Qualitative Analysis of Least Square Regression

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

ABSTRACT

In statistical analysis, the accuracy of approximation is a function of (1) data representation, (2) approximation technique and (3) the metric used for error measurement. For ordinary linear least square approximation (OLA), the existing formulation of error measurement is not satisfactory for many applications. 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 such as polynomial regression for engineering, exponential regression for radioactive decays, bilinear regression for saturated growth, logistic regression for medical prognosis etc. They are all reformulations of OLA using prior domain knowledge, for supervised learning. Singular Value Decomposition (SVD) is also used for ranking, prediction and recommendation systems. 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 supersedes the approximation accuracy and performance of both OLA and SVD. 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 with one independent variable and one dependent variables to demonstrate the hybrid algorithm.

Date of Submission: 27-07-2018 Date of acceptance: 11-08-2018

I. INTRODUCTION

In statistical analysis, the accuracy of approximation is a function of 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. We will show the effect of new metric as compared to OLA and NLA metrics. We show that hybrid algorithm is better in terms of both error metrics. 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. This is also the intent of this paper.

The paper is organized as Section2 describe OLA and an efficient computation by mean-centering data formulation, Section 3 derives new NLA, Section 4 describes SVD and it connection to NLA, Section 5 gives new hybrid approximation algorithm and its implementation, error analysis of OLA,NLA, SVD, and Hybrid algorithms is provided with respect to both metrics , Section 6 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 the 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 to make it the reference point. We will soon see how mean-centering simplifies the computations as well.

Let the data be represented by an m×n matrix A. To mean-center the matrix, if x is column of A, it is translated to x - $\bar{x}$ and if y is row of A, it is replaced with y - $\bar{y}$, where the mean of **x** and **y** is defined as $\bar{x} = \frac{\sum_i x_i}{m}$, and $\bar{y} = \frac{\sum_i y_i}{n}$. For matrix operations most of the linear transformations are performed by means of matrix multiplication, centralization is a linear transformation [4]. There is a clean transformation C_m to mean-center the columns of A as follows. Let I_m be mxm identity matrix, $\mathbf{e_m}$ be a column vector of m ones, and $C_m = I_m - \mathbf{e_m e_m}^T/m$. This C_m is called the column centralizer. Let us see how C_mA centralizes the columns of A. For example, if **x** is a column vector then

$$\begin{aligned} C_m\mathbf{x} &= I_m\mathbf{x} - \mathbf{e_m e_m}^T\mathbf{x}/m \\ &= \mathbf{x} - \mathbf{e_m e_m} \bullet \mathbf{x}/m \\ &= \mathbf{x} - \bar{x}e_m \end{aligned}$$

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

$$\begin{aligned} \mathbf{y}C_n &= (\mathbf{y}\, I_n - \mathbf{y e_n e_n}^T/n) \\ &= \mathbf{y} - \mathbf{y} \bullet \mathbf{e_n e_n}^{\mathbf{T}} /n \\ &= \mathbf{y} - \bar{y}\mathbf{e_n}^T, \end{aligned}$$

mean-centers the row vector **y**.After performing analysis on mean-centered data, reference point can be translated back. This is a standard technique used for visualization [5],[6].

2.1. Ordinary Linear Regression

2.1.1 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 doing any approximation, it is assumed that data is accurate, else prediction will also be inaccurate. First for 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

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

and

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

Let $\bar{x} = \frac{\sum_i x_i}{n}$, $\bar{y} = \frac{\sum_i y_i}{n}$, $\overline{xy} = \frac{\sum_i x_i y_i}{n}$, $\overline{x^2} = \frac{\sum_i {x_i}^2}{n}$, the first equation (1) becomes $\bar{y} = a + b\bar{x}$ which implies that the regression line passes through the centroid ($\bar{x}$,$\bar{y}$). The second equation (2) implies that $\overline{xy} = a\bar{x} + b\overline{x^2}$. These two equations

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

can be solve 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.

