IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 20, Issue 5, Ver. III (Sep - Oct 2018), PP 33-48
www.iosrjournals.org

SVD Adaptive Algorithm for Linear Least Square Regression and Anomaly Reduction

Chaman Lal Sabharwal
Missouri University of Science and Technology
Rolla, MO-65409, USA
Corresponding Author: Chaman Lal Sabharwal

Abstract -- *In data analysis, the accuracy of analysis depends on (1) the data representation, (2) the algorithm and (3) the metric used to measure error. For ordinary linear least square approximation (OLA), the existing formulation measures error along y or vertical direction. Conventionally, ordinary linear least square approximation (OLA) technique has been considered as the best fit regression line for linear trend data. Based on domain knowledge, several versions of OLA have been developed. They are all reformulations of OLA using prior domain knowledge, for supervised learning. Singular Value Decomposition (SVD) is also used least square approximation. The robustness of SVD approximation is attributed to (1) the SVD line is sensitive to temporal variation in time variables whereas OLA is not, it makes OLA less suitable for time sensitive data, and (2) SVD has smaller approximation error than OLA regression line. But SVD has inherent weaknesses. Herein we present a hybrid algorithm that achieves a balance between quantitative and qualitative approximation accuracy of both OLA and SVD. This algorithm is also suitable for noise reduction. Visualization is a preferred way to ascertain the quality of a new algorithm, we use MATLAB R2017b and linear regression in simple two-dimensional space to demonstrate the hybrid algorithm.*

Key-words: *least square regression, singular value decomposition, precision, propensity, noise*

Date of Submission: 10-10-2018 Date of acceptance: 26-10-2018

I. Introduction

For data analysis, most of the time raw data is not suitable to apply analysis algorithms. It becomes mandatary to preprocess data for reliable and accurate data mining and regression. Also, it is necessary to determine if the data is labelled, un-labelled or mixture of both. It is assumed that the data is accurate else the prediction analysis will be inaccurate accordingly. If the data is correlated and noisy, for general algorithm, it is preferable to transform the data into noise free and uncorrelated data as far as possible to avoid overfit. Furthermore, the numerical data may need to be regularized, mean centered and unit standard deviation etc. The approximation is matter of metric used to calculate approximation. For multivariable data, (x, y), x, y are vectors, sometimes y is scaler valued. The simplest case occurs when both x, y are scalar values, it is easy to understand. We use the simple case for examples in the paper. Now linear least square regression is line that represent the data by means of a straight line. This linear representation model approximates the non-parametric data points (x_i, y_i) with points (x_p, y_p) on a parametric line. Since the line depends on two parameters, intercept, a, and slope, b, the line is parametric representation of data. One of the models, measures error along the y-axis. In other words, $x_i = x_{ip}$, $y_{ip} = a + b\, x_{ip}$ such that the sum of squares of errors is minimum, error $E_l=\sum_{k=1,n} (x_k-x_{kp})^2+(y_k- y_{kp})^2 = \sum_{k=1,n}(y_k- a - b\, x_k)^2$.

In statistical analysis, the accuracy of approximation depends on several parameters. One such parameter is the metric used to measure the approximation error. Each metric has its own merits. We do assume that data is accurate, else we get inaccurate approximation. For linear least square approximation regression (OLA), we discuss its merits, and shortcomings of the metric to improve on it. For OLA, there are several issues. First it is least square approximation, it is in fact approximation in y direction, not min distance perpendicular to the approximation line [1],[2],[3]. In order to correct this, we devise a true line at min-distance from the input data, normal distance least square fit line. We refer to it normal linear least square approximation (NLA) similar to ordinary linear least square approximation (OLA). NLA may become complicated for multiple dimensions, we also show that linear algebra SVD can be leveraged to achieve OLA more easily. Finally, we see that OLA is not sensitive to data spread, NLA will also correct this deficiency of OLA. We also define a new metric, propensity scoring metric (PSM) for OLA, NLA and hybrid algorithms. Propensity score has been used in other area for estimating the effect of a treatment, policy or other causal effects [24]. We will show the effect of new metric as compared to OLA and NLA metrics. We show that the hybrid algorithm achieves a balance between quantitative and qualitative approximation accuracy of both OLA and SVD. Also, it will be shown that it can be used for noise and anomaly reduction. Thus, there are several approaches to approximate data linearly: ordinary linear least square regression (OLA), (new) normal linear least square regression (NLA), singular value decomposition linear least square regression (SVD), (new) hybrid linear least square regression (HLA). To measure the accuracy of approximation, there are several metrics: quantitative and qualitative. Knowing what technique and metric to use makes all the difference in analysis and makes most out of data. That way one spends less time on justifying the conclusions. The challenge is the decision making on the metric used to approximate. The intent of this paper is the design a hybrid algorithm that yields better approximation than the OLA and SVD approximate algorithms, also a way to detect and remove anomalies in data.

The paper is organized as SectionII describe OLA and an efficient computation by mean-centering data formulation, Section III derives new NLA, Section IV describes SVD and it connection to NLA, Section V gives new hybrid approximation algorithm and its implementation, error analysis of OLA,NLA, SVD, and Hybrid algorithms is provided with respect to both metrics, it introduces propensity score metric and anomaly reduction, Section VI is conclusion.

II. Background

Data is represented as a matrix of real or discrete values. It is easier to work with data if it is regularized. Simple example of regularization is mean-centered normalized data. It may be standardized to unit standard deviation. Ordinarily the reference point of data is the origin, mean-centering implies that the centroid of data is translated to the origin. We will soon see how mean-centering simplifies the computations.

Let the data be represented by an m×n matrix A, i.e., m rows of n-vectors or n columns of m-vectors. If **x** is column of A, the mean of $\mathbf{x}_i$'s denoted by $\bar{x}$, **x** is translated to **x** - $\bar{x}$. Similarly, if y is row of A, it is replaced with **y** - $\bar{y}$, where the mean of **y** is $\bar{y} = \frac{\sum_i y_i}{n}$. Further, if **x** and **y** are both columns or both rows, the means of dot product of **x** and **y** is denoted by $\overline{xy} = \frac{\sum_i x_i y_i}{n}$, for **x** = **y**, it is denoted by $\overline{x^2} = \frac{\sum_i x_i^2}{n}$. For matrix operations most of the linear transformations are performed by means of matrix multiplication, centralization is a linear transformation [4] for mean-centering a matrix. There is an immaculate transformation T_m to mean-center the columns of A as follows. Let I_m be mxm identity matrix, $\mathbf{e_m}$ be a column m-vector of ones, and $T_m = I_m - \mathbf{e_m e_m}^T/m$. This T_m is called the column centralizer. For example, if **x** is a column vector then

$T_m\mathbf{x}$ $=I_m\,\mathbf{x} - \mathbf{e_m e_m}^T\mathbf{x}/m$
$= \mathbf{x} - \mathbf{e_m e_m} \bullet \mathbf{x}/m$
$= \mathbf{x} - \bar{x}e_m$

or in short **x** -$\bar{x}$ where $\bar{x}$ is the mean of **x**. This T_m applied on the left of A, it centralizes columns of the matrix. Similarly, if T_n is multiplied on the right of A, the AT_n mean-centers the rows of A. For example, for row vector **y**:

$\mathbf{y}T_n$ $= (\,\mathbf{y}\,I_n - \mathbf{y e_n e_n}^T/n)$
$= \mathbf{y} - \mathbf{y} \bullet \mathbf{e_n}\, \mathbf{e_n}^T /n$
$= \mathbf{y} - \mathbf{y} \bullet \mathbf{e_n}/\mathbf{n}\; \mathbf{e_n}^T$
$= \mathbf{y} - \bar{y}\mathbf{e_n}^T$

After preforming analysis on mean-centered data, reference point can be translated back to the centroid. This is a standard technique used for visualization [5],[6].

