 = createData(1990) * (-1)
3  #作図の準備
4  fig, ax = plt.subplots(figsize = (15, 10), facecolor = 'w')
5  xmax = 6000
6  dx = 2000
7  ax.set_xlim(- xmax, xmax)
8  ax.set_xlabel('千人')
9  #0を中心に、左右に xmax まの座標軸
10 #左側にも正の値を表示
11 xticks = [x for x in range(- xmax, xmax + 1, dx)]
12 xlabel=[str(abs(x)) for x in xticks]
13 ax.set_xticks(xticks)
14 ax.set_xticklabels(xlabel)
15 #作図
16 ser.plot.barh(ax = ax)
17 ser2.plot.barh(ax = ax, color = 'red')
18 ax.text(5000, 5, '2015', ha = 'center')
19 ax.text(-5000, 5, '1990', ha = 'center')
20 ax.plot([0, 0], [-1, len(ser)], c = 'black')
21 plt.show()
```

`horizontalBar.ipynb`を開いてください。ソースコード 2.1 です。1 行目と 2 行目で、二つの列のデータを取り出しています。関数 `createData()` の内容はソースコード 2.2 に示します。その戻り値は `Series` です。1990 年に相当するデータは、`-1` を乗じて、左向きにしています。データは、15 行目と 16 行目で、`Series.barh()` を使って横向き

棒グラフとして作図しています。

11 行目から 13 行目が、横軸のラベルの再設定です。set\_xticks() は、横軸の目盛りを設定するメソッドです。最初の引数が目盛りラベルを描く値のリスト、二番目の引数が実際に描くラベルのリストです。今回の例では、リスト内包表記をしています。思い出してください。二番目の引数に渡しているリストが表している描くラベルが、絶対値になっていることに注意してください。

#### ソースコード 2.2 Excel から Series の取り出し

```
1 def createData(year:int) -> pandas.Series:
2     """
3     指定した年に相当する列のデータを返す
4     """
5     with pandas.ExcelFile('KyushuPopulation.xlsx') as f:
6         originalData = pandas.read_excel(f, index_col = 0)
7     return originalData[year]
```

### 3 シートから複数の DataFrame を取得

それでは、いよいよ佐賀県の人口予測のデータをみましょう。

<https://www.ipss.go.jp/pp-shicyoson/j/shicyoson18/3kekka/Municipalities/41.xls>

今回は、5 歳刻みのデータを利用して将来の人口ピラミッドを描きましょう。まず、上の URL にあるエクセルファイルを見ましょう。一枚のシートの中に 3 個の表が入っています。また、総数と年齢層別のデータもあります。幸い、3 個の表は、すべて同じ形式です。人口ピラミッドを描くために、男女のデータを切り出しましょう。

この作業自体は、以前にもやってきたことの繰り返しです。header でどの行から読み出すかを指定し、nrows で行の数を、skiprows で無視する行を指定するだけです。ソースコード 3.1 を見てください。

**課題 3.1** 男女のデータをそれぞれ取り出するようにソースコード 3.1 を変更し、実行結果を確かめなさい。

**課題 3.2** 対象としているエクセルファイルには、複数のシートがあります。最初

### ソースコード 3.1 シートから一部の表を取り出す

```
1 url = 'https://www.ipss.go.jp/pp-shicyoson/j/'\
2     + 'shicyoson18/3kekka/Municipalities/41.xls'
3 header = 3
4 nrows = 19
5 skiprows = [header + 1]
6 with pandas.ExcelFile(url) as f:
7     data = pandas.read_excel(f, header = header,
8                             index_col = 0, nrows = nrows, skiprows = skiprows)
9 data
```

が佐賀県全体を、残りが各市町のデータになっています。`read_excel()` メソッドに、`sheet_name` というパラメタに番号を指定すると、別のシートの情報を読み込むことを確かめなさい。0 が既定値で、最初のシートです。

## 4 佐賀県の将来人口予測ピラミッド

ここまでで、準備はできました。指定した年の人口予測ピラミッドを描きましょう。`sagaFuturePopulation.ipynb` を開いてください。

### ソースコード 4.1 表の読み込み