2.2 For Mean-Centered 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

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

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

$$\equiv \overline{y^2}-2b\overline{xy}+b^2\overline{x^2}$$

That is

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

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

which simpler expression than the raw data computations.
However, if we want to go to the original frame, original reference point, we translate the origin to the centroid then line becomes

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

that is

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

In this case only b is computed, a is automatic.

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

In essence,this is a common sense three step approach. The three steps are, (1) mean-center the data, translate the centroid $(\bar{x},\bar{y})$ to the origin (0,0), (2) find 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. Figure 1 shows that raw data regression line is identical to mean-centered data line after it is translated by the centroid. In Figure 1. Green line is least square regression line on raw data of 20 points, the blue line represents OLA regression line on mean-centered data. Except for computer arithmetic, the two lines are identical. The computations using mean-centered data are simpler.

This regression line is noise sensitive. If one of data points is an outlier, it can create a large adverse effect on the outcome. See the following example Figure 2. Later we will see how to improve on this shortcoming.

Example: Noisy data, vertical distances no not realistic. In the Figure 2, 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.

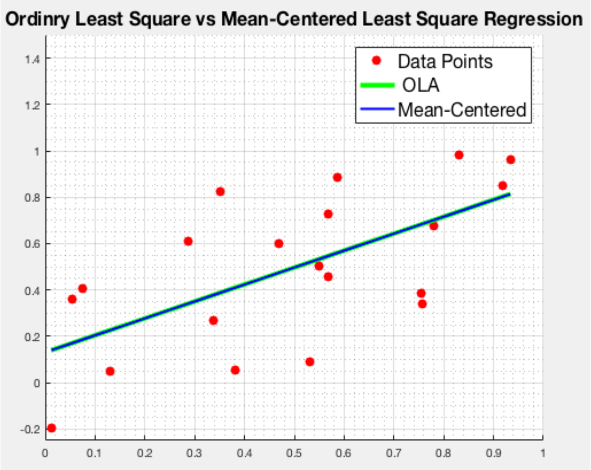


Figure1. Data points and regression line

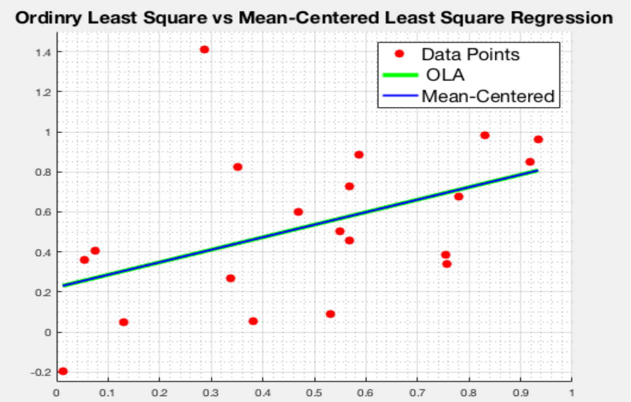


Figure2. Regression line after noise perturbed point5

III. NORMAL LINEAR LEAST SQUARE APPROXIMATION (NLA)

The 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 centralizer transformation can be used to mean-center it. The problem becomes that of finding the value of *only b* that minimizes f(b) where

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

$$f(b) = 1/n \sum_{i=1,n} \frac{({y_i}^2 + b^2 {x_i}^2 - 2bx_iy_i)}{1+b^2} = \frac{\overline{y^2} + b^2\overline{x^2} - 2b\overline{xy}}{1+b^2}$$

Thus, for local minima of $f(b) = \frac{\overline{y^2} + b^2\overline{x^2} - 2b\overline{xy}}{1+b^2}$ (1)

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

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

Since it is a quadratic, it has two local solutions, b_1, b_2

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

If f''(b_1)>0, the b_1 is a local minima. In case f''(b_1)>0 and f''(b_2)>0 , we have two local minima, compare f(b_1) an f(b_2) whichever is smaller that is the value of b we use for 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 ordinary least square line and normal least square line are shown in Figure 3.

