Globus An International Journal of Management & IT
A Refereed Research Journal

Vol 9 / No2 / Jan-Jun 2018 ISSN: 0975-721X

LINEAR ALGEBRA IN MODELING THE PROBABILITIES

***Kuldeep**

****Dr. Seema**

Abstract

Singular Value Decomposition (SVD) also, comparable techniques can be utilized to calculate networks subspaces which portray their conduct. In this paper we survey the SVD and summed up solitary esteem disintegration (GSVD) and a portion of their applications. We give specific regard for how these apparatuses can be utilized to detach vital examples in a dataset and give forecasts of future conduct of these examples. A noteworthy focal point of this task is the examination of a part looking like strategy portrayed by Michael Dettinger which gives assessments of likelihood conveyances to little arrangements of information. We tried the aftereffects of utilizing both the SVD and the GSVD for Dettinger's technique. Also to Dettinger, we found that the strategy tended to give likelihood dispersions a Gaussian shape notwithstanding when this did not appear to be spoken to in the first information. For a few informational indexes, nonetheless, both utilizing the SVD and GSVD gave what give off an impression of being sensible likelihood appropriations. There was not a critical contrast in how well unique likelihood dispersions were assessed when utilizing Dettinger's unique technique or the alterations with the diminished SVD or the GSVD. Utilizing Dettinger's strategy as opposed to a basic histogram dependably gave a higher goals of data, and was a few times equipped for coordinating the state of the first likelihood appropriations all the more nearly.

Keywords: Linear Algebra, Probability

Introduction

The accurate modeling and prediction of time series is becoming increasingly important in a range of applications, from meteorological forecasts to economic models. Along with the ability to predict a pattern comes the need to establish the reliability or accuracy of a prediction, preferably before the modeled event occurs. This paper addresses the issue of predicting the likelihood of an event from either a set of viable models for the event or a set of historical data. For example, figure shows output from six different weather models which each predicted the maximum temperature in each day of a thirty day period. If all six models which contributed to the data in this graph are equally likely to be accurate, then whoever is analyzing this information is faced with the task of deciding how to describe what the maximum temperature is likely to be on each day given six different predictions. Common statistical tools such as means, medians, standard deviations, and histograms are certainly reasonable choices for describing temperature likelihood on any given day. This could lead to problems, however, if the data is heavily skewed to one side of the median, there are outliers, or there are too few data points to calculate a meaningful standard deviation or create a useful histogram. In such cases, simple statistics will not give a very complete picture of the actual probabilities. This paper addresses such problems by asking: How can we use tools such as principal component analysis and independent component analysis to help accurately describe the likelihood of a future event based on an ensemble of models for said event? To explore this question, we will test and expand upon a method proposed by Michael Dettinger, of the U.S. Geological Survey, for estimating probability distributions based on model ensembles. This method provides a means of estimating probability distributions for time series described by small data sets. When there are only a small number of predictions for a future event, there may not be enough data points to provide a meaningful picture of the event's probability.

***Research Scholar, Kalinga University, Naya Raipur, Chhattisgarh**

****Supervisor, Kalinga University, Naya Raipur, Chhattisgarh**

Dettinger's method uses a principal component analysis (PCA) to divide the data into separate components, which are then redistributed in order to provide a large number of new \predictions" which follow the trends of the original but are now large enough in number to be able to provide more detailed information.

Such a method has the potential of being very useful for making policy decisions based on models for such things as weather or hydrology. It provides a means of taking the results of many different models into account without relying on any subjective decisions such as what data points count as outliers or which models appear most reliable. Because of the nature of probabilities, it is impossible to know the underlying probability distribution for a predicted event with certainty. For example, it is al- ways possible to get data points which do not provide a representative sample of a distribution, which would make reconstruction of the original probability distribution unlikely. At the least, however, Dettinger's method provides a means of describing a rough probability distribution from a small random sample. For small data sets on which traditional statistics cannot provide significant results, such an approximation can at least provide some framework for describing a probability on a more detailed level than simply giving means and standard deviations. From a purely mathematical standpoint, this method also provides an interesting example of an application of the SVD, GSVD, and other tools of principal component analysis.

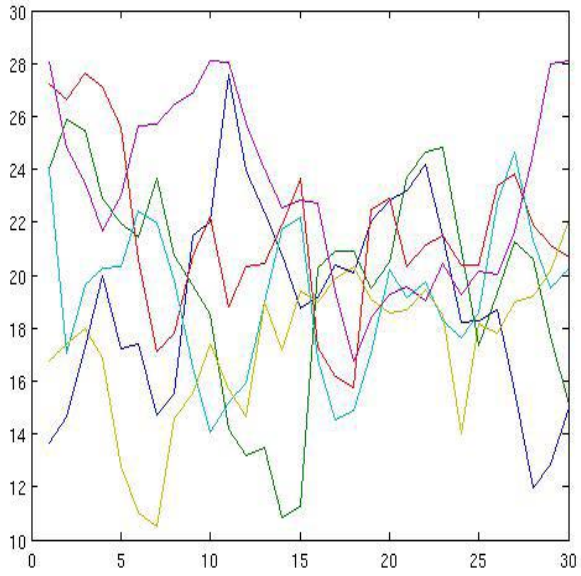

Figure 1: Six different models' predictions for maximum temperature over the samethirty day period in the same location

Review of Literature

J V F Lima et al., (2017) amid this article, we have a tendency to investigate execution and vitality utilization of 5 OpenMP runtime frameworks over a non-uniform access (NUMA) stage. we keep an eye on conjointly first class 3 CPU-level enhancements or procedures to guage their effect on the runtime frameworks: processors alternatives Turbo Boost and C-States, and CPU Dynamic Voltage and Frequency Scaling through Linux CPUFreq governors. we tend to blessing Associate in Nursing test concentrate to portray OpenMP runtime frameworks on the 3 principle parts in thick variable based math calculations (Cholesky, LU, and QR) regarding execution and vitality utilization. Our trial results direct that OpenMP runtime frameworks will be thought of as a substitution vitality use, and Turbo Boost, further as C-States, wedged impressively execution and vitality. CPUFreq governors had a considerable measure of contact with Turbo Boost crippled, since every improvement diminished execution in light of CPU warm breaking points. Relate in Nursing metallic component settling with simultaneous compose augmentation from libKOMP accomplished up to sixty three of execution gain and twenty ninth of vitality diminish over unique PLASMA run exploitation impala C compiler (GCC) libGOMP runtime.

Siddharth Samsi et al., (2017) Analysis of desoxyribonucleic corrosive examples is an indispensable advance in crime scene investigation, and furthermore the speed of research will affect examinations. Correlation of desoxyribonucleic corrosive successions is predicated on the examination of short bike worked for-two rehashes (STRs), that ar short desoxyribonucleic corrosive arrangements of 2-5 base sets. Current criminology approaches utilize twenty STR loci for investigation. the use of single ester polymorphisms (SNPs) has utility for investigation of confounded desoxyribonucleic corrosive blends. the usage of a huge number of SNPs loci for investigation presents fundamental process challenges because of the logical examination scales by the stock of the loci tally and assortment of desoxyribonucleic corrosive examples to be dissected. amid this paper, we tend to examine the usage of a desoxyribonucleic corrosive grouping correlation algorithmic control by re-throwing the algorithmic govern as far as variable based math natives. By building up an over laden grid task way to deal with desoxyribonucleic corrosive correlations, we can use progresses in GPU equipment and algorithms for Dense Generalized Matrix-Multiply (DGEMM) to hustle just a bit desoxyribonucleic corrosive example examinations. we tend to demonstrate that it's

capability to check twenty48 obscure desoxyribonucleic corrosive examples with 20 million noted examples in underneath about six seconds utilizing a NVIDIA K80 GPU.

A HAidar et al., (2011) the objective of this paper is to explore the dynamic arranging of thick polynomial math calculations on shared-memory, multicore models. Current numerical libraries, e.g., LAPACK, indicate clear restrictions on such rising frameworks chiefly as a result of their coarse graininess undertakings. Consequently, a few numerical calculations should be updated to raised work the field style of the multicore stage. The PLASMA library (Parallel polynomial math for ascendable Multi-center Architectures) created at the University of Tennessee handles this test by exploitation tile calculations to achieve a better undertaking graininess. These tile calculations will then be depicted by Directed Acyclic Graphs (DAGs), wherever hubs ar the undertakings and edges ar the conditions between the errands. The prevalent key to accomplish superior is to execute a runtime environment to with productivity plan the DAG over the multicore stage. This paper contemplates the effect on the execution of a few parameters, each at the measure of the equipment, e.g., window size and segment, and furthermore the calculations, e.g., Left attempting (LL) and Right attempting (RL) variations. we tend to reason that some generally acknowledged guidelines for thick variable based math calculations may must be returned to. The logical superior processing network has as of late round-confronted sensational equipment changes with the rise of multicore models. The vast majority of the snappiest superior PCs inside the world, if not all, specified inside the last Top500 list released in Nov 2009 ar as of now upheld multicore designs. This goes up against the logical code network with each an unnerving test and a particular shot. The test emerges from the troubling mate between the arranging of frameworks bolstered this new chip outline – a large number of hubs, 1,000,000 or a great deal of centers, lessened data measure and memory available to centers – and furthermore the components of the ordinary code stack, as numerical libraries, on that logical applications have depended for his or her exactness and execution. The cutting edge, elite thick polynomial math code libraries, i.e., LAPACK [4] have demonstrated impediments on multicore models . The execution of LAPACK relies upon the usage of an average arrangement of Basic polynomial math Subprograms (BLAS) [14], [20] at interims that almost the majority of the correspondence happens following the high-ticket fork-join worldview. In addition, its enormous walk memory gets to have extra exacerbated the issue, and it turns out to be fair to with effectiveness create existing or new numerical variable based math calculations fitting for such equipment.

