Visual Analytics Framework for High Dimensional Data Streams

hafizur.rahman

October 2017

Contents

1 Introduction

Hence the available computing data source increases, our ability to generate and collect a wide variety of large,complex,high-dimensional dataset also continue to grow. In each second new data are being generated in different forms having different structure for example according to youtube company statistics; at present in every minute almost 300 hours of video uploaded[17].Data scientist and researchers are continuously research on collecting data, storing data, analysing data, proposed new algorithm for data mining, finding training methods to train the data , visualising data and many more. Hence data vary in size, structure, format and other factors new challenges are continuously evolved and new research domain being created.

We are living in the world called "Big Data". These big data have diversified sources such as images, videos, different social sites activities like post status,comments etc.Moreover now-a- days a lot of data is generated through sensors applied in different fields, GPS signals and many more[Zic13].The term "big data" is first mentioned on the IEEE 8th conference on visualisation in year 1997[CE97] however since 2011 the interest in big data area had been increased exponentially[WB13].The word "Big" means significance,complexity,challenge, quantification.

There are many definition for the big Data but unfortunately there is no exact one. The most reference one is the one from Gartner report[Lan01].According to that report data those can be termed the "three Vs": volume,variety and velocity are known as big data. Volume means the significant amount of data, variety refers to data sources where structured, semi-structured and unstructured data produced and velocity is the increasing rate at which data is produced. Recently these "three Vs" are extended to "five Vs"; the two new are veracity and value.Veracity include the trustworthy domain i.e where we can trust and uncertainty of the data. Value denotes the added business value for big data users.

Data can be modelled based on some characteristics.Traditionally, data is modelled as persistent relationships and store into database management system. From the system, data can be accessed and analysed based on the requirements.But recently a new class of data intensive application evolved where application does not follow the traditional system rather considered as transient data streams.Examples of such application fields include financial applications, web page visits, network monitoring, security, telecommunication data management, sensor network and many more[Bab+02]. The main characteristics of these sources is from here data produce continuously in multiple, time varies and in almost all cases streams are unpredictable and unbounded.These pattern of data raise various scientific and technical challenges including data capture, storage, transformation, analysis and visualisation.

In many application field real time analysis is very much important.For ex-

ample,in medical domain there are many diagnostic data has been generated from different machine where doctors need to analyse the data and take decision in real time.In financial stock market, price are continuously changing as time changes and the traders need to analyse those data and take decision of doing transaction in real time. The same case also happens with the sensor data attached with the internet of devices, the meteorological data and space data.

In present world, each data sets have a huge number of features or dimensions.These are called as High-Dimensional Data and in almost every fields of study now data are High-Dimensional.The computational complexity and storage complexity increases with the increment of the dimensions.Moreover, if a data is high dimensional then it is also difficult for the human being to reflect their intuitions.Although each feature has the important information of the data set but still we can compare the features based on the important information and skip the less feature and this process is known as dimensionality reduction.

The concept of using pictures to understand data has been around for the centuries and the process to present data in forms of pictures or graphs is known as Data visualization.Through visualisation it is easy to get the inner meaning of the data.It helps us to analyse the data and present in form of patterns, trends, gaps and outliers. It also helps us to compare, make correlations among the data. One of the most important benefit of visualisation is that it encompasses various data set quickly, effectively and efficiently. Scientifically the effectiveness of data visualisation is to maintain a proper balance between perception and cognition through visualisation.

Due to the importance of the visualisation now it is also a demand of the time to visualise the streaming data similar to static data through any visulaisatation techniques.The most common visualisation technique used for high dimensional data are scatter plot, parallel coordinates, Heatmap etc.There is a physical limitation of the display devices and also our visual system are suitable for two or three dimensional pictorial image than the recognition of such high dimensional structures there are already a variety of approaches has been taken for the dimensional reduction.In static data, there are many state-on-art algorithm applied where it is still a challenge for the streaming data.

In this thesis, we will try to implement high dimensional reduction technique and see its performance on streaming data.

2 Outline

There are more four chapters in our thesis. In chapter 3 we will describe the motivation of the thesis. The goal of the thesis is described in chapter 4. In chapter 5 we will present the past related work or background studies and where in chapter 6 we will describe the timeplan of the thesis.

3 Motivation

Due to the growth of streaming data it is now demand of the some application domain to analyse the data and present data to the end user in real time. For example, in stock market data produce continuously about telling the price, volume of the stock etc; if a system can analyse the pattern of the data and predict the price of the data to the trader or broker in real time then it will help them to take decision on transaction.

Hence visualisation is one of the most important aspect for this kind of the real time domain it is always a challenge for the researchers to present the human intuition friendly visualisation.In big data it requires extraneous efforts to reduce the data to uncover knowledgeable patterns. There are number of pre-processing techniques to reduce the data. Some of them are summarization, sketching, anomaly detection, dimension reduction, noise removal, outlier detection etc[Reh+16].Dimension reduction is one of the prominent method among them.

Dimensionality reduction is not only beneficial for the visualisation perspective but also helpful for speeding up any kind of learning algorithm by applying indexing structure to speed up the queering procedure. Dimensionality reduction has also impact on the quality and efficiency of the system.

Dimensionality reduction algorithm can be accomplished either by selecting significant features or by transforming dimension in order to get a new one or reduced set of dimensions.The first approach is known as feature selection and second one is known as feature extraction. Feature Selection is a greedy approach by aiming at finding out the subset of most representative features based on some criteria. Hence FS approaches are greedy so it's a challenge to find the optimal solution. Orthogonal centroid algorithm is a Feature Selection algorithm.This method is widely applied in data mining and machine learning..

Feature extraction algorithm aim to extract features by projecting high-dimensional space into lower dimensional space using algebraic expression.The feature extraction algorithm can be further divided into linear projection and non-linear projection.Linear Discriminant Analysis (LDA), Maximum Margin Classification (MMC) & Principle Component Analysis (PCA) are linear feature extraction algorithm[Yan+06] where non-linear algorithm are kernel PCA, graph-kernel PCA etc.

The conventional dimensionality reduction algorithm use Gaussian Maximum Likelihood estimation which involves different matrices like covariant matrix, scatter matrix which have time & space complexity of $O(n^2)$ or $O(n^3)$.The complexity can be tolerable if the data size is small but if the data size is more than 20000 then this method does not perform well[Reh+16].Due to the high mathematical computation it is not suitable to use those methods on streaming

data specially in real time analysis field.

Researchers have already defined the issue and give many solution on that. They modified the current algorithm and propose almost incremental verison of each algorithm. The term "incremental" means learning from new data without forgetting the prior knowledge. In this mechanism, a system must acquire knowledge from the past data without keeping the original data and also ready to handle the new data [Chu+15].Incremental Principle Component Analysis(IPCA) [Li+03], Incremental Linear Discriminant Analysis [POK05], Candid Covariance-free incremental principle component analysis[WZH03], Incremental Dimension Reduction via QR decomposition[Ye+05] etc. The major drawback is still the involvement of numerical transformation which performs very poor where there is a lot of data & dimensions. The details of each algorithm will be covered on the related work section.

