Neural, Parallel, and Scientific Computations 19 (2011) 397-418

RANKED BASED REGRESSION METHODS TO SOFTWARE RELIABILITY MODELS

A. M. MOSTAFA AND K. M. RAMACHANDRAN

Department of Mathematics and Statistics, University of South Florida
Tampa, FL 33620-5700, USA
ram@cas.usf.edu

ABSTRACT. Statistical approaches such as the linear regression models are used to analyze software failure data. However, most of these regression models depend heavily on their underlying parametric assumptions that in general, are not satisfied for the software failure data sets. In this paper, we propose two regression models that are semi-parametric with unspecified error structures. The first approach is based on the ranks of the residuals, while the second approach is based on the monotone regression methods with the assumption that the ranks of failure data and the time between failures are linear. These methods are quite parallel to the least squares method, but it not only represents a robust alternative to the least square methods but also allows us to deal with nonlinear relationships. Simulation studies and real failure data are used to demonstrate the effectiveness of the proposed models.

AMS Subject Classification: 62N05

Key Words: Rank regression, monotone regression, software reliability

1. INTRODUCTION

In [15], it has been demonstrated that for most of the software failure data sets, the linear regression models outperform other popular models from literature in terms of the predictive accuracy of the Mean Time Between Failure (MTBF) estimates. These estimates were measured using the Mean Square Error (MSE) and Mean Absolute Value Difference (MAVD). However, for many of the software failure data, certain parametric assumptions, such as the normality, does not hold. Also, the assumption of uncorrelated errors is more often violated when applied to software failure data, because it is collected over time in a sequential way. In regression literature, there are many robust alternatives that have been developed in the last decades to deal with these situations [4], [5], [6], [9] and [19]. One of the alternative ways is to model the regression function nonparametrically so as to let the data decide on the functional form. In many cases, the fitted values determined nonparametrically are superior to the fitted values obtained from a parametric model. Nonparametric estimates of the regression coefficients of a linear regression model were proposed by [12]

Received October 14, 2011

and [13] by using the rank-based estimates. In this method, instead of minimizing the least square errors, the regression coefficients are estimated by minimizing the dispersion of the residuals consecutively. These estimates are based on the Wilcoxon scores and the generalized Wilcoxon–Mann–Whitney tests. These procedures make less stringent demands on the data and normality is not required to make inferences about the predicted values.

In this paper, we will analyze some of the software failure data using two specific non-parametric methods: namely, monotonic regression and rank regression. For many real software failure data sets, a simple scatter plot of time of failures versus time between failures show that the relationships need not be linear. The regression procedure proposed in [15] can be considered as a first degree approximation model to the real word data. The main strength of the proposed methods of this work is that they enable us to model response variable which is nonlinearly related to the predictor variable through a simpler linear regression procedure.

In Section 2, we will introduce some preliminary notations, definitions and motivation for using rank based methods. Section 3 will deal with error measurement criteria. Much of the analysis of outliers and influential high leverage are discussed in Section 4. In Sections 5 and 6, we will present proposed models. Numerical studies done with the proposed procedures as well. In Section 7, we will compare the predictive errors, mean square errors (MSE) and mean absolute value difference (MAVD), and the minimum values of the convex function for the new models with those of the least squares models. We will conclude the paper in Section 8.

2. **PRELIMINARIES**

The correlation coefficient is most useful in describing the degree of relationship between the failure time and the time between failures

$$\rho = \frac{n\sum_{i=1}^{n} t_i y_i - \sum_{i=1}^{n} t_i \sum_{i=1}^{n} y_i}{\sqrt{\left(n\sum_{i=1}^{n} t_i^2 - \left(\sum_{i=1}^{n} t_i\right)^2 \left(n\sum_{i=1}^{n} y_i^2 - \left(\sum_{i=1}^{n} y_i^2\right)^2\right)\right)}}$$

We can then determine the probability that the observed correlation occurred by chance by conducting a significance test: Null Hypothesis $H_0 : \rho = 0$ versus Alternate Hypothesis: $H_0 : \rho \neq 0$. We will use the Spearman's rho for the two data sets; System 40 and Project 1, [16]. The coefficient correlation between the failure time and the time between failures of System 40 is $\rho = 0.597$. The test shows that the correlation is significant at the 0.01 level (2-tailed). Similarly for Project 1, the correlation is $\rho = 0.544$ and the test shows that we should reject the null hypothesis that $\rho = 0$ at level 0.01. Thus, we can assume that the relationship between the failure data and

the time between failures exists. A similar test for relationship between the ranks of failure time and the time between failures results in the following: The correlation between ranks for system 40 is $\rho = 0.608$. The result is to reject the null hypothesis that the correlation is zero. The ranks correlation of Project 1 failure data is $\rho = 0.581$ and the conclusion is to reject the null hypothesis.

Now, we will introduce some important terms and definitions. Let $T_1, \ldots, T_n$ be the failure times of the software, and let Y be the random variable representing the time to the next failure. Consider the following linear regression model for software failure prediction:

$$Y_i = \alpha + \beta T_i + \varepsilon_i, \quad 1 \leq i \leq n, \tag{2.1}$$

where α is the intercept parameter, β is the parameter representing the regression coefficient and ε_i represents the random errors.

Definition 1 ([7])**.** Let $D(.)$ be a measure of variability that satisfies the following two properties: (i) $D(T+a) = D(T)$, and (ii) $D(-T) = D(T)$. Then $D(.)$ is called an *even location-free measure of variability*, it follows that $D\left((Y_i - \beta T_i) - \alpha\right) = D\left(Y_i - \beta T_i\right)$. Hence, the intercept α of (2.1) has no effect on the measure of variability and we only need to analyze the residuals $Y_i - \beta T_i$.

Definition 2. The *rank regression estimator* $\widehat{\beta}$ is the value of β that minimizes the sum:

$$D(\beta) = \sum_{i=1}^{n} R_i^c(\beta)\left(Y_i - \beta T_i\right) \tag{2.2}$$

where

$$R_i^c(\beta) = R_i(\beta) - \frac{(n+1)}{2} \tag{2.3}$$

is the *centered ranks* or *mid-ranks* and $R_i(\beta)$ are the ranks of the residuals $Y_i - \beta T_i$ Here $R_i(\beta)$ denotes the rank of the residual as a function of β.