A. Ordinary Linear Least Square Approximation (OLA)

A.1 Conventional formulation

For input data n×2 matrix, columns are x, y coordinates of data points, we find a linear least square approximation line. Before exploiting any approximation, it is assumed that data is accurate, else prediction will also be inaccurate. For linear approximation line y = a + b x, we need to calculate *two parameters* a and b for minimizing of

$$f(a,b)=\sum_{i=1,n}(y_i - a - bx_i)^2.$$

That leads to two equations

$$\frac{\partial f(a,b)}{\partial a} = \sum_{i=1,n}(y_i - a - bx_i) = 0 \qquad (1)$$
$$\bar{y} - a - b\,\bar{x} = 0$$

and

$$\frac{\partial f(a,b)}{\partial b} = \sum_{i=1,n}(y_i - a - bx_i)x_i = 0 \qquad (2)$$
$$\overline{xy} - a\bar{x} - b\overline{x^2} = 0.$$

The first equation (1) becomes $\bar{y} = a + b\,\bar{x}$, which implies that the regression line passes through the centroid ($\bar{x}$,$\bar{y}$). These two equations are

$$\bar{y} = a + b\,\bar{x} \text{ and } \overline{xy} = a\bar{x} + b\overline{x^2}$$

can be solved for a and b to yield

$$b = \frac{\overline{xy} - \bar{x}\bar{y}}{\overline{x^2} - \bar{x}^2} \quad \text{and} \quad a = \frac{\overline{x^2}\bar{y} - \bar{x}\overline{xy}}{\overline{x^2} - \bar{x}^2}$$

However since $\bar{y} = a + b\,\bar{x}$, once b is known, the offset/bias term a can be efficiently computed from a = $\bar{y} - b\,\bar{x}$.

It may be noted that for mean-centered data, $\bar{x}$ = 0, $\bar{y}$ = 0, it results in a=0.

A.2 Mean-Centered data formulation

Mean-centering allows us to consider regression line through the origin because centroid is translated to the origin. The bias term *a* becomes zero automatically and the data becomes unbiased. To take advantage of regularization, the OLA can be reformulated for mean-centered data, we need to compute *only one* parameter b for

minimizing f(b) $=1/n\sum_{i=1,n}(y_i-bx_i)^2$

or

$$\begin{aligned} f(b) &= 1/n\sum_{i=1,n}(y_i-bx_i)^2 \\ &= 1/n\sum_{i=1,n} (y_i^2 -2by_ix_i+ b^2 x_i^2) \\ &= \overline{y^2} - 2b\overline{xy} + b^2\overline{x^2} \end{aligned}$$

That is

$$f(b) = \overline{y^2} - 2\overline{xy}\,b + \overline{x^2}b^2$$

Calculus based critical value criteria requires that f'(b) = 0. This leads to $-2\overline{xy} + \overline{x^2}\,2b$ = 0 or

$$b = \frac{\overline{xy}}{\overline{x^2}}$$

So, for mean-centered data, OLA line is

$$y = bx, \text{ with } b = \frac{\overline{xy}}{\overline{x^2}}$$

which is simpler expression than the raw data computations. Since f''(b) = 2 $\overline{x^2}$ is positive, the critical value is minimum.

However, if we want to go to the original frame, original reference point, we translate the origin back to the centroid

then line translate into original coordinates

$$y - \bar{y} = b(x-\bar{x}) \text{ or } y = \bar{y} - b\bar{x} + b\,x$$

that is

$$y = a+ b\,x \text{ where } a = \bar{y} - b\bar{x}$$

In this case, *only* b is to be computed, a is automatic byproduct.

This gives a line through (0,a) and along the direction $\frac{(1,b)}{\sqrt{(1+b^2)}}$

Non-Calculus based algebraic approach proceeds as follow.

$$f(b) = \overline{y^2} - 2\overline{xy}\,b + \overline{x^2}b^2$$

Since it is convex function, there is *only one* minimum,Figure1. This expression simplifies to

$$\begin{aligned} f(b) &= \overline{y^2} - 2\overline{xy}\,b + \overline{x^2}b^2 \\ &= \overline{x^2}\left(b - \frac{\overline{xy}}{\overline{x^2}}\right)^2 + \frac{\overline{x^2y^2} - \overline{xy}^2}{\overline{x^2}} \end{aligned}$$

Since $\overline{x^2y^2} - \overline{xy}^2 \geq 0$, f(b) is min only if b = $\frac{\overline{xy}}{\overline{x^2}}$. This what we saw above using calculus.

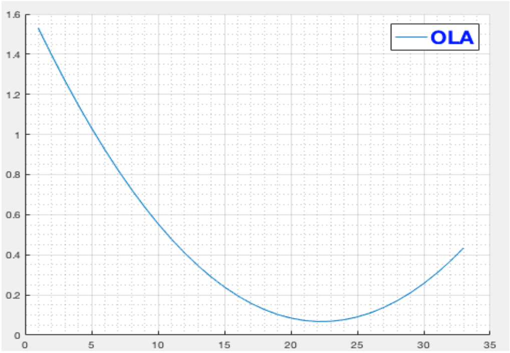


Figure 1. The convex function f(b) has only one global minimum giving the slope for OLA line.

In essence, this is a common sense three step approach to find the OLA line. The three steps are, (1) mean-center the data, translate the centroid ($\bar{x}$,$\bar{y}$) to the origin (0,0), (2) calculate the direction of least square error approximating line through the origin, (3) translate back to centroid ($\bar{x}$,$\bar{y}$) for original frame of reference. The computations using mean-centered data are simpler.In,Figure2, Cyan dots are the raw data, red line is the approximation line, and red dotted lines are errors between the data and corresponding approximations. In Figure3, black dotted lines are normal to the regression line where as red dotted lines are vertical, along the y-axis direction. Clearly the normal lines are shorter than vertical lines. We will explore further whether there are some other lines whose normal distance error is even smaller than this line error. That leads us to next section.

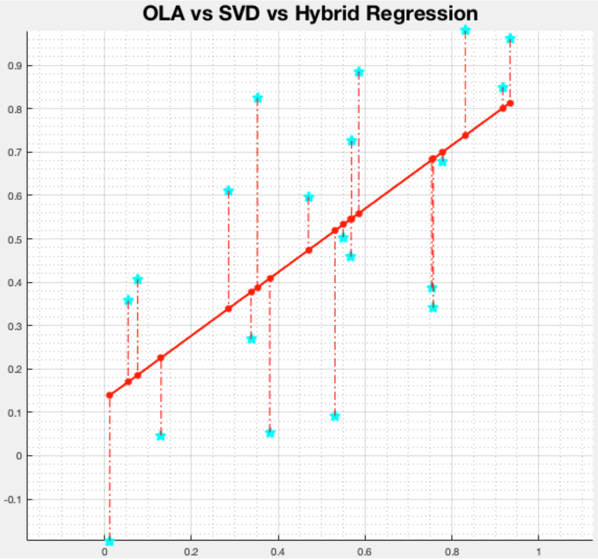


Figure 2. Data points, regression line, approximation errors

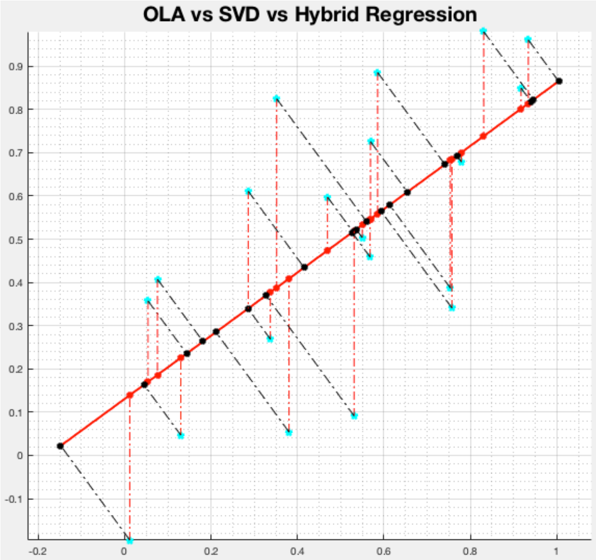


Figure 3. The red vertical dotted lines are error from OLA line along y-axis, the black orthogonal dotted lines are error from OLA line along the normal. Normal distance error is smaller than vertical distance error.

