Basic Data Processing and Visualization Project

Paola Mussida

May 16, 2020

1 University Ranking

In this notebook we analyse the relation between university ranking and the male-female ratio.

We will focus our attention on the Italian university with respect to all the enties in order to investigate the differences.

World University Rankings

Investigate the best universities in the world

1.1 Dataset descriprion

Ranking universities is a difficult, political, and controversial practice. There are hundreds of different national and international university ranking systems, many of which disagree with each other. The dataset we use in this notebook is based on the Times Higher Education World University.

Further details on the ranking system can be found in https://en.wikipedia.org/wiki/College_and_university_rankings.

1.1.1 Dataset source

For this notebook, I used the university ranking dataset from the [KAGGLE] site (https://www.kaggle.com/mylesoneill/world-university-rankings#cwurData.csv).

[1]:
```
# getting the dataset to local (for google colab)
!wget https://raw.githubusercontent.com/Pa-O-La/jupiter/master/timesData.csv
```

```
--2020-05-16 12:54:49--  https://raw.githubusercontent.com/Pa-O-
La/jupiter/master/timesData.csv
Resolving raw.githubusercontent.com (raw.githubusercontent.com)…
151.101.0.133, 151.101.64.133, 151.101.128.133, …
Connecting to raw.githubusercontent.com
(raw.githubusercontent.com)|151.101.0.133|:443… connected.
HTTP request sent, awaiting response… 200 OK
Length: 268231 (262K) [text/plain]
Saving to: ‘timesData.csv.1’

timesData.csv.1     100%[===================>] 261.94K  --.-KB/s    in 0.05s
```

```
2020-05-16 12:54:50 (5.23 MB/s) - ‘timesData.csv.1’ saved [268231/268231]
```

1.1.2 Import dataset into the notebook

[2]:

```
import csv
file_name = './timesData.csv'
fp = open(file_name)
reader = csv.reader(fp)
header = next(reader)
```

1.1.3 Daset entries description (name, datatype)

[3]:

```
import pandas as pd
url = 'https://raw.githubusercontent.com/Pa-O-La/jupiter/master/timesData.csv'
timesdata_pd = pd.read_csv(url)

print(timesdata_pd.dtypes)
```

```
world_rank                 object
university_name            object
country                    object
teaching                  float64
international              object
research                  float64
citations                 float64
income                     object
total_score                object
num_students               object
student_staff_ratio       float64
international_students     object
female_male_ratio          object
year                        int64
dtype: object
```

1.1.4 Dictionary creation

[4]:

```
timesdata = []
for line in reader:
  d = dict(zip(header, line))
  timesdata.append(d)
```

Dataset dimension

[5]:

```
len(timesdata)
```

```
[5]: 2603
```

The first university:

```
[6]: print(timesdata_pd.head())
```

```
  world_rank                       university_name                   country  \
0          1                    Harvard University  United States of America
1          2    California Institute of Technology  United States of America
2          3 Massachusetts Institute of Technology  United States of America
3          4                   Stanford University  United States of America
4          5                  Princeton University  United States of America

  teaching international  research citations income total_score  \
0     99.7          72.4      98.7      98.8   34.5        96.1
1     97.7          54.6      98.0      99.9   83.7        96.0
2     97.8          82.3      91.4      99.9   87.5        95.6
3     98.3          29.5      98.1      99.2   64.3        94.3
4     90.9          70.3      95.4      99.9      -        94.2

  num_students  student_staff_ratio international_students female_male_ratio  \
0       20,152                  8.9                    25%               NaN
1        2,243                  6.9                    27%           33 : 67
2       11,074                  9.0                    33%           37 : 63
3       15,596                  7.8                    22%           42 : 58
4        7,929                  8.4                    27%           45 : 55

   year
0  2011
1  2011
2  2011
3  2011
4  2011
```

1.2 Adding colum with female-male ratio

We handle the exception where the value is not present

```
[7]: for d in range(0, len(timesdata)):
       fmr = None
       try:
         fmr_vett = timesdata[d]['female_male_ratio'].split(':')
         fmr = int(fmr_vett[0]) / int(fmr_vett[1])
         timesdata[d]['fmr'] = fmr
       except:
         timesdata[d]['fmr'] = None
```

Filtering out None values

[8]:

```
timesdata_c = [d for d in timesdata if d['fmr'] is not None]
```

Number of removed None entry

[9]:

```
print('Removed ', len(timesdata) - len(timesdata_c), ' entries')
```

```
Removed  238  entries
```

1.3 Preparation of world_rank data

We count the number of null value of the total score.