It follows that $\sum_{i=1}^{n} R_i^c(\beta) = 0$, $R_1^c(\beta) \leq R_2^c(\beta) \leq \cdots \leq R_n^c(\beta)$ (monotone), and $R_i^c(\beta) = -R_{n-i+1}^c(\beta)$ (antisymmetric). Also, the dispersion function $D(\beta)$ is a linear combination of ordered residuals. Assume that $D(.)$ satisfies the assumptions of Definition 1 and let $\widehat{\beta}$ be the value that minimizes $D\left(Y - \beta T_i\right)$. The corresponding minimum is denoted by $V = D\left(Y_i - \widehat{\beta} T_i\right)$. This is a semi-parametric approach to the problem of estimating the regression coefficients in a linear model. For the rank procedure, the residuals are no longer squares as the least squares method, but are weighted according to their ranks. This helps to reduce the effects of outliers and makes it desirable for heavy tailed residual distributions (see [10]).

Theorem 1. *$D(\beta)$ defined in equation (2.2) is indeed an even location-free measure of variability.*

Proof. From (2.1), it is enough to prove that: (i) $D(\beta, \alpha) = D(\beta)$, where $D(\beta, \alpha) = D\left((Y_i - \beta T_i) - \alpha\right)$, and (ii) $D(-\beta) = D(\beta)$.

(i) Now,

$$\begin{aligned} D(\beta, \alpha) &= D\left((Y_i - \beta T_i) - \alpha\right) \\ &= \sum_{i=1}^{n} R_i^c(\beta)\left(Y_i - \beta T_i - \alpha\right) \\ &= \sum_{i=1}^{n} R_i^c(\beta)\left(Y_i - \beta T_i\right) - \alpha \sum_{i=1}^{n} R_i^c(\beta) \\ &= \sum_{i=1}^{n} R_i^c(\beta)\left(Y_i - \beta T_i\right) = D(\beta). \end{aligned}$$

(ii) To show that $D(-\beta) = D(\beta)$,

$$\begin{aligned} D(-\beta) &= D\left(-\left(Y_i - \beta T_i\right)\right) \\ &= \sum_{i=1}^{n} -R_i^c(\beta)\left(Y_i - \beta T_i\right) \\ &= \sum_{i=1}^{n} R_{n-i+1}^c(\beta)\left(Y_i - \beta T_i\right) \text{ (by antisymmetry)} \\ &= \sum_{i=1}^{n} R_i^c(\beta)\left(Y_i - \beta T_i\right) = D(\beta). \end{aligned}$$

Thus, we have an even and location-free measure of dispersion. □

3. ERROR MEASUREMENTS PREDICTION CRITERIA

We denote the estimated value of a measure of the time between failures with $\widehat{MTBF}$ and the actual value with TBF. We will be using the following metrics to evaluate the accuracy of estimates and to compare the software models.

Mean Squared Error (MSE). The MSE of an estimator T of an unobservable parameter θ is defined by

$$MSE(T) = E\left((T - \theta)^2\right).$$

Then,

$$MSE = \frac{1}{n} \sum_{i=1}^{n} \left(TBF_i - \widehat{MTBF}_i\right)^2. \tag{3.1}$$

Mean Absolute Value Difference (MAVD). $MAVD$ is defined as the average of the difference between predicted mean time between failures and actual time between failure values,

$$MAVD = \frac{1}{n}\sum_{i=1}^{n}\left|TBF - \widehat{MTBF}\right|. \tag{3.2}$$

Magnitude of Relative Error (MRE) ([17])**.** MRE is the absolute value of the relative error, defined by

$$MRE = \left|\frac{\widehat{MTBF} - TBF}{TBF}\right| \tag{3.3}$$

Mean Magnitude of Relative Error (MMRE). $MMRE$ is the mean of MRE.

Conte et al. [3] considered $MMRE \leq 0.25$ to be an acceptable value for prediction models effort. There are advantages for this assessment: (i) Comparisons can be made easily across failure time data sets, (ii) the mean magnitude of relative error is independent of units of data, (iii) comparisons can be made across all types of prediction models, [3], and (iv) since $MMRE$ is independent of scale, which is the expected value of MRE does not vary with size. Kitchenham et al. [14] proposed another measure, the magnitude of error relative to the estimate.

Magnitude of Error Relative to the Estimate (MER). The MER is defined as

$$MER = \frac{\left|\widehat{MTBF} - TBF\right|}{\widehat{MTBF}} \tag{3.4}$$

Mean Magnitude of Error Relative to the Estimate (MMER). $MMER$ is defined as the mean of MER.

In [17], it was shown that MER measure seems preferable to MRE because it measures the error relative to the estimate value of mean time between failures.

Median of Absolute Residual (MdAR). Another measure proposed in [14] is median of the absolute error AR instead of $MMRE$, where the absolute error is defined as

$$AR = \left|TBF - \widehat{MTBF}\right| \tag{3.5}$$

Then, $MdAR$ is the median of the values of AR. Also, MAR, which is the mean of AR. MAR is nothing but $MAVD$, which we have already calculated. Table 8 shows error values of different methods of measurement (MSE,$MAVD$,$MMRE$,$MdAR$) for some of the software failure time data sets such as System 40, Project 1, and Project 5 in [16].

The Measure of Dispersion (V).

$$D\left(\widehat{\beta}\right) = \sum_{i=1}^{n} R_i^c\left(\widehat{\beta}\right)\left(Y_i - \widehat{\beta}X_i\right) \tag{3.6}$$

where $R_i^c\left(\widehat{\beta}\right)$ is the centered rank of the residuals.

