Cairo University

The 56th Annual International

Conference of Data science

Department of Applied Statistics & Econometrics

4 – 6, Dec, 2023

Index

Department of Applied Statistics & Econometrics

A New Regression Model for Transmuted Weibull with application

Mostafa Abd-elwahab Hamed[1], Salah Mahdy Ramadan[2], Amal Mohamed Abdel Fattah[3]

[1]Department of Applied Statistics and Econometrics, Faculty of Graduates Studies for Statistical Research (FSSR), Cairo University, Egypt. Faculty of commerce, Aswan University

[2]Department of Applied Statistics and Econometrics, Faculty of Graduates Studies for Statistical Research (FSSR), Cairo University, Egypt.

[3]Department of Applied Statistics and Econometrics, Faculty of Graduates Studies for Statistical Research (FSSR), Cairo University, Egypt.

Corresponding author. Abdelwahabmostafa35@gmail.com

1.Abstract

In this research paper, a new regression model for the transmuted Weibull distribution was proposed, and the parameters of the new model were estimated by using the Maximum likelihood method, the different types of residuals were calculated for the new model, such as, Cox and Snell Residual, Pearson Residuals, Deviance component Residual, and the martingale residual. and then the proposed model was applied to a set of real data, and the proposed model was also compared with some other models such as kumaraswamy Lindley regression model and MG weibull regression model.

Key words: weibull distribution, transmutation map, regression approach, survival analysis and residual analysis.

2.Introduction:

Regression analysis is statistical modeling that can be used in financing, marketing, investing and other fields that utilized to infer the strength and direction of relationships between dependent variable and one or more independent variables. The dependent variable Y is also known as response variable or outcome, and the independent variables X_k where k=(1,2,…,p) as predictors, explanatory variables, or covariates. Regression analysis, More precisely, attempts identifying the mathematical formula that explains Y in terms of X as, Y=f(x). **Sarstedt. M., & Mooi. E.(2014)**. Transmutation maps are generally a handy tool for creating a new distributions, especially; survival ones. Transmutation maps, defined by **Shaw and Buckley (2009),** are the functional composition of one distribution's inverse cumulative distribution (quantile) function and one distribution's cumulative distribution function.

Recently; some studies involving the type of quadratic rank transmutation map (QRTM) and its application such as survival analysis. by applying QRTM **Aryal and Toskes (2011)** proposed Transmuted Weibull distribution, **Aryal and Tsokos (2009)** presented a generalization of the extreme value distribution applied this new distribution to analyze snowfall data at Midway Airport in the US state of Illinois, **Merovci, F. (2013a)** introduced transmuted Lindley distribution and applied the proposed distribution to an uncensored data of random sample from 128 bladder cancer patients, **Merovci, F. (2013b)** generalized the Rayleigh distribution and applied to Real data of nicotine measurements obtained from several brands of cigarettes for the year 1998. **Louzada and Granzotto, (2016)** proposed the transmuted log-logistic model and the regression one in the study of the time until the first calve of a polled Tabapua race. **Dey. S, et al. (2021)** stated that with increasing the variability of applications, the standard distributions were not sufficient for modeling complicated phenomena, therefore the need to generalize standard distributions is required. It is necessary to emphasis transmutation map as generalization tool for obtaining new distributions that are more flexible, the new generalized distributions have the capacity for modeling complicated phenomena more accurately. These generalizations made a considerable advancement in the direction of constriction flexile distributions in order to facilitate good modeling of lifetime data. Recently some generalizations of Weibull distribution including exponentiated Weibull, modified Weibull, extended Weibull and transmuted Weibull that can be derived by using QRTM. The generalizations and modifications of Weibull distribution have been approved by **Cordeiro, G. M. et al (2011), Elbatal. I and Aryal. G. (2013), Aryal and Toskes (2013), Eltehiwy, et al. (2013), Merovice, et al. (2013), Afify. A. Z, et al. (2014), Al-Kadim, K. A. et al. (2017), Afify. A. Z, et al. (2017), Ahmed. Z. Afify, et al. (2018), Nofal, Z. M., et al. (2018), Khan. (2019).**

In this paper we introduced a New Regression Model for Transmuted Weibull with application. The background of transmuted weibull distribution will be presented at section(3) are divided into three subsections including the log transmuted regression weibull, inference about the parameters of regression model by using maximum likelihood estimation, and finally subsection is residual analysis. At section (3) the numerical experiment will be presented and classified to several subsections involving simulation study in order to assess the performance of T-W regression model, applications of T-W to real data set, goodness of fit to ensure that real data more fitted to the proposed model, and then comparing the T-W regression model with kumaraswamy Lindley regression model and MG weibull regression

model based on some goodness of fit criteria, and also the global influence will be presented to shows the sensitive analysis for T-W regression model.