In all of the Incremental algorithm researchers use static data sets.They divide the data set into two portion where first portion they used to calculate all the mathematical computation and remember values irrespective of the dataset and later they used the remaining portion to update those values incrementally and see the final output of the algorithm.In our thesis, we will use time based streaming data having a fixed time window which is unpredictable in nature and also data will not available after the time pass.

Though dimensionality reduction has many advantages but it has some demerits too. The main drawback is the possibility of information loss. It is undoubtedly true that when we left of one dimension that means we have the trade off of loosing some data but in some cases we have lost the important information.The essential idea of dimensionality reduction is to preserve the instrinsic meaning of the data by keeping similar data points close and maintaining a distance among dissimilarity data. Researchers is now also looking for different approaches to get the same or better visualisation as ouput without doing information loss.One of them is reordering the columns based on similarities.Djuric et.all proposed an algorithm based on the reordering approaches in their paper [DV13].The details will be covered in the "Entropy Minimisation ordering" of the related work section.

4 Thesis Goals

The aim of this thesis is to implement the Incremental version of traditional dimensionality reduction algorithm known as Linear Discriminant Analysis.The name of the algorithm is Incremental Fast Batch Linear Discriminant Analysis.We implement this algorithm on streaming data and present the output through visualisation technique for example Heatmap. The output of the algorithm will be evaluated based on some evaluation criteria. The following figure will describe the framework in step by step wise:

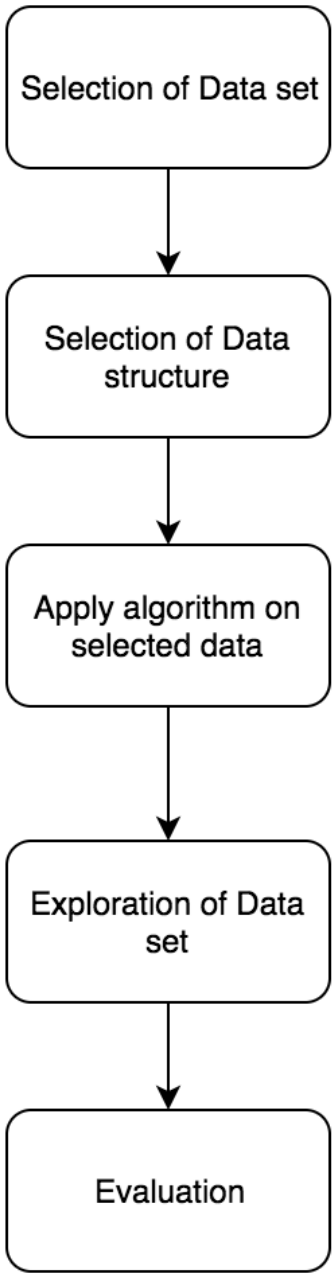

Figure 1: Thesis goal

4.1 Selection of Data Set

In this thesis,we are looking for the high dimensional data set.There is no fixed threshold number of the dimension we can call as the high but while we look for the data set we considered from visualisation perspective where the data set can be defined as High or not. If required we can convert the batch data to streaming data using any kind of state on art tool.

4.2 Selection of Data Structure

The main difference between static data and streaming data is in static data we can save data into the disk and access that whenever we need that for any purpose but in streaming data the situation is not the same. In streaming data hence the volume and velocity of the data is huge so we can not save the data rather we need to do operations with the flow of the data. Sliding window structure means we will divide the data in a block and only have access for once; there is no scope of viewing the data for the second time.The block can be done based on the time interval.

4.3 Implementation of the Incremental Linear Discriminant Analysis

There are so many state-on-art algorithm on ILDA but there is no open source version of the code. In my thesis, I choose one of the algorithm known as IFLDA/QR and will implement that algorithm in this thesis.

4.4 Visualisation

There are many state-on-art visualisation techniques for visualise the data. we will visualise the output of the framework through one of the common procedure known as Heatmap.

4.5 Evaluation

There are basically two way to evaluate the system.One is known as the quantitative evaluation and the other is qualitative evaluation.In Quantitative evaluation we define the metrics for example calculating the entropy or the matrix reordering quantitatively. Qualitative evaluation is the evaluation for example after showing the visualisation by deciding which one is more informative. This can be done by doing user studies or selecting streaming data from different fields.

5 Related work

There are three forms of complexities to handle big data systems[Reh+16]. These are:

- Data complexity
- Computational complexity
- System Complexity

Data complexity arises due to the multiple formats & unstructured of the data.Moreover data have multiple of dimensions and also there is a complexity between inter-dimensional and intra-dimensional relationships.The data complexity also increase the computational complexity in big data systems.Moreover,The extensive computational requirements of big data systems increase the system level complexity.

In the following two subsections we will describe the details of the past work of dimensionality reduction algorithm and Entropy minimisation based reordering algorithm respectively.

5.1 Related work with Dimensionality Reduction

Efficient storage and retrieval of high-dimensional data certainly one of the major issue in database and data mining research [Ye+05]. In the past,many attempts took to design multi dimensional indexing structure for example R-trees,R^*-trees, X-trees,SR-tree, etc. in order to speed up the query procedure. But these procedures fail to do so in as the number of dimensions increases. The more is the dimension the more performance deteriorates.To overcome this issue, one way is to transform from high dimensional data to low dimensional data with the trade-off of limited information loss.

Though Dimensionality reduction is a very old problem but still this problem exists.Principle Component Analysis(PCA) and Linear Discriminant Analysis(LDA) are two mostly used one to reduce the number of dimensions. Both PCA and LDA seek directions of the component but the main difference among them is PCA seeks direction for representation where LDA seek directions for efficient discrimination.In both cases it assumes that training data set are available in advance but in reality a complete training set might not given beforehand.

5.1.1 Principle Component Analysis

PCA is a multivariate technique to analyze the data table.Here the observations are described by several inter-correlated quantitative dependent variables [AW10]. The goal is to find a subspace whose basis vectors correspond to the direction with maximal variances [Yan+06]. Mathematically, it depends on eigen decomposition by computing eigenvectors and eigenvalues and upon singular value decomposition (SVD) of rectangular matrices.

Lets denote $C = \frac{1}{n}\sum_{i=1,2,\ldots,n}(x_i - m)(x_i - m)^T$ as the covariance matrix of the sample data. We define the objective function as $J(W) = traceW^TCW$. PCA aims to maximise the objective function $J(W)$ in a solution space $H^{d*p} = W \in R^{d*P}, W^TW = I$

5.1.2 Linear Discriminant Analysis

The goal of the Linear Discriminant Analysis (LDA) is used to find a lower dimensional space that best discriminants the samples from different classes [Yan+06].In Linear Discriminant Analysis (LDA), we need to compute a linear transformation by maximising the ratio of the between-class distance to the within-class distance in target of achieving maximal discrimination[DHS00].After that we need to find eigen value decomposition of both matrix to select new feature subspace.