Dettinger's Method

The basic idea behind Dettinger's component resembling method is that often there are multiple models for predicting a certain event, such as temperature or rainfall (these sets of models are called \forecast ensembles" by Dettinger) The models can either use different algorithms, have different input parameters, (i.e. climate models using different estimates for future atmospheric CO_2 concentrations) or both. Hopefully these models' predictions are fairly similar, though of course there will be some discrepancies, especially as the time from the initial conditions increases.

Dettinger's goal is to provide a way of estimating what outcome within the range given by the forecast ensemble is most likely the mean (not the sample mean, which is easily computed, but the actual mean of the unknown underlying probability distribution), and especially to estimate what outcomes are the most likely.

Predicting the likelihood of an event requires a method of estimating its probability distribution function (PDF) from a set of data. A PDF is a measure of the probabilities of obtaining various outcomes for a given event (such as the roll of a die or the temperature at a given time), assigning a unique probability to each possible outcome in the domain. Describing a distribution involves some method of applying a regression to a set of data in order to find an algebraic expression for its PDF. If there are many models, then a fairly accurate PDF can be found by simply using a histogram at each time step and creating a mathematical expression to describe the histograms' shapes. Unfortunately, for small numbers of models this will not be very precise since the number of histogram bins it is possible to fill will be limited by the number of data points. To address this problem, Dettinger suggests decomposing the original forecasts into individual component vectors. Specific linear combinations of the component vectors will recreate the original data, but recombining the vectors randomly will fill in the gaps by creating a large number of forecasts which capture the same ranges and variations as the originals but are still distinct, such as those in figure. This large new set of data allows for more meaningful statistics because the sample size is so much larger. The data may be artificial, but it is still related to the original data, and is designed, essentially, to interpolate the information between the given data points.

Dettinger describes this method as analogous to filtering the original ensembles through many narrow, non-overlapping \frequency bands" to separate the components into bins according to their frequencies. Each separate forecast will have a different\power" in each frequency bin. Imagine that instead of dividing up groups of vectors the goal is to separate out the colors of many different beams of light by shining them through various polarized media. Light beam A might contain mostly red light, so it would have a relatively high power when looked at through something which only allowed red light through. Light beam B, however, might contain mostly blue light and only a little red, so it would have a higher power in the blue bin than the red. To make a new forecast, you take a power from each frequency bin, regardless of which model it originally came from, and put these all together (for the light analogy, a possible reconstruction would be a new beam of light created using the amount of red light from beam A and the amount of blue light from beam B). After doing this many times, powers which appear more often will appear in a proportionally larger number of the reconstructed models, and the trends from the original forecasts will be represented in this new, larger forecast ensemble.

This chapter both analyzes results from using the component resembling method just described and extends upon it. The first goal is to test how accurately this method can reproduce PDFs from a limited sample. Next, we explore the use of different forms of principal component analysis and independent component analysis to possibly improve Dettinger's method. The main question we consider is whether removing noise from the data set will give more useful estimated PDFs, or if instead it would remove too much data to give a reliable prediction.

First, Dettinger's method is outlined in order to introduce the relationships between the topics which will be discussed in this paper. Further details on the justification for each step are given in later sections. We also discuss the background and applications of component resampling methods including Singular Value Decomposition and Independent Component Analysis.

DETTINGER'S COMPONENT RESAMPLING METHOD

Consider an ensemble of n different forecasts for the same set of m events. Each forecast x^j, $1 \le j \le n$, is m time steps long with elements $\{x^j_1, x^j_2, \ldots, x^j_m\}$. For example, figure shows a set of 6 different forecasts for the daily maximum temperature over 30 days; in this case n = 6 and m = 30. These forecasts can be compiled into an m x n matrix X, with a forecast in each column.

$$X_{m\times n} = \begin{bmatrix} x_1^1 & x_1^2 & \dots & x_1^n \\ x_2^1 & x_2^2 & \dots & x_2^n \\ \vdots & \vdots & \ddots & \vdots \\ x_m^1 & x_m^2 & \dots & x_m^n \end{bmatrix}$$