[10]:

```
counter = 0
for d in timesdata_c:
  try:
    (float(d['total_score']))
  except:
    counter +=1

print('%f', counter/len(timesdata_c))
```

```
%f 0.5441860465116279
```

Instead of throwing the 54% of the dataset entries we use as the score value the word ranking position.

Some entries needs homogenization

[11]:

```
for d in timesdata_c:
  if (d['world_rank'].isdigit()):
    rank_num = float(d['world_rank'])
  elif ('-' in d['world_rank']):
    #split on '-'
    tmp = d['world_rank'].split('-')
    rank_num = float(tmp[0]) * 0.5 + float(tmp[1]) * 0.5
  elif ('=' in d['world_rank']):
    tmp = d['world_rank'].split('=')
    rank_num = float(tmp[1])
  else:
    rank_num = -1
    print(d['world_rank'])

  d['wr_num'] = rank_num
```

1.4 Creation of the Italian data subset

[12]:

```
data_ita = [d for d in timesdata_c if d['country'] == 'Italy']
```

Number of Italian entries (over the yers)

[13]:

```
len(data_ita)
```

[13]:

```
92
```

Retreiving years list of the global dataset

[14]:

```
from collections import defaultdict
years_g = defaultdict(int)
for d in timesdata_c:
  years_g[d['year']] += 1

print (years_g.keys())
```

```
dict_keys(['2011', '2012', '2013', '2014', '2015', '2016'])
```

Numbers of global entries per year

[15]:

```
for d in years_g:
  print (d, ' - ', years_g[d])
```

```
2011  -  178
2012  -  362
2013  -  364
2014  -  364
2015  -  362
2016  -  735
```

1.5 Create annual female-male ratio (fmr)

We store the global and italian average fmr per year in vectors fmr_g and fmr_i, respectively

[16]:

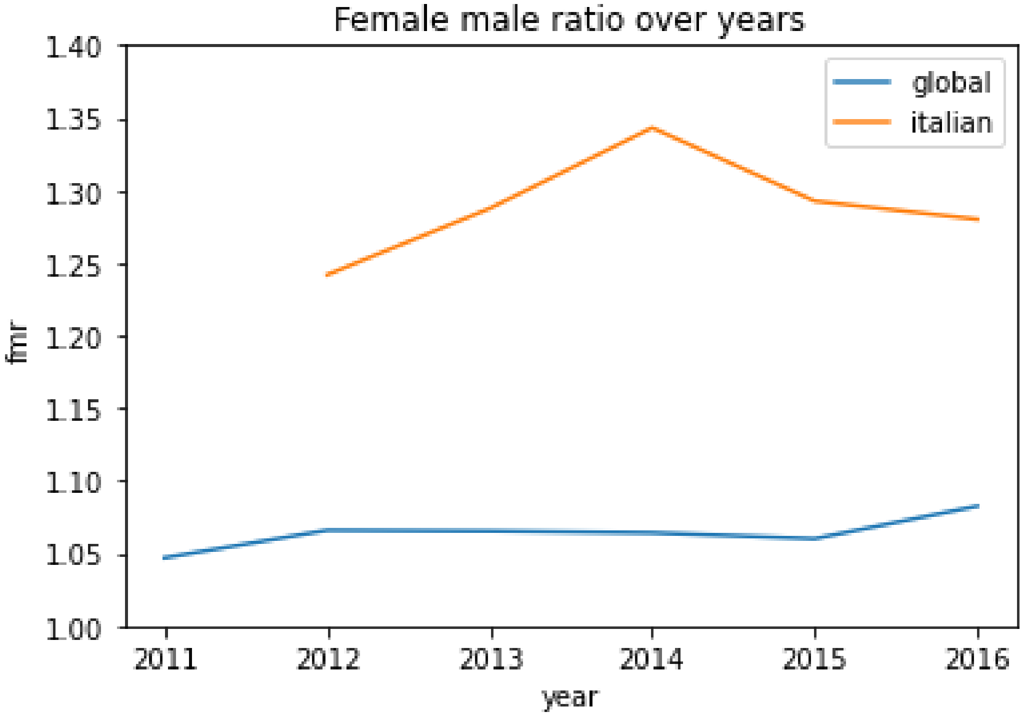
```
import numpy
fmr_g = []
fmr_i =[]
for y in years_g.keys():
  tmp = [d['fmr'] for d in timesdata_c if d['year'] == y]
  fmr_mean = numpy.array(tmp).mean()
  fmr_g.append(fmr_mean)

  tmp_ita = [d['fmr'] for d in data_ita if d['year'] == y]
  fmr_mean_ita = 0
  if (len(tmp_ita)>0):
    fmr_mean_ita = numpy.array(tmp_ita).mean()
  fmr_i.append(fmr_mean_ita)
```