Mathematically the aim is to maximize the Fisher criterion an objective function:

$$J(W) = \frac{W^T S_b W}{W^T S_w W}$$

where $S_b = \sum_{i=1}^{c} p_i (m_i - m)(m_i - m)^T$ and $S_w = \sum_{i=1}^{c} p_i \underset{x \in c_i}{\mathrm{E}} (x - m_i)(x - m_i)^T$ are called Interclass scatter matrix and Intraclass scatter matrix respectively. Here E denotes the expectation and $p_i = \frac{n_i}{n}$ is the prior probability of class i. We can get W by solving $W^* = \arg\max \mathrm{J(W)}$ in the solution space $H^{d*p} = W \in R^{d*p}, W^T W = I$. This is done by solving the generalised eigenvalue decomposition problem: $S_b w = \lambda S_w w$

5.1.3 Limitation of PCA & LDA

By considering the availability of the data before applying algorithm is the main challenge in stream data application. In streaming data, data comes in a lot of numbers and continuous speed as a result it is not possible to store the data before applying algorithm. Moreover the traditional LDA and PCA perform in batch mode which is computationally expensive when dealing with large scale problems.In streaming data if there is a new data then both the LDA and PCA algorithm starts from the scratch to learn it from beginning which increase the computational complexity and large memory [Chu+15].

Therefore,researchers look for the solution and one of the way to get rid of this to collect data whenever new data are presented and apply batch learning approach for the collected data so far. But the drawback of this approach is that it requires a large memory to store the data and high computational expenses are required. Moreover the system also forget about the knowledge that it acquires in past batch mode.Researchers work on this issue and some of their work are presented in the following section.

The another main challenge in streaming data is data may come into chunk from and also may be a single data in allowed form. That means the rate of data passing is unpredictable in nature therefore the dimensionality reduction algorithm should be adaptable with this issue.

5.1.4 Principle Component Analysis for Streaming data

Traditionally PCA efficiently represent high dimensional vectors with a small number of orthogonal basis vectors but this method is usually perform in batch-

mode which is computationally expensive for large scale problems. To address this issue, researchers developed several incremental algorithms in their previous studies[Li+03].

Artac Jogan et. al[AJL02] describe the way of remembering data from the sample and then delete the data to optimize the storage complexity. They used image as input where the representation of the image consists only of the corresponding coefficients stored as per image then the image is discarded. Here the performance is almost similar with the batch method but the learning method helps us to relearn data.Hall et.all [HMM98], Chandrasekaran et.all [Cha+97], DeGroat et.all also propose based on gaining information from the past and learn from the past and delete the data after acquiring the knowledge.

Li Xu et al. proposed an algorthm by removing the outliers. The estimation is done using the likelihood function. All Incremental PCA algorithm proposed so far is how to effectively handle with the covariance matrix but all of their effectiveness and computational complexity is more or less similar. Weng et all. [WZH03] proposed a candid covariance free incremental analysis by using a well-known statistical concept efficient estimate like some well-known distribution for example Gaussian distribution.This algorithm also compute the principal component of samples incrementally without estimating the covariance matrix by keeping the scale of observations and computes the mean of observations incrementally.The main problem of this technique is to run into convergence problems in high dimensions.

5.1.5 Linear Discriminant Analysis for Streaming data

Similar to IPCA researchers also modify the LDA algorithm to run an incremental fashion to accomodate the new data. Here also one of the major concern is not forgetting prior knowledge.

Pang et. all [POK05] propose the algorithm by adopting the system ready for any new data arrival in basis of single or chunk basis and termed separately Sequential Incremental LDA and chunk incremental LDA.In this paper they handle the eigenvectors by providing ranking and select the top most eigen vectors.This proposed algorithm will confront difficulty as the dimension if the data is very high.Specially it will require large memory hence it needs to solve a high dimensional generalised eigen value problem [YLO07]

Ye et.al [YLO07] proposed an algorithm called Streaming LDA consisting three steps: first it compute the centroid matrix then update within-class scatter matrix.Finally the between-class scatter matrix is updated.After this steps, it also solve the eigenvalue problem.

kim et.all [Kim+11] propose a new concept to update between -class and within-class scatter matrix. They used sufficient spanning set to do so.In every step both matrices are kept and updated and minor components are removed in every step.

The main difficulty in all of the propose solution is the presence of the eigenvalue problem of scatter matrices which makes it difficult to maintain it incrementally.

5.1.6 QR decomposition

LDA algorithm use Singular Value Decomposition(SVD). It is difficult to design an incremental solution for the eigenvalue problem on the product of scatter matrices.

To solve this, Ye Li et.al [Ye+05] propose an LDA based incremental dimension reduction algorithm called IDR/QR which applies QR decomposition rather than Singualr Value Decomposition.The reason for using this technique is it does not require the whole data sets in memory before implementation. The algorithm is also computational cost efficient when new data item is inserted. The classification accuracy of this algorithm is very close in compare with the other best described LDA algorithm but it has much less cost when new items are inserted compare to others. Moreover hence it is computed some approximate matrices there is a chance of accumulating the approximate error as new data are appended sequentially. The larger the error the more the opportunity of information loss. [YLO07].

5.1.7 Fast Online incremental learning on mixture streaming data

In [Wan+17] they proposed an algorithm for streaming data known as Fast Batch LDA algorithm known as FLDA/QR learning algorithm.In the algorithm they use the cluster centres to solve a lower triangular system which is optimized by the Cholesky-factorization. They also develop this algorithm for an exact incremental algorithm called IFLDA/QR.For reorthogonalization they use the Gram-Schmidt process which saves the space and time expenses compared with the rank-one QR-updating of most existing methods.In there paper they mentioned their contributions are twofold:

- They use the advantage of the QR-decomposition on a lower triangular matrix and propose a new fast batch method called FLDA/QR.In this algorithm, they take the centroid of each cluster to constitute the matrix decomposition.Because of this, they require a smaller storage and less computation. They also use the Cholesky-factorization which surpass in performance specially the computation load of the FLDA/QR method.
- Next they develop an exact incremental version of the FLDA/QR known as the IFLDA/QR.It is mathematically possible to update the Gram-Schmidt reorthogonalization process. According to them, this process is faster than the rank-one updating in many other ILDA algorithms based on QR-decomposition.

The algorithm described in a paper is presented in the following figure2 as a flow-chart.