3. Transmuted Weibull distribution:

3.1 Background

The Weibull distribution is play very important role for modeling lifetime data in many filed for example medicine, biology, engineering and finance, a very small amount of the massive applications of Weibull model. let's random variable x is said to have Weibull distribution with 2 parameters α and β if its probability function (pdf) is given by **(Aryal and Tsokos 2009):**

$$g(x) = \frac{\alpha}{\beta^{\alpha}} \; x^{\alpha-1} \, e^{-(\frac{x}{\beta})^{\alpha}} \qquad x > 0 \; and \; \alpha, \beta > 0 \tag{1}$$

The coressponding cumulative distribution function of Weibull distribution (CDF) is given by:

$$G(X) = 1 - e^{-(\frac{x}{\beta})^{\alpha}} \qquad x > 0 \; and \; \alpha, \beta > 0 \tag{2}$$

By using methodology of QRTM shown by Shaw and Buckley (2007), **Aryal and Toskes (2011)** proposed transmuted weibull distribution based on weibull distribution as baseline distribution. Let X is said to have transmuted distribution if its cumulative distribution (CDF) is given by:

$$F(x) = (1 + k) \, G(x) - k[G(x)]^2 \tag{3}$$

Where $k \in [-1,1]$, *F(x)* is CDF of transmuted distribution, And *G(x)* is CDF of parent distribution. When K=0, the Transmuted distribution is the same as base distribution. The corresponding probability density function (pdf) is given by:

$$f(x) = g(x)[(1 + k) \; - 2kG(x)] \tag{4}$$

Hence; the pdf and CDF of Transmuted weibull distribution is given by respectively:

$$f(x) = \frac{\alpha}{\beta^{\alpha}} \; x^{\alpha-1} \, e^{-(\frac{x}{\beta})^{\alpha}} \left[(1 + k) \; - 2k(1 - e^{-(\frac{x}{\beta})^{\alpha}})\right] \tag{5}$$

And

$$\boldsymbol{F(x) = (1+k)\left(1-e^{-(\frac{x}{\beta})^{\alpha}}\right) - k\left[1-e^{-(\frac{x}{\beta})^{\alpha}}\right]^{2}} \quad \textbf{(6)}$$

According to **Aryal and Tsokos (2011);** Transmuted Weibull distribution is extended model for analyzing data that is more complex and it generalizes some of widely used distributions. From equation (5) note that when parameters vary, there are special cases from Transmuted Weibull distribution will be placed as following:

- When $\alpha = 1$, transmuted exponential distribution (skew exponential) with two parameters β and k will be placed. Shaw, et al. (2009).
- By butting $k = 0$, Weibull distribution is generated which is the base distribution.

When $\alpha = 1$ and $k = 1$, the Exponential distribution with parameter $(\frac{\beta}{2})$ is given.

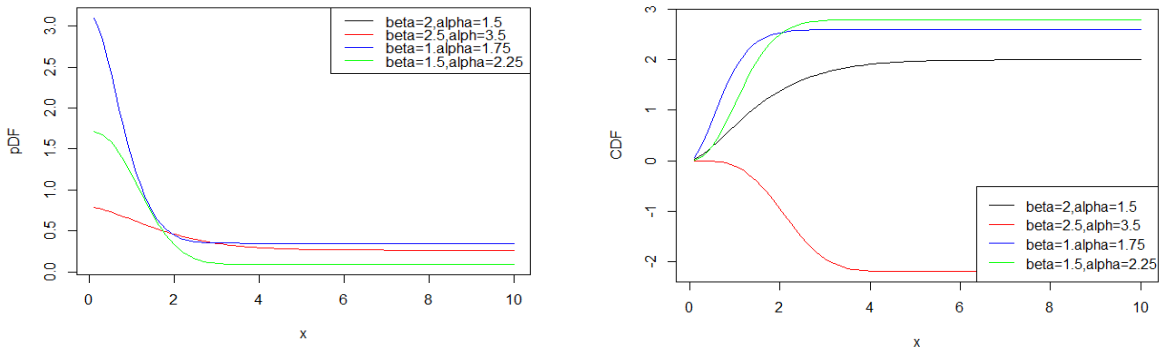


Figure (1) the pdf and CDF of the transmuted weibull distribution at different value with k=0.6

