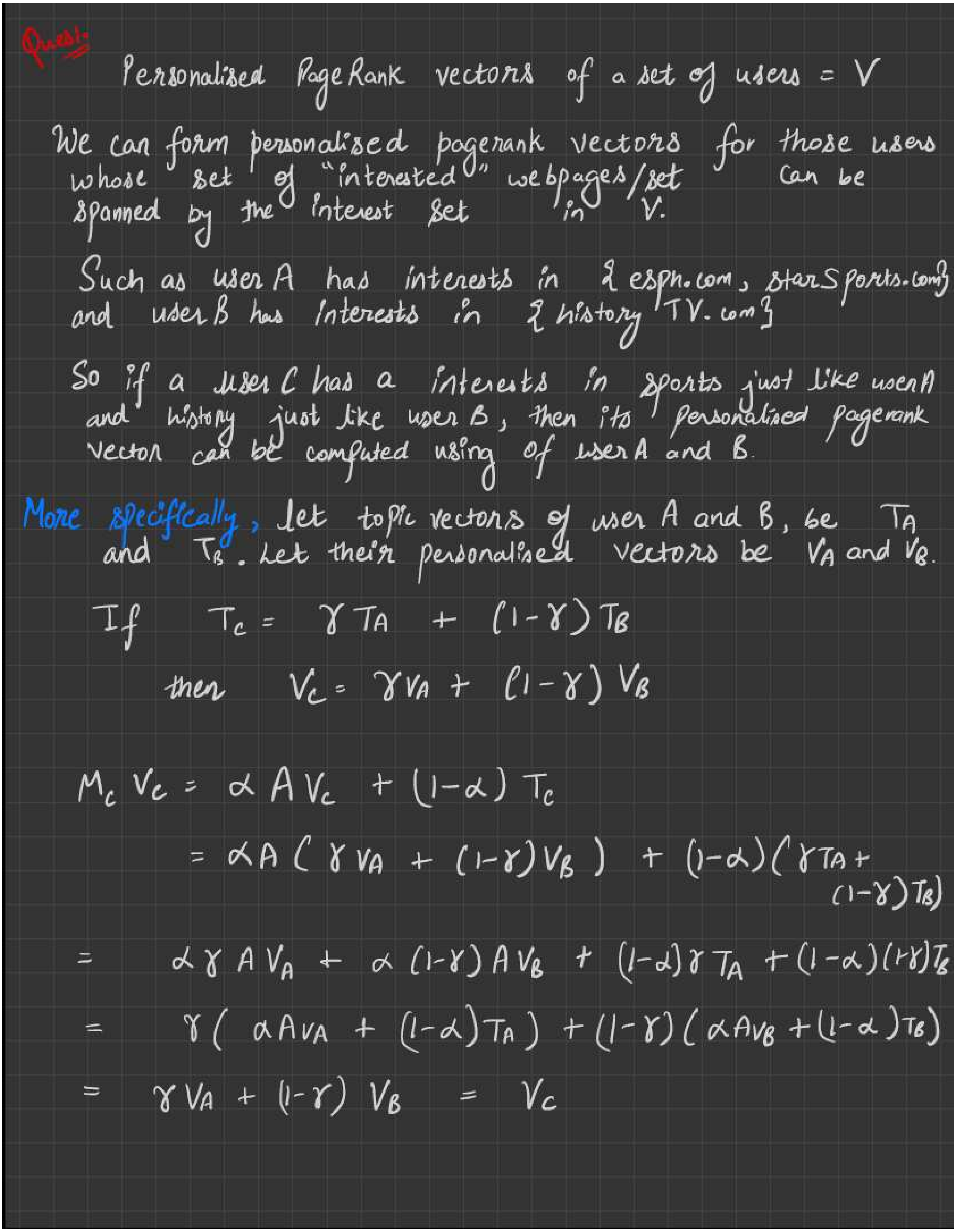
CS 328 : Homework 2

Sachin Yadav 18110148

GitHub Link: https://github.com/yadav-sachin/CS328-Assignment-2

Ques 1.



Ques:

Personalised PageRank vectors of a set of users = V

We can form personalised pagerank vectors for those users whose set of "interested" webpages/set can be spanned by the interest set in V.

Such as user A has interests in { espn.com, starSports.com} and user B has interests in { history TV.com }

So if a user C has a interests in sports just like user A and history just like user B, then its personalised pagerank vector can be computed using of user A and B.

More specifically, let topic vectors of user A and B, be T_A and T_B. Let their personalised vectors be V_A and V_B.

If $T_c = \gamma T_A + (1-\gamma) T_B$

then $V_c = \gamma V_A + (1-\gamma) V_B$

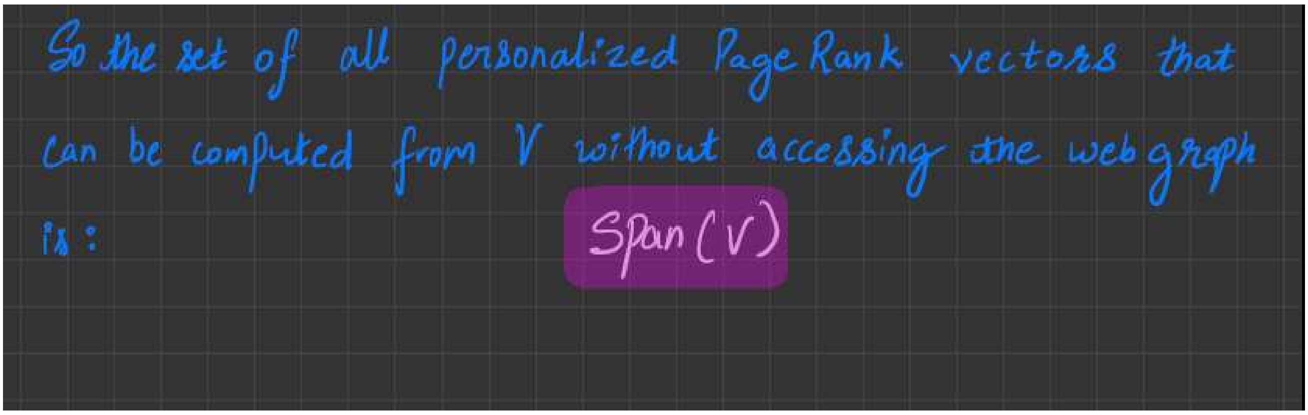
$$M_c V_c = \alpha A V_c + (1-\alpha) T_c$$

$$= \alpha A (\gamma V_A + (1-\gamma) V_B) + (1-\alpha)(\gamma T_A + (1-\gamma) T_B)$$

$$= \alpha \gamma A V_A + \alpha (1-\gamma) A V_B + (1-\alpha)\gamma T_A + (1-\alpha)(1-\gamma) T_B$$

