International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:2682
Vol.4, No.8, August 2015 DOI:10.15693/ijaist/2015.v4i8.5-8

Quadratic regression for complex software predictions

DVSS.SUBRAHMANYAM

Professor, Dept. of CSE, Malla Reddy Engineering. College (Autonomous)
Dhulapally, Maisammaguda, Secunderabad – 500 100 Telangana India

ABSTACT: **Many times it has become very difficult to make software predictions for the given data. Regression analysis provides many tools for software predictions. Its predictions, in most of the cases, are more than 95% accurate. One of the important tools is quadratic regression which can be used to design many experiments in complex situations. Even though quadratic regression is rarely used but it plays very important role to resolve many tropical problems. Quadratic regression belongs to the family of non-linear regression and it dominates all other types of non-linear regression techniques. In most of the cases quadratic regression resolved many ambiguous conditions where designing and predictions of systems was very difficult. Quadratic regression has become an alternative tool in cases where no possible solution is found.**

Key words: **linear regression, non-linear regression, quadratic regression, inverse of the matrix**

I. INTRODUCTION

Mainly software risks can be classified as scheduling risks, cost risks, requirements risks, quality risks and business risks. Entire software business risks are based on cost risks and quality risks as quality risks always have to be met with users needs and requirements from time to time[3][4]. There is a direct proportionality between quality risks and cost risks of any software product. Software products' viability is purely based on their quality. This is the reason why software industries spend more money for quality of products. Indian software companies mainly depend upon foreign companies for their business. Software companies always undergone many fluctuations as they are being influenced by the conditions that happen globally. Software quality always helps companies to protect from global affects by meeting software global requirements, conditions and standards from time to time. The factor of software quality risks decides the overall future of software industries[3][4][6]. Software companies have to consider many components internally and externally to meet software quality of products. Software companies are needed to have an eagle eye on the overall relation and structure of components between cost risks and quality risks. Companies are needed to have a proper system design for their products with respect to software costs and software quality of products. Linear regression[2] techniques helped a lot in designing many models for software markets. But the same techniques may not be suitable to meet all the existing conditions which may not be frequent. Non-linear regression techniques[2] enter the situations where linear-regression techniques fail to meet the requirements of the data models. Even though rarely used but non-linear techniques play a prominent role in designing and modeling many complex market conditions. Once software companies are able to design their system models then they will be in a position to predict future outcomes of their products *a priori*[1]. This .tool is very essential for companies for estimating their strategies of past and future. Many statistical software are existing in the market to evaluate the software system models but a very easy and inexpensive tool can be used to evaluate the entire system needs and requirements. Some times it may not be possible to evaluate system designs for the existing data. Quadratic regression provides this unique feature to design complex issues in order to overcome many deficiencies[3][4][5].

Two medium level software companies are taken into consideration to model the relation in terms of dependency between cost risks and quality risks of their software products. This model will help these companies to predict their software business with an approximate expectations. The key role of quadratic regression is discussed here where normal predictions cannot be done by linear regression methods. Any non-linear regression can be expressed as[2]

$P = C_0 + C_1 R_{i1} + C_2 R_{i2}^{\ 2} + C_3 R_{i3}^{\ 3} +$ ---------- $+ C_n R_{in}^{\ n} + \epsilon$ ....(1) where i = 1,2,3,……..n and ϵ is error coefficient and here it is assumed to be very negligible. R is independent variable and P is dependent variable. C_0, C_1, C_2, C_3, C_4, ……….. and C_n all are Regression coefficients. If the degree of (1) is 2 then it is quadratic regression ,if degree is 3 then it is cubic regression etc., But quadratic regression method is more reliable than cubic, biquadratic methods etc., as quadratic regression gives better prediction than other alternatives[1][2][5]. Thus equation (1) can be considered as

$$P = C_0 + C_1 R_{i1} + C_2 R_{i2}^{\,2} + C_3 R_{i3}^{\,3} + \text{----------} + C_n R_{in} \qquad \ldots\ldots(2)$$

Here equation (2) can be solved by using Matrix Method. This equation (2) can be expressed simply as $P = RC$ ……………(3)

$$\begin{pmatrix} P_1 \\ P_2 \\ P_3 \\ P_n \end{pmatrix} = \begin{pmatrix} 1 & R_{11} & R_{12}^2 & R_{13}^3 & R_{1n}^n \\ 1 & R_{21} & R_{22}^2 & R_{23}^3 & R_{2n}^n \\ 1 & R_{31} & R_{32}^2 & R_{33}^3 & R_{3n}^n \\ 1 & R_{k1} & R_{k2}^2 & R_{k3}^3 & R_{kn}^n \end{pmatrix} \begin{pmatrix} C_0 \\ C_1 \\ C_2 \\ C_k \end{pmatrix}$$

……………………… (4)

Here P,R and C are matrices such that $P = n \times 1$ matrix, $R = k \times (n+1)$ matrix and $C = (k+1) \times 1$ matrix. Thus the above relation (3) can be rewritten as $RC = P$ ……………..….(5)

Now multiplying both sides of (4) by R^T (where R^T is the Transpose of the matrix R) i.e.,$(R^TR)C=R^TP$ …………..(6)

Here (R^TR) is again a matrix. Now multiplying both sides of (6) by $(R^TR)^{-1}$ the above relation (5) can be written as $(R^TR)^{-1} (R^TR)C = (RTR)^{-1} R^T P$ …….(7) The above equation becomes as $IC = (RTR)^{-1} R^T P$, where I is unit matrix, and it is clear that $IC = C$ in Matrix Algebra. The above equation (7) becomes as
$\mathbf{C = (R^TR)^{-1} R^T P}$ **……..(8)**

A Software company spent 40% , 55% and 41% more than the budget to maintain 81% , 88% and 96 % quality of software products for the years 2010, 2011 and 2012 respectively due to various factors in domestic and global markets[6]. Assuming all cost risks (P-values) on y-axis ad all quality risks (R-values) on x-axis

year	Quality risks (R-values)	Cost risks (P-values)
2010	81%	40%
2011	88%	55%
2012	96%	41%

Table 1. quality and cost risks

percentages in the Table 1 can be converted into real numbers as in Table 2.

year	Quality risks (R-values)	Cost risks (P-values)
2010	0.81	0.4
2011	0.88	0.35
2012	0.96	0.41

Table 2. quality and cost risks in real numbers

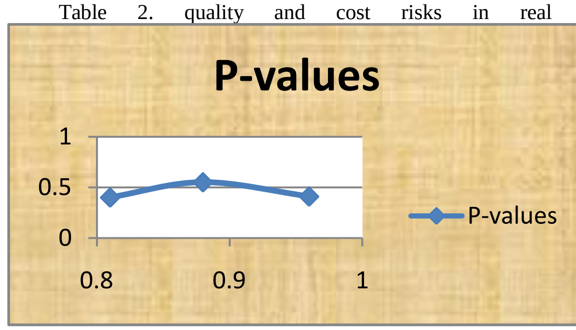


Figure 1: graphical representation of P and R values

In the above Figure 1 it is very clear that linear regression methods cannot be applied here to design the relation between quality and cost risks. There is one hump (U-shaped curve) in the above Figure 1. These type of data are common in real life tendencies[5][6]. Here it is advisable to use quadratic regression method to solve this situation. The above information in Table 2 in non-linear regression fitting[2] , then these can be written as follows :

$$0.4 = C_0 + C_1 (0.81) + C_2 (0.81)^2 \quad \ldots\ldots\ldots\ldots\ldots(9)$$

$$0.35 = C_0 + C_1 (0.88) + C_2 (0.88)^2 \quad \ldots\ldots\ldots\ldots\ldots(10)$$

$$0.41 = C_0 + C_1 (0.96) + C_2 (0.96)^2 \quad \ldots\ldots\ldots\ldots\ldots(11)$$

The above (5),(6) and (7) equations can be expressed in matrix form by using (4) as

$$\begin{pmatrix} 0.4 \\ 0.35 \\ 0.41 \end{pmatrix} = \begin{pmatrix} 1 & 0.81 & (0.81)^2 \\ 1 & 0.88 & (0.88)^2 \\ 1 & 0.96 & (0.96)^2 \end{pmatrix} \begin{pmatrix} C_0 \\ C_1 \\ C_2 \end{pmatrix}$$

……………………………………..(12)

It is of the form $P = RC$. Here

$$R = \begin{pmatrix} 1 & 0.81 & 0.6561 \\ 1 & 0.88 & 0.7744 \\ 1 & 0.96 & 0.9216 \end{pmatrix}$$

International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:2682
Vol.4, No.8, August 2015 DOI:10.15693/ijaist/2015.v4i8.5-8

$$R^T = \begin{bmatrix} 1 & 1 & 1 \\ 0.81 & 0.88 & 0.96 \\ 0.6561 & 0.7744 & 0.9216 \end{bmatrix}$$

where R^T is transpose of the matrix R

………………………… (13)

Then $R^T . R$ =

$$\begin{bmatrix} 3 & 2.65 & 2.3521 \\ 2.65 & 2.3521 & 2.0976 \\ 2.3521 & 2.0976 & 1.8795 \end{bmatrix}$$

………………….. (14)

$$\text{Then } (R^T R)^{-1} = \left[\frac{\text{Adjoint}(R^T R)}{\text{Determinant}(R^T R)} \right]$$

………………………(15)

Determinant $(R^T R)$ = 0.0000063

Adjoint $(R^T R)$ =

$$\begin{bmatrix} 0..2084619 & -0.04691 & 0.0262656 \\ -0.04691 & 0.1061256 & -0.059735 \\ 0.0262656 & -0.059735 & 0.0338 \end{bmatrix}$$

