'''

DS 5230

Summer 2022

HW1_Problem_1_and_Problem_2

Hongyan Yang

'''

Question 1

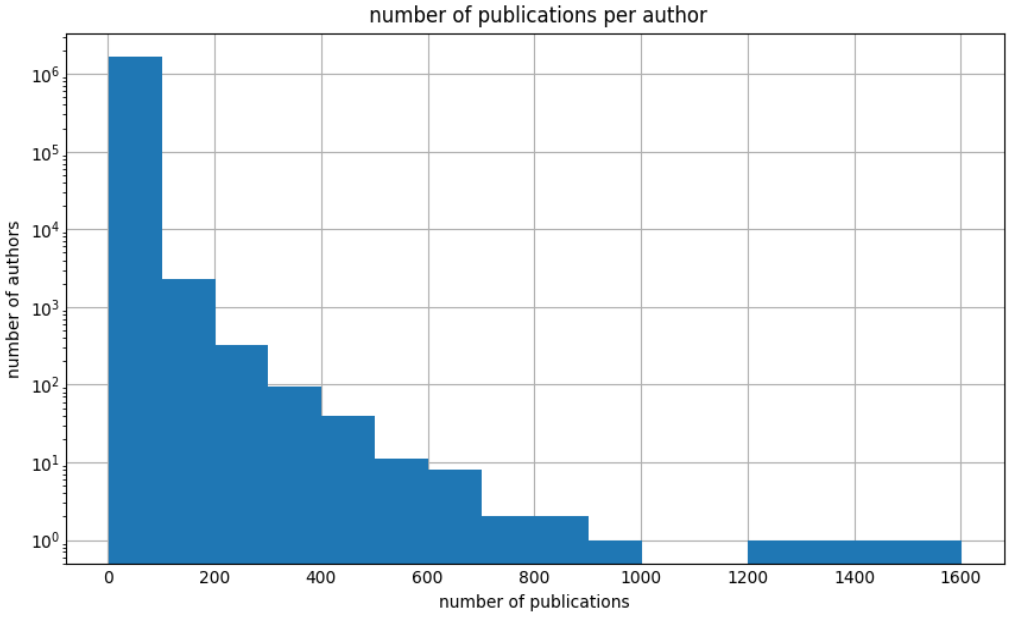
A. There are 1651650 distinct authors, 273329 distinct publication venues, 2385057 distinct publications, and 1007495 distinct citations/references.

B. {'Knowledge Discovery in Databases: PKDD 2006: 10th European Conference on Principles and Practice of Knowledge Discovery in Databases, Berlin, Germany, September ... (Lecture Notes in Computer Science)', 'Knowledge Discovery in Databases: PKDD 2005: 9th European Conference on Principles and Practice of Knowledge Discovery in Databases, Porto, Portugal, October ... / Lecture Notes in Artificial Intelligence)', "PKDD '04 Proceedings of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases", 'PKDD 2007 Proceedings of the 11th European conference on Principles and Practice of Knowledge Discovery in Databases', "PKDD'05 Proceedings of the 9th European conference on Principles and Practice of Knowledge Discovery in Databases"}

These numbers are not likely to be accurate. As can be seen in the publication venues listed above, the same conference has been assigned different names due to its yearly publication and thus produces duplicated records in the dataset.

C. ['Jr.', '-', 'III', 'Computer Staff', 'Staff', 'Wei Wang', 'Linux Journal Staff', 'II', 'Lei Zhang', 'Wei Zhang', 'Wei Li', 'Jun Wang', 'Yang Liu', 'Jun Zhang', 'Li Zhang', 'Lei Wang', 'Ming Li', 'Yan Zhang', 'Wei Liu', 'Elisa Bertino']