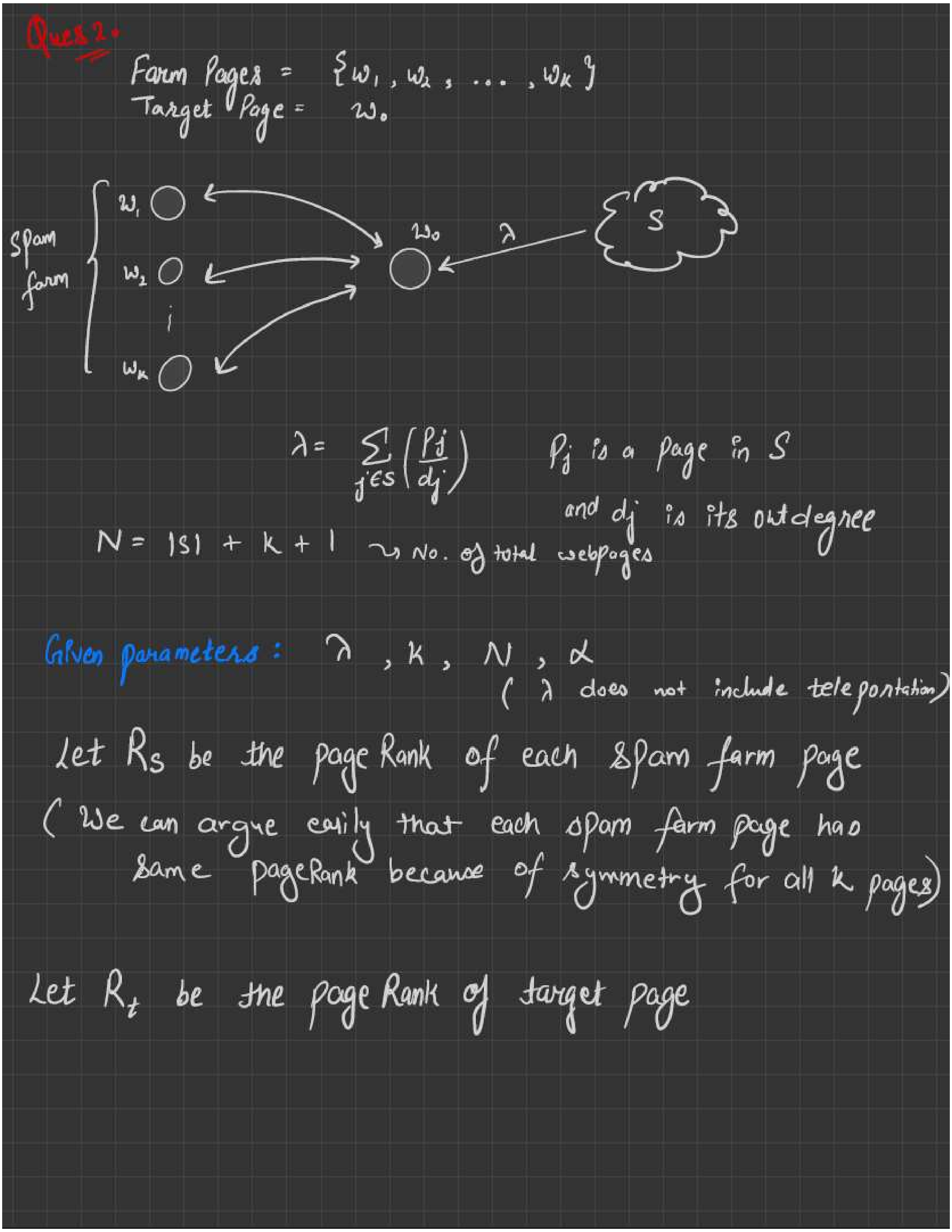
$$= \gamma (\alpha A V_A + (1-\alpha) T_A) + (1-\gamma) (\alpha A V_B + (1-\alpha) T_B)$$

$$= \gamma V_A + (1-\gamma) V_B = V_c$$



So the set of all personalized PageRank vectors that can be computed from V without accessing the web graph is: $\boxed{\text{Span}(V)}$

Ques 2.



spam farm
w_1
w_2
w_k
w_0
λ
S

Ques 2.

Farm Pages = $\{w_1, w_2, \dots, w_k\}$
Target Page = w_0

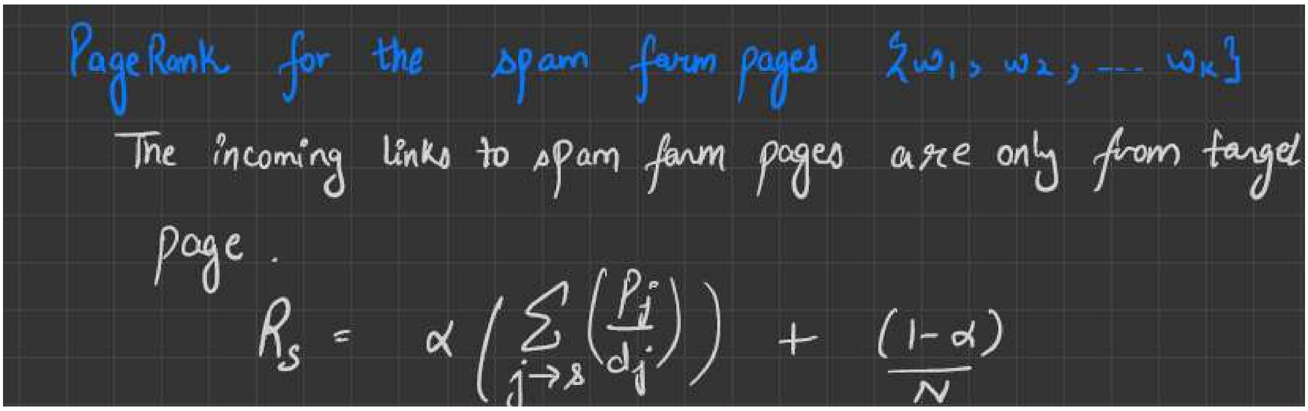
$\lambda = \sum_{j \in S} \left(\frac{P_j}{d_j} \right)$ P_j is a page in S and d_j is its outdegree

$N = |S| + k + 1$ ⤳ No. of total webpages

Given parameters: λ, k, N, α
(λ does not include teleportation)

Let R_s be the page Rank of each spam farm page
(We can argue easily that each spam farm page has same pageRank because of symmetry for all k pages)