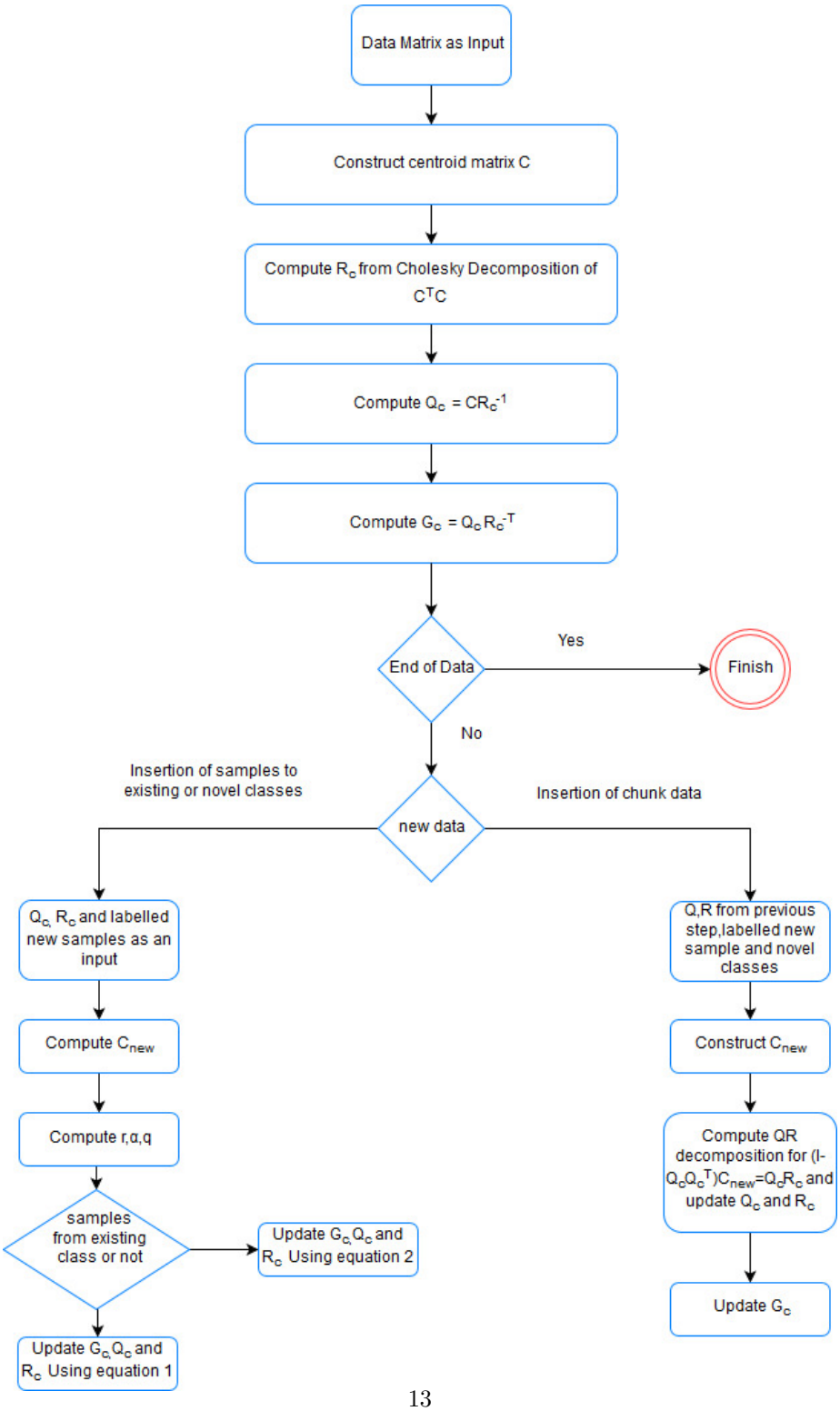

Figure 2: IFLDA/QR algorithm

The key characteristic of their algorithm which separates it from the other is the space and time complexity.It is space and time efficient compare to others while updating the algorithm because of the arrival of the data.This algorithm is 2 to 10 times faster than the state-of-the-art algorithm where the classification performance is same like the others.

Fast Batch Linear Discriminant Analysis: After having the data matrix as an input first the algorithm try to compute the global centroid matrix C consisting of the K centers.In this algorithm, lower triangular linear system is used for the calculation.Since the input training data is the centroid matrix therefore the scatter matrix within-class scatter is 0.They used the Cholesky-factorization to generate the matrix of R for the QR decomposiiton.The algorithm complexity is *O(dn)*

Incremental Linear Discriminant Analysis: After development of the FLDA algorithm they extend the algorithm for the incremental development of the new incoming streaming data.The streaming data may come in three different forms:

1. new labeled samples to the existing classes
2. samples from an entirely new (novel) class
3. a chunk of samples mixed with those as 1) and 2)

In cases 1 & 2, the samples from the existing classes and from an entirely new classes the updated calculation of the new centroid matrix is same.The r,α,q is also calculated in the same way for both case.The difference lies within the update G_c,Q_c and R_c.The update version of any *variable* is represented as $\hat{variable}$. If the the new labeled samples from the existing classes then the Q_c & R_c are updated using equation 1

$$\hat{C} = Q_c * R_c \tag{1}$$

For new novel classes data are updated by the equation 2

$$\hat{G_c} = \begin{bmatrix} G_c - qc_{new}{}^T G_c/\alpha & q/\alpha \end{bmatrix} \tag{2}$$

There may be a chunk of new data which contain samples from the existing classes and novel classes. The challenge in this case is to extract the information from these mixed data and also they need to preserve the previous learned ones.The current algorithms fail to perfom in this scenario.after constructing C_{new} they compute the QR decomposition of $(I - Q_c{Q_c}^T)C_{new} = \hat{Q}_c\hat{R}_c$ and update $\hat{Q_c}$ and $\hat{R}_c$.Update the $\hat{G_c}$ using equation 3

$$\hat{G_c} = \begin{bmatrix} G_c - \hat{Q_c}(\hat{R_c}^{-T}({C_{new}}^T)G_c) & 0 \end{bmatrix} + \hat{Q_c}(\hat{R_c}^{-T}Z) \tag{3}$$

5.2 Related work with Entropy Minimisation Algorithm:

Petrie introduced the model of how to order a data matrix in 1899. Later this is named as data reordering or seriation. This methodology is used in many different application disciplines for example archaeology,anthropology etc.Data ordering has huge impact on some cases for instance gene expression data analysis in bio-informatics, geographical data analysis, bandwidth minimization or data compression[DV13].

The main assumption of seriation in data visualisation is based on the assumption of permuting either rows or columns of the dataset without loss of information. Therefore, data reordering is done in such a way that similar examples or features are close to each other.Closeness of data helps to improve the quality of visualisation without loss of any information as data dimension reduction methods do.

The seriation of the dataset can be formalised as if there is a dataset having n objects O_1,...., O_j one can construct an n*n symmetric dissimilarity matrix $D=(d_{i,j})$ where $d_{i,j}$ for $1 \leq i,j \leq n$ represents the dissimilarity between objects O_i and O_j,and $d_{i,i} = 0$ for all i.The major challenge in seriation problem is to find a permutation function.