3.2 The Log Transmuted Weibull (T-W) Regression Model:

The log-location-scale regression models are popular models to analyze the censored response variable with some covariates. In the last decade, researchers have introduced flexible location scale regression models to analyze the different characteristics of the data sets. The important paper on location-scale can be cited as follows: log-generalized Transmuted-Weibull, Suppose x is a random variable following the **T-W** density function in equation (5). and y is defined $y = logx$, $x = e^{y}$, $\beta = e^{\mu}$, $\alpha = \frac{1}{\sigma}$, It is easy to verify that the density function of y reduces to:

$$f(y) = (|J|).f^{-1}(x) \quad \textbf{(7)}$$

To obtain the value of the Jacobean transformation, the differential is with respect to x and we get $\frac{dy}{dx} = \frac{1}{x}$ and then substituting in equation (7).

$$\boldsymbol{f(y,\mu,\sigma) = \frac{1}{\sigma} exp\left(\frac{y-\mu}{\sigma}\right) * e^{-exp\left(\frac{y-\mu}{\sigma}\right)} * \left[(1+k) - 2k(1-e^{-exp\left(\frac{y-\mu}{\sigma}\right)})\right]} \quad \textbf{(8)}$$

By substituting in equation (8), let $\boldsymbol{Z = \frac{y-\mu}{\sigma}}$ **&** $\boldsymbol{\sigma Z = y-\mu}$ **&** $\boldsymbol{y = \mu + \sigma Z}$

$$\boldsymbol{f(z) = \frac{1}{\sigma} exp(z) * e^{-exp(z)} * \left[(1+k) - 2k(1-e^{-exp(z)})\right]} \quad \textbf{(9)}$$

where $-\infty < \mu < \infty$ and σ > 0. The parameters μ and σ are the location and scale parameters of the **T-W** distribution. Hereafter, the pdf in (9) is called as log-transmuted weibull regression model. The corresponding survival function is given by:

$$\boldsymbol{S(y,\mu,\sigma) = 1-(1+k)\left(1-e^{-exp\left(\frac{y-\mu}{\sigma}\right)}\right) + k\left[1-e^{-exp\left(\frac{y-\mu}{\sigma}\right)}\right]^2} \quad \textbf{(10)}$$

Consider the following regression model:

$$\boldsymbol{y_i = x_i^T \beta + \sigma_i z_i} \quad \textbf{(11)}$$

where the response variable y_i has the density function is given in (8). The covariates are linked to location of y_i with identity link function $\mu = x^T\beta$.

Where $X = (x_1, x_2, \ldots\ldots\ldots\ldots x_p)^T$ is the model matrix consists of the observations of independent variables, and $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots\ldots\ldots \beta_p)$ is the unknown regression coefficients.

3.3.Maximum Likelihood Estimation For T-W Regression Model:

Let the random sample$(y_1, y_2, \ldots, y_n)$ follow a T-W distribution and the response variable is defined as $y_n = min(x_i, c_i)$. Where c_i is the censoring time and x_i is the observed lifetime. Assume that the censoring times and lifetimes are independent. Let *F* and *C* are the sets representing the observed lifetimes and censoring times. The general formulation of the log-likelihood function for the model given in (8) is given by:

$$L(\theta) = \sum_{i=1}^{n} \delta_i \log\big(f(y)\big) + \sum_{i=1}^{n} (1-\delta_i)\log(s(y)) \quad \textbf{(12)}$$

$$\log\big(f(y)\big) = \delta_i log\left[\frac{1}{\sigma}\left(\frac{y-\mu}{\sigma}\right) * \ e^{-exp\left(\frac{y-\mu}{\sigma}\right)} \ * \left[(1+k) \ - 2k(1-e^{-exp\left(\frac{y-\mu}{\sigma}\right)})\right]\right]$$

$$\sum_{i=1}^{n} log\big(f(y)\big) = \delta_i \left[\sum_{i=1}^{n} log\left(\frac{1}{\sigma}\right) + \sum_{i=1}^{n} log\left(\frac{y-\mu}{\sigma}\right) - \sum_{i=1}^{n} exp\left(\frac{y-\mu}{\sigma}\right) + \sum_{i=1}^{n} log\left[(1+k) \ - 2k(1-e^{-exp\left(\frac{y-\mu}{\sigma}\right)})\right]\right]$$

$$\sum_{i=1}^{n} log\big(f(y)\big) = \delta_i \left[\sum_{i=1}^{n} log\left(\frac{1}{\sigma}\right) + \sum_{i=1}^{n} log\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right) - \sum_{i=1}^{n} exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right) + \sum_{i=1}^{n} log\left[(1+k) \ - 2k(1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)})\right]\right]$$

$$\sum_{i=1}^{n} \log(s(y)) = (1-\delta_i) \sum_{i=1}^{n} log\left[1-(1+k)\,(1-e^{-exp\left(\frac{y-\mu}{\sigma}\right)}) + k\left[1-e^{-exp\left(\frac{y-\mu}{\sigma}\right)}\right]^2\right]$$

$$\sum_{i=1}^{n} \log(s(y)) = (1-\delta_i) \sum_{i=1}^{n} log\left[1-(1+k)\,(1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}) + k\left[1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}\right]^2\right]$$

To estimate the coefficients of the proposed regression model, we deferential equation No. (12) with respect to the regression coefficients.