III. Normal Linear Least Square Approximation (NLA)

The ordinary linear approximation(OLA) line is not as close to the data points because distances are measured along the y-axis. If distances are measured along the normal (perpendicular) to the approximation line, then line is more representative of data. The normal (perpendicular, orthogonal) distance problem is formulated below. For the reasons stated above, we assume that the data is mean-centered, else it can be transformed by centralizer transformation to mean-center it. The problem becomes that of finding the value of *only b* that minimizes f(b) where

$$f(b) = 1/n \sum_{i=1,n} (\frac{y_i - bx_i}{\sqrt{1+b^2}})^2 \quad \text{or}$$

$$f(b) = 1/n \sum_{i=1,n} \frac{({y_i}^2 + b^2 {x_i}^2 - 2bx_i y_i)}{1+b^2}$$

$$= \frac{\overline{y^2} + b^2\overline{x^2} - 2b\overline{xy}}{1+b^2}$$

Thus, for local minima of

$$f(b) \quad = \frac{\overline{y^2} + b^2\overline{x^2} - 2b\overline{xy}}{1+b^2} \qquad (1)$$

$$f(b) \quad = \frac{b^2\overline{x^2} - 2b\overline{xy} + \overline{y^2}}{1+b^2} = \frac{b^2\overline{x^2} - 2b\overline{xy} + \frac{\overline{xy}^2}{\overline{x^2}} - \frac{\overline{xy}^2}{\overline{x^2}} + \overline{y^2}}{1+b^2}$$

$$= \frac{b^2\overline{x^2} - 2b\overline{xy} + \frac{\overline{xy}^2}{\overline{x^2}} - \frac{\overline{xy}^2}{\overline{x^2}} + \overline{y^2}}{1+b^2}$$

$$= \frac{\overline{x^2}(b \ - \frac{\overline{xy}}{\overline{x^2}})^2 \ - \ \frac{\overline{xy}^2}{\overline{x^2}} + \overline{y^2}}{1+b^2}$$

$$= \frac{\overline{x^2}(b \ - \frac{\overline{xy}}{\overline{x^2}})^2 \ + \ \frac{\overline{x^2y^2} - \overline{xy}^2}{\overline{x^2}}}{1+b^2}$$

Note $\overline{x^2y^2} \ - \ \overline{xy}^2$always ≥ 0. It is equivalent to standard result |**x•y**|≤|**x**||**y**| which can be quickly derived from triangle inequality.

We saw that in the unnormalized case, f(b) is minimum when

$$b \ - \frac{\overline{xy}}{\overline{x^2}} = 0$$

or

$$b \ = \frac{\overline{xy}}{\overline{x^2}}$$

This is not true in this case, see Figure 4. For OLA, f(b) is convex and has only one extreme/minima. For NLA, f(b) is not convex. It has two extrema, one maxima and one minima. In both cases, the minima are close to each other, but not identical.

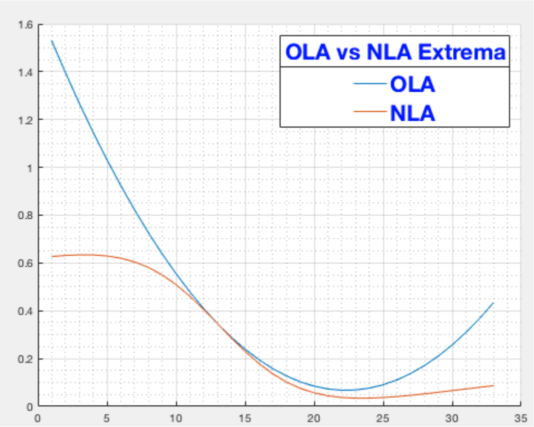


Figure 4. The minimization function f(b) is convex for OLA case.

f(b) is not convex for NLA case.

For NLA, f(b) is never negative. As b approaches zero, f(b) becomes $\overline{y^2}$ and as b approaches infinity, f(b) becomes $\overline{x^2}$.

To calculate the minimum, setting the first derivative of f(b) w.r.t b to zero, f'(b)=0, we get quadratic

$$\overline{xy}\, b^2 + \left(\overline{x^2} - \overline{y^2}\right) b - \overline{xy} = 0 \qquad (2)$$

Since it is a quadratic, it has two critical values, b_1, b_2

$$b = \frac{-\,(\overline{x^2} - \overline{y^2}) \pm \sqrt{(\overline{x^2} - \overline{y^2})^2 + 4\,\overline{xy}^2}}{2\,\overline{xy}} \qquad (3)$$

f(b) can't have both local minima, see Figure 4. If f'(b_1)>0, the b_1 is a local minima else f'(b_2)>0 , then b_2 is a local minima. Once b is computed, we have a line through the origin (0,0) along the *direction* $\frac{(1,b)}{\sqrt{(1+b^2)}}$

The normal least square line (NLA) are shown in Figure5. This is not the same as OLA regression line as seen in Figure2 and Figure3.

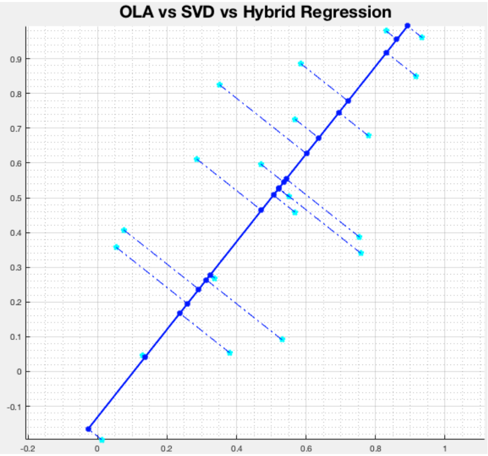


Figure 5. Cyan dots are the data points blue line is NLA line. Blue dots are the approximation, Blue dotted lines are normal errors from NLA line.

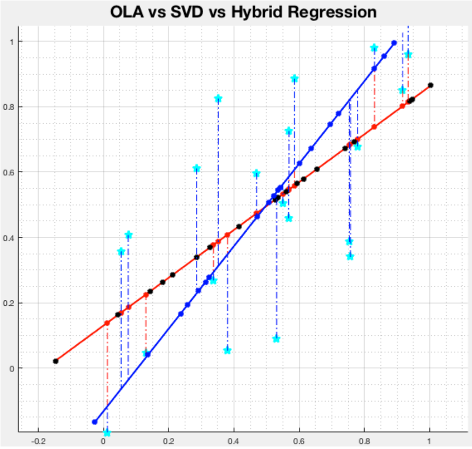


Figure 6. Red line is OLA, Blue line is NLA. Red dotted lines and Blue dotted lines are vertical errors form the Cyan data points. NLAvertical error from Blue line is *more* than OLA error from red line.

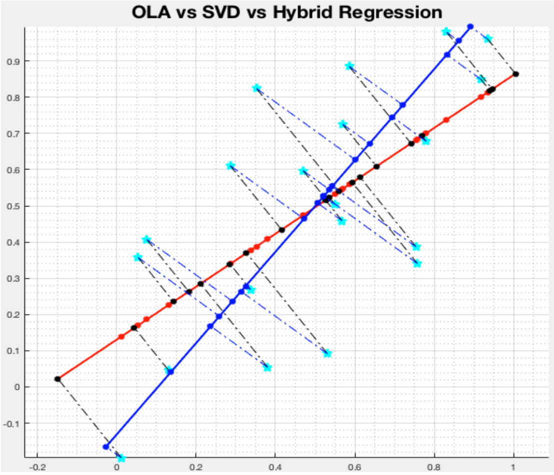


Figure 7. Red line is OLA, Blue line is NLA. Red dotted lines and Blue dotted lines are orthogonal errors form the Cyan data points. NLA normal error from Blue line is *less* than OLA error from red line.

Further, the approximation error in both cases (OLA and NLA) is minimum depending on how the error is measured. Visual inspection shows that *majority* of the cyan dots are *closer* to blue line dots than the cyan dots to red line dots, see Figure6, Figure7. This visualization justifies, to some extent, to prefer NLA over OLA. We will give formal justification later in section V. Since NLA is based on calculus, its derivative is complex, the second derivative is quite complex, we explore an easier implementation of this idea by means of linear algebra, singular value decomposition (SVD).

IV. Singular Value Decomposition (SVD)

This normal least square approximation (NLA) line can also be obtained directly by using linear algebra singular value decomposition (SVD). Today, singular value decomposition is used in many branches of science: computer science and engineering, psychology and sociology, atmospheric science and astronomy, health and medicine etc. [7],[8],[9],[10],[11],[12],[13]. It is also extremely useful in machine learning and in both descriptive and predictive statistics. For the sake of completeness, we give brief description of SVD.

Singular Value Decomposition (SVD) is a matrix factorization technique generalizing eigen-decomposition. Every positive semi-definite real matrix can be decomposed into three matrix factors: left singular vectors matrix, right singular vectors matrix and a diagonal matrix of singular values in descending order on main diagonal. The goal is not to recreate the matrix, but to create the *best linear least square approximation* [14], [15]. There are various advantages of SVD. First, 150 years old *Principal Component Analysis* (PCA) is a generalization of eigen-decomposition to symmetric matrices with orthogonal eigenvectors such that $A = VDV^{-1} = VDV^T$. In our case, A is data matrix, it not a square. But A^TA is a symmetric square positive semi-definite matrix, then $A^TA = VDV^T$, [16],[17],[18]. Besides other benefits of this factorization, we are interested in *direction vector* only. The columns of V are eigenvectors of A^TA corresponding to eigenvalues arranged in descending order. Since we are interested in direction of approximation line, we show that direction vector of NLA corresponds to first eigenvector of SVD [19], [20],[21].

We derive the direction **v** so that sum of squares of distances of points from **v** is least. Note, v passes through the origin and the data is mean-centered. Since data is mean-centered, the approximation line passes through the origin. By default, vectors **P** are column vectors in linear algebra, thus rows of A are position vectors [x,y] = $\mathbf{P^T}$. The vector **P** can be written as the sum of a vector along unit vector **v** and a unit vector **w** orthogonal to **v**, that is, using vector notation **P = P•v v+ (P-P•v v) = v**v+w**w.** This means that minimizing the distance w amounts to maximizing the component v. We are to maximize over all data points $\mathbf{P_i}$. The problem becomes that of maximizing

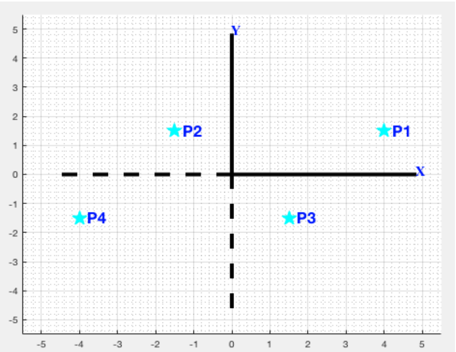
$$\sum_{i=1,n} |\mathbf{Pi{\bullet}v}|^2$$

for *all* P_i and *some* vector **v**, that is of interest to us. Now

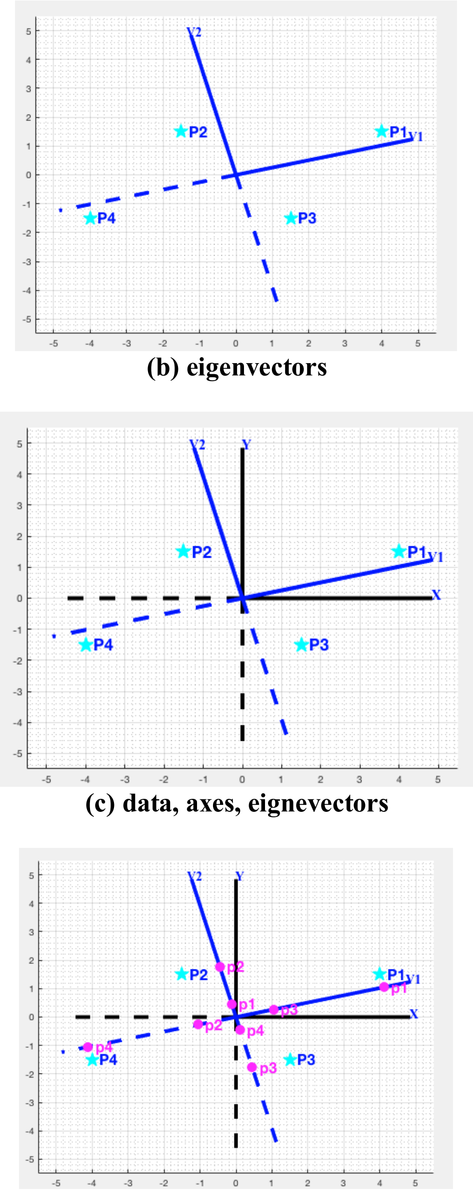
$$\begin{aligned}\sum_{i=1,n} |\mathbf{Pi{\bullet}v}|^2 &= \sum_{i=1,n} \mathbf{Pi{\bullet}v\ Pi{\bullet}v} = \sum_i \mathbf{v{\bullet}Pi\ Pi{\bullet}v}\\ &= \sum_{i=1,n} \mathbf{v^TPi\ Pi^Tv} = \mathbf{v^T} (\sum_{i=1,n} \mathbf{Pi\ Pi^T})\mathbf{v}\\ &= \mathbf{v^T} (A^TA)\mathbf{v}.\end{aligned}$$

This means that $\sum_i |\mathbf{Pi{\bullet}v}|^2$ is maximum if **v** is an eigenvector of A^TA and corresponds to largest eigenvalue of A^TA. Similarly, all the other eigenvectors can be obtained incrementally one at a time, constraining each vector orthogonal to the previous eigenvectors. Thus, SVD is computed iteratively in descending order of eigenvalues and corresponding eigenvectors orthogonal to the previously computed eigenvectors. It may be noted that largest eigenvalue refers to the largest spread of data along the eigenvector. The spread of projections of data on $\mathbf{v_1}$ is larger than that on$\mathbf{v_2}$, see Figure8(d).

For example, $\mathbf{P^T}$'s are data points in 2D, $\mathbf{v_1}$**,** $\mathbf{v_2}$ are eigenvectors corresponding to largest eigenvalues of A^TA . For this consideration, the NLA requires only $\mathbf{v_1}$, the direction with largest eigenvalue, and with largest data spread.



(a) data points,



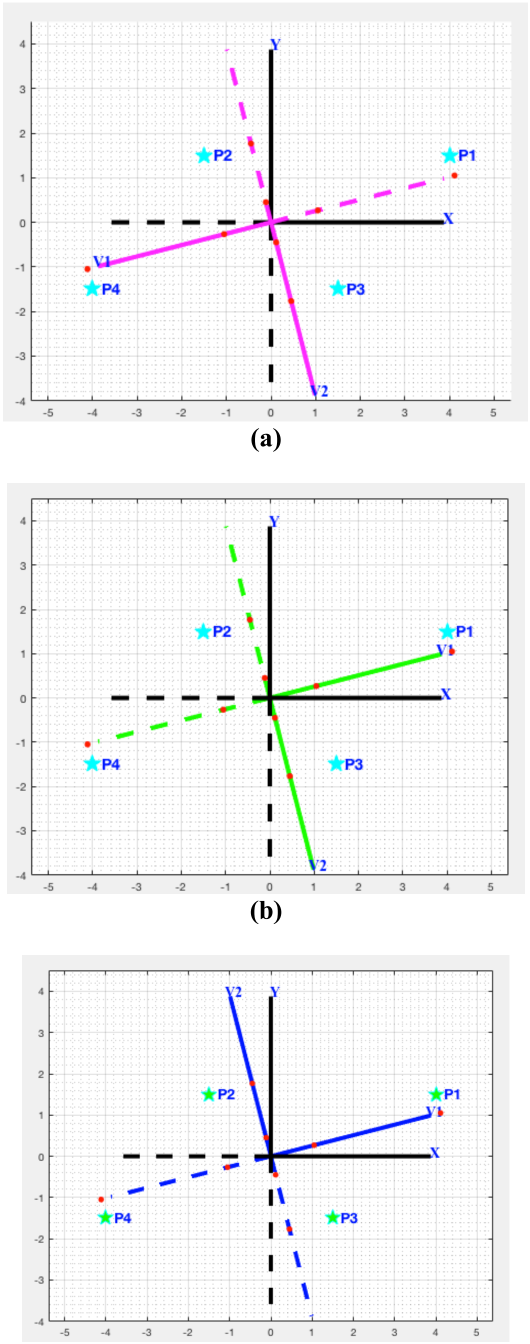
(b) eigenvectors

(c) data, axes, eignevectors

(d) everything with projections on eigenvectors.

Figure 8. (a) Four data points {P_1, P_2, P_3, P_4} with standard axes, (b) Four data points {P_1, P_2, P_3, P_4} with eigenvectors, axes of data trend, (c) data points, standard xy-axes, eigenvectors frame, (d) both xy and v_1v_2, frames with data points and projections on v_1 and v_2.

Uniqueness of Eigenvectors. As a side remark, for the matrix, any non-zero multiple of an eigenvector is again an eigenvector. To make the eigenvectors unique, they are normalized to unit vectors. But if **u** is unit eigenvector, then –**u** is also a unit vector, see Figure 9(a) for MATLAB SVD computed eigenvectors [19], [20]. In the literature. It is an accepted convention to make the first non-zero component positive in the eigenvector, see Figure 9(b). Since eigenvectors are ordered, we use ordering to make the k-th element of k-th vector to be positive, see Figure 9(c) that makes the vectors look more natural like a right-handed system. In case, the kth element is zero, then the first non-zero element is made positive. This is the approach we prefer to use [21].Incidentally, recall that the direction vectors in OLA and NLA had first component as positive.



(a)

(b)

(c)

Figure 9. (a) Eigenvectors as computed by MATLAB SVD, (b) each vector has first no element positive by convention, (c) first eigenvector has first component positive, second eigenvector has second component positive on using ordering of eigenvectors, so the eigenvectors form a right handed system.

V. Hybrid Algorithm Design

We design a hybrid algorithm leveraging the best of OLA and NLA/SVD approximation lines in two forms: non-parametric overfitting and parametric in general. For each observed point, (x_0,y_0), we have seen in Figure 6 and Figure 7 that there is a corresponding predicted point (x_R, y_R) on regression line and a predicted point (x_S, y_S) on SVD line. If (x_0,y_0) is an observed value, (x_R, y_R) is predicted point value corresponding to the OLA line y=a+bx. The vertical distance is along y direction. The distance between (x_0,y_0) and (x_R,y_R) is the y-distance, the OLA regression error $e_R = |y_0-y_R|$. For normal distance from NLA or SVD approximation line, it is along perpendicular to the line, it turns out that $x_S \neq x_0$ in (x_S, y_S) , the distance between (x_0, y_0) and (x_S,y_S) is Euclidian normal distance $e_S = \sqrt{((x_0-x_S)^2 + (y_0-y_S)^2)}$.It is clear from Figure 6 and Figure 7 that for some points in observed data, $e_R < e_S$ while for some other points $e_S < e_R$. In each method, the total error E is sum of squares of pointwise distances (errors) for all data points, question arises which one (E_R for OLA and E_S for SVD) is acceptable due to the dual nature on error computation. There is no denying the fact if vertical distances are used for *both* lines, then $E_R<E_S$ and if normal distances are used for *both* lines, then $E_S < E_R$. Then how does the user determine which one preferable to use: OLA or NLA/SVD? For hybrid algorithm, define the approximation point (x_H, y_H) to be that point which is closer to the observed point (x_0,y_0) in both ways. Euclidean distance is used to measure closeness. For each input, we will determine approximate line that represents the input data no matter how the error is computed, see Figure11for green color dots, these are closer to cyan dots than red line dots or blue line dots. Instead of measuring the quantitative distance we define a qualitative metric that is more useful in visualization and cognitively acceptable.

A. Non-Parametric Hybrid algorithm

Algorithm A:

```
Input: array of x and y mean-centered data values
Output: hybrid approximation points ($x_H$, $y_H$),where($x_R y_R$) is on OLA, ($x_S$, $y_S$) is on SVD line
1.  Calculate a and b for OLA regression for observed x, y
    Calculate predicted values by linear regression $y_R$ = a+bx
    Calculate  approximation error $E_R$
    Test Goodness of the regression line
2.  Calculate A=[x, y], x, y are columns of matrix A.
    Calculate SVD  [U S V] = svd(A)
    Use first column of V to get b.  a is automatic
    Calculate $x_S$, $y_S$ of projected points [$x_S$, $y_S$] on column vectors  of V that is AVV'
    Calculate approximation error $E_S$
    Compare error $E_R$ and $E_S$
3.  Calculate hybrid $x_H$, $y_H$ using variation of relaxation method
    for all point pairs($x_R$, $y_R$),($x_S$, $y_S$)
            if d( ($x_S$, $y_S$), ($x_0$, $y_0$))<= d( ($x_R$, $y_R$), ($x_0$, $y_0$))
                    ($x_H$, $y_H$) = ($x_S$, $y_S$);
            else
                    ($x_H$, $y_H$) = ($x_R$, $y_R$);
            end
    end
    Calculate error $E_H$ from pointwise $e_H$
    Compare error $E_S$, $E_R$, $E_H$
    Calculate and Compare by propensity values
4.  $x_H$, $y_H$ are arrays of predicted coordinates on hybrid polygonal line.
```

This algorithm gives non-parametric polygonal approximation and overfitting. The next algorithm parametrizes it by using SVD, see Table 1. In practice we do not need to store for prediction it is more efficient to retain the line parameters of OLA and NLA/SVD for in real time calculations.

B. Parametric Hybrid algorithm

This algorithm is of theoretical interest, Algorithm A is sufficient for practical use. The non-parametric *polygonal* approximation algorithm gives insight for improving the accuracy, Figure 12. It has two shortcomings it does conserve space, and it is overfitting the input. Here we explore double approximation to design a general algorithm which conserves space as well as it is parametric, see Figure 12.

Algorithm B

```
Input: array of x and y mean-centered data values
Output: hybrid approximation line parameters for points ($x_H$, $y_H$),where($x_R$, $y_R$) is on OLA, ($x_S$, $y_S$) is on SVD line
1.  From algorithm A,  ($x_H$, $y_H$) is polygonal hybrid approximation
2.  Use SVD to fit points ($x_H$, $y_H$) with SVD algorithm  to derive parameters for the direction of the line
3.  Use parameters of SVD line to compute ($x_D$, $y_D$) approximation based on this line
4.  Calibrate to determine the pointwise mean of predicted values
```

Now almost all observed points are closer to hybrid line than OLA and NLA/SVD approximation lines. It satisfies the general parametric and space conservation requirements, see Table1.

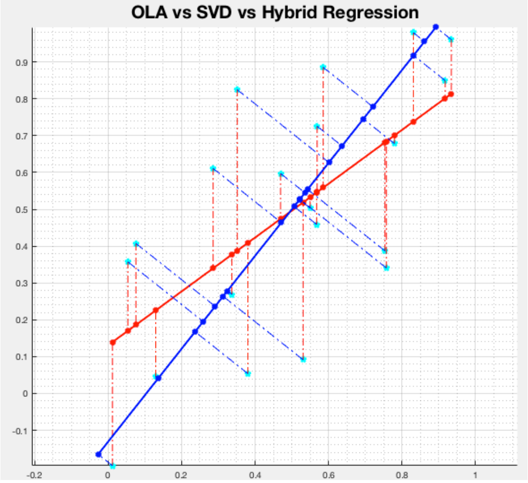


Figure 10. Cyan dots are data points, Red line is OLA line , Blue line is NLA/SVD line, Green dots are hybrid approximation dots

Note over the entire data set, *red dots have smallest error* from cyan dots when distances are measured along y, while *blue dots have smallest error* from cyan dots when distances are measured along the normal to the line. Each green dot is at a smaller of the two distances from cyan dot, interestingly, it *does not mean* that green dots have *overall* smaller error than the two, in fact it will be bigger than each. The green dots can be connected by a polygonal line see Figure11 or an SVD straight line approximation. We have seen that NLA is better than OLA. We may use SVD to approximate data (x_H, y_H) to a line, see Figure12.

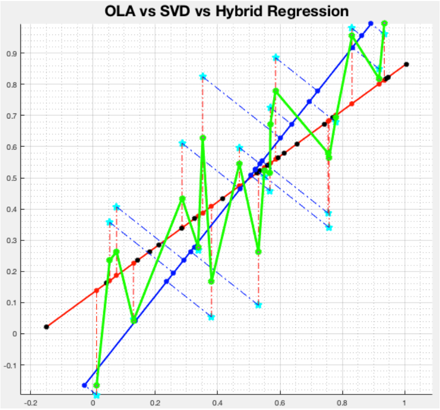


Figure 11. Non-Parametric polygonal Hybrid data points, Cyan dots are points which are closer to green dots than red or blue dots. Hybrid polygonal line, green polygonal line connects the green hybrid points (x_H, y_H).

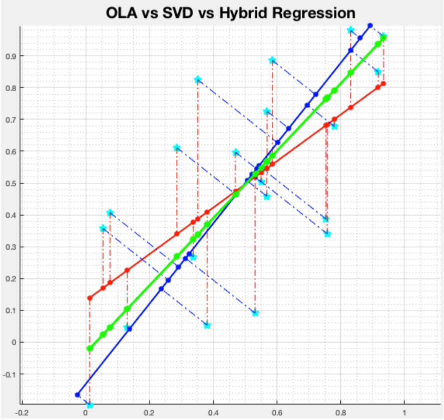


Figure 12. Parametric line Green dots in Figure 11 are not shown here for clarity. SVD line is created to corresponding green points into the green Hybrid parametric line.

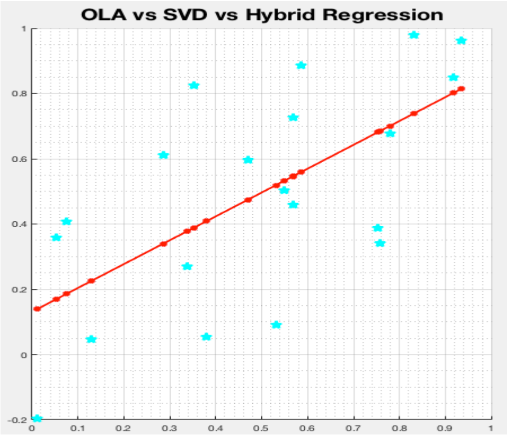
B. Precision and Propensity

The linear least square approximation error is *quantitative* measure. The precision and propensity are a *qualitative* measure of accuracy [22],[23],[24]. Quantitative error is a function of the location of data points, propensity depends on percentage of data points for pointwise binary outcome from comparing error due to a pair of methods. This is similar to precision metric used in data mining community. For percentage of data truly closer to OLA, SVD, Hybrid lines pairwise, see Table 1 and Table 2. From Figure 11, it is clear that green construction is preferable, but the quantitative error comparison is inconclusive. However, we use propensity metric to determine the level of accuracy that hybrid line has as compared to OLA and SVD. When errors are measured in the respective methods, we can calculate the propensity value for one line relative to the other line to conclude the preference irrespective of which method is used to calculate errors. It is determined that overall SVD/NLA approximation is better approximation than OLA, see Figure 11.Similarly,propensity metric shows, that hybrid line is preferable to both OLA and SVD lines, see Figure 12, Table 1. Table 2.

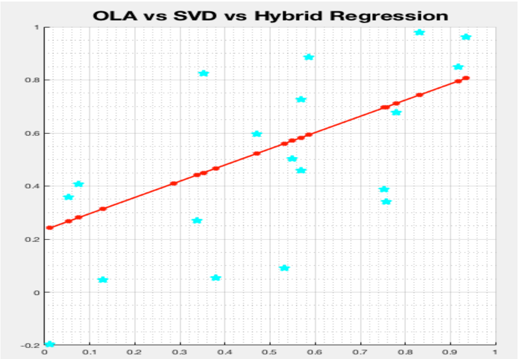
C. Anomaly Detection and Removal

It is clear that vertical distance, E_R, is always greater than normal distance, E_S, from a line. Since sum of squares of errors for OLA line, E_R is smallest in the vertical distance metric, the regression error from any other line is bound to be larger than error, E_R, from OLA line. Pointwise error between OLA and NLA is not deterministic, Propensity score metric(PSM) is a qualitative measure to differentiate for better approximation line, where the distance metric fails. Not only that, PSM can also leveraged identify the anomalies. To detect anomalies accurately, we create a confusion matrix for number of points within one standard deviation of both the lines. Any point which is not within this band about any of the two lines, is probably an anomaly. Such is point is candidate for further scrutiny. After clipping suspicious points for the data, we reapplied our algorithm to ascertain that reduced data set gives better accuracy, see Table 1, and Table 2.

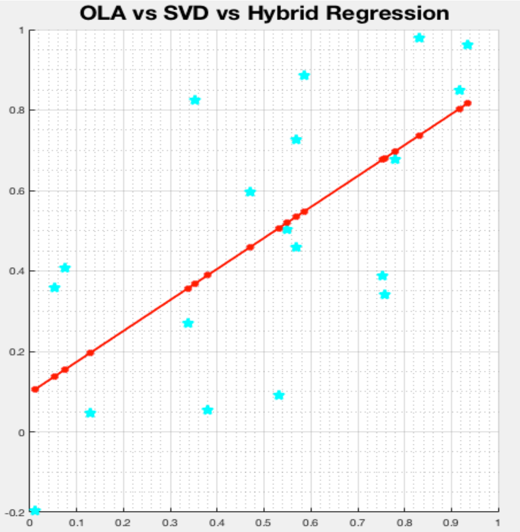
Example: Noisy data, vertical distances not realistic. In the Figure 13(b), we can see that if fifth point is noisy, it has affected the entire approximation line. In particular for the neighboring points, there is glaring offset. Experiments show that one noise point can adversely affect the approximation line in the immediate neighborhood of noisy point, see Figure 13. Red line is least square regression line on raw data of 20 points. This regression line is noise sensitive, see Figure 13(a), (b). If one of data points is an outlier, it can create a large adverse effect on the outcome. Figure 13(c) shows the improvement on this shortcoming after removing noise.



(a) No noise



(b) Noise introduced in position 5, direction of line changes



(c) Noise removal, position 5 removed from the data, data has one less point.

Figure 13. (a) has no noise, (b) has noise in position 5, as a result the regression lines are different, (c) here noise is removed, now (a) and (c) are same , but (c) has one less points as point 5 has been removed. We do not see any major difference in the regression lines.

The goal is to determine the prediction capability rather than numeric value. The hybrid algorithm achieves a balance between quantitative and qualitative approximation accuracy of both OLA and NLA/SVD. Numeric error is a measure of divergence from the true value. We use STD-standard deviation for confidence interval about the approximation lines. If A is the set of points outside the confidence interval and B is the set of points where $e_R > e_S$, the AB is a candidate set of anomalies.

Table 1 Comparison of Algorithms

	OLA	SVD	Hybrid
Approximation Line Directi	[0.81, 0.59]	[0.62, 0.78]	[0.68, 0.73]
Approximation Error	7.06%	5.09%	3.32%
Confidence in one std Interv	75.00%	100.00%	100.00%
closeness OLA vs SVD	25.00%	75.00%	
closeness OLA vs Hybrid	0.00%		100.00%
closeness SVD vs Hybrid		15.00%	85.00%

Table 2 Comparison of Algorithms 5% Noise Removal

	OLA	SVD	Hybrid
Approximation Line Directi	[0.79, 0.62]	[0.66, 0.75]	[0.68, 0.73]
Approximation Error	6.46%	4.71%	3.32%
Confidence in one std Interv	88.89%	100.00%	100.00%
closeness OLA vs SVD	0.00%	100.00%	
closeness OLA vs Hybrid	0.00%		100.00%
closeness SVD vs Hybrid		16.70%	83.30%

D. Temporal Sensitivity

If the time interval for a treatment is changed, we expect to see the temporal change in response. Using OLA, we see that there is no change, that is error computation remains unchanged, see Figures 10-13. Figure 14 is the visual summary of quantitative and qualitative error in the methods. Using the same data set, on scaling the time interval, the NLA/SVD and Hybrid algorithms respond positively to the changes. This suggests that OLA is not suitable for such applications. In the example we also notice that as the slope of the hybrid line increase, the error decreases. Experiments confirm that slop of 45 degrees is break-even point with maximum error. Slope below or above accounts for reduction in error. For comparison of the three algorithms, see Table 1, Table 2. It shows the computed direction vectors of the approximation lines, approximation error in the Euclidean distance metric, and propensity value, how close is data to one formulation vs the other formulation.

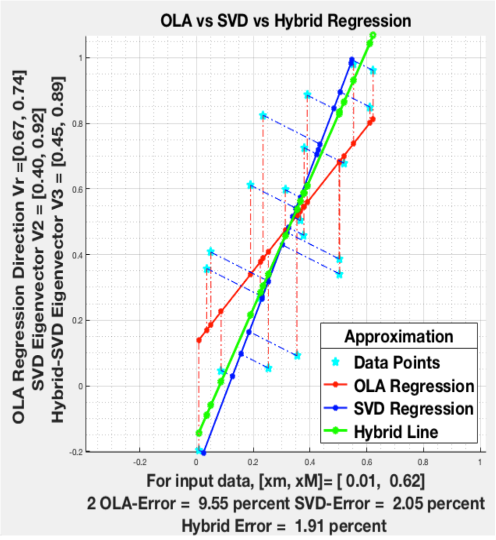


Figure 14. Relative errors one-time interval [0.01,0.62]

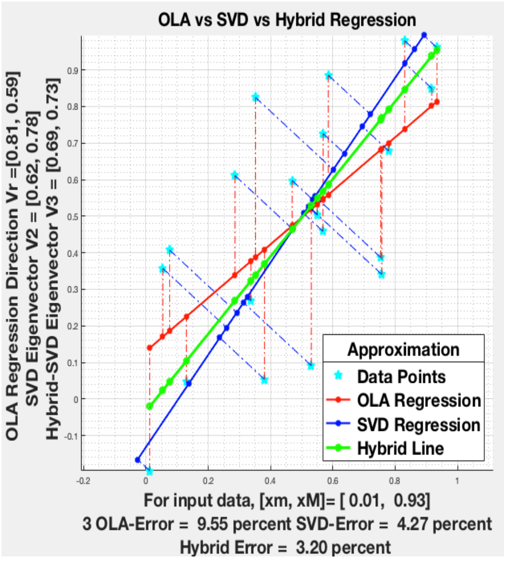


Figure 15. Relative errors on time interval [0.01,0.93]

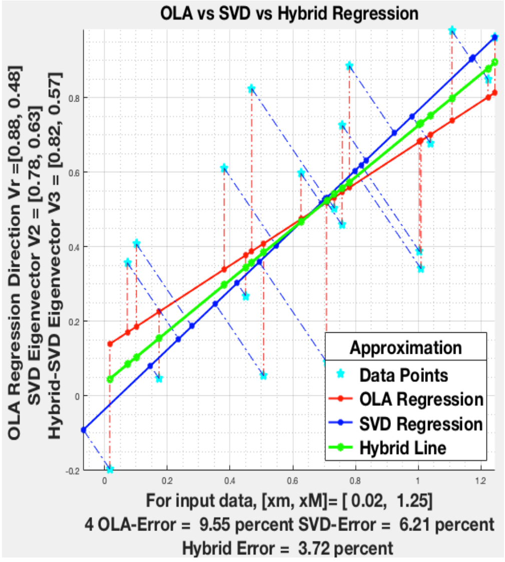


Figure 16. Relative errors on time interval [0.02,1.25]

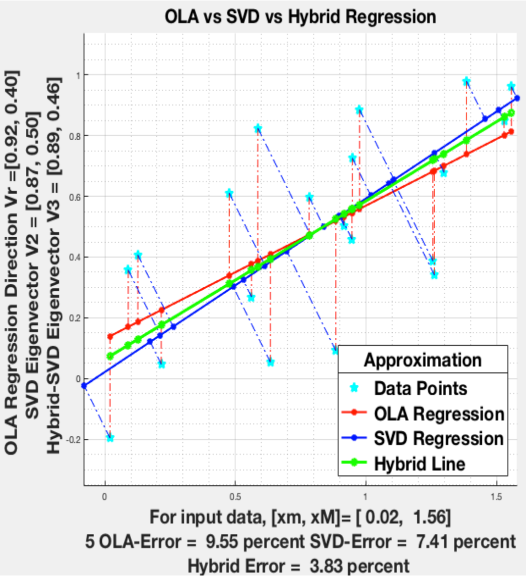


Figure 17. Relative errors on time interval [0.02,1.56]