Researchers proposed different permutation function for reordering data. Hahsler et al.[HHB08] reviewed a larger number of permutation function such as column gradient measure, anti-robinson effects, Hamilton path length, inertia criterion, least squares criterion,linear seriation criterion, measure of effectiveness and stress measure.

To find the most loss/merit functions in discrete optimization problem is a complex problem. An exchaustive serach is infeasible because the number of possible permutations for n objects is n!.Researchers proposed different heuristics and seriation methods that are briefly covered in the following subsections.

5.2.1 Entropy Minimization

There are some loss functions used for data seriation which are related with entropy.For instance, in [You+05] author defined the stress measure as a sum of local entropy of each data item.To minimize the sum,Wilkinson proposed a heuristic approach [WF09] and Niermann proposed a genetic evolutionary algorithm [You+05].The another way to encode the dataset using Differential Predictive Coding (DPC).In this method, each item is encoded as a difference between current and previous items.Djuric et al. proposed an efficient algorithm to minimize the entropy of the encoded dataset in [DV13].The deatils of this algorithm will be covered in section 5.2.7.

5.2.2 Travelling salesman problem solver

Data reordering using the length of a Hamiltonian path as a loss function is equal to solving a Travelling Salesman Problem (TSP), which is a well known and extensively researched combinatorial optimization problem. The aim of an Travelling Salesman Problem Solver (TSP-solver) is to find the shortest tour that, starting from a specific city, visits each city exactly once and then returns to the starting point. As the general seriation problem, solving the TSP is also complex. In case of seriation with n + 1 cities, n! tours have to be checked. In order to avoid exhaustive search, different heuristics were proposed, from simple nearest neighbour methods to complex approaches like the Lin-Kernighan (LK) algorithm [HHB08]. Recently, Djuric and Vucetic [DV13] introduced a fast *O(nlog2n)* TSP-solver, called the TSP-means.

5.2.3 Hierarchical clustering

Hierarchical clustering and its extension named Hierarchical clustering with optimal leaf ordering(HC-olo) are most commonly used methods in bioinformatics [DV13]. The output of HC is a series of nested clustering which are stored in a tree. The order of leaf nodes in a tree is used to produce the linear order of the example.The problem is to find the optimal leaf ordering because a binary tree having n leafs and fixed tree structures have 2^{n-1} different linear orderings.To solve this issue, Bar-joseph et al.[Bar+02] proposed an optimization algorithm called HC-olo.The time complexity of $O(n^3)$ is not at all suitable for big data.

5.2.4 Low-dimensional projection algorithm

Researchers also target to use the state-on-art dimensionality reduction algorithm for ex. PCA,LLE,LDA to project the original dataset into one-dimensional subspace.The linear ordering of the examples is that new subspace can be used for reordering. The problem is the intention of any dimensional reduction method is not the reordering rather it is a byproduct of manifold search so the quality of produced ordering is not at all satisfactory.

5.2.5 Sugiyama'salgorithm

To draw a bipartile graph with as few edge crossing as possible is a well-known NP-complete problem[MS00].Sugiyama's algorithm is based on average heuristics approach for drawing problem.The backbone of the algorithm is to order nodes according to the average of their adjacent nodes in the opposite node set. To apply Sugiyama's algorithm for data reordering first Makinen et.al proposed first to make data into a binary format from the original dataset.Then we consider rows and columns of the matrix as two separate nodes sets, where binary values represent edges between nodes. In figure 3 shows applying the average heuristic to a simple bipartite graph.

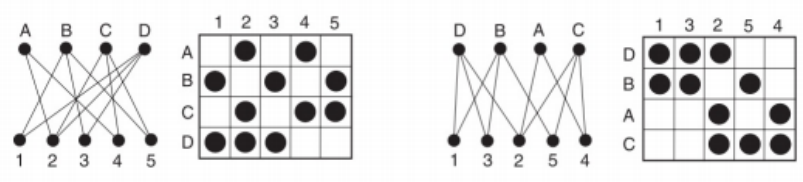

Figure 3: Applying the average heuristic to a simple bipartite graph

5.2.6 Entropy-Minimization-based data reordering

In this section we will cover the algorithm that we will implement from data reordering representative. The algorithm will be presented shortly in this section. The approach that has been proposed in [DV13] we will implement in the algorithm and modify it to adopt for data streaming. According to Djuric et.al [DV13]:

- Ordering can be done by doing permutation of rows or columns that ensures maximally compressible data set. The maximally can be determined by the entropy of the residuals of predictive coding.
- The problem is determined by an Expectation-Maximization algorithm which alternatively solves a TSP and assign reweights features based on the quality of the resulting tour.
- The proposed TSP solver known as TSP-means find the path comparable to those by LK algorithm. By applying K-means (k=2) recursively the algorithm construct a Binary tree. The runtime of TSP solver is *O(nlog(n))*.

In next following paragraph we will describe the differential predictive coding, entropy minimisation reordering and TSP-means algorithm respectively.

Differential Predictive Coding: We assume our dataset D is stored in a form of n*m data table.

$$\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1j} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2j} & \dots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i1} & x_{i2} & \dots & x_{ij} & \dots & x_{im} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nj} & \dots & x_{nm} \end{bmatrix}$$

where ith row vector represent an example with m features.
Differential predictive coding replaces each example with its difference from the previous one, $\varepsilon_i = x_i - x_{i-1}$, where ε_i is called DPC residual. As a result, the

initial data table D is transformed into $\mathrm{D_{DPC}}$ without loss of information since the original dataset can be retreived from the encoding.

$$\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1j} & \dots & x_{1m} \\ \varepsilon_{21} & \varepsilon_{22} & \dots & x_{2j} & \dots & \varepsilon_{2m} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \varepsilon_{i1} & \varepsilon_{i2} & \dots & \varepsilon_{ij} & \dots & \varepsilon_{im} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \varepsilon_{n1} & \varepsilon_{n2} & \dots & \varepsilon_{nj} & \dots & \varepsilon_{nm} \end{bmatrix}$$

Entropy Minimization Reordering: In data compression theory entropy is used as a measure of randomness of the dataset, where the larger value of entropy denotes large randomness and small compression ratio. Small entropy denoted by $H_{DPC(\varepsilon)}$ means DPC residuals are small which implies D is a well ordered dataset.$H_{DPC(\varepsilon)}$ can be estimated as

$$H_{DPC}(\varepsilon) = -\frac{1}{n-1}\sum_{i=2}^{n} logP_{\varepsilon}(x_i - x_i - 1) \tag{4}$$

where $p_{\varepsilon}(\varepsilon)$ is the probability density of vectors ε.
The permutation of examples π^* whose has minimize entropy of DPC residuals is the optimal one.