2 Data Visualization

2.1 Plot of the fmr_g and fmr_i over years

[17]:

```
import matplotlib.pyplot as plt

X = list(years_g.keys())
plt.plot(X, fmr_g, label = 'global')
plt.plot(X[1:], fmr_i[1:], label = 'italian')
plt.ylim([1, 1.4])
plt.legend()
plt.ylabel('fmr')
plt.title('Female male ratio over years')
plt.xlabel('year');
```



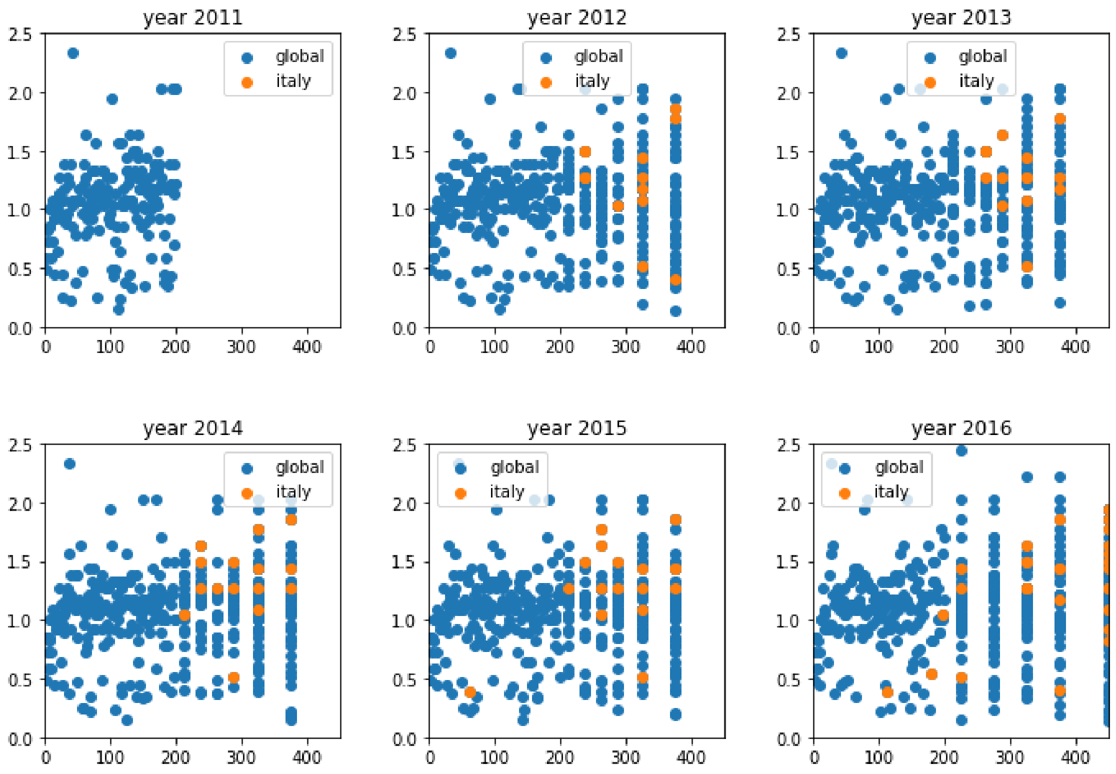
The plot show that the italian average fmr is higher than the global one