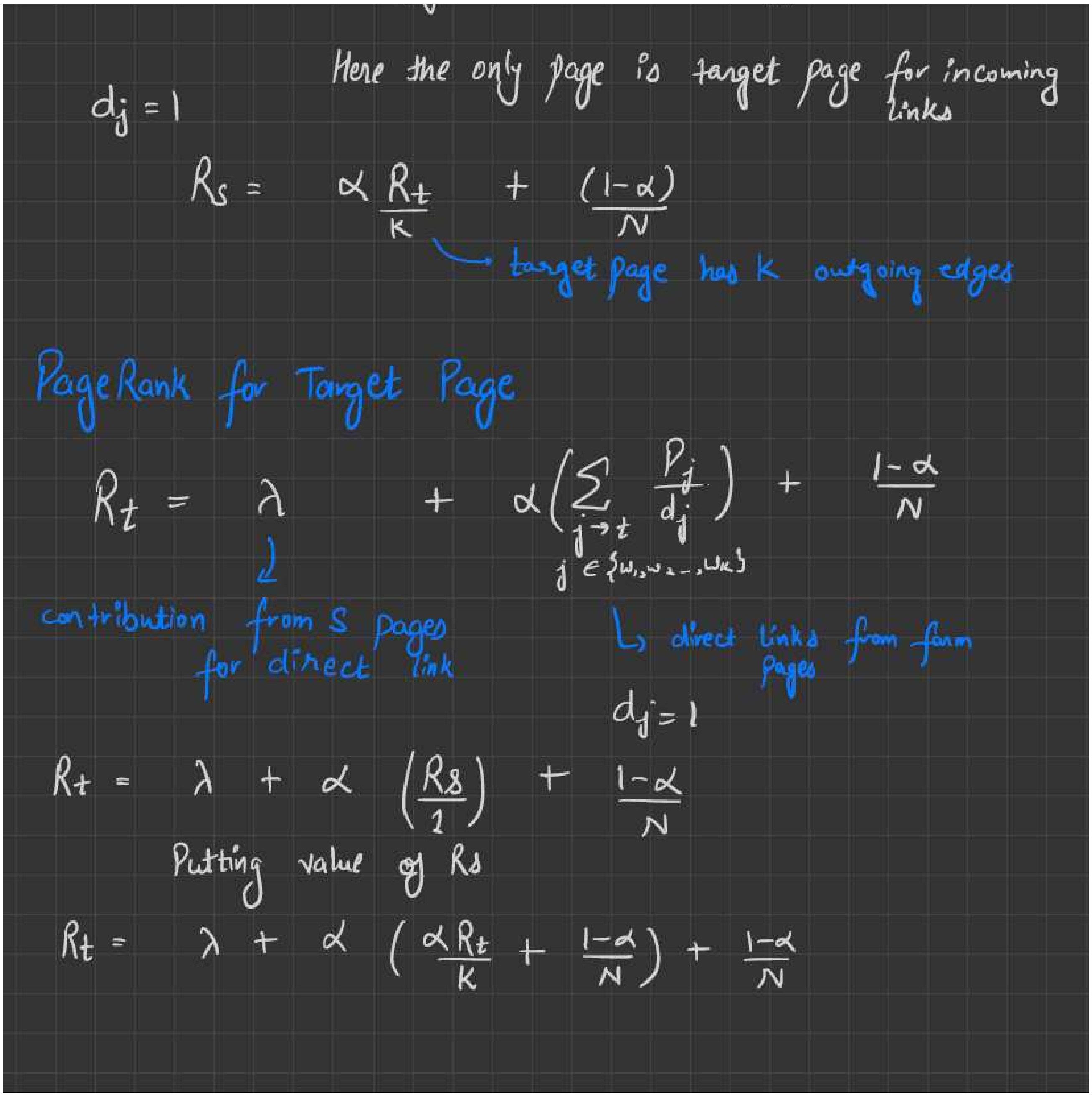
Let R_t be the page Rank of target page



PageRank for the spam farm pages $\{w_1, w_2, \dots w_k\}$

The incoming links to spam farm pages are only from target page.

$$R_s = \alpha \left(\sum_{j \to s} \left(\frac{P_j}{d_j} \right) \right) + \frac{(1-\alpha)}{N}$$



$d_j = 1$ Here the only page is target page for incoming links

$$R_s = \alpha \frac{R_t}{K} + \frac{(1-\alpha)}{N}$$

target page has K outgoing edges

PageRank for Target Page

$$R_t = \lambda + \alpha\left(\sum_{\substack{j \to t \\ j \in \{w_1, w_2, \dots, w_k\}}} \frac{P_j}{d_j}\right) + \frac{1-\alpha}{N}$$

contribution from S pages for direct link

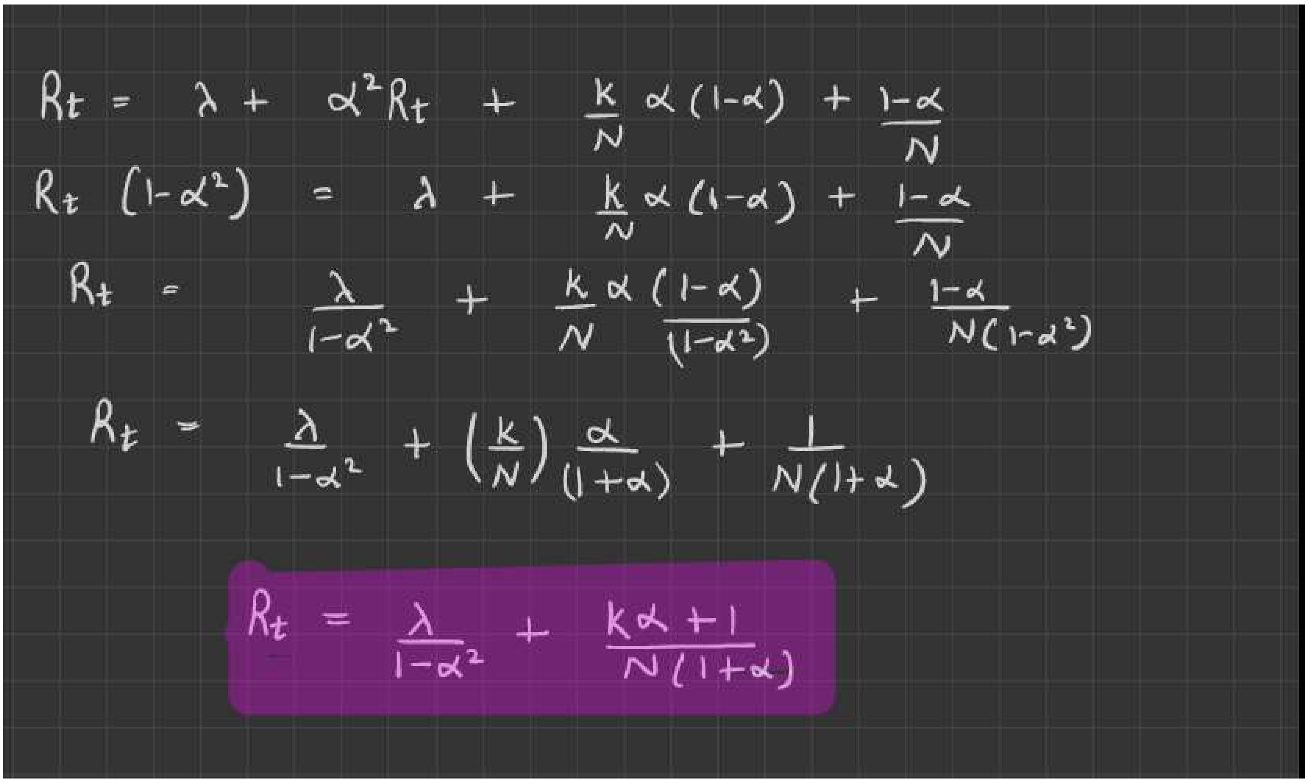
direct links from farm pages

$d_j = 1$

$$R_t = \lambda + \alpha\left(\frac{R_s}{1}\right) + \frac{1-\alpha}{N}$$

Putting value of R_s

$$R_t = \lambda + \alpha\left(\frac{\alpha R_t}{K} + \frac{1-\alpha}{N}\right) + \frac{1-\alpha}{N}$$



$$R_t = \lambda + \alpha^2 R_t + \frac{k}{N}\alpha(1-\alpha) + \frac{1-\alpha}{N}$$