$$\pi^* = \underset{\pi}{\operatorname{argmin}}(H^{\pi}_{DPC}) \tag{5}$$

Djuric et al. [DV13] proposed a model P_{ε} as Gaussian or Laplacian distribution that results in introduction of $\sigma = [\sigma_1\sigma_2...\sigma m]$- vector of m parameters. According to his paper equation 2 can be restated as:

$$(\pi^*, \bar{\sigma}^*) = \underset{\pi^*,\bar{\sigma}^*}{\operatorname{argmin}}(H^{\pi}_{DPC}) \tag{6}$$

For data reordering the reason for using expectation-maximization-like algorithm is used to find a permutation of examples π which gives the minimize DPC residuals.
In M-step we need to find a permutation of examples π,which assumes σ_j are known and find the entropy. This way is equivalent of solving TSP where features are downscaled using σ_j and then TSP-means algorithm can be applied.
When the current ordering π is found,the goal of E-step is to calculate new values of $\bar{\sigma}$ which in return minimizes H^{π}_{DPC}.We can derive the new parameters using following formula:

$$\sigma_j^2 = \frac{1}{n-1}\sum_{i=2}^{n}(x_{\pi(i),j} - x_{pi(i-1),j})^2 \tag{7}$$

TSP-means: To start TSP-means we need to recursively apply k-means clustering (k=2) to initial dataset to create binary tree T which follows a conversation from binary tree to 2^l-ary tree T^l by keeping only nodes at every l^{th} tree level.After the conversation the algorithm traverse the tree in a breadth first way from left to right. The goal is to reorder internal and leaf nodes so that similar examples (leafs) and clusters (internal nodes) are closer to each other.To achieve this,we are going to use a TSP-solver for example: LK algorithm to reorder the children of the node together with their neighbours and replace the node by its reordered children.

6 Solution

At present the world is shifted to streaming data from the static data due to the involvement of the mobile networks, social media and video cameras.To analyse this kind of data we need to implement a proper data structure to handle the data sets and also define fast pre-processing steps to analyse those data sets.One of the important preprocessing technique is Dimensionality Reduction.In this section, I will describe the solution approach of reducing the high dimensional streaming data to low dimensional streaming data.

The following figure 4 contains the general process model of our Framework.

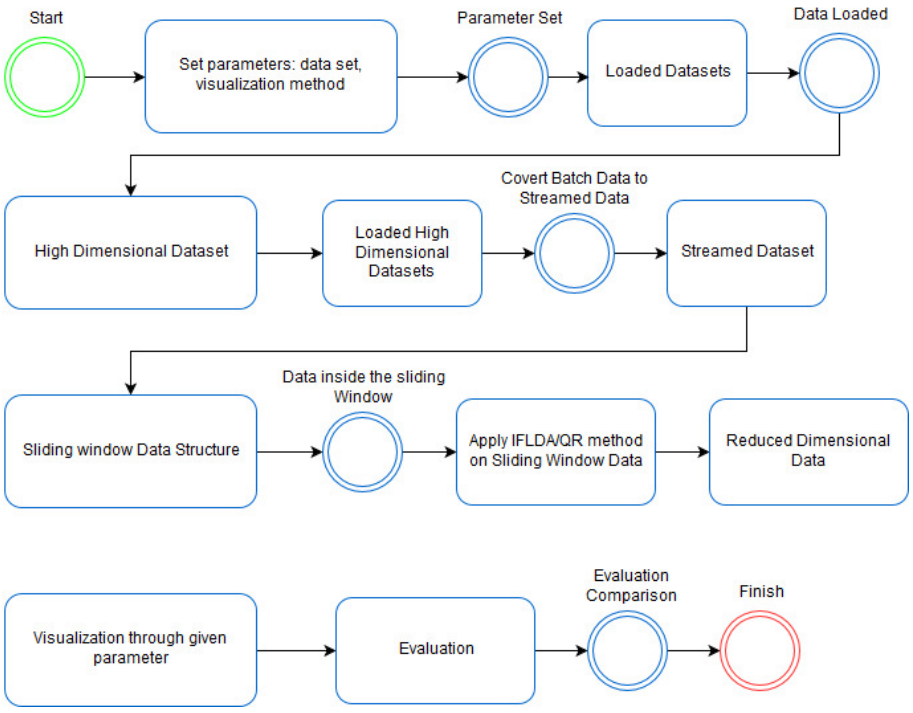

Figure 4: Dimensionality Reduction Framework of Streamed Data

6.1 Selection of Data Sets

The framework will be flexible for both the online available Streaming API to consider as a Data source and also have the ability to upload the static data.Hence we are considering the streaming data it will be good if the chosen data domain have some impacts on the change of the data with the passage of the time.
If the chosen data set is not the streaming data then the framework will be capable enough to convert the static data to the streaming data using any available free state-on-art tools.

6.2 Selection of Data Structure

The main characteristics of the streaming data that the data comes continuously with an unbounded structure and pattern as time progresses.For this reason,to handle the streaming data there should be some mechanism to hold the data for a minimum period of time and within that time all necessary work related with the data should be done then the data is thrown off.One of the common mechanism to handle those data is known as sliding window approach.Here the window is defined based on the time like in particular interval for example in

each 500 ms the data comes are considered one window. The window will be moved after each fixed period of time and that's the reason it is known as sliding window approach.Our framework will be capable of considering the streaming data in a sliding window basis.

6.3 IFLDA/QR algorithm

As previously mentioned we will implement the IFLDA/QR algorithm.At the first window, we will implement the FLDA algorithm to calculate the centroid matrix and implement the Cholesky Decomposition of the centroid matrix.The output of this mechanism is the optimal transformation matrix.
From second window and so on we will implement the algorithm of the insertion of the chunk data.Here as an input of the process we will consider labeled new samples and also the novel class.Here the centroid matrix for an existing class will be updated as well as the new centroid of the new cluster will be calculated.In following figure 5 the whole process is shown step by step wise.