4. DETECTING OUTLIERS AND LEVERAGE VALUES

In regression analysis, outliers, or over-influential observations represent observations that need careful examination. The residuals are the most commonly used measures for detecting outliers. There are two common ways to calculate the standardized residual for the ith observation. One uses the residual mean square error from the model fitted to the full data set (internally studentized residuals). The other uses the residual mean square error from the model fitted to all of the data except the ith observation (externally studentized residuals). The externally studentized residuals follow a t-distribution [1]. The following formula gives the amount of contribution of the ith observation to its own fitted value in least squares regression. It is useful in determining the influence of each observation

$$h_i = \frac{1}{n} + \frac{\left(t_i - \overline{t}\right)^2}{S_{tt}} = \frac{1}{n} + \frac{\left(t_i - \overline{t}\right)^2}{\sum\limits_{i=1}^{n}\left(t_i - \overline{t}\right)^2}. \tag{4.1}$$

Note that

$$h_i \geq \frac{1}{n} \text{ and } \sum_{i=1}^{n} h_i = 2. \tag{4.2}$$

We used $h_i \geq 6/n$ as a threshold condition to obtain leverage observations. A good leverage point is a point that is unusually large or small among the observations, and is not a regression outlier. In the Project 1 data set, there are two outliers (0.69 and 1.10) and three leverage values (0.69, 1.10 and 1.39). We can consider the observation 1.39 as a good leverage point. Additionally, in System 40, there are no outliers but one leverage point that is considered to be a good leverage value. A bad leverage point is a point that has an unusually large residual corresponding to regression line. A bad leverage point is an outlier among all observations as well. Bad leverage points can adversely affect the estimated value of the slope. Such an effect has been seen in the case of Apollo 8, with the leverage value 33. The normality test reveals that Apollo 8 data is not normal; Projects 1 and 5 have outliers and slightly depart from the normality. In such cases, it can substantially reduce our ability to detect a true association between the failure time and time between failures. However, the outliers among the failure time T values will inflate the sample variance s_t^2, and this will decrease the standard error of the least squares estimate of the slope. This suggests that outlier points are beneficial in terms of increasing our ability to detect regression

lines which have a nonzero slope. However, there is another concern that must be taken into consideration. If the T value is a leverage point and the time between failures is an outlier, then we have a regression outlier that might completely distort how the bulk of the points are related. In a similar manner, we can fail to detect a situation where the slope differs from zero. This occurs not because the slope is indeed zero, but because regression outliers mask an association among the bulks of the points under study. Also, the error measurements of the least squares regression line is not resistant to outliers as we have seen in the following tables described in [15] for the Apollo 8, System 40, Project 1 and Project 5 data sets. Now, we will demonstrate in Table 1 the effect of outlier and leverage values in Apollo 8 data on the error measurements of the software reliability regression model of the first degree (SRRM1) using the least square procedure.

TABLE 1. The Effects of Outlier and Leverage Values on SRRM1 (A8) Model

Model	Measurement Criteria	Complete Data	Data Without Leverage Value '91'	Data Without Outliers '33, 91'
SRRM1	MSE	151, 9420	29, 3014	7.1063
SRRM1	MAVD	6, 3229	3.1597	1.9085
SRRM1	MMER	0.5914	0.57	0.5147
SRRM1	MMRE	2.2680	1.01	0.7385
SRRM1	MdAR	2.2045	2.18	2.0853
SRRM1	V(LSR)	341.25	11.64	73.6

We have been shown that the least squares regression line has an estimated slope of 0.1132, MSE = 151.9420, and MAVD = 6.3229 in the case of complete data, while the slope of the regression is -0.0428, MSE = 7.1063, and MAVD = 1.9805 in the case of removing the two outliers, 33 and 91, of TBF. This negative association is missed by the least squares regression line because of the outliers. In fact the two outliers, 33 and 91, caused problems. Even if we remove the most extreme outlier, 91 and keep the other less hazardous outlier, 33, we get the following results: the slope of the regression line is 0.0124, MSE = 29.3014, and MAVD = 3.1597. Besides, the mean square error is improved by removing the outliers. Therefore, the least squares regression line offers no hint of a strong association even though the mean square error is low compared to previous models. The slope estimator is not resistant to outliers and if there is a small departure from normality, devastating consequences would erupt when trying to use the least squares regression method. In other words, this method is not robust and the slope of the regression line can be extremely sensitive

to small changes in the probability curve or to the existence of outliers. The removal of the outliers from Apollo 8 data improves the SRRM1 model by decreasing its MSE and MMRE values. This model is not robust for Apollo 8 effort prediction, even if it has better results than both of Roberts [18] and Suresh [22] models.

The System 40 software failure data has no outliers, but it has one leverage value. Table 2 demonstrates the effect of even one single leverage value on the error measurements of the least squares regression model. The SRRM1 model is very sensitive to leverage values too.

Since MMRE $= 0.12 \leq 0.25$, applying this model on System 40 data can be considered as acceptable for effort prediction. Therefore, there is no need for us to eliminate the leverage values from System 40 failure data.

TABLE 2. The Effects of Outlier and Leverage Values on SRRM1 Model (S 40)

Model	Measurement Criteria	Complete Data	Data Without Leverage Value '4.70'	Data Without Outliers
SRRM1	MSE	2.5657	2.3428	No Outliers
SRRM1	MAVD	1.2370	1.1736	No Outliers
SRRM1	MMER	0.1224	0.1138	No Outliers
SRRM1	MMRE	0.1320	0.1223	No Outliers
SRRM1	MdAR	1.0701	1.0336	No Outliers
SRRM1	V(LSR)	404	400	No Outliers

TABLE 3. The Effects of Outlier and Leverage Values on SRRM1 (P1) Model

Model	Measurement Criteria	Complete Data	Data Without Leverage Value '91'	Data Without Outliers '33, 91'
SRRM1	MSE	1.7965	1.6384	1.7209
SRRM1	MAVD	0.9479	0.9195	0.9428
SRRM1	MMER	1.1691	0.1628	0.1694
SRRM1	MMRE	0.2733	0.2277	0.2423
SRRM1	MdAR	0.6596	0.6125	0.6577
SRRM1	V(LSR)	343	357.65	355

This model is sensitive to outliers. These analyses of the data sets show the need to use more robust methods in lieu of the least square method for software reliability analysis. Even though, the least squares regression methods in terms of error measures introduced in [15] gave very good results, there was an issue of small values. This casts doubt about the robustness of a simple linear regression model. Since we are dealing with real life software failure time, it is not acceptable to drop the outliers from analysis. We will now utilize two methods, the monotone and rank regression that are less sensitive to outliers and high leverage values.