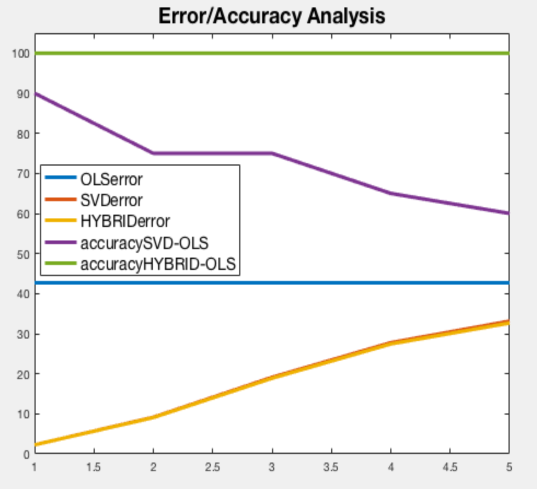


Figure 18. Green line shows percentage of Hybrid points closer to data points as compared to OLA. Purple line shows percentage of SVD points closer to data points as compared to OLA. Blue line shows percentage of error in OLA. Yellow and red (on top of each other) percentage of error in SVD and Hybrid algorithms.

VI. Conclusion

In the paper we have described 1. various ways to approximate numerical data, 2. Temporal versions of prediction, 3. how to reduce noise. For approximation, the ordinary linear least square approximation (OLA) regression is suitable for continuous real data, normal linear least square approximation (NLA), Singular Value Decomposition (SVD) may be used for continuous data for best approximation, and for compression. Here we used OLA and NLA/SVD first to compare and remove noise by virtually using OLA and NLA. The hybrid data is then approximated by using NLA/SVD. It is determined that hybrid algorithm outperforms the two algorithms when applied individually. The statistician in the area will benefit from the hybrid linear least approximation algorithm.

OLA was found to be insensitive to data spread, whereas SVD was implicitly modifying the independent (temporal) variable of the original input in pursuit of lower error. We designed a hybrid algorithm that overcomes the shortcomings and supersedes the accuracy of existing algorithms. From the experiments, it follows that error is least for lines that are almost horizontal or vertical, the breakeven point occurs as the slope of the line becomes closer to 45 degrees. NO matter what the slope is, the new hybrid regression line error is always bounded above by the error in OLS regression line. It is interesting to note that OLA remains unchanged while new regression line approximation error responds to the slope variation. We also showed how to improve MATLAB SVD with correct directions of eigenvectors, a natural technique. We designed and implemented a hybrid algorithm that supersedes both accuracy and efficacy. The algorithm was implemented on MAC OS Seirra v 10.13.4, Intel Cire i5, 8GB 1600MHZ using Matlab R1700b.

References

[1]. Steven C Chapra and Raymond P Canale, Numerical Methods for Engineers, 7th Edition, ISBN: 978 0073397924 , McGraw-Hill Publishers, 2015.

[2]. Cohen, J., Cohen P., West, S.G., & Aiken, L.S. (2003). Applied multiple regression/correlation analysis for the behavioral sciences. (2nd ed.) Hillsdale, NJ: Lawrence Erlbaum Associates.

[3]. Draper, N.R.; Smith, H. (1998). Applied Regression Analysis (3rd ed.). John Wiley. ISBN 0-471-17082-8.

[4]. Gwowen Shieh, Clarifying the role of mean centering in multicollinearity of interaction effects,British Journal of Mathematical and Statistical Psychology (2011), 64, 462–477.

[5]. Jim Hefferon, Linear Algebra, Free Book, http://joshua.smcvt.edu/linearalgebra, 2014.

[6]. John F. Hughes, AndriesVanDam,Morgan McGuire, David F. Sklar, James D. Foley, Steven K. Feiner, Kurt Akler Computer Graphics: principle and Practice, 3rd edition , Addison Wesley, 2014.

[7]. Matlab, https://www.mathworks.com/downloads/

[8]. P. Groves, B. Kayyali, D. Knott, S. V. Kuiken, “The ‘Big Data’ Revolution in Healthcare”, *Center of US Health System Reform Business Technology Office,* pp. 1-20, 2013.

[9]. C. C. Yang, L. Jiang, H. Yang, M. Zhang, “Social Media Mining for Drug Safety Signal Detection" *ACM SHB’12*, October 29, 2012, Maui, Hawaii, USA.

[10]. Jure Leskovec, Anand Rajaraman, Jeffrey D Ullman, Datamining of Massive Datasets, 2014.

[11]. Patrick J.F. Groenen, Michel van de Velden, Multidimensional Scaling, Econometric Institute EI 2004-I5, Erasmus University Rotterdam, Netherlands, 2015.

[12]. Chaman Sabharwal, Principal Component Analysis and Qualitative Spatial Reasoning, 28th International Conference on Computer Applications in Industry and Engineering, CAINE 2015, October 12-14, 2015, San Diego, California, USA pp.23-28.

[13]. Sebastian Raschka Principal Component Analysis in 3 Simple Steps LSA-Least Squares Approximation http://sebastianraschka.com/Articles/2015_pca_in_3_s teps.html, 2015.

[14]. JonthanShlens A Tutorial on Principal Component Analysis, arXiv:1404.1100 [cs.LG], pp. 1-15,2014[Stephen] Stephen Vaisey, Treatment Effects Analysis,https://statisticalhorizons.com/seminars/public-seminars/treatment-effects-analysis-spring17.

[15]. Abdi, Hervé, Beaton, Derek, Principal Component and Correspondence Analyses Using R, Springer, ISBN 978-3-319-09256-0, Digitally watermarked, DRM-free, 2017.

[16]. Caroline J Anderson, Psychology Lecture Notes: Principal Component Analysis, 2017.

[17]. K.Baker,SingularValueDecompositionTutorial https://www.ling.ohio-state.edu/~kbaker/pubs/Singular_Value_Decomposition_Tutorial.pdf , January 2013.

[18]. H. Y. Chen, R. LiÅLegeois, J. R. de Bruyn,and A. Soddu, “Principal Component Analysis of Particle Motion”, *Phys. Rev.* E 91, 042308 - 15 April 2015.

[19]. Karen Bandeen-Roche Nov 28, 2007, An Introduction to Latent variable Models, http://www.biostat.jhsph.edu/~kbroche/Aging/Intro to Latent VariableModels.pdf.

[20]. Yusuke Ariyoshi and JunzoKamahara. 2010. A hybrid recommendation methodwith double SVD reduction. In International Conference on Database Systems forAdvanced Applications. Springer, 365–373.

[21]. Chaman Sabharwal, Hybrid Linear Least Square and Singular Value Decomposition Approximation, International Journal of Trend in Research and Development, Volume 5(3), ISSN: 2394-9333 www.ijtrd.com May-Jun 2018, pp. 1-8.

[22]. Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. Computer 42, 8 (2009).

[23]. Mark Tygert Regression-aware decompositions, arXiv1710.04238v2, 12 Feb 2018.

[24]. Stephen Vaisey, Treatment Effects Analysis,https://statisticalhorizons.com/seminars/public-seminars/treatment-effects-analysis-spring17.

IOSR Journal of Computer Engineering (IOSR-JCE) is UGC approved Journal with Sl. No. 5019, Journal no. 49102.

Chaman Lal Sabharwal, "SVD Adaptive Algorithm For Linear Least Square Regression and Anomaly Reduction." IOSR Journal of Computer Engineering (IOSR-JCE) 20.5 (2018): 33-48.