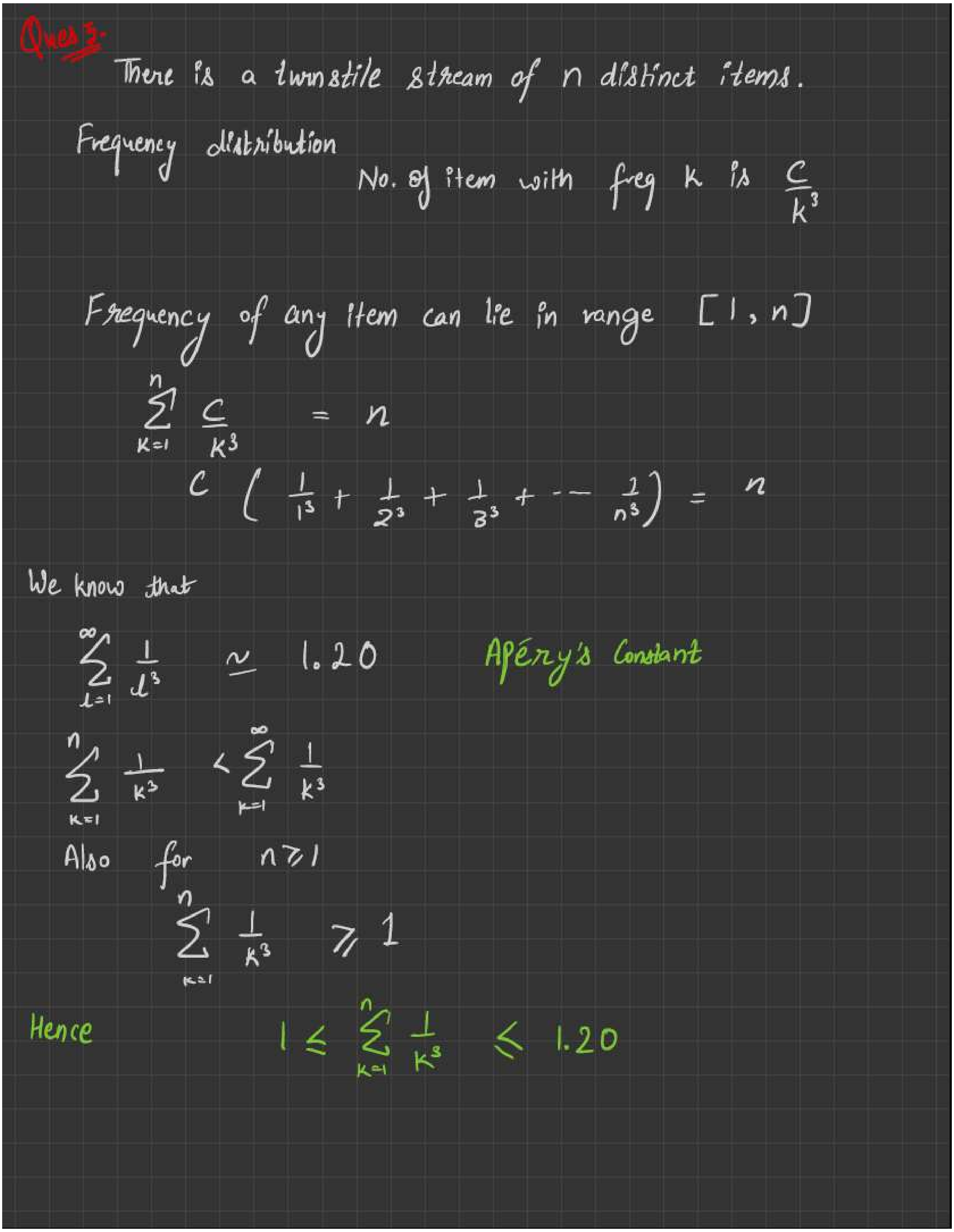
$$R_t(1-\alpha^2) = \lambda + \frac{k}{N}\alpha(1-\alpha) + \frac{1-\alpha}{N}$$

$$R_t = \frac{\lambda}{1-\alpha^2} + \frac{k}{N}\frac{\alpha(1-\alpha)}{(1-\alpha^2)} + \frac{1-\alpha}{N(1-\alpha^2)}$$

$$R_t = \frac{\lambda}{1-\alpha^2} + \left(\frac{k}{N}\right)\frac{\alpha}{(1+\alpha)} + \frac{1}{N(1+\alpha)}$$

$$\boxed{R_t = \frac{\lambda}{1-\alpha^2} + \frac{k\alpha+1}{N(1+\alpha)}}$$

Ques 3.



Ques 3.

There is a turnstile stream of n distinct items.

Frequency distribution

No. of item with freq k is $\frac{c}{k^3}$

Frequency of any item can lie in range $[1, n]$

$$\sum_{k=1}^{n} \frac{c}{k^3} = n$$

$$c\left(\frac{1}{1^3}+\frac{1}{2^3}+\frac{1}{3^3}+\cdots\frac{1}{n^3}\right) = n$$

We know that

$$\sum_{l=1}^{\infty} \frac{1}{l^3} \simeq 1.20$$

Apéry's Constant

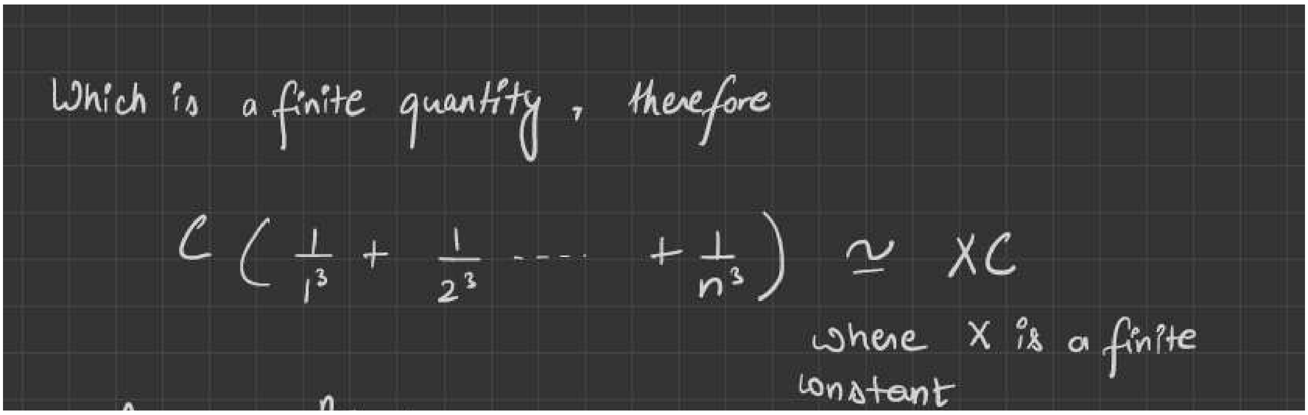
$$\sum_{k=1}^{n} \frac{1}{k^3} < \sum_{k=1}^{\infty} \frac{1}{k^3}$$