5. MONOTONE REGRESSION

Iman and Conover [11] introduced the idea of rank transform procedure in regression as an alternative method to formulating a nonlinear data [20]. In software failure models, as the bugs are removed, it is assumed that the time between failures will increase on average. That is, assume that $E(Y \mid T)$ increases (at least, it does not decrease) as T increases. Since there is a monotonic relationship between T and Y for Apollo 8, Project 1, and System 40 failure data sets, the relationship between T and Y is nonlinear. The procedure used in this work is to replace the dependent variable Y by $R(Y_i)$, the assigned rank to the ith value of Y. Similarly, replace each of the failure times T with its corresponding ranks $R(T_i)$. Ties are assigned by their average ranks. The monotonic regression procedures depend on the fact that the ranks of these two variables have a linear relationship if the corresponding variables have a monotonic relationship. The linear regression equation based on ranks is given by

$$R(Y_i) = \alpha + \beta R(T_i) + \varepsilon_i \tag{5.1}$$

The least squares regression analysis is performed on the ranks of T and Y. The regression equation which expresses $\widehat{R}(Y_i)$ in terms of $R(T_i)$ is

$$\widehat{R}(Y_i) = (n+1)/2 + \widehat{\beta}\,(R(T_i) - (n+1)/2) + \varepsilon_i. \tag{5.2}$$

Since these monotonic procedures can be viewed as the usual parametric procedures applied to ranks, the hypothesis test of the ranked data suggests using the linear regression. We will apply the following algorithms on software failure data to obtain the regression curve.

Algorithm of Obtaining the Estimate of $E(Y \mid T)$ at a Particular Point. In order to estimate the regression of the time between failures Y on T at a particular failure time $T = t_0$, apply the following algorithm [2].

1. Acquire the ranks $R(T_i)$ and $R(Y_i)$ of the T and Y, respectively. In case of ties use average of tied ranks.

2. Identify the least squares regression estimators of equation (5.1),

$$\widehat{\beta} = \frac{\sum_{i=1}^{n} R(T_i) R(Y_i) - n(n+1)^2/4}{\left[\sum_{i=1}^{n} [R(T_i)]^2 - n(n+1)^2/4\right]} \tag{5.3}$$

$$\widehat{\alpha} = \left(1 - \widehat{\beta}\right)(n+1)/2, \tag{5.4}$$

3. A rank $R(t_0)$ for t_0 can be acquired by applying the next algorithms:
   (a) If t_0 equals one of the observed points T_i, let $R(t_0)$ equal the rank of that T_i.
   (b) If t_0 exists between two adjacent values T_i and T_j where $T_i < t_0 < T_j$, interpolate between their respective ranks to get $R(t_0)$

$$R(t_0) = R(T_i) + \frac{t_0 - T_i}{T_j - T_i}\left[R(T_j) - R(T_i)\right]. \tag{5.5}$$

   Note that this "rank" is not necessarily an integer.
   (c) If t_0 is less than the smallest observed T or greater than the largest observed T, do not attempt to extrapolate. Information on the regression of Y on T is available within the observed range of T.
4. Substitute $R(t_0)$ for t in (5.1) to get an estimated rank $R(y_0)$ for the corresponding value of $E(Y \mid T = t_0)$

$$\widehat{R}(y_0) = \widehat{\alpha} + \widehat{\beta} R(t_0). \tag{5.6}$$

5. Transform $R(y_0)$ into $\widehat{E}(Y \mid T = t_0)$, an estimate of $E(Y \mid T = t_0)$, by referring to the observed Y_i as follows:
   (a) If $R(y_0)$ equals the rank of one of the observations Y_i, let the estimate $\widehat{E}(Y \mid T = t_0)$ equal observation Y_i.
   (b) If $R(y_0)$ is between the ranks of two adjacent values of Y, say Y_i and Y_j where $Y_i < Y_j$, so that $R(Y_i) < R(y_0) < R(Y_j)$, interpolate between Y_i and Y_j:

$$\widehat{E}(Y \mid T = t_0) = Y_i + \frac{R(y_0) - R(Y_i)}{R(Y_j) - R(Y_i)}(Y_i - Y_j). \tag{5.7}$$

   (c) If $R(y_0)$ is greater than the largest observed rank of Y, let $\widehat{E}(Y \mid T = t_0)$ equal the largest observed value Y. If $R(y_0)$ is less than the smallest observed rank of Y, let $\widehat{E}(Y \mid T = t_0)$ equal the smallest observed value Y.

The Estimate of the Regression of Y on T. The following procedure is used to get the entire regression curve consisting of all points.

1. Obtain the end points of the regression curve by using the smallest $T^{(2.1)}$ and the largest $T^{(n)}$ observations in the preceding procedure to obtain $\widehat{E}\left(Y \mid T = t^{(2.1)}\right)$ and $\widehat{E}\left(Y \mid T = t^{(n)}\right)$.

2. For each rank of Y, find the estimated rank of T_i and $\widehat{R}(T_i)$ from (5.2)

$$\widehat{R}(T_i) = [R(Y_i) - \widehat{\alpha}]/\widehat{\beta}, \quad i = 1, \ldots, n. \tag{5.8}$$

3. Transform each $\widehat{R}(T_i)$ to an estimate $\widehat{T}_i$ in the manner of the preceding Step 5. More specifically:
   (a) If $\widehat{R}(T_i)$ equals the rank of some observation T_i, let $\widehat{T}_i$ equal that observed value.
   (b) If $\widehat{R}(T_i)$ is between the ranks of two adjacent observations T_j and T_k, where $T_j - T_k < 0$, obtain $\widehat{T}_i$ by using the following equation:

$$\widehat{T}_i = T_j + \frac{\widehat{R}(T_i) - R(T_j)}{R(T_k) - R(T_j)}[T_k - T_i]. \tag{5.9}$$

   (c) For all observed ranks of T values, if $\widehat{R}(T_i) < \min(R(T_i))$ or $\widehat{R}(T_i) > \max(R(T_i))$, then there is no estimate for $\widehat{T}_i$.
4. Plot each of the points found in Step 3, with Y_i as the ordinate and $\widehat{T}_i$ as the abscissa. Also plot the end points found in Step 1, with $E(\widehat{Y} \mid T)$ as the ordinate and $T^{(2.1)}$ or $T^{(n)}$ as the abscissa. All these points are monotonic, increasing if $\widehat{\beta} > 0$ and decreasing if $\widehat{\beta} < 0$.
5. The estimate of the regression of Y on T is represented by lines joining points in Step 4.

Results of Monotone Regression. Now we will apply the monotone regression procedure to some of the software failure data sets.

TABLE 4. Analysis of Software Monotone Regression Method (Apollo 8)

Measurement Criteria	Complete Data	Data Without Leverage Values '33', '91'	Data Without Leverage Value '91'
MSE	326.7174	9.8692	42.9201
MAVD	6.9378	2.5810	3.7246
MMER	1.4040	0.5796	0.7987
MMRE	1.2132	0.9779	1.1072
MdAR	2.3256	2.1249	2.4462

Table 4 demonstrates its sensitivity to outliers and leverage values.