We want to factor X into

$$X = EP^T$$

with the columns of E_{mxm} being orthogonal vectors (the \frequency bins" or light polarizations in the above explanation). Each column of E is a vector $\mathbf{e}^k \in \mathbb{R}^m$, $1 \le k \le m$, written as

$$\begin{bmatrix} e_1^k & e_2^k & \dots & e_m^k \end{bmatrix}^T$$

Each vector has a corresponding coefficient in $P_{n\,x\,m}$ (the "strengths" of each forecast in each bin). The k^{th}columnof P is written

$$\mathbf{p}^k = \begin{bmatrix} p_1^k & p_2^k & \dots & p_n^k \end{bmatrix}^T$$

For example, the j^{th} element of the k^{th} coefficient vector, p^k_j, is the projection (strength) of the j^{th} ensemble member (column of X) on the k^{th} column of E. The goal is to resample the components in E according to the weights in P to get new reconstructed forecasts. We will call the set of reconstructed forecasts R.

Conclusion

These are the steps in Dettinger's component-resampling method. Justification for the more complex steps will be given in separate sections.

1. Calculate the mean values $\bar{x^i}$ for each time i across all forecasts:

$$\bar{x}_i = \frac{1}{n}\sum_{j=1}^{n} x_i^j$$

2. Calculate the ensemble standard deviations s_i of the centered forecasts at each time i:

$$s_i = \left[\frac{1}{n}\sum_{j=1}^{n}\left(x_i^j - \bar{x}_i\right)^2\right]^{1/2}$$

Separation the focused figure vectors at each time advance by the relating standard deviation. This will make an institutionalized conjecture troupe, X' (mean of zero and standard deviation of one at each time step). Every passage in X' is given by

$$x_i^{\prime j} = \frac{\left(x_i^j - \bar{x}_i\right)}{s_i} \text{ for } 1 \le j \le n \text{ and } 1 \le i \le m.$$

Subtracting the mean removes the mean shared by all the forecasts, and ensures that we reconstruct only the variations from this mean. Normalizing the standard deviation ensures that the variations between forecasts are treated in the same detail no matter how much the forecasts vary. After the ensemble is resampled we will reintroduce the mean and standard deviation.

3. Compute the m x m Cross Correlation Matrix $C = \frac{1}{n}X'X'^T$ which summarizes the covariance of each forecast with itself and each of the other forecasts at each time step. Defining c^s_r as the entry in the r^{th} row and s^{th} column of C, then each entry of C is found via

$$c_r^s = \frac{1}{n}\sum_{i=1}^{n} x_r^{\prime j} x_s^{\prime j} \text{ for } 1 \le r \le m \text{ and } 1 \le s \le m.$$

4. Construct the matrix E by setting each vector e^k equal to an eigenvector of C, i.e. $C\mathbf{e}^k = \lambda_k \mathbf{e}^k$. The corresponding eigen values, λ_k, represent the variance in X captured by the k^{th} vector of E.

Construct the coefficient matrix P by setting the columns of P equal to the projections of X' onto the vectors of E, so

$$\mathbf{p}^k = X'^T \mathbf{e}^k \text{ for } 1 \le k \le m$$

Construct additional "forecasts," $r^{\prime}$, for l values ranging from 0 to whatever number of reconstructed "forecasts" is desired. Do this by recombining the columns from E and P randomly for each time step.

Because of the way we created E and P, an exact reconstruction of each (standardized) data point would be found via $x_i^{\prime j} = \sum_{s=1}^{m} e_i^s p_j^s$, but the goal is to create distinct data sets. To do this, we will redistribute the individual components by randomly choosing the index j at each step in the equation, so the strengths of each eigenvector (or "frequency bin") in the constructed forecast is randomly chosen from all the strengths represented throughout the ensemble.

References

- Bryan et al., "Relationships between limit cycles and algebraic invariant curves for quadratic systems", *Journal of Differential Equations*, vol. 229, no. 2, pp. 529-537,1994.
- Chipman, George, and McCulloch, "The positive definite completion problem revisited", *Linear Algebra and its Applications*, vol. 429, no. 7, pp. 1442-1452, 2002.
- Kooperberg, Ruczinski, LeBlanc, and Hsu, An extremal sparsity property of the Jordan canonical form", *Linear Algebra and its Applications*, vol. 429, no. 10, pp. 2367-237, 2014.
- Ferryansyah et al., The analysis of students' difficulty in learning linear algebra, Journal of Physics, vol. 5, issue 4, pp. 87-98, 2017.
- D Plaxo et al., Analyzing student understanding in linear algebra through mathematical activity, Journal of Mathematical Behavior, vol. 38, issue 4, pp. 87–100, 2015.