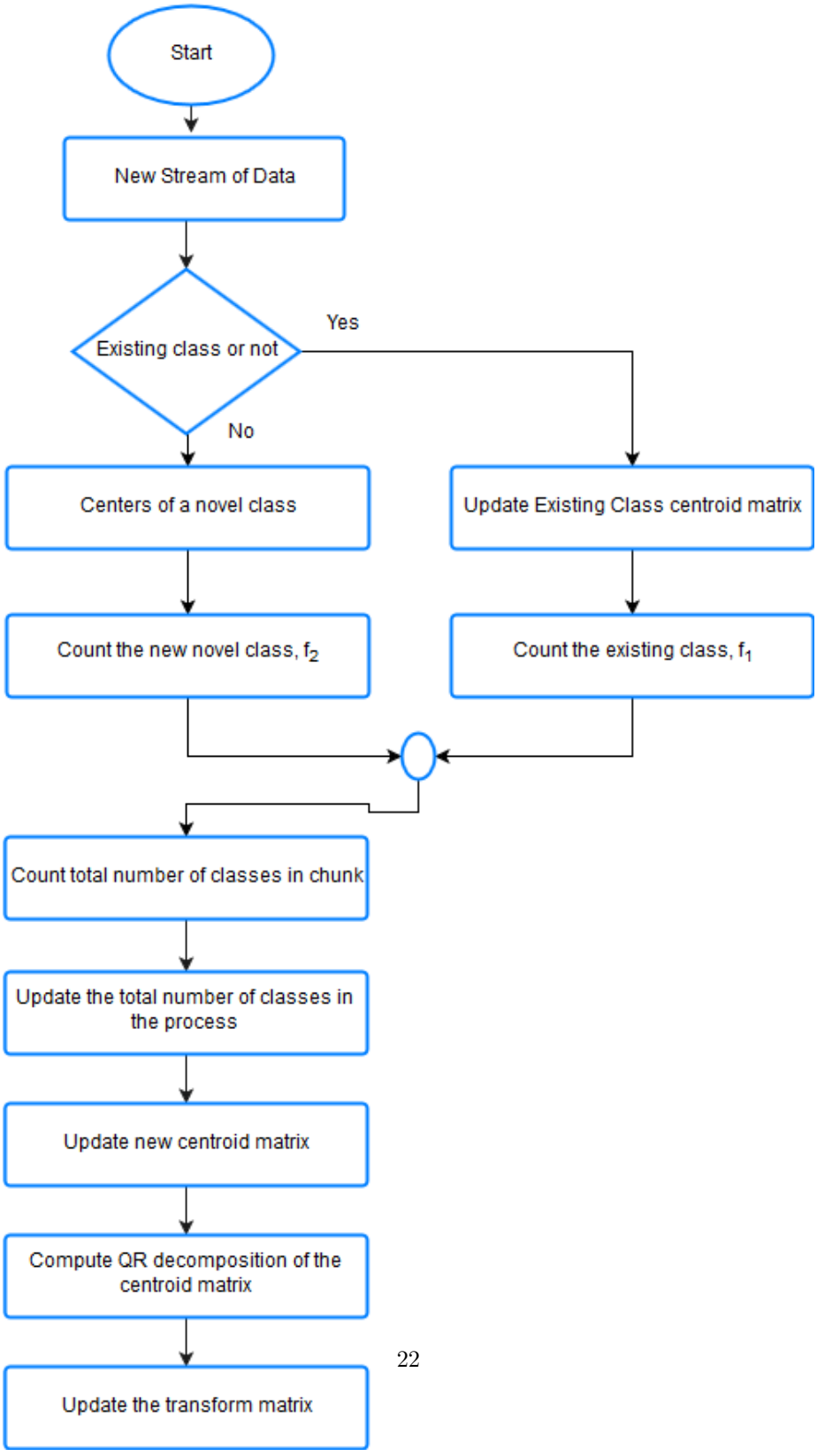

Figure 5: Process of Handling Stream Data

7 Evaluation

There are number of challenges which could arise in reduction of the dimension of the data.The varies algorithm from both PCA & LDA domain is still not the time coefficient and as the number of dimension increases the time and space complexity increases mot in an optimized way.In almost all example they divide the static data into different portions. The first portion of the data they used for the calculation of all the parameter of their algorithm and the rest portion are passed by the user that means the data flow in controlled by the user.In our case the data will not be controlled by the user rather the data will be unstructured and uncontrolled.
To appraise the quality of the algorithm,a comparative methodology with an evaluation method known as Qualitative evaluation will be done.Here the evaluation will be totally based on the comparative visualisation.The comparison will be done in three different ways and another evaluation will be by using different data sets having different number of dimensions.We will calculate a graph of using different numbers of dimensions and its reflection on the performance. This evaluation will help us to determine the scalability of the framework.The all evaluation criteria of the thesis is listed below:

1. Comparison of Vizualisation without dimensionality reduction and applying dimensionality reduction
2. Comparison of performance of the algorithm between using incremental data and streaming data
3. Comparison of the performance between IPCA algorithm from Scikit Python library and our algorithm
4. Scalability of the Framework

8 Timeplan

The time plan as shown in figure 6 is to achieve the goals of this thesis. The thesis will be structured into the following way:

- Literature review is one of the important activities of the thesis. It will be carried out in parallel with the other phase of the thesis.
- Analysis of the state-on-algorithm will be done with the progress of the literature review.
- Modify of the current EMDR algorithm to make it faster and scalable and also adopt the system with the stream data is the second most important task of the thesis.This is planned to be finished after selection of the tools and between November to January.

- implementation of the Incremental Linear Discriminant Analysis is the main task of the thesis. In this thesis, a fast,scalable framework should be developed for the streaming data.This will be continued with the parallel of other tasks of the thesis.
- Evaluation is another big topic for this thesis.The evaluation will be done after the implementation.
- Thesis writing will be done through out the whole time of the thesis and preparation for the presentation will be done at the last month of thesis.

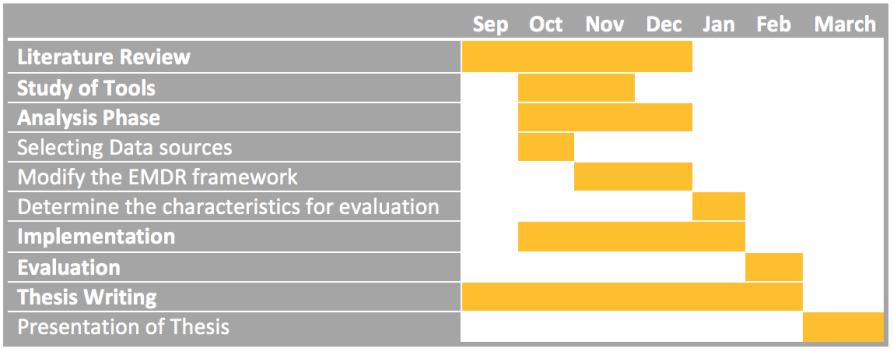

Figure 6: Thesis plan

9 References:

References

[17] *Youtube Company Statistics*. http://www.statisticbrain.com/youtube-statistics/. 2017.

[AJL02] M. Artac, M. Jogan, and A. Leonardis. "Incremental PCA for online visual learning and recognition". In: *Object recognition supported by user interaction for service robots*. Vol. 3. 2002, 781–784 vol.3. DOI: 10.1109/ICPR.2002.1048133.

[AW10] Hervé Abdi and Lynne J. Williams. "Principal component analysis". In: *Wiley Interdisciplinary Reviews: Computational Statistics* 2.4 (2010), pp. 433–459. ISSN: 1939-0068. DOI: 10.1002/wics.101. URL: http://dx.doi.org/10.1002/wics.101.

[Bab+02] Brian Babcock et al. "Models and Issues in Data Stream Systems". In: *Proceedings of the Twenty-first ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. PODS '02. Madison, Wisconsin: ACM, 2002, pp. 1–16. ISBN: 1-58113-507-6. DOI: 10.1145/543613.543615. URL: http://doi.acm.org/10.1145/543613.543615.