2.2 Evolution of the global and the italian fmr through years

[18]:

```
# in the years

fig, axs = plt.subplots(2, 3)
fig.set_size_inches(10, 7)

```

```
c = 0
for yy in years_g.keys():

  data_per_year_g = [d for d in timesdata_c if d['year'] == yy]
  ranking_y_g = numpy.array([float(d['wr_num']) for d in data_per_year_g])
  fmr_y_g = numpy.array([d['fmr'] for d in data_per_year_g])

  data_per_year_i = [d for d in data_ita if d['year'] == yy]
  ranking_y_i = numpy.array([float(d['wr_num']) for d in data_per_year_i])
  fmr_y_i = numpy.array([d['fmr'] for d in data_per_year_i])


  axs[c//3, c%3].scatter(ranking_y_g, fmr_y_g, label= 'global')
  axs[c//3, c%3].scatter(ranking_y_i, fmr_y_i, label = 'italy')

  axs[c//3, c%3].set_title('year %s' %yy)
  axs[c//3, c%3].set_xlim([0, 450])
  axs[c//3, c%3].set_ylim([0, 2.5])
  axs[c//3, c%3].legend()
  c += 1

fig.subplots_adjust(left=0.08, right=0.98, bottom=0.05, top=0.9, hspace=0.4,␣
 ↪wspace=0.3)
```



The panel shows that through years the average score of italian university increases.

We can also notice that the majority of the entries are clustered around the average value.