```
1 def getData(url:str, h:int, s = 0) -> pandas.DataFrame:
2     with pandas.ExcelFile(url) as f:
3         df = pandas.read_excel(f, header = h, sheet_name = s,
4                               index_col = 0, nrows = 19, skiprows = [h + 1])
5     df.index.name = '年齢層'
6     return df
```

ソースコード 4.1 は、ヘッダ行 `h` を指定して、`DataFrame` へ読み込む関数です。ヘッダ行に 30 を指定すると男性のデータを、57 を指定すると女性のデータを読み込みます。`s` を省略すると、最初のシート、つまり佐賀県全体のデータを読み込みます。

ソースコード 4.2 に示す関数 `plotPopulation()` では、引数 `year` に整数で描画する年を指定します。6 行目と 7 行目で、引数に渡された男女の `DataFrame` から対応する列の `Series` を取り出しています。8 行目と 9 行目で `barh()` メソッドを使って横棒グラフとして描きます。女性のデータは、マイナス側になっていることに注意してください。7

#### ソースコード 4.2 横棒グラフ作図

```
1 def plotPopulation(year:int,  
2     dataMale:pandas.DataFrame, dataFemale:pandas.DataFrame,  
3     ↪ ax:plt.Axes):  
4     """  
5     指定した年の人口予想を作図する  
6     """  
7     yStr = f'{year}年'  
8     male:pandas.Series = dataMale[yStr]  
9     female:pandas.Series = dataFemale[yStr] * (-1)  
10    male.plot.barh(ax=ax)  
11    female.plot.barh(ax=ax,color = 'red')  
    ax.plot([0, 0], [-1, len(male.index)], color = 'black')
```

#### ソースコード 4.3 作図準備

```
1 def plotPrepare(year:int)->plt.Axes:  
2     """  
3     作図準備  
4     """  
5     fig, ax = plt.subplots(figsize = (15, 10), facecolor = 'w')  
6     ax.set_title(f'佐賀県人口構成予想 ({year})')  
7     #横軸の設定  
8     xMax = 40000  
9     xTic = 10000  
10    ax.set_xlim(- xMax, xMax)  
11    xt:list[int] = [x for x in range(- xMax, xMax + 1, xTic)]  
12    #左側にも正の数字を書くためのラベル  
13    xl:list[int] = [abs(x) for x in range(- xMax, xMax + 1, xTic)]  
14    xt.extend(xl)  
15    ax.set_xticks(xt)  
16    ax.set_xlabel('人')  
17  
18    ax.text(- xMax + xTic, 18, '女性', ha = 'center')  
19    ax.text(xMax - xTic, 18, '男性', ha = 'center')  
20    return ax
```

行目で Series に -1 を乗じている部分です。10 行目は、左右の区別のための縦軸を描いています。

ソースコード 4.3 は、作図の準備をする関数です。横軸の座標を 11 行目から 14 行目で書き換えています。11 行目で横軸の座標のリストを、13 行目でそのラベルを作成してい

ます。ラベルには、絶対値が入っていることに注意してください。

#### ソースコード 4.4 メイン部分

```
1 url = 'https://www.ipss.go.jp/pp-shicyoson/j/'\
2     + 'shicyoson18/3kekka/Municipalities/41.xls'
3 dataMale = getData(url,30)
4 dataFemale = getData(url, 57)
5 year = 2025
6 plotPrepare(year)
7 plotPopulation(year, dataMale, dataFemale)
8 plt.savefig('sagaFuturePopulation.pdf')
9 plt.show()
```

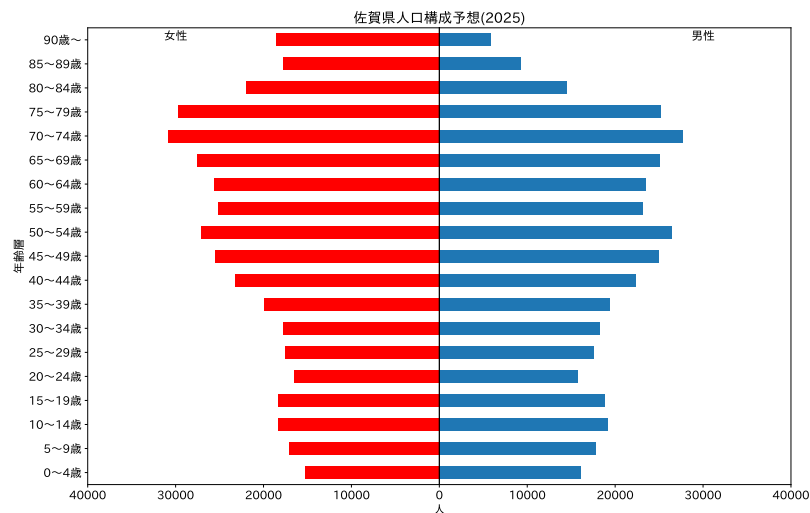


図 1 佐賀県の人口予想

## 5 次回

Web ページの中に表があるものがあります。佐賀市の毎月の最高気温がある Web ページから、表のデータを取り出して作図しましょう。