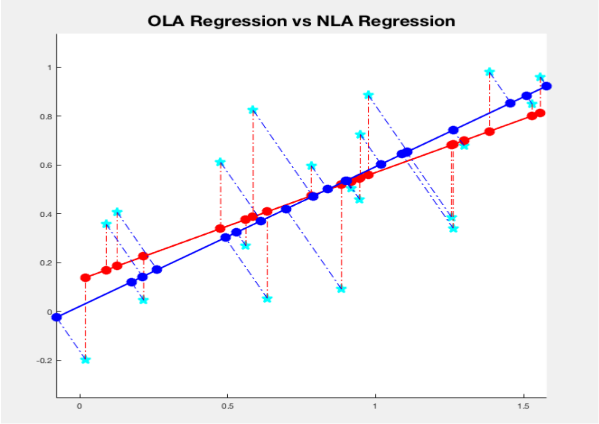


Figure 3. Cyan dots are the data points, red line is OLA line and blue line is NLA line.Red dots and blue dots are the predictions by the respective methods (OLA,NLA).

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 Figure 3. This visualization justifies, to some extent, to prefer NLA over OLA. We will give formal justification later. Since NLA is based on calculus, it is complex due to derivatives, 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 singular value decomposition (SVD). Today, singular value decomposition is used in many branches of science, in particular 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 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, *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 $A^{T}A$ is a symmetric square positive semi-definite matrix, then $A^{T}A = VDV^{T}$, [16],[17],[18]. Besides other benefits of this factorization, we are interested in *direction vector* only. The columns of V are eigenvectors of $A^{T}A$ corresponding to eigenvalues arranged in descending order. Since we are interested in direction of approximation line, it is first proved 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. Since data is mean-centered the line passes through the origin. As a standard, vectors **P** are column vectors, thus rows of A are row vectors, $\mathbf{P}^{\mathbf{T}}$. 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 $\mathbf{P} = \mathbf{P}\bullet\mathbf{v}\ \mathbf{v} + (\mathbf{P}-\mathbf{P}\bullet\mathbf{v}\ \mathbf{v}) = x\mathbf{v}+y\mathbf{w}$**.** This means that minimizing the distance y amounts to maximizing the component x. We are to maximize over all data points $\mathbf{P}_{i}$. The problem becomes that of maximizing

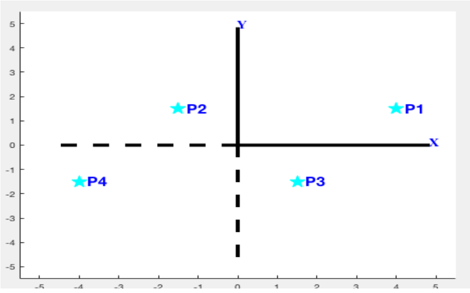
$$\sum_{i} |\mathbf{P}i\bullet\mathbf{v}|^{2}$$

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

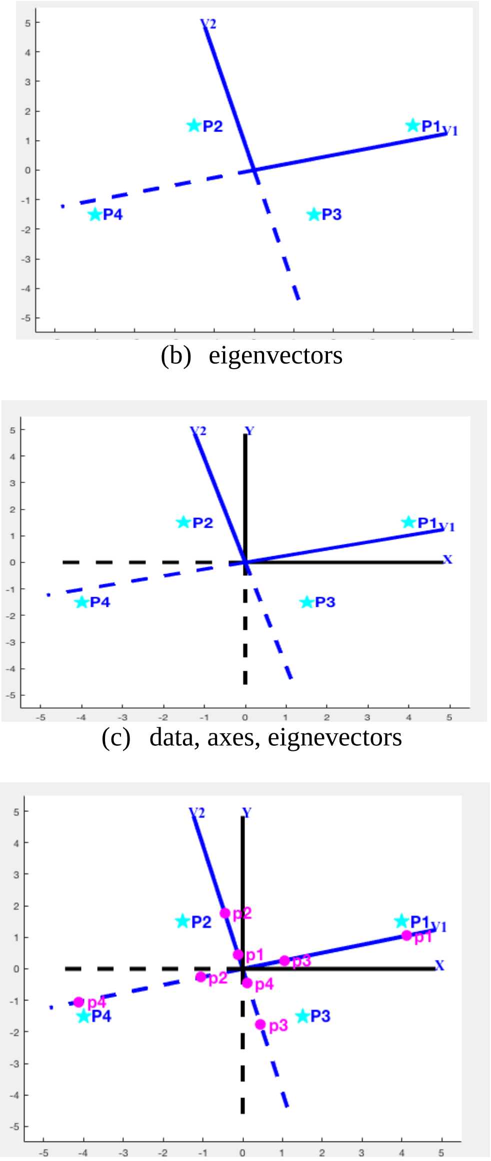
$$\begin{aligned}\sum_{i} |\mathbf{P}i\bullet\mathbf{v}|^{2} &= \sum_{i}\mathbf{P}i\bullet\mathbf{v}\mathbf{P}i\bullet\mathbf{v}= \sum_{i}\mathbf{v}\bullet\mathbf{P}i\mathbf{P}i\bullet\mathbf{v}\\ &= \sum_{i}\mathbf{v}^{\mathbf{T}}\mathbf{P}i\mathbf{P}i^{\mathbf{T}}\mathbf{v}= \mathbf{v}^{\mathbf{T}} (\sum_{i}\mathbf{P}i\mathbf{P}i^{\mathbf{T}})\mathbf{v}\\ &= \mathbf{v}^{\mathbf{T}} (A^{\mathbf{T}}A)\mathbf{v}.\end{aligned}$$

This means that $\sum_{i} |\mathbf{P}i\bullet\mathbf{v}|^{2}$ is maximum if **v** is an eigenvector of $A^{T}A$ and corresponds to largest eigenvalue of $A^{T}A$. 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. Along the direction $\mathbf{v_1}$ the spread of projections of data on $\mathbf{v_1}$ is larger than that for $\mathbf{v_2}$, see Figure 4(d).

For example, **P**’s are data points in 2D, $\mathbf{v_1}$**,** $\mathbf{v_2}$ are eigenvectors corresponding to largest eigenvalues. The NLA requires only $\mathbf{v}_{1}$, the direction with largest eigenvalue, and with largest data spread.



(a)   data points,



(b) eigenvectors

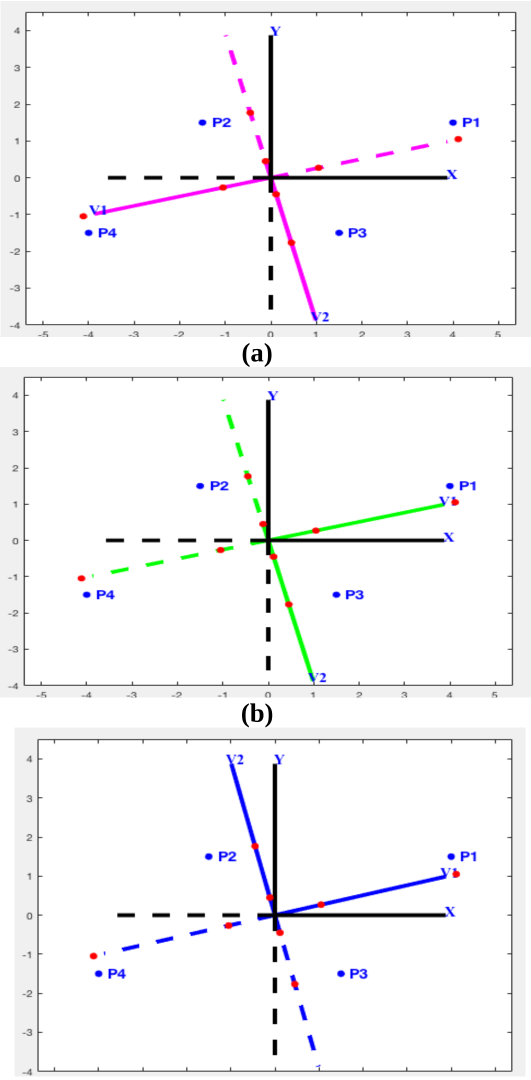
(c) data, axes, eignevectors

(d) everything with

projections on eigenvectors.

Figure 4. (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, (3) data points, standard xy-axes, eigenvectors frame, (4) both xy and $\mathbf{v}_1\mathbf{v}_2$, frames with data points and projections on $\mathbf{v}_1$ and $\mathbf{v}_2$.

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 5(a) for Matlabsvd 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 5(b). Since eigenvectors are ordered, we use ordering to make the k-th element of k-th vector to be positive, see Figure 5(c) that makes the vectors look more natural like a right handed system. In case, the kth elements 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 5. (a) Eigenvectors as computed by Matlabsvd, (b) each vector has first no element positive, (c) first eigenvector has first component positive, second eigenvector has second component positive, so the eigenvectors form a right handed system.

V. HYBRID ALGORITHM DESIGN

We design a hybrid algorithm taking the best of OLA and NLA/SVD approximation lines. For each observed point, (x_0,y_0), we have seen in Figure 6 that there is a corresponding estimated point (x_R, y_R) on regression line and an estimated point (x_S, y_S) on SVD line. For hybrid algorithm, define the approximation point (x_H, y_H) to be that point which is both ways closer to the observed point (x_0,y_0).If (x_0,y_0) is an observed value, $(x_H,y_H) = (x_R, y_R)$ is estimated value corresponding to the OLA line y=a+bx. The vertical distance is along y direction, $x_H= x_0$. The distance between (x_0,y_0) and (x_H,y_H) is the y-distance, the OLA regression distance $d_R= |y_0-y_R|$. For normal distance from NLA or SVD approximation line, it is along perpendicular to the line, it turns out that $(x_H,y_H)= (x_S, y_S)$ implying $x_H\neq x_0$, the distance between (x_0,y_0) and (x_H,y_H) is Euclidian normal distance $d_S = \sqrt{(} (x_0-x_S)^2 + (y_0-y_S)^2$).It is clear from Figure 6 that for some points in observed data, $d_R<d_S$ while for some points $d_S<d_R$. However, green dot are closer to cyan dots than corresponding dots on red line or blue line. In each method, the total error E is sum of squares of distances (errors) for all data points, question arises which one (E_R for OLA and E_S for SVD) is better. 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? For each input we will determine approximate line that represents the input data no matter how the error is computed,see Figure 6 for 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 aqualitative metric that is more useful in visualization.

5.1. Hybrid algorithm

Input: array of x and y mean-centered data values

Output: hybrid approximation line points (x_H,y_H),where(x_Ry_R) is on OLA, (x_Sy_S) is on SVD line

Algorithm:

```
1.  Calculate a and b for OLA regression for observed values x,y
    Calculate predicted values by linear regression $y_R = a+bx$
    Calculate  approximation error $E_R$
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 columns 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 points$(x_R, y_R),(x_S, y_S)$
        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 $E_H$
Compare error $E_S$, $E_R$, $E_H$
Double approximation, using estimated data point find an SVD line $(x_H, y_H)$
Calculate and Compare by propensity values
4.  $x_H$, $y_H$ are arrays of predicted coordinates on hybrid line.
```

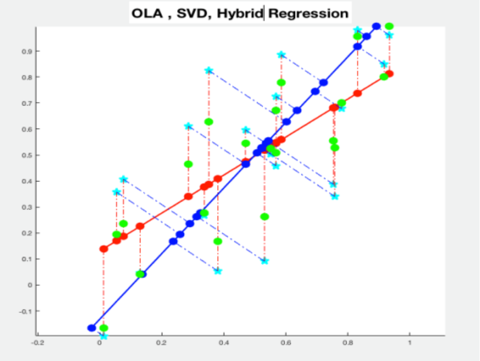


Figure 6. Cyan dots is data points, Red line is OLA line , Blue line is NLA/SVD line, Green dots are hybrid approximation dots

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 Figure 7 or a 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 Figure 8.

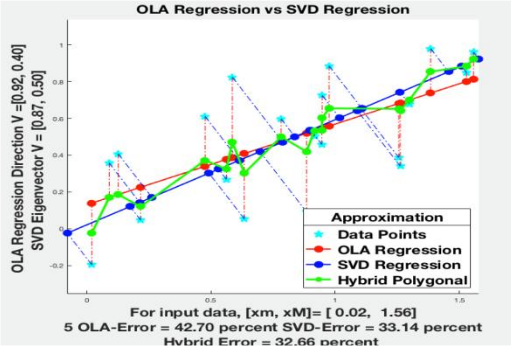


Figure 7. Hybrid points connected by a polygonal line.

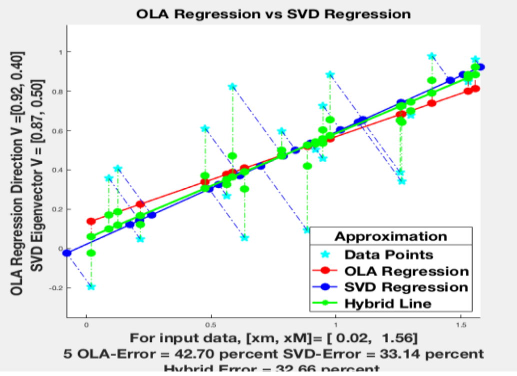


Figure 8. Hybrid points approximated by SVD line.

5.1. Precision and Propensity

The linear least square approximation error is *quantitative* measure. The precision and propensity is a *qualitative* measure of accuracy[22],[23],[24]. Quantitative error is a function of the location of data points, propensity depends on count 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. Percentage of data truly more close to OLA, SVD, Hybrid lines pairwise. Figure 8, 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 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 9. Similarly propensity metric shows, that hybrid line is preferable to OLA and SVD lines.

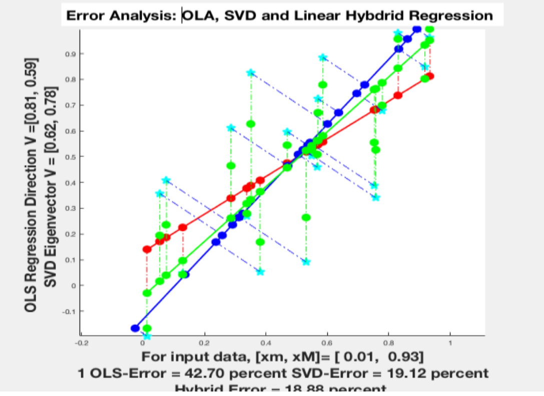


Figure 9. The data points, OLA,SVD, and Hybrid approximation lines

5.3. 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 if brake even point with maximum error. Slope below or above accounts for reduction in error. For comparison of the three algorithms, see Table 1. It shows the computed direction vectors of the approximation lines, approximation error in the Euclidean distance metric, and propensity how close is data to one formulation vs the other formulation.

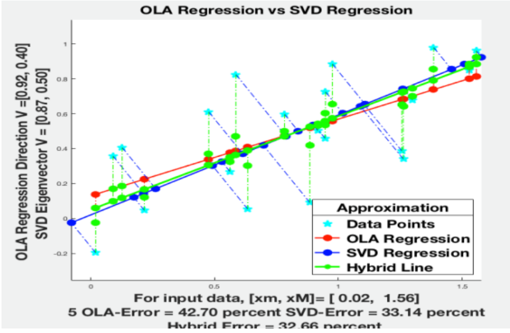


Figure 10 Relative errors one time interval [0.01,1.56]

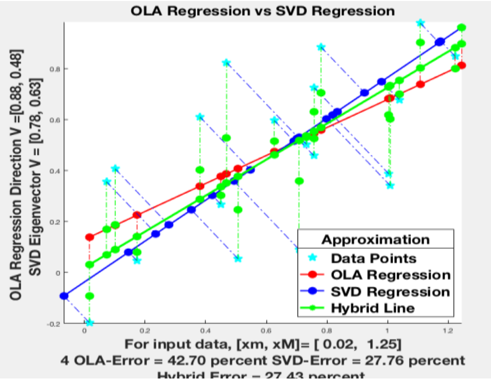


Figure 11 Relative errors one time interval [0.01,1.25]