Also for $n \geq 1$

$$\sum_{k=1}^{n} \frac{1}{k^3} \geq 1$$

Hence

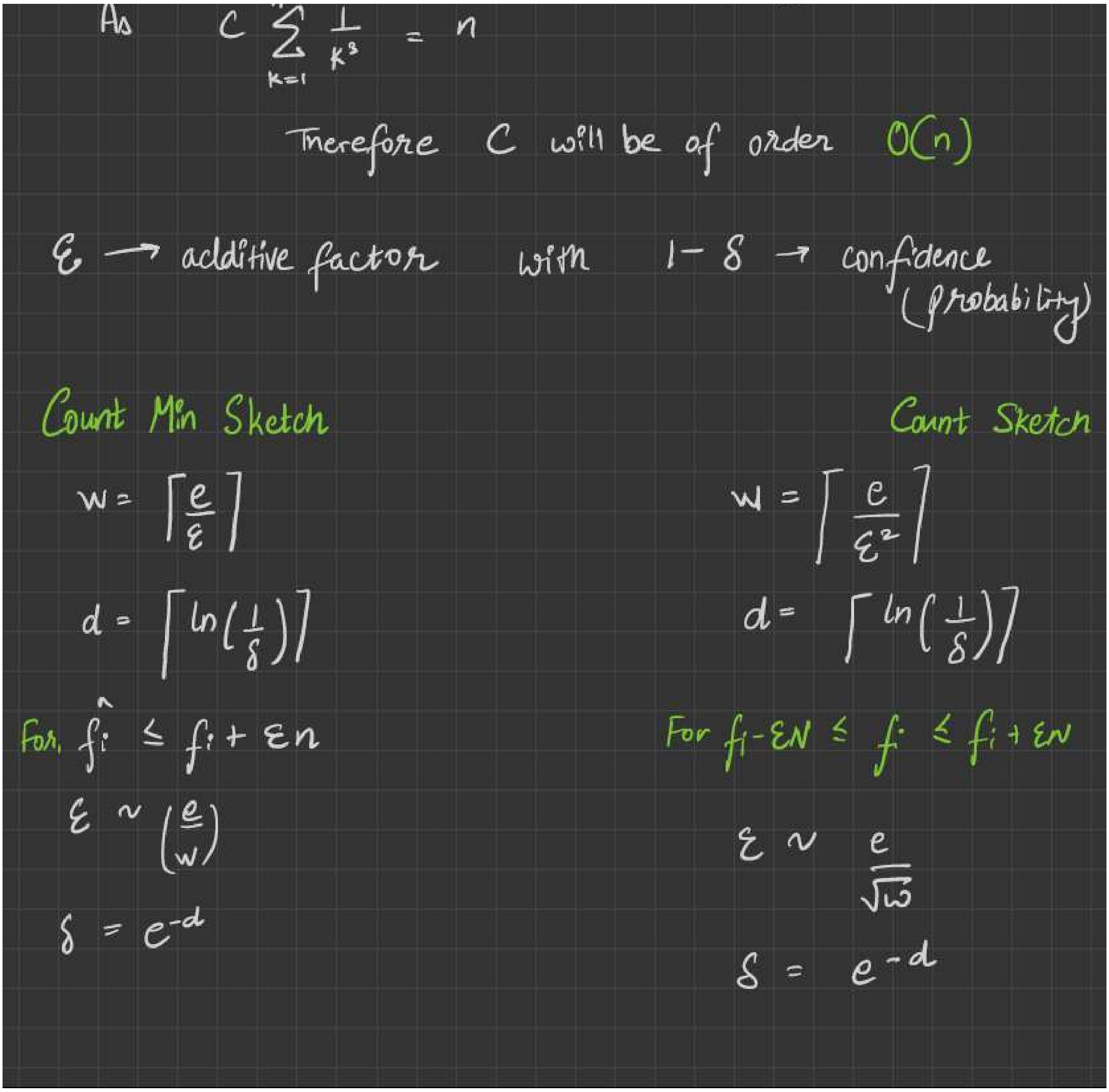
$$1 \leq \sum_{k=1}^{n} \frac{1}{k^3} \leq 1.20$$



Which is a finite quantity, therefore

$$C\left(\frac{1}{1^3}+\frac{1}{2^3}\cdots+\frac{1}{n^3}\right) \simeq XC$$

where X is a finite constant



As $C \sum_{k=1} \frac{1}{k^3} = n$

Therefore C will be of order $O(n)$

$\varepsilon \rightarrow$ additive factor with $1-\delta \rightarrow$ confidence (probability)

Count Min Sketch

$w = \lceil \frac{e}{\varepsilon} \rceil$

$d = \lceil \ln(\frac{1}{\delta}) \rceil$

For, $\hat{f_i} \le f_i + \varepsilon n$

$\varepsilon \sim (\frac{e}{w})$

$\delta = e^{-d}$

Count Sketch

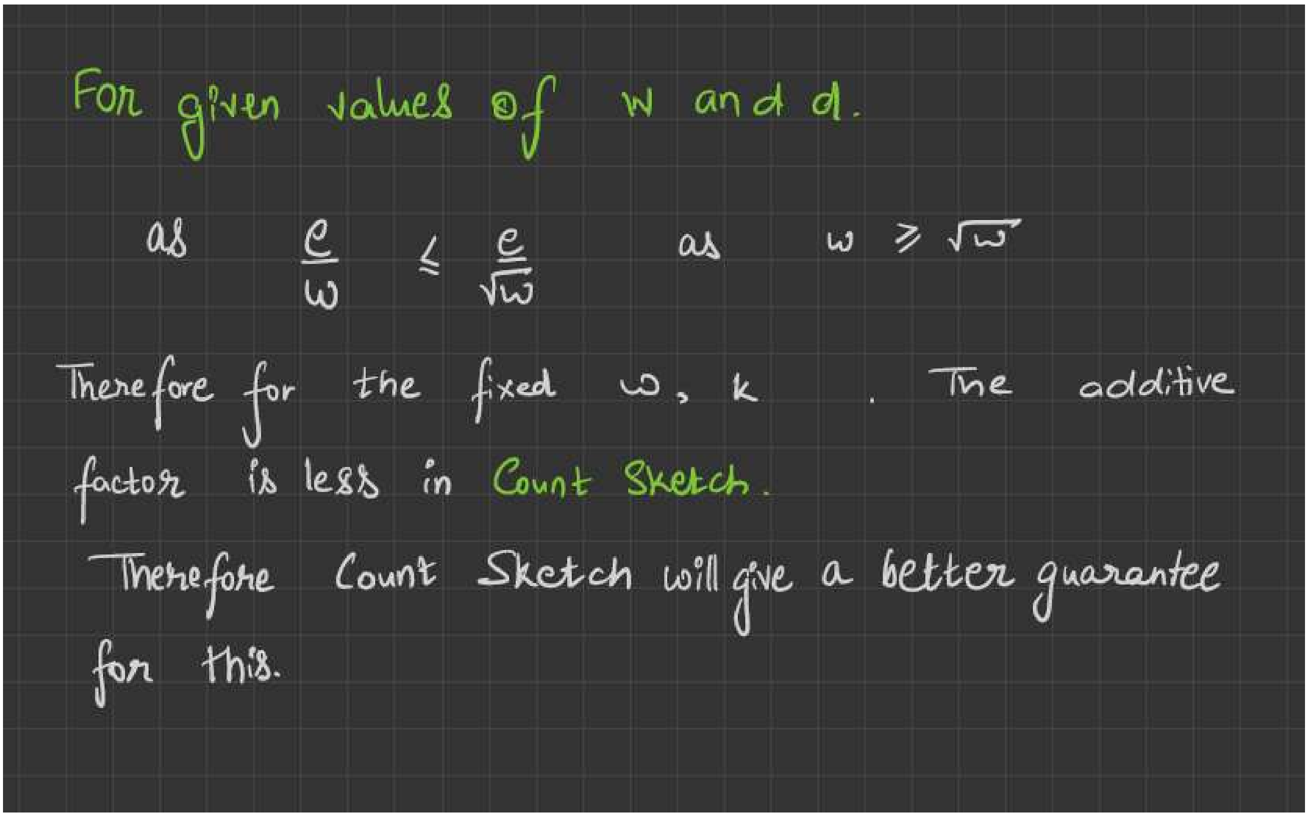
$w = \lceil \frac{e}{\varepsilon^2} \rceil$

$d = \lceil \ln(\frac{1}{\delta}) \rceil$

For $f_i - \varepsilon N \le f_i \le f_i + \varepsilon N$

$\varepsilon \sim \frac{e}{\sqrt{w}}$

$\delta = e^{-d}$



For given values of w and d.

as $\frac{e}{w} \leq \frac{e}{\sqrt{w}}$ as $w \geqslant \sqrt{w}$

Therefore for the fixed w, k . The additive factor is less in Count Sketch.

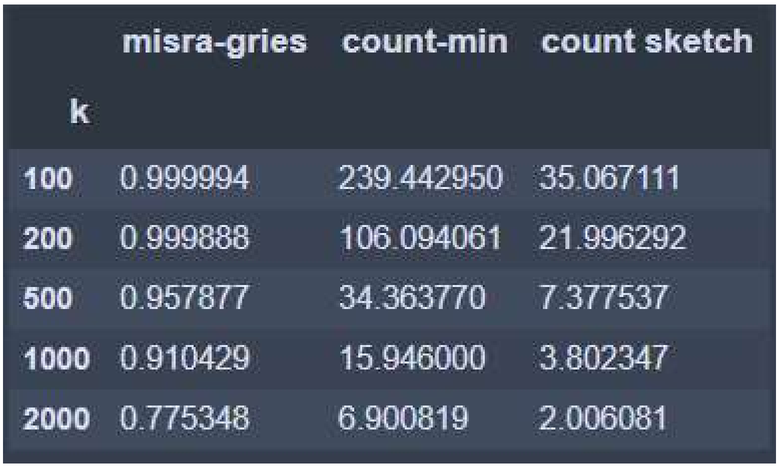
Therefore Count Sketch will give a better guarantee for this.

Ques 4.

File: ques4.ipynb

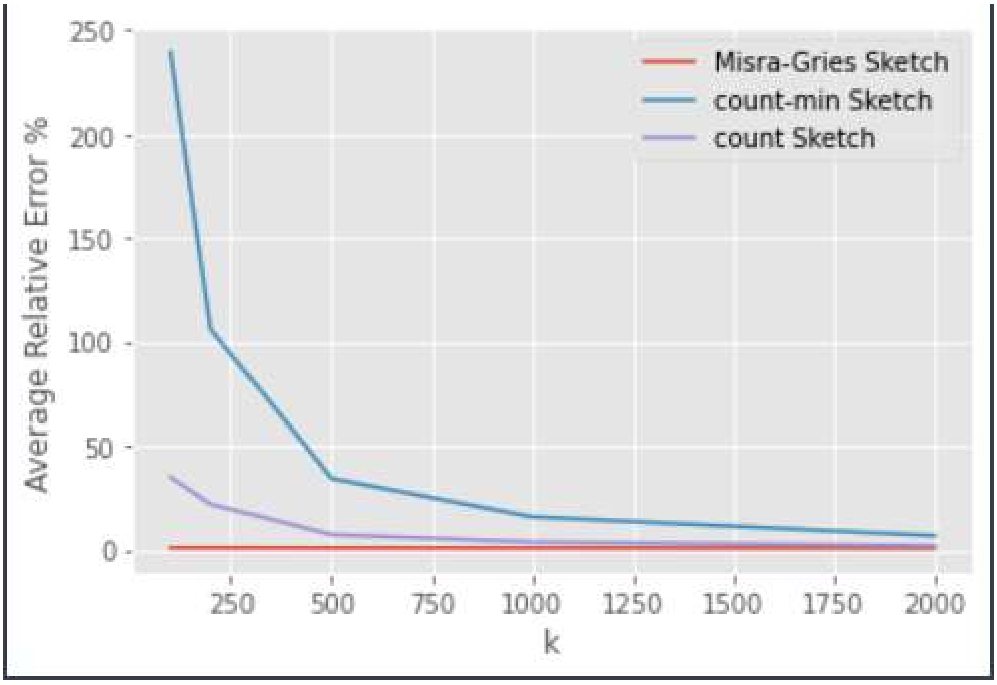
Colab Link: https://colab.research.google.com/drive/1b5oGbPErcwkGiypdSz1tUTgNGijX8978?usp=sharing

Average Relative Error Percentage Table



	misra-gries	count-min	count sketch
k			
100	0.999994	239.442950	35.067111
200	0.999888	106.094061	21.996292
500	0.957877	34.363770	7.377537
1000	0.910429	15.946000	3.802347
2000	0.775348	6.900819	2.006081

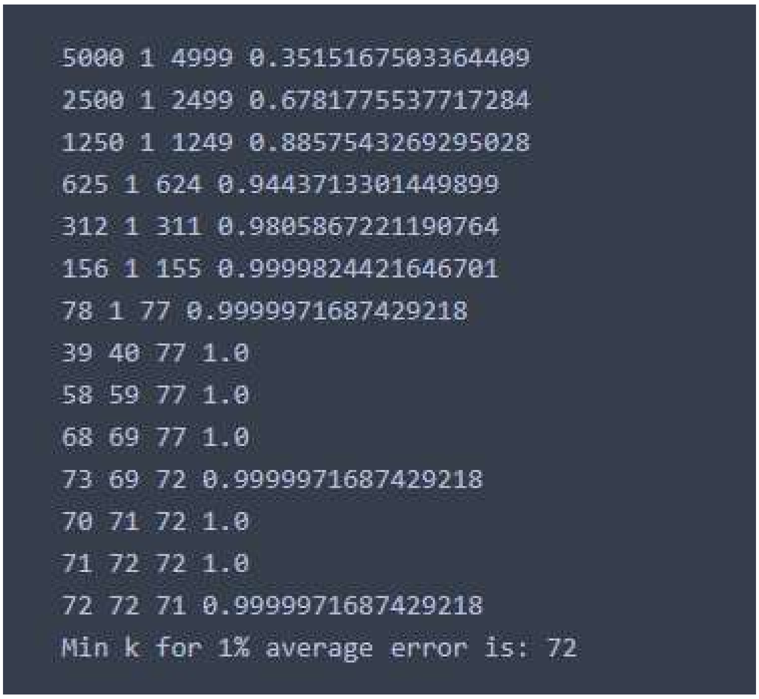
Average Relative Error Plot



minimum w or k for the average error less than 1%.

Misra-Gries Sketch

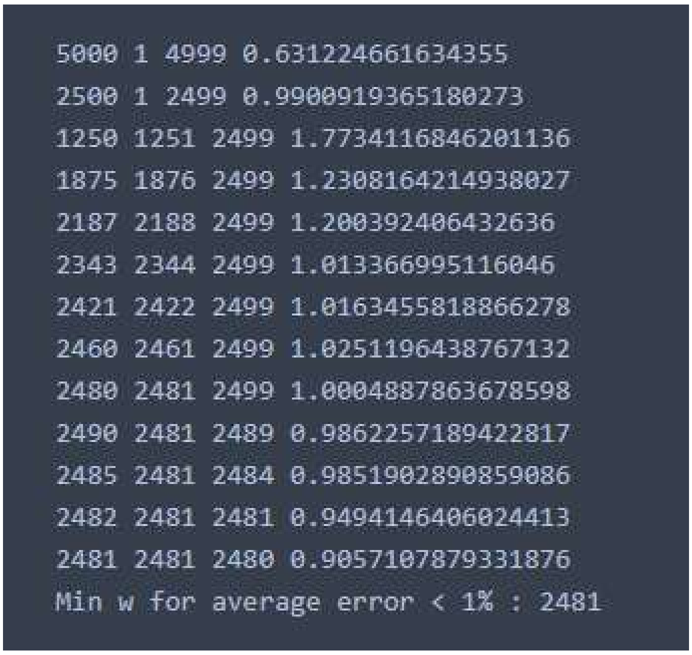
Min k for 1% average error is: **72**



```
5000 1 4999 0.3515167503364409
2500 1 2499 0.6781775537717284
1250 1 1249 0.8857543269295028
625 1 624 0.9443713301449899
312 1 311 0.9805867221190764
156 1 155 0.9999824421646701
78 1 77 0.9999971687429218
39 40 77 1.0
58 59 77 1.0
68 69 77 1.0
73 69 72 0.9999971687429218
70 71 72 1.0
71 72 72 1.0
72 72 71 0.9999971687429218
Min k for 1% average error is: 72
```

count-min Sketch

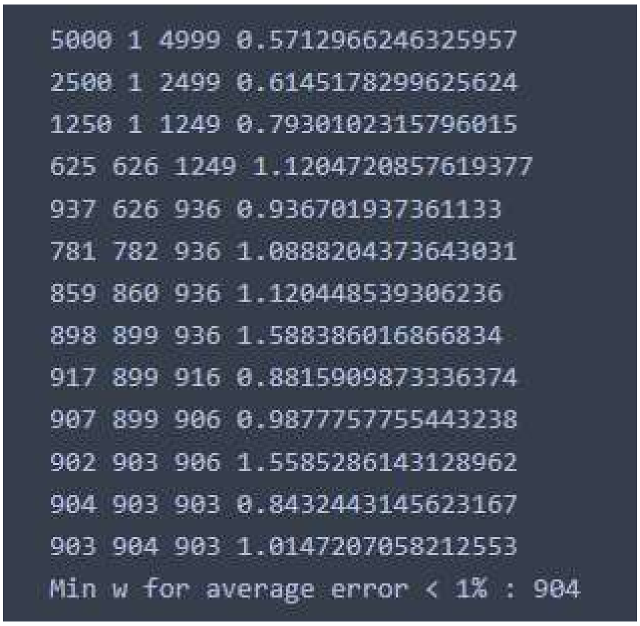
Min w for average error < 1% : **2481**



```
5000 1 4999 0.631224661634355
2500 1 2499 0.9900919365180273
1250 1251 2499 1.7734116846201136
1875 1876 2499 1.2308164214938027
2187 2188 2499 1.200392406432636
2343 2344 2499 1.013366995116046
2421 2422 2499 1.0163455818866278
2460 2461 2499 1.0251196438767132
2480 2481 2499 1.0004887863678598
2490 2481 2489 0.9862257189422817
2485 2481 2484 0.9851902890859086
2482 2481 2481 0.9494146406024413
2481 2481 2480 0.9057107879331876
Min w for average error < 1% : 2481
```

Count Sketch

Min w for average error < 1% : **904**



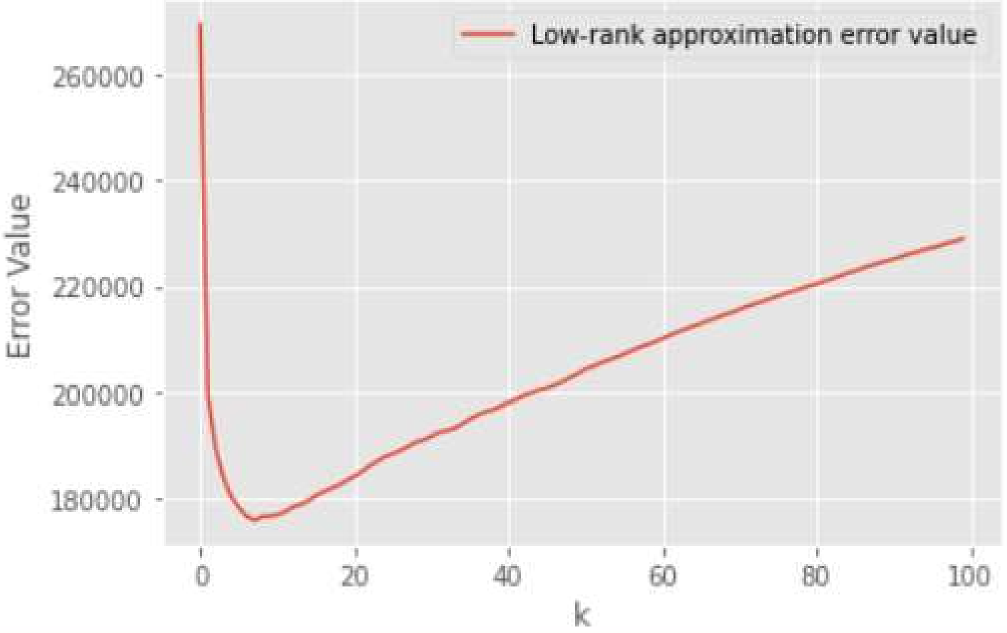
```
5000 1 4999 0.5712966246325957
2500 1 2499 0.6145178299625624
1250 1 1249 0.7930102315796015
625 626 1249 1.1204720857619377
937 626 936 0.936701937361133
781 782 936 1.0888204373643031
859 860 936 1.120448539306236
898 899 936 1.588386016866834
917 899 916 0.8815909873336374
907 899 906 0.9877757755443238
902 903 906 1.5585286143128962
904 903 903 0.8432443145623167
903 904 903 1.0147207058212553
Min w for average error < 1% : 904
```

Ques 5.

File : ques5.ipynb

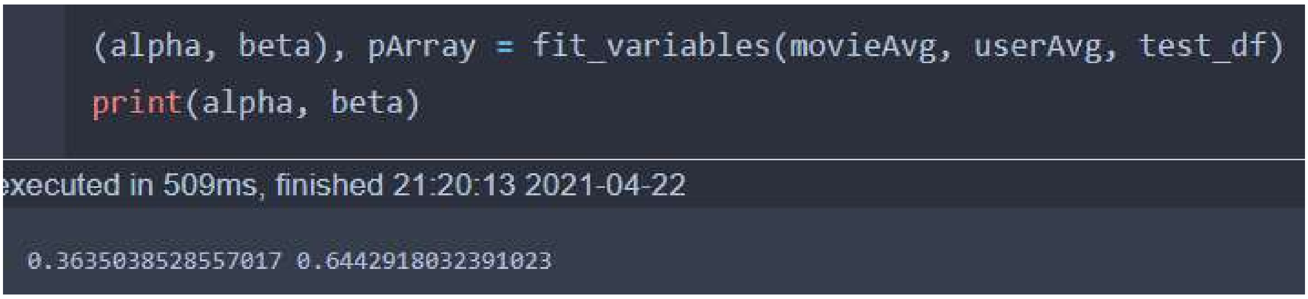
Colab Link: https://colab.research.google.com/drive/1q__OQZLYr9lCjcPjrD7Scmk0Ykqdgi5-?usp=sharing

Low-Rank Approximation Error vs k plot



BaseLine Algorithm

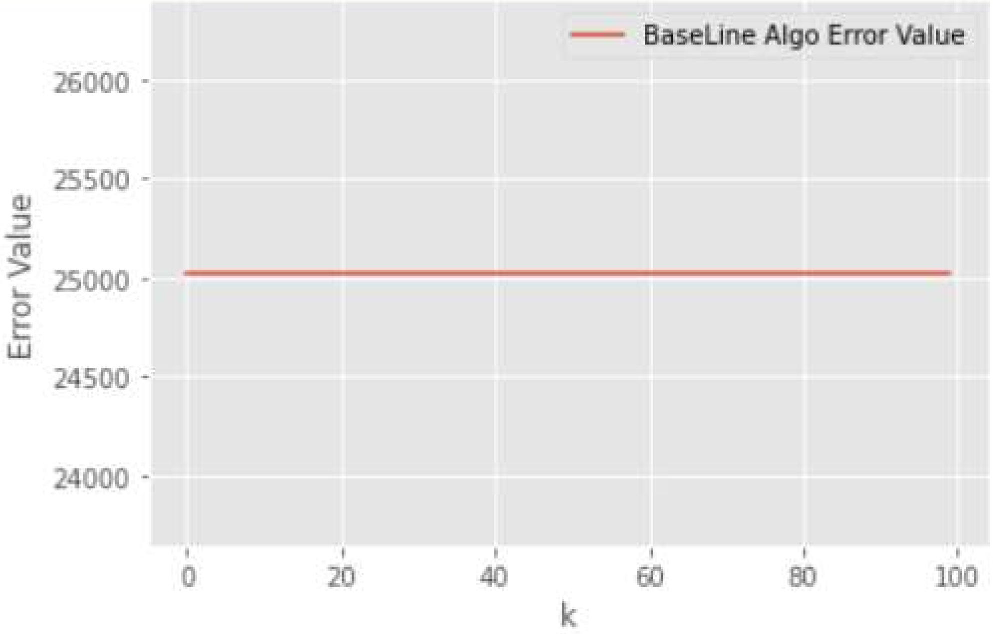
alpha = 0.3678348482223803
beta = 0.639915746842154



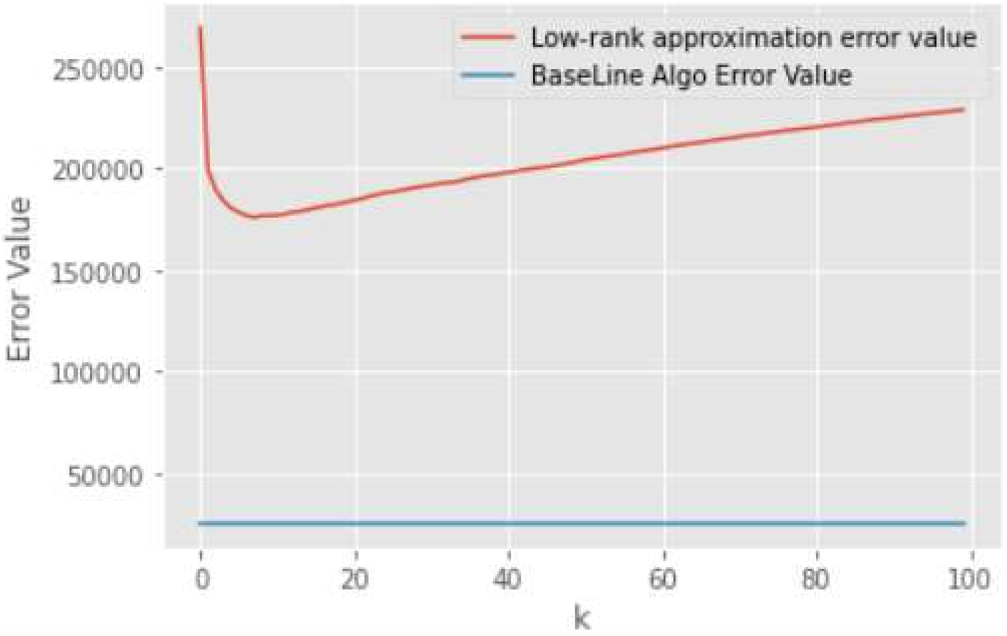
```
(alpha, beta), pArray = fit_variables(movieAvg, userAvg, test_df)
print(alpha, beta)
```

executed in 509ms, finished 21:20:13 2021-04-22

```
0.3635038528557017 0.6442918032391023
```



Comparison


Low-rank approximation error value
BaseLine Algo Error Value
Error Value
250000
200000
150000
100000
50000
0
20
40
60
80
100
k