Thus by using (15), $(R^T R)^{-1}$ =

$$\begin{bmatrix} 3308.919 & -7446.032 & 4169.143 \\ -7446.032 & 16845.333 & -9481.746 \\ 4169.143 & -9481.746 & 5365.0794 \end{bmatrix}$$

By using equation (8), C can be evaluated as C $= (R^TR)^{-1} R^T P$ thus

$$C = \begin{bmatrix} C_0 \\ C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} 1.206 \\ -2.1174 \\ 1.1428 \end{bmatrix}$$

Equating the above two matrices, it is clear that

C_0 = 1.206 , C_1 = - 2.1174 and C_2 = 1.1428 ….(16) substituting these values in (2) , it can be expressed as

P = 1.206 - 2.1174 R + 1.1428 R^2 …………….(17) is the required design for the system where R stands for quality risks and P stands for cost risks. Thus (17) can be explained easily as

Cost risks = 1.206 – 2.1174 (quality risks) + 1.1428(quality risks)^2 ………………..(18)

II. RESULTS ANALYSIS

In 2012 company wanted to maintain a minimum of 98.5 % of quality requirements for the product. Thus it can be calculated as

In 2012 cost risks will be = 1.206 – 2.1174 (0.985) + 1.1428 $(0.985)^2$ = 1.206 – 2.085639 + 1.108773 = 0.23 i.e., 23 % more than the actual budget requirement is needed to meet a minimum of 98.5% quality of software product . Thus company will be in a position to meet the future requirements in a correct standard way. Company prepared in this way exactly. In 2012 they spent nearly 23.55% more than the budget requirement. i.e., the company is able to predict the future needs in advance and it achieved. Thus non-linear regression method provides a very sophisticated way to predict the future aspects of companies in a well planned manner. Here all calculations are approximated values but not exact values but these approximate values are nearer to the exact values calculated.

Here a very few values are considered for the design of the system as the size of the matrix (R^T R) will be large for further calculations. The output of the result will be deviated if the size of the matrix (R^T R) is increased which is a practical problem. Calculations will be clumsy if the size of the above matrix is increased. Another advantage of this system is even though some observations are considered, there will be no remarkable impact on the entire system. Non-linear regression provides a very helpful hand in complex situations where no solution may be found[3][4]. Results given by this method are very reliable in all aspects which pave the way to build future steps regarding any software products[5][6]. It is a manual method but many software are also available to find this. It makes to design any model with respect to any complicated problems. There is only one hump in the above Figure 1 that is why quadratic regression is recommended, if the above Figure 1 has two humps then cubic regression can be recommended. Predictions can be deviated if regression goes beyond quadratic[3][4]. This is the why quadratic regression mostly dominates the entire non-linear regression domain.

III. CONCLUSION

Taking the size of the matrix (R^T R) into consideration a few values are observed here. If more values have been taken into consideration then the size of the matrix is increased. If size of the matrix is increased then output prediction can be deviated. This is the reason why the size of the matrix (R^T R) is always small, within the constraints, but it will have no impact on the overall output of the system. It is a powerful tool which is very frequently used for big data analysis too. Quadratic can further be extended to cubic regression but it will have less impact on the output of the system which is not advisable. But still quadratic regression has its own prominence in designing experiments and making predictions. quadratic regression takes place wherever linear regression fails to find an appropriate solution to the given situation. Most of the non=linear regression techniques move around quadratic regression. This technique has been, now, considering for data mining techniques, statistical analysis and bid data analytics.

REFERENCES

[1].Bayesian Statistics and Marketing by Peter, Rossi and Allenby (Wiley Series)

[2] Probability and Statistics 1 for Cambridge International A Level by Janet Crawshaw

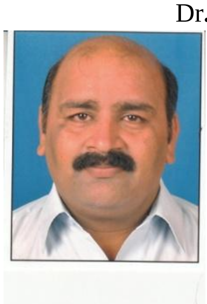
[3] Professional Software Development by Steve McConnell

[4] Software Estimation by Steve McConnell

[5]Steve McConnell Code Complete: A Practical Handbook of Software Construction

[6] Steve McConnell : Rapid Development

AUTHOR'S PROFILE



Dr.D.V.S.S.Subrahmanyam is Professor at Dept. of Computer Science & Engineering at Malla Reddy Engineering College (Autonomous) , Hyderabad. He did his graduation AMIETE, M.Tech and Ph.D in computer science & engineering, Also did M.Sc in Industrial Mathematics another M.Sc in Mathematics and M.Phil in Mathematics. Are as of interest include Digilal Logic Design, Data Structures, Algorithms, Software Engineering, Neural Networks, Fuzzy Logic and Big Data Analysis. Presently research work is going on Big Data Analysis.