problem_1_c.txt

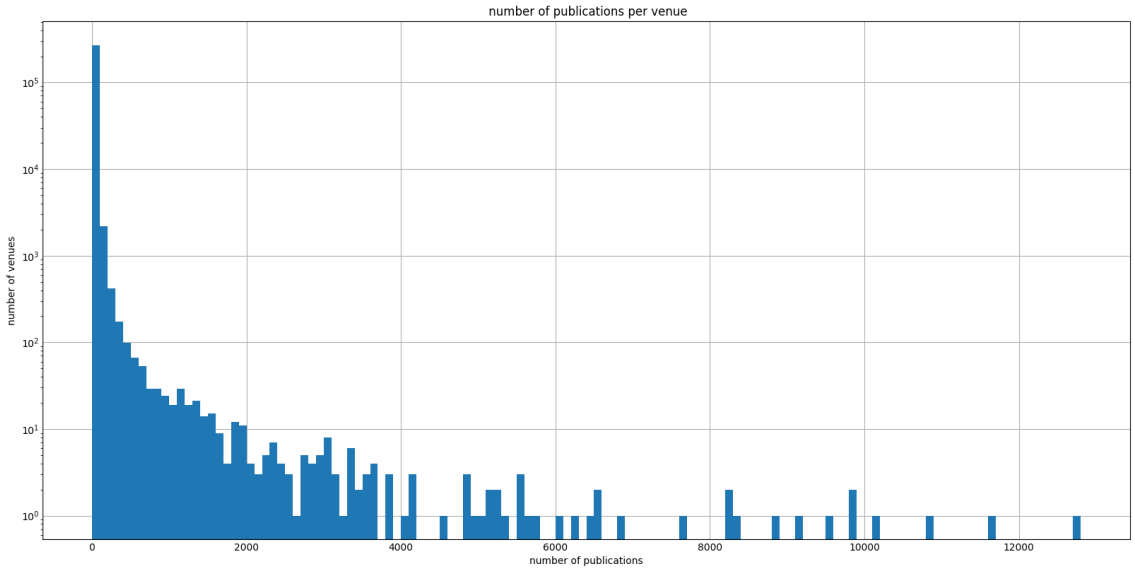


D. mean of publications per author: 3.45
std of publications per author: 10.05
Q1 of publications per author: 1.0
median of publications per author: 1.0
Q3 of publications per author: 3.0

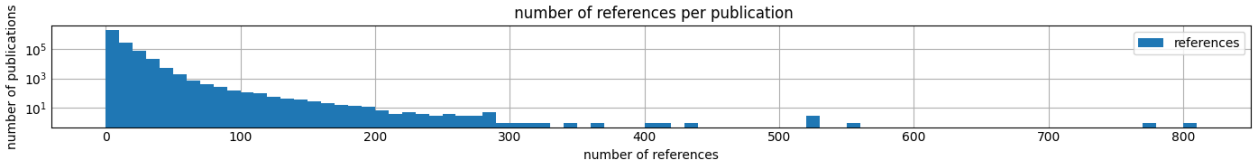
The median of publications per author is 1.0 and the mean of publications per author is 3.45. It can be explained by the large std of the dataset, which is 10.05, indicating that the dataset is really spread out. It can also be explained by the dataset's high skewness to the right as can be verified by a 3rd quartile of 3.0, which is still lower than dataset's mean.

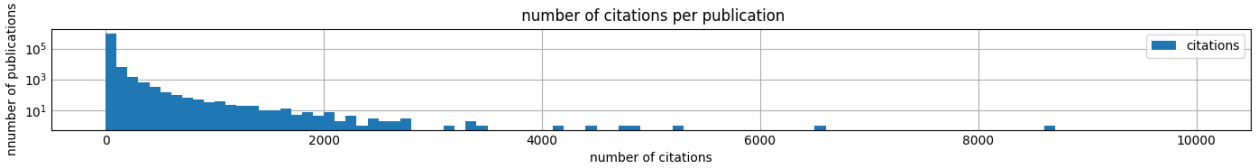
E. mean of publications per venue: 9.0
std of publications per venue: 107.0
Q1 of publications per venue: 1.0
median of publications per venue: 1.0
Q3 of publications per venue: 1.0

Venue IEEE Transactions on Information Theory has the largest number of publications.

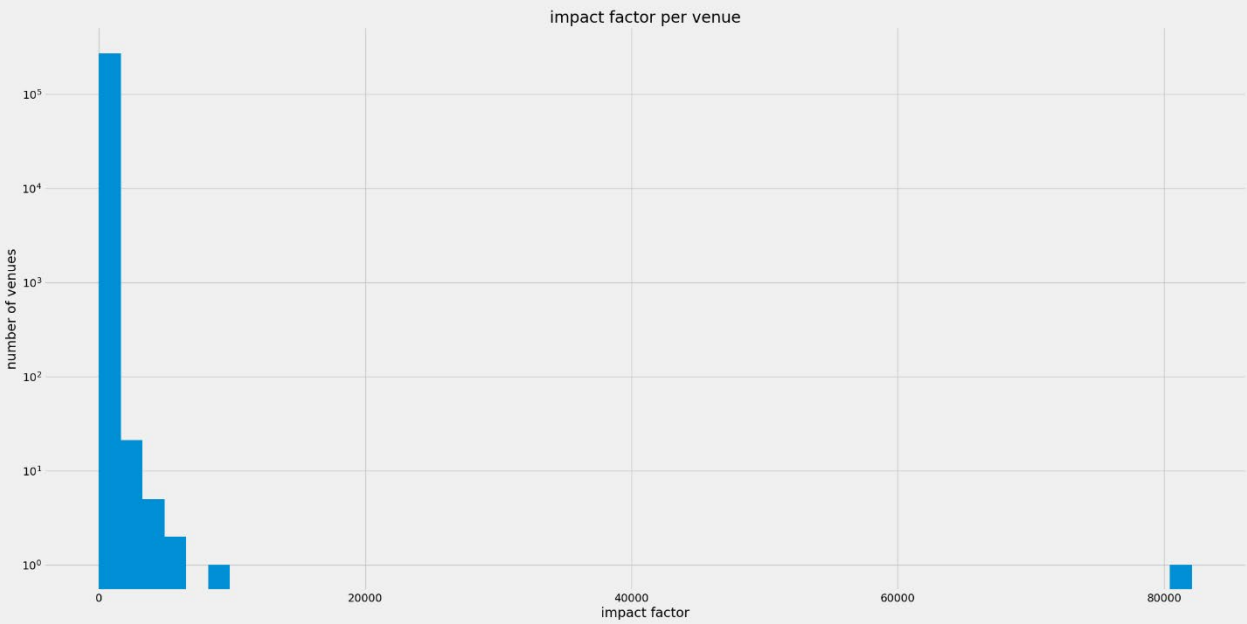


F. Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles has the largest number of references, INFORMS Journal on Computing has the largest number of citations. These make sense to me because I referred to each publication by its index in the dataset and INFORMS Journal on Computing with an index 2135000 does have that many citations, which indicates that INFORMS Journal on Computing must be an important and fundamental publication for the whole subject or domain area.





G. problem_1_g.png

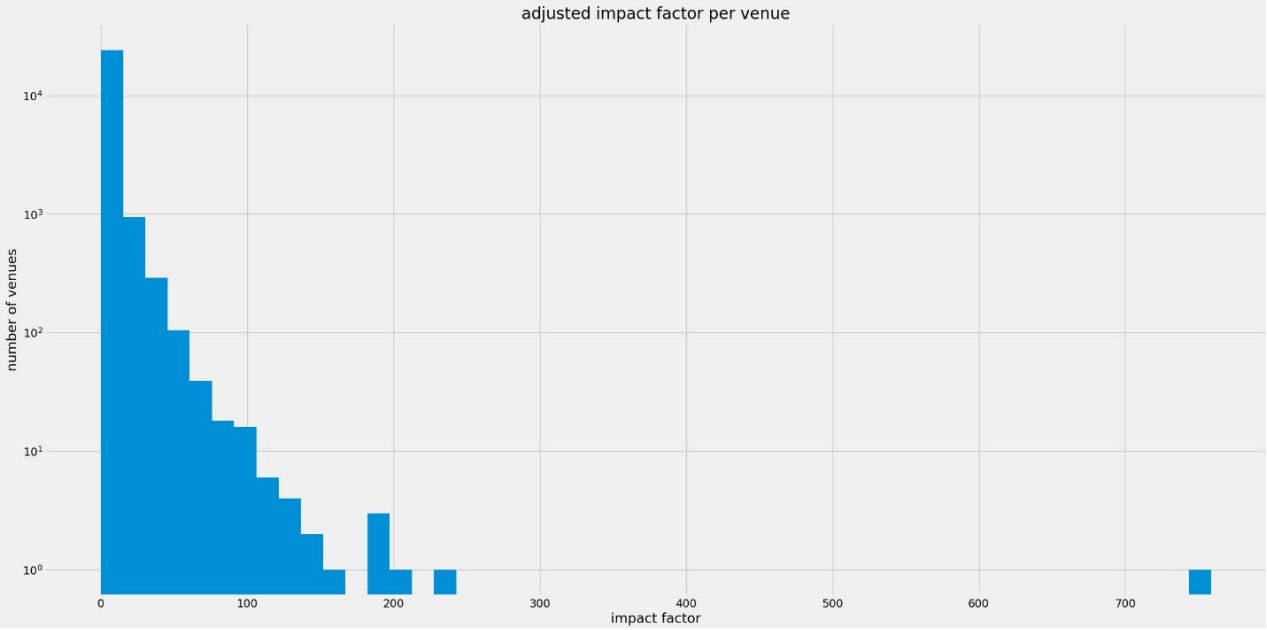


H. IJIRR: International Journal of Information Retrieval Research. has the highest apparent impact factor of 82080.0. I do not believe this number because it is far beyond a reasonable impact factor for an even top-ranking journal.

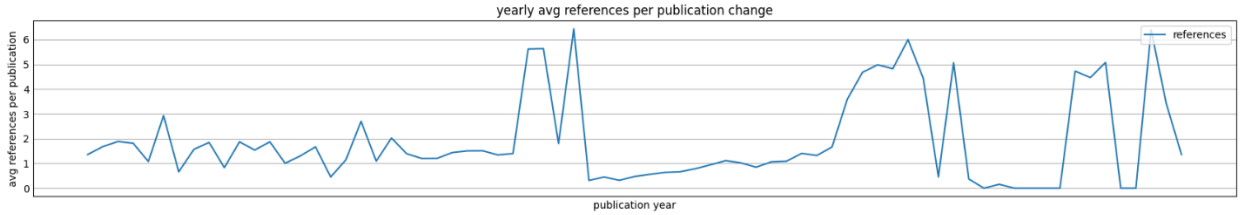
I. INFORMS Journal on Computing has the highest apparent impact factor of 758.81.

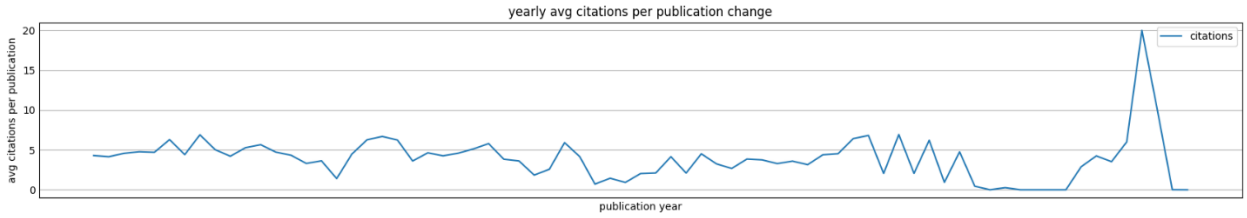
After restricting the calculation to venues with at least 10 publications, the histogram has less outliers and all publication venues' impact factors are relatively less spread out.

The impact factor (mean number of citations) of INFORMS Journal on Computing is 758.81, the median number of 0.00. It can be inferred that publications have a wide spread out citation numbers even in the same most influential publication venue.



J. From the plot I see there's a relatively close relationship between yearly avg references per publication and yearly avg citations per publication. If we assume the more references a publication has the more important it is. Then it makes sense to witness a cointegration between these two subplots.





Question 2

A. *HW1_Problem_2.py*

B. It takes around 6.82 seconds to run the program to convert the kosarak.dat file to a sparse kosarak.arff file.

C. According to the log below:
01:36:00: Started on Tuesday, 24 May 2022
01:36:05: Base relation is now Kosarak (990002 instances)

It takes about 5 seconds to load the kosarak.arff file to Weka.

D. *=== Run information ===*

Scheme: weka.associations.FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.99 -D 0.05 -U 1.0 -M 49500.0
Relation: Kosarak
Instances: 990002
Attributes: 41270
[list of attributes omitted]
=== Associator model (full training set) ===

FPGrowth found 2 rules (displaying top 2)

1. [news_item_11=1, news_item_218=1, news_item_148=1]: 50098 ==> [news_item_6=1]: 49866 <conf:(1)> lift:(1.64) lev:(0.02) conv:(84.4)
2. [news_item_11=1, news_item_148=1]: 55759 ==> [news_item_6=1]: 55230 <conf:(0.99)> lift:(1.63) lev:(0.02) conv:(41.3)

E. According to the log below:

01:36:00: Started on Tuesday, 24 May 2022

01:36:05: Base relation is now Kosarak (990002 instances)

01:38:57: Started weka.associations.FPGrowth

01:38:57: Command: weka.associations.FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.99 -D 0.05 -U 1.0 -M 49500.0

01:39:00: Finished weka.associations.FPGrowth

01:39:02: Started weka.associations.FPGrowth

01:39:02: Command: weka.associations.FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.99 -D 0.05 -U 1.0 -M 49500.0

01:39:05: Finished weka.associations.FPGrowth

01:39:09: Started weka.associations.FPGrowth

01:39:09: Command: weka.associations.FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.99 -D 0.05 -U 1.0 -M 49500.0

01:39:12: Finished weka.associations.FPGrowth

01:39:13: Started weka.associations.FPGrowth

01:39:13: Command: weka.associations.FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.99 -D 0.05 -U 1.0 -M 49500.0

01:39:15: Finished weka.associations.FPGrowth

01:39:16: Started weka.associations.FPGrowth

01:39:16: Command: weka.associations.FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.99 -D 0.05 -U 1.0 -M 49500.0

01:39:19: Finished weka.associations.FPGrowth

It takes about 3 seconds to apply Weka's FP-Growth implementation to find association rules. It takes even shorter time (3s) to find association rules than to convert the dataset and then load into Weka (total 11.82s).