

Fox2015_solution

January 23, 2016

0.1 Solution of Fox et al. 2015

```
In [1]: import pandas
import numpy as np
```

First, we read the input, and take a look at the column names

```
In [2]: fox = pandas.read_csv("../data/Fox2015_data.csv")
```

```
In [3]: fox.columns
```

```
Out[3]: Index(['MsID', 'Year', 'HandlingEditorSex', 'ReviewerSex', 'ReviewerRegion',
              'ReviewerInvited', 'ReviewerAgreed', 'ReviewerScore', 'FinalDecision'],
              dtype='object')
```

Extract the unique manuscripts and count them

```
In [4]: unique_ms = list(set(fox['MsID']))
num_ms = len(unique_ms)
print(num_ms)
```

6720

Now we want to elaborate the data, by saving into lists the number of reviewers, the final decision, and the year for each manuscript. At the end of the code, we convert the lists into np.arrays, as it is much easier to subset them.

```
In [5]: num_reviewers = []
final_decision = []
year = []

for ms in unique_ms:
    # extract the rows
    subset = fox[fox['MsID'] == ms]
    # count number of reviewers by summing ReviewerAgreed
    num_reviewers.append(sum(subset['ReviewerAgreed']))
    # extract final decision
    if list(subset['FinalDecision'])[0] == 1:
        final_decision.append(1)
    else:
        final_decision.append(0)
    # extract year
    year.append(list(subset['Year'])[0])

# convert to np.array
num_reviewers = np.array(num_reviewers)
final_decision = np.array(final_decision)
year = np.array(year)
```

Now we write a function that takes a year as input, and prints the rejection rate for each number of reviewers, along with some other summary information. If we call the function with 'all' instead of a year, then the analysis is performed on the whole data set.

```
In [6]: def get_prob_rejection(my_year = 'all'):
        # subset the data
        if my_year != 'all':
            my_num_reviewers = num_reviewers[year == my_year]
            my_final_decision = final_decision[year == my_year]
        else:
            my_num_reviewers = num_reviewers
            my_final_decision = final_decision
        # start printing output
        print("=====")
        print("Year:", my_year)
        print("Submissions:", len(my_final_decision))
        print("Overall rejection rate:",
              round(my_final_decision.mean(),3))
        print("NumRev", '\t', "NumMs", '\t', "rejection rate")
        for i in range(max(my_num_reviewers) + 1):
            print(i, '\t',
                  len(my_final_decision[my_num_reviewers == i]), '\t',
                  round(my_final_decision[my_num_reviewers == i].mean(), 3))
        print("=====")
```

Compile a table measuring the probability of rejection given the number of reviewers. Does having more reviewers increase the probability of being rejected?

```
In [7]: get_prob_rejection('all')

=====
Year: all
Submissions: 6720
Overall rejection rate: 0.807
NumRev      NumMs      rejection rate
0           2875        0.978
1            91         0.527
2           2667        0.685
3           1012        0.674
4            72         0.708
5             3         1.0
=====
```

It seems so. Especially, look at the difference between one and two reviewers...

0.1.1 Repeat the analysis above for each year represented in the database.

We can simply call the function for each year. For example:

```
In [8]: get_prob_rejection(2009)

=====
Year: 2009
Submissions: 626
Overall rejection rate: 0.827
NumRev      NumMs      rejection rate
```

| | | |
|---|-----|-------|
| 0 | 306 | 0.977 |
| 1 | 2 | 0.5 |
| 2 | 228 | 0.68 |
| 3 | 86 | 0.698 |
| 4 | 4 | 0.75 |

=====

```
In [9]: for yr in range(2004, 2015):
        get_prob_rejection(yr)
```

=====

Year: 2004

Submissions: 390

Overall rejection rate: 0.741

| NumRev | NumMs | rejection rate |
|--------|-------|----------------|
| 0 | 55 | 0.836 |
| 1 | 8 | 0.5 |
| 2 | 302 | 0.735 |
| 3 | 25 | 0.68 |

=====

=====

Year: 2005

Submissions: 467

Overall rejection rate: 0.745

| NumRev | NumMs | rejection rate |
|--------|-------|----------------|
| 0 | 117 | 0.897 |
| 1 | 17 | 0.471 |
| 2 | 299 | 0.692 |
| 3 | 34 | 0.824 |

=====

=====

Year: 2006

Submissions: 548

Overall rejection rate: 0.712

| NumRev | NumMs | rejection rate |
|--------|-------|----------------|
| 0 | 171 | 0.918 |
| 1 | 17 | 0.353 |
| 2 | 322 | 0.634 |
| 3 | 36 | 0.611 |
| 4 | 2 | 0.5 |

=====

=====

Year: 2007

Submissions: 557

Overall rejection rate: 0.79

| NumRev | NumMs | rejection rate |
|--------|-------|----------------|
| 0 | 207 | 0.981 |
| 1 | 12 | 0.5 |
| 2 | 255 | 0.678 |
| 3 | 75 | 0.693 |
| 4 | 8 | 0.75 |

=====

=====

Year: 2008

Submissions: 604

Overall rejection rate: 0.768

| NumRev | NumMs | rejection rate |
|--------|-------|----------------|
| 0 | 254 | 0.961 |
| 1 | 5 | 0.6 |
| 2 | 285 | 0.639 |
| 3 | 56 | 0.589 |
| 4 | 4 | 0.5 |

=====
=====

Year: 2009

Submissions: 626

Overall rejection rate: 0.827

| NumRev | NumMs | rejection rate |
|--------|-------|----------------|
| 0 | 306 | 0.977 |
| 1 | 2 | 0.5 |
| 2 | 228 | 0.68 |
| 3 | 86 | 0.698 |
| 4 | 4 | 0.75 |

=====
=====

Year: 2010

Submissions: 670

Overall rejection rate: 0.846

| NumRev | NumMs | rejection rate |
|--------|-------|----------------|
| 0 | 341 | 0.997 |
| 1 | 1 | 1.0 |
| 2 | 116 | 0.724 |
| 3 | 198 | 0.672 |
| 4 | 13 | 0.615 |
| 5 | 1 | 1.0 |

=====
=====

Year: 2011

Submissions: 740

Overall rejection rate: 0.82

| NumRev | NumMs | rejection rate |
|--------|-------|----------------|
| 0 | 370 | 0.997 |
| 1 | 5 | 0.6 |
| 2 | 118 | 0.653 |
| 3 | 227 | 0.626 |
| 4 | 20 | 0.8 |

=====
=====

Year: 2012

Submissions: 783

Overall rejection rate: 0.844

| NumRev | NumMs | rejection rate |
|--------|-------|----------------|
| 0 | 392 | 0.992 |
| 1 | 3 | 0.667 |
| 2 | 185 | 0.686 |
| 3 | 188 | 0.691 |
| 4 | 13 | 0.846 |
| 5 | 2 | 1.0 |

=====
=====

```

=====
Year: 2013
Submissions: 872
Overall rejection rate: 0.847
NumRev      NumMs      rejection rate
0           436          0.995
1            14          0.571
2           366          0.691
3            51          0.804
4             5           0.6
=====

```

```

=====
Year: 2014
Submissions: 463
Overall rejection rate: 0.862
NumRev      NumMs      rejection rate
0           226          0.996
1             7          0.857
2           191          0.749
3            36          0.667
4             3          0.333
=====

```

In []: