L1 Norm SVD based Ranking Scheme: A Novel Method in Big Data Mining

Rahul Aedula[1], Yashasvi Madhukumar[1], Snehanshu Saha[1], Archana Mathur[2], Kakoli Bora[1] and Surbhi Agrawal[1]

[1]PES Institute of Technology, Bangalore South Campus, Bangalore, India
rahulaedula95@gmail.com, yash21496@gmail.com, snehanshusaha@pes.edu,
k_bora@pes.edu,surbhiagrawal@pes.edu
[2]Indian Statistical Institute, 8th Mile, Mysore Road, Bangalore, India
archana_plp@isibang.ac.in

Abstract. Scientometrics deals with analyzing and quantifying works in science, technology, and innovation. It is a study that focuses on quality rather than quantity. The journals are evaluated against several different metrics such as the impact of the journals, scientific citation, SJR, SNIP indicators as well as the indicators used in policy and management context. The practice of using journal metrics for evaluation involves handling a large volume of data to derive useful patterns and conclusions. These metrics play an important role in the measurement and evaluation of research performance. Due to the fact that most metrics are being manipulated and abused, it becomes essential to judge and evaluate a journal by using a single metric or a reduced set of significant metrics. We propose l_1-norm Singular Value Decomposition(l_1-SVD) to efficiently solve this problem. We evaluate our method to study the emergence of a new journal, Astronomy and Computing, by comparing it with 46000 journals chosen from the fields of Computing, Informatics, Astronomy and Astrophysics.

Keywords: l_1-norm, Sparsity Norm, Singular Value Decomposition, Journal Ranking, Astronomy and Computing, Big Data

1 INTRODUCTION

Scientometrics evaluates the impact of the results of scientific research by placing focus on the work's quantitative and measurable aspects. Statistical mathematical models are employed in this study and evaluation of journals and conference proceedings to asses their quality. The implosion of journals and conference proceedings in the science and technology domain coupled with the insistence of different rating agencies and academic institutions to use journal metrics for evaluation of scholarly contribution present a big data accumulation and analysis problem. This high volume of data requires an efficient metric system for fair rating of the journals. However, certain highly known and widely used metrics such as the Impact Factor and the H factor have been misused lately through practices like non-contextual self-citation, forced citation, copious-citation etc. [7] Thus, the way this volume of data is modeled needs improvement because it influences the evaluation and processing of this data to draw useful conclusions. One effective way to deal with this problem is to characterize a journal by a single metric or a reduced set of metrics that hold more significance. The volumes of data scraped from various sources are organized as a rectangular mxn matrix where m is the rows representing the number of articles in a journal and n columns of various Scientometric parameters. An effective dimensionality and rank reduction technique such as the Singular Value Decomposition (SVD) applied on the original data matrix not only helps to obtain a single ranking metric(based on the different evaluation parameters enlisted as various columns) but also identifies pattern used for efficient analysis of the big data. Apache Mahout, Hadoop, Spark, R, Python, Ruby are some tools that can be used to implement SVD and other similar dimensionality reduction techniques. [5]

One notable characteristic of the Scientometric data matrix is its sparsity. The matrix is almost always rectangular and most metric fields (columns) do not apply to many of the articles(rows). For instance, a lot of journals may not have patent citations. Similarly, a number of other parameters might not apply to a journal as a whole. Usually, n and m differ from each other by a good integer difference. Thus, by virtue of this sparsity, the efficiency of the SVD algorithms can be enhanced when coupled with norms like l_1-norm, l_2-norm or the group norms. In general, both sparsity and structural sparsity regularization methods utilize the assumption that the output $\boldsymbol{Y}$ can be described by a reduced number of input variables in the input space $\boldsymbol{X}$ that best describe the output. In addition to this, structured sparsity regularization methods can be extended to allow optimal selection over groups of input variables in $\boldsymbol{X}$.

2 The depths of Dimensionality Reduction

Dimensionality reduction has played a significant role in helping us ascertain results of the analysis for voluminous data set [2]. The propensity to employ such methods comes from the phenomenal growth of data and the velocity at which it is generated. Dimensional reduction such as Singular Value Decomposition and Principle component Analysis solves such big data problems by means of extracting more prominent features and obtaining a better representation of the data. This data tends to be much smaller to store and much easier to handle to perform further analysis. These dimensionality reduction methods are very often found in most of the tools which handle large data sets and perform rigorous data analysis. Such tools include Apache Mahout, Hadoop, Spark, R, Python etc. The ease of employing such methods is directly dependent on the performance of such tools to be able to compute and assess the results quickly and store it efficiently, all this while managing resources available at an optimal rate. The divergence in the methods used in these tools to compute such algorithms gives us scope to study and evaluate such case scenarios and help us choose the right kind of tools to perform these tasks.

2.1 *PCA*

Principal Component Analysis, a technique mostly used in statistics to transform a set of observations of possibly correlated variables into a set of linearly uncorrelated variables called as Principal Components. These Principal Components are the representation of the underlying structure in the data or the directions in which the variance is more and where the data is more concentrated.

The procedure lays emphasis on variation and identification of strong patterns in the dataset. PCA extracts a low dimensional set of features from a higher dimension dataset, simultaneously serving the objective of capturing as much useful information as possible. PCA is most commonly implemented in two ways:-

- **Eigenvalue Decomposition** of a data covariance(or correlation) matrix into canonical form of eigenvalues and eigenvectors. However, only square/diagonalizable matrices can be factorized this way and hence it also takes the name Matrix Diagonalization.

- **Singular Value Decomposition** of the initial higher dimension matrix. This approach is relatively more suitable for the problem being discussed since it exists for all matrices: singular, non-singular, dense, sparse, square or rectangular.

2.2 *Singular Value Decomposition*

Singular Value Decomposition is the factorization of a real or complex matrix. Large scale of Scientometric data is mined using suitable web scraping techniques and is modeled as a matrix in which the rows represent the articles in a journal published over the years, and the columns represent various Scientometrics or indicators proposed by experts of evaluation agencies [3]. The original data matrix, say $\mathbf{A}$ of dimension mxn and rank k is factorized into three unique matrices $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{W^H}$.

- $\mathbf{U}$ - Matrix of Left Singular Vectors of dimension mxr

- $\mathbf{V}$ - Diagonal matrix of dimension rxr containing singular values in decreasing order along the diagonal

- $\mathbf{W^H}$ - Matrix of Right Singular Vectors of dimension nxr. The Hermitian, or the conjugate transpose of $\mathbf{W}$ is taken, changing its dimension to rxn and hence the original dimension of the matrix is maintained after the matrix multiplication. In this case of Scientometrics, since the data is represented as a real matrix, Hermitian transpose is simply the transpose of $\mathbf{W}$.

r is a very small number numerically representing the approximate rank of the matrix or the number of "concepts" in the data matrix $\mathbf{A}$. *Concepts* refer to latent dimensions or latent factors showing the association between the singular values and individual components [3]. The choice of r plays a vital role in deciding the accuracy and computation time of the decomposition. If r is equal to k, then the SVD is said to be a Full Rank Decomposition of $\mathbf{A}$. Truncated SVD or Reduced Rank Approximation of $\mathbf{A}$ is obtained by setting all but the first r largest singular values equal to zero and using the first r columns of $\mathbf{U}$ and $\mathbf{W}$ [4].

Therefore, choosing a higher value of r closer to k would give a more accurate approximation whereas a lower value would save a lot of computation time and increase efficiency.

2.3 *Regularization Norms*

In the case of Big Data, parsimony is central to variable and feature selection, which makes the data model more intelligible and less expensive in terms of processing.

l_p-norm of a matrix or vector $\mathbf{x}$, represented as $||\mathbf{x}_p||$ is defined as, $||\mathbf{x}_p|| = \sqrt[p]{\Sigma_i |x|_i^p}$ i.e the p$^{\text{th}}$ root of summation of all the elements raised to the power p. Hence, by definition, l_1 norm $= ||\mathbf{x}||_1 = \Sigma_i |x|_i$

Sparse approximation, inducing structural sparsity as well as regularization is achieved by a number of norms, the most common ones being l_1 norm and the mixed group l_1-l_q norm. The relative structure and position of the variable in the input vector, and hence the inter-relationship between the variables is inconsequential as a variable is chosen individually in l_1 regularization. Prior knowledge aids in improving the efficacy of estimation through these techniques.

The l_1 norm concurs to only the cardinality constraint and is unaware to any other information available about the patterns of non-zero coefficients.[1]

2.4 *Sparsity via the l_1 norm*

Most variable or feature selection problems are presented as combinatorial optimization problems. Such problems focus on selecting the optimal solution through a discrete, finite set of feasible solutions. Additionally, l_1 norm turns these problems to convex problems after dropping certain constraints from the overall optimization problem. This is known as convex relaxation. Convex problems classify as the class of problems in which the constraints are convex functions and the objective function is convex if minimizing, or concave if maximizing.

l_1 regularization for sparsity through supervised learning involves predicting a vector $\mathbf{y}$ from a set of usually reduced values/observations consisting a vector in the original data matrix $\mathbf{x}$. This mapping function is often known as the hypothesis $\mathbf{h : x \rightarrow y}$. To achieve this, we assume there exists a joint probability distribution P(x,y) over $\mathbf{x}$ and $\mathbf{y}$ which helps us model anomalies like noise in the predictions.

In addition to this, another function known as a loss function L(y',y) is required to measure the difference in the prediction y'=h(x) from the true result y. Consider the resulting vectors consisting of the predicted value and the true value to be $\mathbf{y'}$ and $\mathbf{y}$ respectively. A characteristic called *Risk*, R(h) associated with loss function, and hence in turn with the hypothesis-h(x) is defined as the expectation of the loss function.

$$R(h) = \mathbf{E}[L(y', y)] = \int L(y', y)\, dP(x, y)$$

Thus, the hypothesis chosen for mapping should be such that the risk, R(h) is minimum. This refers to as risk minimization. However, in usual cases, the joint probability distribution of the problem in hand, P(x,y) is not known. So, an approximation called *empirical risk* is computed by taking the average of the loss function of all the observations. Empirical Risk is given by :

$$R_{emp}(h) = \frac{1}{n} \sum_{i=1}^{n} L(\mathbf{y'}_i, \mathbf{y}_i)$$

The empirical risk minimization principle states that the hypothesis(h') selected must be such it that reduces the empirical risk $R_{emp}(h)$:

$$h' = \min_h R_{emp}(h)$$

While mapping observations x in n dimensional vector $\mathbf{x}$ to outputs y in vector $\mathbf{y}$, we consider p pairs of data points - $(\mathbf{x}_i, \mathbf{y}_i) \in \mathbb{R}^n \times \mathbf{y}$ where i = 1,2...p.

Thus the optimization problem for the data matrix in Scientometrics takes the form:

$$\min_{\mathbf{w} \in \mathbb{R}^n} \frac{1}{p} \sum_{i=1}^{p} L(\mathbf{y'}_i, \mathbf{w}^T \mathbf{x}_i) + \lambda \Omega(\mathbf{w})$$

L is a loss function which can either be square loss for least squares regression, $L(y', y) = \frac{1}{2}(y' - y)^2$, or a logistic loss function. Now, the problem thus takes the form:

$$\min_{\mathbf{w} \in \mathbb{R}^n} ||\mathbf{y'} - \mathbf{A}\mathbf{w}||^2$$

Since the variables in the vector space/groups can overlap, it is ideal to choose $\Omega(\mathbf{w})$ to be a group norm for better predictive performance and structure. The m rows of data matrix $\mathbf{A}$ are treated as vectors or groups(g)

of these variables, forming a partition equal to the vector dimension, $[1{:}n]$. If $\boldsymbol{G}$ is the set of all these groups and d_g is a scalar weight indexed by each group g, the norm is said be a l_1-l–q norm where $q \in [2, \infty)$. [1]

$$\Omega(\mathbf{w}) = \sum_{g \in \boldsymbol{G}} d_g ||\mathbf{w}_g||_q$$

The choice of the indexed weight d_g is critical because it is responsible for the discrepancies of sizes between the groups. It must also compensate for the possible penalization of parameters which can increase due to high-dimensional scaling. The factors that affect the selection are the choice of q in the group norm and the consistency that is expected of the result. In addition to this, accuracy and efficiency can be enhanced by weighing each coefficient in a group rather than weighing the entire group as a whole. The initial sparse data matrix is first manipulated using the l_1-norm. [1]

3 Methodology

An estimate of a journal's scholastic indices is necessary to judge its effective impact. The nuances of scientometric factors such as Total Citation Count and Self-citation Count come into play when deciding the impact of a journal. However, these factors unless considered in ideal circumstances don't by themselves become a good indicator to represent the importance of a journal. Many anomalies arise when considering these indices directly which may misrepresent or falsify a journal's true influence. The necessity to use these indices in context with a ranking algorithm is imperative to better utilize these indices. The resulting transformation of l_1-norms gives rise to a row matrix which is of the length equal to the number of features of the pristine Scientometric data. This row matrix effectively represents the entire dataset at any given iteration. The application of the Singular Value Decomposition operation on this row matrix is key in determining the necessary norm values to remove through a recursive approach.

The *singval* array contains the Normalized Singular Values of all the individual l_1-norm transformed columns. These values act as scores while addressing the impact of any given journal. In the context of Singular Values the one with the lowest *singval* score is the most influential journal. Utilizing these scores we can for-

Algorithm 1 Recursive l_1-norm SVD

```
A ← Input Transposed Feature Matrix A
procedure LASSO
    row_matrix ← Coefficents of Lasso Regression
    return row_matrix
procedure SVD
    U,Σ,V ← Matrices of SVD
    return Σ
procedure NORMALIZE
    Norm_Data ← Normalized using l1-norms
    return Norm_Data
procedure RECURSIVE
    L1_row ← LASSO(A)
    singval [] ← SVD(L1_row)
    Row_Norm ← Normalize(L1_row)
    Col_Norm ← Normalize(All columns of A)
    Col_i ← Closest Col_Norm Value to Row_Norm
    Delete Col_i from A
    goto RECURSIVE
```

mulate a list of Journals which give preference to subtle factors such as high or low Citation Counts and give an appropriate ranking. Identifying the influential journals from a column norm and contrasting it with the Singular values is the equivalent of recursively eliminating the a low impact journal by comparing it's Singular Value to its Frobenius norm. This allows the algorithm to repeatedly eliminate the journals and find the score simultaneously to give a more judicious ranking system. Our method is different from the SCOPUS journal rank (SJR) algorithm. The SJR indicator computation uses an iterative algorithm that distributes prestige values among the journals until a steady-state solution is reached. The method is similar to eigen factor score [9] where the score is influenced by the size of the journal so that the score doubles when the journal doubles in size. Our method, on the contrary, adopts a recursive approach and doesn't assume initial prestige values. Therefore, the eigen factor approach may not be suitable for evaluating the short-term influence of peer-reviewed journals. In contrast, our method works well under such restrictions.

4 The Big Data Landscape

The appeal of modern-day computing is its flexibility to handle volumes of data through an aspect of coordination and integration. Advancements in Big Data frameworks and technologies has allowed us to break the barriers of memory constraints for computing and implement a more scalable approach to employ methods and algorithms. [5] The aforementioned journal ranking scheme is one such algorithm which thrives under the improvements made to scalability in Big Data. With optimized additions such as Apache Spark to the distributed computing family, the enactment of l_1 Regularization and Singular Value Decomposition has reached an all new height. Implementing the SVD algorithm with the help of Spark can not only improve spatial efficiency but temporal as well. The l_1-norm SVD scheme utilizes the SVD and regularization implementation of *ARPACK* and *LAPACK* libraries along with a cluster setup to enhance the speed of execution by a magnitude of at least three times depending on the configuration. Collecting data is also a very important aspect of Big Data topography. The necessity of a cluster based system is rendered useless without the requisite data to substantiate it. Scientometric data usually deals with properties of the journals such as Total Citation, Self-Citation etc. This data could be collected using Web Scraping methodologies but also can be found by most journal ranking organizations, available for open source use; SCOPUS and SCIMAGO. For the l_1-norm SVD scheme, we used SCOPUS as it had an eclectic set of features which were deemed appropriate to showcase the effectiveness of the algorithm. The inclusion of the two important factors such as CiteScore and SJR indicators gave a better enhancement over just considering one over the other. For more information about the data and code used to develop this algorithm (please refer to [8], Github repository of the project).

4.1 *Case Study*: Astronomy and Computing

SCOPUS and SCIMAGO hold some of the best journal ranking systems to this day, using their CiteScore and SJR indicators respectively to rank journals. However, due to the manner in which both these indicators are considered, it is often the case that the ranking might not display the true potential of a specified scientific journal. To demonstrate this we considered the case of the Journal *Astronomy and Computing* within the context of SCOPUS Journals in the relevant domain of Astronomy and Astrophysics.

The primary focus of this case study is to determine where the Journal *Astronomy and Computing* stand with respect other journals which were established prior to it. The algorithm also tests the validity of the ranking and suggests an alternative rank which used a more holistic approach towards the features.

Journal Name	L1 Scheme Rank	SJR based Rank	Year
Astronomy and Computing	39	31	2013
Astronomy and Astrophysics Review	40	5	1999
Radiophysics and Quantum Electronics	41	51	1969
Solar System Research	42	48	1999
Living Reviews in Solar Physics	43	3	2005
Astrophysical Bulletin	44	45	2010
Journal of Astrophysics and Astronomy	45	55	1999
Revista Mexicana de Astronomia y Astrofisica	46	23	1999
Acta Astronomica	47	20	1999
Journal of the Korean Astronomical Society	48	32	2009
Cosmic Research	49	58	1968
Geophysical and Astrophysical Fluid Dynamics	50	46	1999
New Astronomy Reviews	51	12	1999
Kinematics and Physics of Celestial Bodies	52	65	2009
Astronomy and Geophysics	53	67	1996
Chinese Astronomy and Astrophysics	54	72	1981

Table 1. Case Study: Astronomy and Computing, SJR and L1-SVD ranks

Using the publicly available SCOPUS dataset we implemented the aforementioned l_1-norm SVD scheme to rank all its corresponding journals and simultaneously determine the potency of the algorithm. SCOPUS contains approximately around 46k Journals listed in different domains. Discarding few redundancies, SCOPUS effectively covers a large range of metrics and provides adequate resources for verification. For this demonstration, we have considered SCOPUS's 7 different metrics to be used as features in our algorithm. These features include *Citation Count*, *Scholarly Output*, *SNIP*, *SJR*, *CiteScore, Percentile and Percent Cited.*

To cross verify the results of the algorithm they were compared to SJR based ranking of SCIMAGO to articulate the discrepancies. The l_1-norm SVD scheme worked brilliantly in rating the journals and approached

the data in a more wholesome sense. The result was a ranking system which ranked *Astronomy and Computing* much higher than most older journals and also at the same time highlighting the niche prominence of the particular journal. Similarly, this method also highlighted the rise of other journals which were underrepresented due to the usage of the aforementioned SCOPUS and SCIMAGO indicators. This was method was largely successful in rectifying the rank of such journals. This l_1-norm SVD scheme can be extrapolated to other data entries as well. It can also be used to study the impact of individual articles. Utilizing similar features such as Total Citation, Self Citation, and NLIQ. The algorithm can be used to rank articles within a journal with great accuracy along with a holistic consideration.

4.2 *Contrasting Performances of l_1 and l_2 Norms*

Being recursive in nature the Norm-based algorithms are subjected to some lapse while parallelizing its execution. However, they can be improved by using the right kind of suitable norm to enhance its running time. The decision of using l_1-norm over the l_2-norm was made because of a pragmatic choice for the following recursive scheme. The facet of the l_1-norm to use a loss function over the l_2-norm's squared data approach proves to be significantly better in structuring the data for a high-density computation. This type of method allows the overall dataset to reduce to a row matrix the size of the smallest dimension of the original data. This gives the added benefit of having a very consistent execution time and scale accordingly with the increase in data size.

Norm	**Time per row**
l_1 Norm	0.172s
l_2 Norm	0.188s

Table 2. Performance time for a row matrix of size 46k.

The execution time mentioned in Table 2 of this article gives the time-based performance of the different norms. This will only get significant with the increase in the size of the rows. This dereliction in parallelization can be compensated by the expected speed increase in the execution of the l_1-norm and SVD routines in a cluster setup. Optimized settings like Apache Spark which uses the aforementioned *LAPACK* and *ARPACK* libraries are able to boost the speed even further. The biggest benefit of opting such Big Data settings is that by increasing the size of the cluster the overall speed of the algorithm also scales appropriately.

Data Framework	**Overall Time**
Python	2hrs +
R	58 mins
L1 SVD	15 mins

Table 3. Performance time for SVD of size 100k X 100k.

Table 3 indicates the performance time for the SVD algorithms in different ecosystems. The usage of SVD function in the algorithm to determine the individual singular values of the reduced row matrices of the columns can also be enhanced by using the corresponding Eigen Value optimization which are usually provided within the Big Data environment. Algorithms such as Lanczo's algorithm can not only enhance the speed of the operation but also can be very easily parallelized.

Hence, this combination of l_1-norm and SVD can effectively make the best version of algorithm; being fast in execution at the same time delivering a holistic approach.

5 Knowledge Discovery from Big Data Computing: The Evolution of ASCOM

Even though Astronomy and Computing (ASCOM) has been in publication for five years only, its reputation has grown quickly as seen from the ranking system proposed here. This is despite the fact that ASCOM is severely handicapped in size. ASCOM is ranked 39 according to our method, slightly lower than its 31 rank in SCOPUS. This is due to the fact that we haven't used " citations from more prestigious journals" as a feature. Nonetheless, it is ranked higher than many of its peers which have been in publication over 20 years. This is also due to the fact that ASCOM is "one of its kind" and uniquely positioned in the scientific space shepherded by top notch editors. Such qualitative feature, regrettably is not visible from the big data landscape.

There is another interesting observation to take note of. By ignoring the "size does matter" paradigm, the ranks of some journals (many years in publication with proportionate volumes and issues) suffered. A few

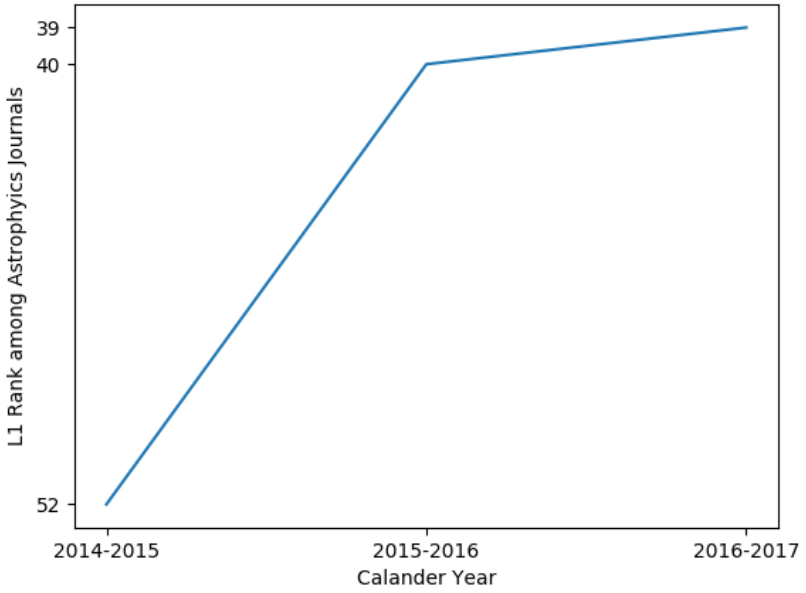

Fig. 1. l_1 Rank Progression of ASCOM based on SCOPUS data computed by the proposed method. The steady ascendancy in the journal's rank is unmistakable. it will be interesting to investigate the behavior of the journal rank in the long run once enough data is gathered.

examples include Living Reviews in Solar Physics, ranked 43 according to our scheme while it is ranked 3 in SCOPUS and Astronomy and Astrophysics Review, ranked 40 in our scheme while it is ranked 5 according to SCOPUS. This is important as our goal was to investigate the standing of a journal relatively new and in a niche area. This indicates that years in publication may sometimes dominate over other quality indicators and may not capture the growth of journals in "short time windows". Our study also reveals that ASCOM is indeed a quality journal as far as early promise is concerned.

6 Conclusion

The Big Data abode adds a new dimension to the already existing domain of Machine Learning; where the computation aspect is as important as the algorithmic and operational facet. The l_1-norm SVD scheme does just that, it introduces a brand new way of ranking data by considering all the features to its entirety. The added benefit of optimizing the required norms and methodologies in terms of a Big Data domain suggests its vast flexibility in the area of Big Data Mining. This article covered its application in the Scientometric Domain. However it can be extended to any type of data, provided that the nuances are well understood. The aforementioned recursive methodology of the scheme allows us to carefully consider the important feature of the dataset and make prudent decisions based on the outcome of an iteration. This allows us to take a more wholesome approach which is very similar to the page rank algorithm which gives a specific importance to each one of the features under computation.

In the context of Scientometrics, this scheme is also applicable as a way to rank specific articles in a given journal with the result that their respective scholastic indices are available. We can conduct similar data experiments using indicators like *Total Citations, Self Citations* etc to categorize them of their various other features available for articles. We have also done some extensive studies based on the scholastic indices of the ACM journal whose case study lies outside the scope of this article and were able to successfully rank the corresponding journals and article. The scheme proved to be successful in evaluating the parameters with their nuances intact. More often than not, most Scientometric indicators do not apply to the journal being evaluated. As a consequence of this, the data matrix in which the rows represent the articles in the journal and the columns represent the different evaluation metrics is clearly sparse. Exploiting this sparsity, using certain structural sparsity inducing norms and applying recursive Singular Value Decomposition to eliminate metrics can make the process more efficient. Sparse approximation is ideal in such cases because although the data is represented as a matrix in a high-dimensional space, it can actually be obtained in some lower-dimensional subspace due to it being sparse.

With the ever-expanding necessity to process voluminous amounts of data, there needs to be a need to provide solutions which can adapt to the fluctuating technological climate. The l_1-norm SVD scheme tries to

achieve similar potency, the usage of norm-based dimensionality reduction enhances the over-all efficiency on how we interpret data. The usage of techniques like sparsity norms suppresses outliers and only highlights the most meaningful data in store. The evolution of such methods will prove to be an absolute prerequisite in the future to compute copious amounts of data. Moving forward Dimensionality Reduction based techniques will become the foundation of salient data identification and the l_1-norm SVD scheme is such a step along that direction.

References

1. Francis Bach, Rodolphe Jenatton, Julien Mairal and Guillaume Obozinski, *Structured Sparsity through Convex Optimization*, *Statistical Science*. 27:450-468 (2011).
2. Golub, G.H., Van Loan, C.F.: *Matrix Computations.* 3rd ed. Baltimore, MD: John Hopkins University (2012).
3. Kalman D.: *A singularly valuable decomposition: The SVD of a matrix.* College Mathematics Journal 27:223 (1996).
4. Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C. & Byers, A. H.: *Big Data: The Next Frontier for Innovation, Competition, and Productivity (). McKinsey Global Institute* (2011)
5. Ginde G,Aedula R, Saha S, Mathur A, Dey S R, Sampatrao G S, Sagar B.: *Big Data Acquisition, Preparation and Analysis using Apache Software Foundation Projects, Somani, A. (Ed.), Deka, G. (Ed.)., Big Data Analytics, New York: Chapman and Hall/CRC. (2017)*
6. Bora, K., Saha, S., Agrawal, S., Safonova, M., Routh, S., Narasimhamurthy, A.: *CD-HPF: New Habitability Score Via Data Analytic Modeling, Astronomy and Computing*, 17, 129-143 (2016)
7. Ginde, G., Saha, S., Mathur, A., Venkatagiri, S., Vadakkepat, S., Narasimhamurthy, A., B.S. Daya Sagar.: *Sciento-BASE: A Framework and Model for Computing Scholastic Indicators of Non-Local Influence of Journals via Native Data Acquisition Algorithms, J. Scientometrics*, 107:1, 1-51 (2016)
8. Aedula, R.: rahul-aedula95/L1_Norm, https://github.com/rahul-aedula95/L1_Norm.
9. Ramin, S., Shirazi, A.S.: Comparison between Impact factor, SCImago journal rank indicator and Eigenfactor score of nuclear medicine journals. Nuclear Medicine Reviews. (2012).