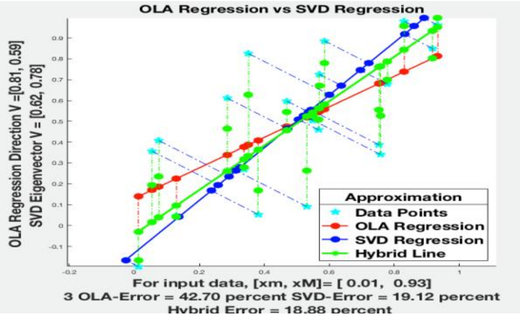


Figure 12 Relative errors one time interval [0.01,0.93]

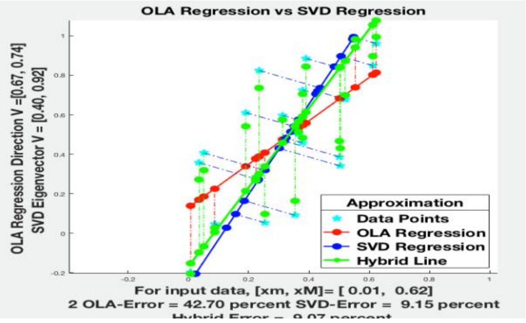


Figure 13 Relative errors one time interval [0.01,0.62]

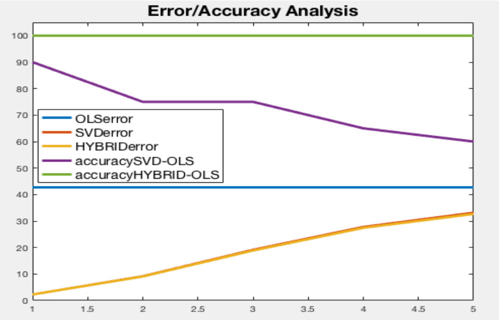


Figure 14. Green line shows percentage of Hybrid points closer to data points as compared to OLA.

Purple lineshows 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.

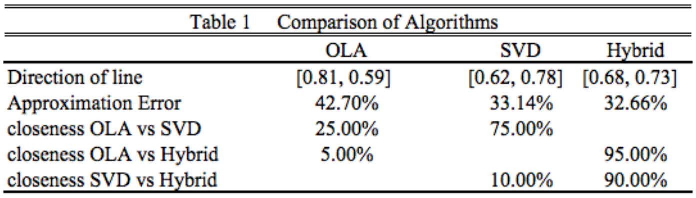


Table 1 Comparison of Algorithms

	OLA	SVD	Hybrid
Direction of line	[0.81, 0.59]	[0.62, 0.78]	[0.68, 0.73]
Approximation Error	42.70%	33.14%	32.66%
closeness OLA vs SVD	25.00%	75.00%	
closeness OLA vs Hybrid	5.00%		95.00%
closeness SVD vs Hybrid		10.00%	90.00%

VI. CONCLUSION

For approximation, the ordinary linear least square approximation (OLA) regression is suitable for continuous real data, classification is used for discrete data, normal linear least square approximation (NLA), SVD may be used for discrete and continuous data best approximation, and for compression. Here we used OLA and NLA first to compare and remove noise by virtually using OLA and NLA. The hybrid data is then approximated by using NLA. It is determined that hybrid algorithm outperforms the two algorithms when applied individually. The statistician in this area will benefit from the hybrid linear least square 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 of 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 Matlabsvd with correct directions of eigenvectors, a natural technique. We designed and implemented a hybrid algorithm that supersedes both accuracy and efficacy. The algorithm was implement on MAC OS Seirra v 10.13.4, IntelCire 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, Singular Value Decomposition Tutorial 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

Chaman Lal Sabharwal “Qualitative Analysis of Least Square Regression” International Journal of Computational Engineering Research (IJCER), vol. 08, no. 08, 2018, pp. 39-50.