TABLE 5. Analysis of Software Monotone Regression Method (System 40)

Measurement Criteria	Complete Data	Outliers	Data Without Leverage Value '91'
MSE	2.9877	No outliers exist	2.7345
MAVD	1.3286	No outliers exist	1.2868
MMER	0.1295	No outliers exist	0.1249
MMRE	0.1460	No outliers exist	0.1365
MdAR	1.0331	No outliers exist	1.0166

There are no outliers in System 40, but one can observe that in Table 5 the impact of removing the leverage values on the measurements output is very little. This means that the monotone regression is insensitive to leverage values for System 40 data. At the same time, notice that MMRE $= 0.1460 < 0.25$ for complete System 40 data, indicating that the monotone regression is an acceptable predictive model for applying System 40.

TABLE 6. Analysis of Software Monotone Regression Method (Project 1)

Measurement Criteria	Complete Data	Data Without Outliers	Data Without Leverage Values
MSE	2.0517	19.6718	16.05
MAVD	1.0592	4.1360	3.77
MMER	0.0408	2.9756	2.10
MMRE	0.1981	0.7150	0.64
MdAR	0.7494	4.31	3.91

From Table 6, we see that the Monotone regression method is very good for complete data. MSE and MAVD values are small, and MMRE = 0.1981. This information proves that the monotone regression for the Project 1 data set is an acceptable prediction model. While if we remove the outliers or the leverage values, then MMRE = 0.7150 and MMRE = 0.64 respectively lead us to decide that this model is not acceptable for prediction in the absence of outliers and leverage values. Because we are dealing with real life software data, it is crucial to keep these values.

TABLE 7. Analysis of Software Monotone Regression Method (Project 5)

Measurement Criteria	Complete Data	Data Without Outliers	Data Without Leverage Values
MSE	2.7975	2.4576	2.3085
MAVD	1.3348	1.2819	1.2525
MMER	0.0148	0.0109	0.0090
MMRE	0.0597	0.0451	0.0399
MdAR	1.1471	1.1108	1.1

Since the value MMRE $= 0.0597 \leq 0.25$ in Table 7, it is significant, that the monotone regression model is an acceptable model for prediction for complete data. The percent effect of outliers is 12.15 percent, while the percentage effect of leverage values is 17.48 percent. This result shows that the monotone regression is insensitive to outliers for Project 5.

6. RANK REGRESSION BASED ON ROBUST SLOPE ESTIMATION

For the monotone regression, we were investigating errors by assuming that the ranks of the failure data and the time between failures were linearly related. However, in this section we are taking into consideration the ranks of the residuals. Ranking the residuals reduces the effects of outliers [9, 10]. Iman and Conover [11], Sawyer [21], and Theil [23, 24, 25] introduced the rank regression method. In this section, we will adopt this method to study the software reliability models. Recall equations (2.1), (2.2), and (2.3). Since

$$\sum_{i=1}^{n} R_i^c(\beta) = \sum_{i=1}^{n} \left(R_i(\beta) - \frac{(n+1)}{2} \right) = 0$$

then

$$D(\beta, \alpha) = \sum_{i=1}^{n} R_i^c(\beta) \left(Y_i - \beta T_i - \alpha \right)$$

$$D(\beta, \alpha) = \sum_{i=1}^{n} R_i^c(\beta) \left(Y_i - \beta T_i \right) = D(\beta). \tag{6.1}$$

Equation (5.6) represents the sum that involves the residuals $(Y_i - \beta T_i - \alpha)$ of the regression equation (2.1). Instead of equation (5.6), the least-squares estimators (LSE) of α and β are found by minimizing:

$$C(\alpha, \beta) = \sum_{i=1}^{n} \left(Y_i \alpha - \beta T_i \right)^2 . \tag{6.2}$$

The LSE of $\widehat{\beta}_c$ from equation 5.7 is:

$$\widehat{\beta}_c = \frac{\sum_{i=1}^{n} \left(Y_i - \overline{Y}\right)\left(T_i - \overline{T}\right)}{\sum_{i=1}^{n} \left(T_i - \overline{T}\right)^2}. \tag{6.3}$$

Equation (5.6) can be minimized for β, but the computations are very difficult. However a natural generalization of the LSE $\widehat{\beta}_c$ in equation (5.8) is to minimize

$$E(\alpha, \beta) = \sum_{i=1}^{n} |Y_i - \alpha - \beta T_i| \tag{6.4}$$

instead of $C(\alpha, \beta)$ in equation (5.7). But the parameter estimates $\widehat{\alpha}_1$, and $\widehat{\beta}_1$ are computed by minimizing the function $E(\alpha, \beta)$ in equation (5.9) are not easy to analyze for the proposed rank regression model. This leads us to use Theil's statistics procedure [8], [23], [24], and [25] to compute the slope estimator of equation (2.1). Theil's estimator is:

$$\widehat{\beta}_{TH} = \text{median}\{S_{ij}\}, \quad 1 \le i < j \le n \tag{6.5}$$

where

$$S_{ij} = \frac{(Y_j - Y_i)}{(T_j - T_i)}, \quad 1 \le i < j \le n.$$

Compute $N = n(n-1)/2$ individual sample slope values of S_{ij}. Let $S^{(2.1)} \le S^{(2.2)} \le \cdots \le S^{(N)}$ be the ordered sample slope values of S_{ij}. If $N = 2k+1$ is odd, then $\widehat{\beta}_{TH} = S^{(k+1)}$, where $k = (N-1)/2$. If $N = 2k$, then $k = N/2$ and $\widehat{\beta}_{TH} = \left[S^{(k)} + S^{(k+1)}\right]/2$. If the values of T_i are equally spaced, then $\widehat{\beta}_{TH}$ and the rank regression estimator $\widehat{\beta}$ from [8] can be shown to be asymptotically equally powerful for estimating β, [8]. If T_i are not equally spaced as in the software failure time data sets [16], then the rank-regression estimator $\widehat{\beta}$ is asymptotically more powerful, or is more accurate for the same size of failure data, [21].

Theorem 2. *If $R_i^c(\beta) = R_i(\beta) - \frac{n+1}{2}$ is the centered rank of $Y_i - \beta T_i$, then $D(\beta) = \sum_{i=1}^{n} R_i^c(\beta)\,(Y_i - \beta T_i)$ is continuous, piecewise linear, and convex upwards. Let*

$$k_T = \min\left\{k : S_k = -Q + \sum_{p=1}^{k} |T_{jp} - T_{ip}|\right\} > 0. \tag{6.6}$$

For $k = k_T$, the rank regression estimator $\widehat{\beta}$ is:

$$\widehat{\beta} = W_k = \frac{Y_{jk} - Y_{ik}}{T_{jk} - T_i}, \quad \text{if } S_{k-1} < 0 < S_k. \tag{6.7}$$

Equation (6.3) *corresponds to the case of a unique minimum value for $D(\beta)$.*

$$\widehat{\beta} = \frac{W_{k-1} + W_k}{2}, \quad \text{if } S_{k-1} = 0 < S_k. \tag{6.8}$$

Equation (6.4) *corresponds to an interval of minimum values,* [8], [12] *and* [21].

Proof. Here we present our proof, according to the possible values/conditions of $D(\beta)$:

(i) $D(\beta)$ **is a piecewise linear function**. For simplicity, assume that there are no ties among the failure data pairs (T_i, Y_i). Let $R_\alpha(\beta)$ be the ranks of $Z_\alpha = Y_\alpha - \beta T_\alpha$, for $1 \leq \alpha \leq n$. If there is a solution of $Y_j - \beta T_j = Y_i - \beta T_i$ for $i \neq j$, then the values of Z_α have tied values of (T_α, Y_α)

$$Y_j - \beta T_j = Y_i - \beta T_i \Leftrightarrow Y_j - Y_i = \beta\,(T_j - T_i)$$

$$\beta = Z(i,j) = \frac{Y_j - Y_i}{T_j - T_i}; \quad T_i \neq T_j. \tag{6.9}$$

Equation (6.5) implies $\{Z_\alpha\} = \{Y_\alpha - \beta T_\alpha\}$ has no tied values as

$$\beta \neq Z(i,j) = \frac{Y_j - Y_i}{T_j - T_i},\ (T_j \neq T_i)$$

which are at most $N = \frac{n(n-1)}{2}$ in number. The sorted values $(Y_\alpha - \beta T_\alpha)$ keeps the same relative order as soon as

$$\beta \neq Z(i,j) = \frac{Y_j - Y_i}{T_j - T_i}.$$

In other words, the ranks $R_i(\beta)$ are constant in each of the $N + 1 = \frac{n(n-1)}{2} + 1$ intervals (W_k, W_{k+1}) for $0 \leq k \leq N$, where W_k are the sorted values $Z(i,j)$, such that it is sorted in an increasing order with $W_0 = -\infty$, and $W_{N+1} = +\infty$. Since the ranks $R_i(\beta)$ are constant in each interval (W_k, W_{k+1}), then the function

$$D(\beta) = \sum_{i=1}^{n} R_i^c(\beta)\,(Y_i - \beta T_i)$$

is piecewise linear

$$D(\beta) = A_k + \beta B_k \tag{6.10}$$

where $W_k < \beta < W_{k+1}$, $A_k = \sum_{i=1}^{n} R_i^c(\beta) Y_i$, and $B_k = -\sum_{i=1}^{n} R_i^c(\beta) T_i$.

(ii) $D(\beta)$ **is convex upward**. If $\beta_1 < W_1$, then the relative order of $Y_i - \beta T_i$ stays the same as β becomes arbitrarily large and negative. This implies that $Y_i - \beta T_i$ has the same relative order as T_i, or $R\,(T_i)$. With this said, the slope then the slope of $D(\beta)$ is $B_0 = -Q < 0$, where $Q = \sum_{i=1}^{n} R_i^c(T) T_i > 0$. If $\beta > W_N$, where $N = n(n-1)/2$, then the slope of $D(\beta)$ is $B_N > 0$, $\sum_{i=1}^{n} R_i^c(T)$, $T_i > 0$. Therefore, $D(\beta)$ has a minimum value, i.e., $D(\beta)$ is convex upwards.

(iii) $D(\beta)$ **is continuous**. Now what happens if $W_1 < \beta < W_N$, and β crosses one of the values of W_k, $1 \leq k \leq N$? If there is a single tie, then: $Y_j - \beta T_j = Y_i - \beta T_i$ at $\beta = W_k$ with $T_i < T_j$

$$Y_a - \beta T_a = Y_a - W_k T_a + (W_k - \beta)\,T_a$$

if $\beta < W_k$ then $Y_j - \beta T_j > Y_i - \beta X_i$, and if $\beta > W_k$ then $Y_j - \beta T_j < Y_i - \beta X_i$. If β is sufficiently close to W_k, then no other values $Y_a - \beta T_a$ lie between the previous two values. Therefore, there exists an integer m such that for sufficiently small $b > 0$, we have:

$$R_j(\beta) = m + 1 \text{ and } R_i(\beta) = m \text{ for } W_k - b < \beta < W_k \tag{6.11}$$

$$R_j(\beta) = m \text{ and } R_i(\beta) = m + 1 \text{ for } W_k < \beta < W_k + b \tag{6.12}$$

and we have no other change in ranks $R_a(\beta)(a \neq i, j)$ in the interval $W_k - b < \beta < W_k + b$ implies

(a) $D(\beta) = \sum_{a=1}^{n} R_a(\beta)\,(Y_a - RT_a)$ is a continuous function across $\beta = W_k$. By equations (6.10)–(6.12), there is only one change in the ranks of $R_a(\beta)$. This change happens when both ranks $R_i(\beta)$ and $R_j(\beta)$ switch their values at $\beta = W_k$. However, since the coefficients $R_i^c(\beta)$, $R_j^c(\beta)$ have the tied values $Y_j - \beta T_j = Y_i - \beta T_i$ for $D(\beta)$ under the condition $\beta = W_k$, $D(\beta)$ is continuous. We can then follow the same procedures if $\{Y_a - \beta T_a\}$ has multiple tied values at $\beta = W_k$.

(b) By assuming $T_i < T_j$ and using equations (6.10)–(6.12) as β increases through W_k the change in the slope parameter $D(\beta)$ is

$$\begin{aligned} B_k - B_{k-1} &= -\left(\sum_{a=1}^{n} R_a^c\,(\beta + a)\,T_a - \sum_{a=1}^{n} R_a^c(\beta - a)T_a\right) \\ &= -\left((mT_j + (m+1)T_i) - ((m+1)T_j + mT_i)\right) \\ &= T_j - T_i. \end{aligned} \tag{6.13}$$

However, if we assume $T_i > T_j$, then $B_k - B_{k-1} = T_i - T_j$. In general for all the values of T_i and T_j, we have $B_k - B_{k-1} = |T_i - T_j|$.

Assume that T_i are increasing, then the slope of $D(\beta)$ increases by $T_j - T_i > 0$ at each $\beta = W_k$ for $1 \leq k \leq N$. If $\beta < W_1$, then the slope of $D(\beta) = -Q$, and if $\beta > W_N$, then the slope of $D(\beta) = Q$. For verification, consider the pair (i, j) for $W_k = Z(i, j)$ and (6.5), where

$$\{W_k : 1 \leq k \leq N\} = (\text{sorted})\left\{\frac{Y_j - Y_i}{T_j - T_i} : 1 \leq i < j \leq n, T_i \neq T_j\right\}. \tag{6.14}$$

If the values of T_i are strictly increasing, then

$$\begin{aligned} \sum\sum_{(i,j)} (T_j - T_i) &= \sum \sum_{1 \leq i < j \leq n} (T_j - T_i) = \sum_{j=1}^{n}\sum_{i=1}^{j-1} T_i - \sum_{i=1}^{n}\sum_{j=i+1}^{n} T_j \\ &= \sum_{j=1}^{n} (j - 1 - (n - j))\,T_j = 2\sum_{j=1}^{n}\left(j - \frac{(n+1)}{2}\right)T_j = 2Q. \end{aligned} \tag{6.15}$$

This is another proof that the slope of $D(\beta)$ is $B_N = Q$. To find the minimum value of $D(\beta)$. Using equation (6.8) we have that the slope of $D(\beta)$ in (W_k, W_{k+1})

is:

$$B_k = B_0 + \sum_{p=1}^{k} (B_p - B_{p-1}) = B_0 + \sum_{p=1}^{k} (T_{jp} - B_{ip}) = S_k \tag{6.16}$$

Thus, $S_k = B_k$ in equation (6.10) is the slope of $D(\beta)$ in (W_k, W_{k+1}):

(i) If $S_{k-1} < 0 < S_k$ then $\beta = \widehat{\beta} = W_k$ is the unique minimum value of $D(\beta)$. Resulting in: $\widehat{\beta} = W_k = \frac{Y_{jk} - Y_{ik}}{T_{jk} - T_i}$.

(ii) $S_{k-2} < 0$, $S_{k-1} = 0$, $S_k > 0$, then $\widehat{\beta} = \frac{W_{k-1} + W_k}{2}$. Note that equation (6.3) represents a unique minimum value of $D(\beta)$, and equation (6.4) corresponds to an interval estimation of minimum values. Similar to the monotone regression case, an algorithm can be developed for computing the slope estimator $\widehat{\beta}$.

□

7. NUMERICAL AND SIMULATION RESULTS AND COMPARISONS

In this section, we will present errors resulting from the application of rank regression to the software failure data sets.

TABLE 8. Analysis of Software Rank Regression (Apollo 8)

Measurement Criteria	Complete Data	Data Without Outlier Values	Data Without Leverage Values
MSE	429.4732	19.4206	73.7114
MAVD	8.2009	3.299	4.8218
MMER	7.5311	2.0602	0.1770
MMRE	1.5341	N/A	N/A
MdAR	NA	NA	NA
V(RRS)	245.7090	209.9932	243.2727

TABLE 9. Analysis of Software Rank Regression (System 40)

Mesurement Criteria	Complete Data	Data Without Outlier Values	Data Without Leverage Values
MSE	28.56	No outlier exists	24.41
MAVD	4.59	No outlier exists	4.26
MMER	0.73	No outlier exists	0.58
MMRE	0.46	No outlier exists	0.42
MdAR	4.44	No outlier exists	4.27
V(RRS)	1274.97	No outlier exists	1199.35

TABLE 10. Analysis of Software Rank Regression (Project1)

Mesurements Criteria	Complete Data	Data Without Outlier Values	Data Without Leverage Values
MSE	21.8342	21.1284	20.1449
MAVD	3.9773	3.9337	3.8370
MMER	6.7568	0.2167	0.1967
MMRE	0.9864	0.8452	0.7964
MdAR	3.8216	3.7546	3.6655
V(RRS)	1343.90	1334	1305

Examining Tables 8, 9 and 10, we can conclude that the rank regression is insensitive to outliers and to leverage values. This is in contrast to the results obtained in [15], where it has been shown that the linear regression models are very sensitive to the presence of outliers and to leverage values. Since the data are real, removing the outliers and to leverage values are not an option, and hence rank based methods seem to be more appropriate.

Simulation and other studies. Table 11 represents the output of three models: least squares regression, Monotone regression, and the rank regression by applying the three procedures on the monotonic data that is taken from [20]. The results support the point-of-view that if the data is monotonic then the monotonic regression method is the better choice for software reliability.

TABLE 11. Comparison of Models Using Different Measurement's Criteria (for a Monotonic Data)

Model	Least Squares Method	Monotone Regression	Rank Regression
MSE	3.2814	0.2011	7.1656
MAVD	1.4008	0.2829	1.8184
MMER	0.1020	0.0188	0.1175
MMRE	0.1454	0.0193	0.2450
V	36.9	116.10	99.7500
R-Square	0.9512	0.9692	0.5767
R-adj	0.9485	0.9675	0.5532

The results in Table 11 support this study for the need of the monotone regression model. Since the hypothetical data is normal, we are expecting the least squares regression model to be acceptable model for prediction. The MMRE of the three

models is smaller than 0.25, meaning that all three models are considered to be acceptable for prediction for a normal data. At the same time the error measurements of the monotonic regression model is the best among these three. The value of R-square is 0.9512 for the least squares method, and $R^2 = 0.9692$ for the monotonic regression method, while the value R-square for the rank regression method is 57.67 percent. These results are very good for the simulated Monotonic normal data. The error measurements of the rank regression illustrate an acceptable model for prediction for the data drawn from a normal distribution. In order to facilitate further study, we simulated two different normal data sets of sizes 20 and 50. There were no monotonicity assumptions.

TABLE 12. MSE and V Percentage Effect of Outlier Values ($n = 20$) for Normal Simulated Failure Data

Models	Complete Data	Data Without Outliers	Percentage Effect of Outliers
Least Square Regression	MSE = 0.0197	MSE = 0.0154	21.8274
Monotone Regression	MSE = 0.0240	MSE = 0.0188	21.6667
Rank Regression	V = 57.1451	V = 52.1208	8.7922

TABLE 13. MSE and V Percentage Effect of Outlier values ($n = 50$) for Normal Simulated Failure Data

Models	Complete Data	Data Without Outliers	Percentage Effect of Outliers
Least Square Regression	MSE = 0.0157	MSE = 0.0114	27.3885
Monotone Regression	MSE = 0.0192	MSE = 0.0129	32.8125
Rank Regression	V = 157.2260	V = 142.9994	9.0485

Looking at the percentage effect of outliers in Tables 12 and 13, even if both data sets are normal, we have better results for the rank regression model in both cases, while the monotone regression is better than the least squares regression for the data of sample size 20. The above results show that the rank regression is insensitive to outliers and is a better model even when the data is coming from a normal distribution. In order to see the effects of these methods on a skewed data, we have simulated 50 data values from a truncated skewed Laplace distribution. The following table gives the results.

TABLE 14. MSE and V% Effect of Outlier Values (TSL $n = 50$) for Simulated Truncated Skewed Laplace Failure Data of Size 50

Models	Complete Data	Data Without Outliers	Percentage Effect of Outliers
Least Square Regression	MSE = 1.2079	MSE = 0.2709	77.5726
Monotone Regression	MSE = 3.6117	MSE = 1.2747	64.7063
Rank Regression	V = 223.3501	V = 107.0164	52.0858

Again, looking at the percentage effect, we see that the rank regression outperforms the other methods.

8. CONCLUSION

In this paper two non-parametric approaches were proposed, the monotone regression and the rank robust regression models. The results illustrate that if we relax the assumptions of the least square procedure described in [15], the monotone regression and rank regression models perform well for: System 40, Project 1, and Project 5 failure data sets. Both of these models are capable of predicting the next failures with smaller error measurements. They are also less sensitive to outliers and leverage values of the time between failures, in comparison with the parametric models. The percentage effects of outliers and leverage values illustrate that the rank regression model is less sensitive to outliers and leverage values than either the least squares regression or the monotone regression.

REFERENCES

[1] David A. Belsley, Edwin Kuh, and Roy E. Welsch, *Regression diagnostics: identifying influential data and sources of collinearity*, Wiley Series in Probability and Mathematical Statistics, John Wiley & Sons, New York-Chichester-Brisbane, 1980.

[2] W. J. Conover, *Practical nonparametric statistics*, second ed., Wiley Series in Probability and Statistics, John Wiley & Sons, New York, 1980.

[3] S. D. Conte, H. E. Dunsmore, and V. Y. Shen, *Software engineering metrics and models*, Benjamin/Cummings Pub. Co., Menlo Park, Calif., 1986.

[4] Frank R. Hampel, Elvezio M. Ronchetti, Peter J. Rousseeuw, and Werner A. Stahel, *Robust statistics: The approach based on influence functions*, Wiley Series in Probability and Statistics, John Wiley & Sons, New York, 1986.

[5] T. J. Hastie and R. J. Tibshirani, *Generalized additive models*, Statist. Sci. **1** (1986), no. 3, 297–318.

[6] ———, *Generalized additive models*, Monographs on Statistics & Applied Probability, vol. 43, Chapman and Hall/CRC, London–New York, 1990.

[7] Thomas P. Hettmansperger and Joseph W. McKean, *A robust alternative based on ranks to least squares in analyzing linear models*, Technometrics **19** (1977), no. 3, 275–284.

[8] Myles Hollander and Douglas A. Wolfe, *Nonparametric statistical methods*, second ed., Wiley Series in Probability and Statistics, John Wiley & Sons, New York, 1999.

[9] Peter J. Huber, *Robust regression: asymptotics, conjectures and Monte Carlo*, Ann. Statist. **1** (1973), no. 5, 799–821.

[10] Peter J. Huber, *Robust statistics*, Wiley, New York, 1981.

[11] Ronald L. Iman and W. J. Conover, *The use of the rank transform in regression*, Technometrics **21** (1979), no. 4, 499–509.

[12] Louis A. Jaeckel, *Estimating regression coefficients by minimizing the dispersion of the residuals*, Ann. Math. Statist. **43** (1972), no. 5, 1449–1458.

[13] Jana Jureckova, *Nonparametric estimate of regression coefficients*, The Annals of Mathematical Statistics **42** (1971), no. 4, 1328–1338.

[14] B. A. Kitchenham, S. G. MacDonell, L. Pickard, and M. J. Shepperd, *What accuracy statistics really measure*, IEE Proceedings — Software **148** (2001), no. 3, 81–85.

[15] A. M. Mostafa, K. M. Ramachandran, and A. N. V. Rao, *Regression approach to software reliability models*, Int. J. Pure Appl. Math. **26** (2006), no. 2, 217–240.

[16] John D. Musa, *Software failure data*, site is not currently available.

[17] I. Myrtveit, E. Stensrud, and M. Shepperd, *Reliability and validity in comparative studies of software prediction models*, IEEE Trans. Software Eng. **31** (2005), no. 5, 380–391.

[18] Henry W. Roberts, *Predicting the performance of software systems via the power law process*, Ph.D. dissertation, University of South Florida, Tampa, FL, 2000.

[19] Peter J. Rousseeuw and Annick M. Leroy, *Robust regression and outlier detection*, Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics, John Wiley & Sons Inc., New York, 1987.

[20] Thomas P. Ryan, *Modern regression methods*, Wiley, New York, 1997.

[21] Stanley Sawyer, *Linear rank regression: robust estimation of regression parameters*, 2003, revised 2009.

[22] Nalina Suresh, *Modeling and analysis of software reliability*, Ph.D. dissertation, University of South Florida, Tampa, FL, 1992.

[23] H. Theil, *A rank-invariant method of linear and polynomial regression analysis. I*, Nederl. Akad. Wetensch., Proc. **53** (1950), 386–392 = Indagationes Math. 12, 85–91 (1950).

[24] ———, *A rank-invariant method of linear and polynomial regression analysis. II*, Nederl. Akad. Wetensch., Proc. **53** (1950), 521–525 = Indagationes Math. 12, 173–177 (1950).

[25] ———, *A rank-invariant method of linear and polynomial regression analysis. III*, Nederl. Akad. Wetensch., Proc. **53** (1950), 1397–1412 = Indagationes Math. 12, 467–482 (1950).