[Bar+02] Ziv Bar-Joseph et al. "K-ary Clustering with Optimal Leaf Ordering for Gene Expression Data". In: *Algorithms in Bioinformatics: Second International Workshop, WABI 2002 Rome, Italy, September 17–21, 2002 Proceedings*. Ed. by Roderic Guigó and Dan Gusfield. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 506–520. ISBN: 978-3-540-45784-8. DOI: 10.1007/3-540-45784-4_39. URL: https://doi.org/10.1007/3-540-45784-4_39.

[CE97] Michael Cox and David Ellsworth. "Application-controlled Demand Paging for Out-of-core Visualization". In: *Proceedings of the 8th Conference on Visualization '97*. VIS '97. Phoenix, Arizona, USA: IEEE Computer Society Press, 1997, 235–ff. ISBN: 1-58113-011-2. URL: http://dl.acm.org/citation.cfm?id=266989.267068.

[Cha+97] S. Chandrasekaran et al. "An Eigenspace Update Algorithm for Image Analysis". In: *Graphical Models and Image Processing* 59.5 (1997), pp. 321–332. ISSN: 1077-3169. DOI: http://dx.doi.org/10.1006/gmip.1997.0425. URL: http://www.sciencedirect.com/science/article/pii/S1077316997904251.

[Chu+15] D. Chu et al. "Incremental Linear Discriminant Analysis: A Fast Algorithm and Comparisons". In: *IEEE Transactions on Neural Networks and Learning Systems* 26.11 (Nov. 2015), pp. 2716–2735. ISSN: 2162-237X. DOI: 10.1109/TNNLS.2015.2391201.

[DHS00] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification (2Nd Edition)*. Wiley-Interscience, 2000. ISBN: 0471056693.

[DV13] N. Djuric and S. Vucetic. "Efficient Visualization of Large-Scale Data Tables through Reordering and Entropy Minimization". In: *2013 IEEE 13th International Conference on Data Mining*. Dec. 2013, pp. 121–130. DOI: 10.1109/ICDM.2013.63.

[HHB08] Michael Hahsler, Kurt Hornik, and Christian Buchta. "Getting Things in Order: An Introduction to the R Package seriation". In: *Journal of Statistical Software* 25.3 (Mar. 2008), pp. 1–34. ISSN: 1548-7660. DOI: 10.18637/jss.v025.i03.

[HMM98] Peter M. Hall, David Marshall, and Ralph R. Martin. "Incremental Eigenanalysis for Classification". In: *in British Machine Vision Conference*. 1998, pp. 286–295.

[Kim+11] Tae-Kyun Kim et al. "Incremental Linear Discriminant Analysis Using Sufficient Spanning Sets and Its Applications". In: *International Journal of Computer Vision* 91.2 (Jan. 2011), pp. 216–232. ISSN: 1573-1405. DOI: 10.1007/s11263-010-0381-3. URL: https://doi.org/10.1007/s11263-010-0381-3.

[Lan01] Douglas Laney. *3D Data Management: Controlling Data Volume, Velocity, and Variety*. Tech. rep. META Group, Feb. 2001. URL: http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf.

[Li+03] Y. Li et al. "An integrated algorithm of incremental and robust PCA". In: *Proceedings 2003 International Conference on Image Processing (Cat. No.03CH37429)*. Vol. 1. Sept. 2003. DOI: 10.1109/ICIP.2003.1246944.

[MS00] Erkki Mäkinen and Harri Siirtola. "Reordering the Reorderable Matrix As an Algorithmic Problem". In: *Proceedings of the First International Conference on Theory and Application of Diagrams*. Diagrams '00. London, UK, UK: Springer-Verlag, 2000, pp. 453–467. ISBN: 3-540-67915-4. URL: http://dl.acm.org/citation.cfm?id=645970.674914.

[POK05] Shaoning Pang, S. Ozawa, and N. Kasabov. "Incremental linear discriminant analysis for classification of data streams". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 35.5 (Oct. 2005), pp. 905–914. ISSN: 1083-4419. DOI: 10.1109/TSMCB.2005.847744.

[Reh+16] Muhammad Habib ur Rehman et al. "Big Data Reduction Methods: A Survey". In: *Data Science and Engineering* 1.4 (Dec. 2016), pp. 265–284. ISSN: 2364-1541. DOI: 10.1007/s41019-016-0022-0. URL: https://doi.org/10.1007/s41019-016-0022-0.

[Wan+17] Yi Wang et al. "Fast Online Incremental Learning on Mixture Streaming Data." In: *AAAI*. 2017, pp. 2739–2745.

[WB13] Jonathan Stuart Ward and Adam Barker. "Undefined By Data: A Survey of Big Data Definitions". In: *CoRR* abs/1309.5821 (2013). URL: http://arxiv.org/abs/1309.5821.

[WF09] Leland Wilkinson and Michael Friendly. "The History of the Cluster Heat Map". In: *The American Statistician* 63.2 (2009), pp. 179–184. DOI: 10.1198/tas.2009.0033. eprint: http://dx.doi.org/10.1198/tas.2009.0033. URL: http://dx.doi.org/10.1198/tas.2009.0033.

[WZH03] Juyang Weng, Yilu Zhang, and Wey-Shiuan Hwang. "Candid covariance-free incremental principal component analysis". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25.8 (Aug. 2003), pp. 1034–1040. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2003.1217609.

[Yan+06] Jun Yan et al. "Effective and Efficient Dimensionality Reduction for Large-Scale and Streaming Data Preprocessing". In: *IEEE Trans. on Knowl. and Data Eng.* 18.3 (Mar. 2006), pp. 320–333. ISSN: 1041-4347. DOI: 10.1109/TKDE.2006.45. URL: http://dx.doi.org/10.1109/TKDE.2006.45.

[Ye+05] Jieping Ye et al. "IDR/QR: an incremental dimension reduction algorithm via QR decomposition". In: *IEEE Transactions on Knowledge and Data Engineering* 17.9 (Sept. 2005), pp. 1208–1222. ISSN: 1041-4347. DOI: 10.1109/TKDE.2005.148.

[YLO07] M. Ye, X. Li, and M. E. Orlowska. "Supervised Dimensionality Reduction on Streaming Data". In: *Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)*. Vol. 1. Aug. 2007, pp. 674–678. DOI: 10.1109/FSKD.2007.548.

[You+05] S. Stanley Young et al. "Niermann, S. (2005), "optimizing the ordering of tables with evolutionary computation," the American statistician, 59, 41-46: Comment by Young, Liu, and Hawkins and reply [2] (multiple letters)". In: *American Statistician* 59.4 (Nov. 2005), pp. 353–354. ISSN: 0003-1305. DOI: 10.1198/000313005X72171.

[Zic13] Roberto V Zicari. "Big Data computing and clouds: Trends and future directions". In: (2013), pp. 103–128.