2.2.1 Focusing

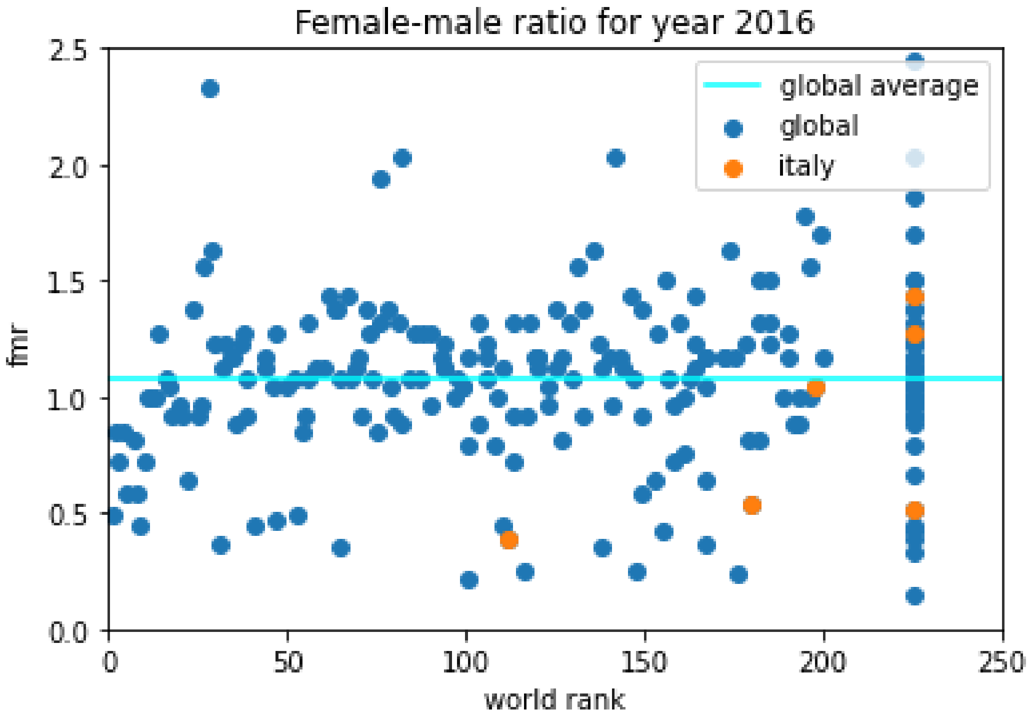
We now focus the attention on 2016

[19]:

```
# 2016
yy='2016'
data_2016_g = [d for d in timesdata_c if d['year'] == yy]
ranking_2016_g = numpy.array([float(d['wr_num']) for d in data_2016_g])
fmr_2016_g = numpy.array([d['fmr'] for d in data_2016_g])

data_2016_i = [d for d in data_ita if d['year'] == yy]
ranking_2016_i = numpy.array([float(d['wr_num']) for d in data_2016_i])
fmr_2016_i = numpy.array([d['fmr'] for d in data_2016_i])

plt.plot([0, 250], [fmr_g[-1],fmr_g[-1]], color= 'cyan' , label = 'global␣
 ↪average')
plt.scatter(ranking_2016_g, fmr_2016_g, label= 'global')
plt.scatter(ranking_2016_i, fmr_2016_i, label = 'italy')

plt.title('Female-male ratio for year %s' %yy)
plt.xlim([0, 250])
plt.ylim([0, 2.5])
plt.legend()
plt.xlabel('world rank')
plt.ylabel('fmr')
```

[19]:

```
Text(0, 0.5, 'fmr')
```



We remark that the lower the rank, the higher the position.

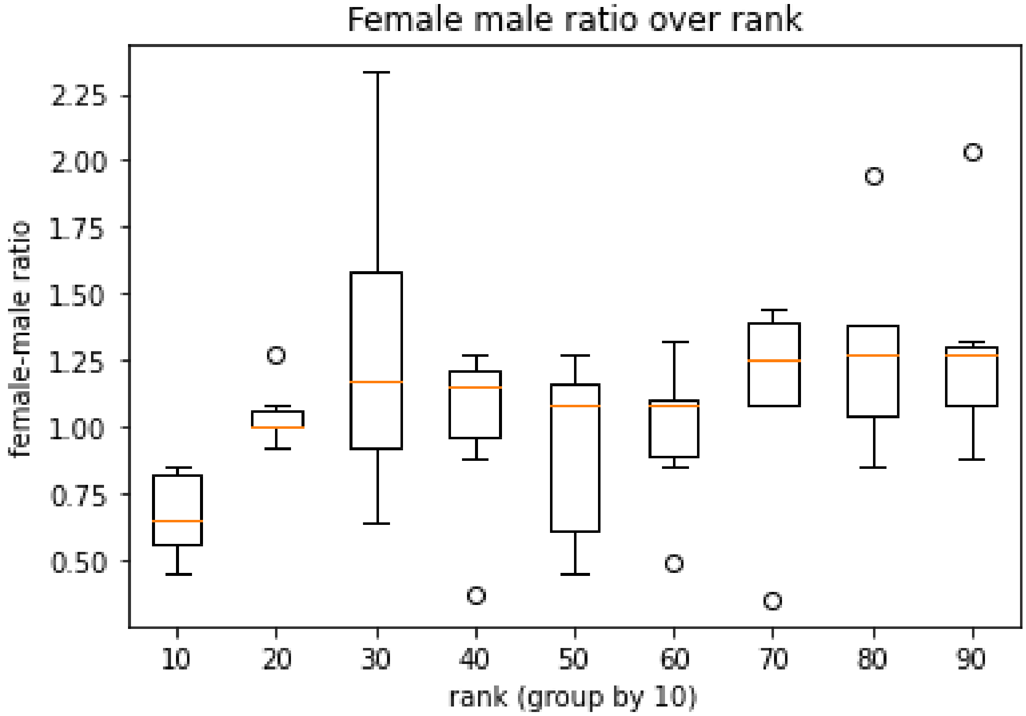
We notice that while the majority of the entries are round the average value, the firsts entries are remarkably below.

We also notice that all the italian entries, within the first 200, are below the global average.

2.3 Looking for a trend

We split the dataset in group of 10 entries. Each group contains 10 entries of the university sorted by the rank.

[20]:
```
c_fmr = []

lowlim = 0
for c in range (10,100,10):
  uplim = c
  temp = [d['fmr'] for d in data_2016_g if d['wr_num'] > lowlim and d['wr_num']␣
 ↪< uplim ]
  c_fmr.append(temp)
  lowlim = c

plt.boxplot(c_fmr);
plt.title('Female male ratio over rank')
plt.ylabel('female-male ratio')
plt.xlabel('rank (group by 10)')
plt.xticks(range(1, 10), range (10,100,10));
```



We notice that by grouping the data, the first entries exhibit a lower average fmr with respect to the global one.