First; Estimation of parameter $\boldsymbol{\beta_0}$:

By using partials' derivatives of log likelihood function of Transmuted Weibull at (12) with respect to $\boldsymbol{\beta_0}$ as the following:

$$\frac{\partial L(\theta)}{\partial \beta_0} = \sum_{i=1} \frac{\partial \log\big(f(y)\big)}{\partial \beta_0} + \sum_{i=1}^{n} \frac{\partial \log(s(y))}{\partial \beta_0}) = 0$$

$$\sum_{i=1}^{n} log\big(f(y)\big) = (\delta_i) \left[\sum_{i=1}^{n} log\left(\frac{1}{\sigma}\right) + \sum_{i=1}^{n} log\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right) - \sum_{i=1}^{n} exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right) + \sum_{i=1}^{n} log\left[(1+k) \ - 2k(1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)})\right]\right]$$

$$\frac{\partial \sum_{i=1}^{n} log\big(f(y)\big)}{\partial \beta_0} = (\delta_i) \sum_{i=1}^{n} \left(\frac{\sigma}{y-\beta_0-x\beta_1}\right) * \left(\frac{-1}{\sigma}\right) + (\delta_i) \sum_{i=1}^{n} exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right) * \left(\frac{-1}{\sigma}\right) + (\delta_i) \sum_{i=1}^{n} \left[\frac{+\frac{2}{\sigma} k(e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)} * -exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right))}{(1+k) \ - 2k(1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)})}\right]$$

$$\frac{\partial \sum_{i=1}^{n} \log(s(y))}{\partial \beta_0}$$

$$= (1-\delta_i)\ \frac{(1+k)e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}\left(-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)\right)\frac{-1}{\sigma}-\frac{1}{\sigma}e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)} * exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right) *}{\sum_{i=1}^{n} log\left[1-(1+k)\ (1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)})+1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}\right]}$$

second; Estimation of parameter $\boldsymbol{\beta_1}$:

By using partials' derivatives of log likelihood function of Transmuted Weibull at (12) with respect to $\boldsymbol{\beta_1}$ as the following:

$$\frac{\partial L(\theta)}{\partial \beta_1} = (\delta_i)\sum_{i=1}\frac{\partial \log\left(f(y)\right)}{\partial \beta_1} + (1-\delta_i)\sum_{i=1}^{n}\frac{\partial \log(s(y)}{\partial \beta_1})=0$$

$$\frac{\partial log\left(f(y)\right)}{\partial \beta_1} = \sum_{i=1}^{n}\left(\frac{(\delta_i)}{y-\beta_0-x\beta_1}\right) * \left(\frac{-x}{\sigma}\right) + \sum_{i=1}^{n} exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right) * \left(\frac{-x}{\sigma}\right)$$

$$+ \sum_{i=1}^{n}\left[\frac{+\frac{2x}{\sigma}k(e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)} * -exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}{(1+k)\ -2k(1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}}\right]$$

$$\frac{\partial \sum_{i=1}^{n} \log(s(y)}{\partial \beta_1}$$

$$= (1-\delta_i)\frac{(1+k)e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}\left(-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)\right)\frac{-x}{\sigma}-\frac{x}{\sigma}e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)} * exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right) *}{\sum_{i=1}^{n} log\left[1-(1+k)\ (1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)})+1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}\right]}$$

third; Estimation of parameter σ:

By using partials' derivatives of log likelihood function of Transmuted Weibull at (12) with respect to σ as the following:

$$\frac{\partial L(\theta)}{\partial \sigma} = \sum_{i=1}\frac{\partial \text{og}\left(f(y)\right)}{\partial \sigma} + \sum_{i=1}^{n}\frac{\partial \log(s(y)}{\partial \sigma})=0$$

$$\frac{\partial \text{og}\left(f(y)\right)}{\partial \sigma} = \sum_{i=1}^{n}\left(\frac{(\delta_i)}{y-\beta_0-x\beta_1}\right) * \left(\frac{-1}{\sigma}\right) + \sum_{i=1}^{n} exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right) * \left(\frac{-(\delta_i)}{\sigma^2}\right)$$

$$+ \sum_{i=1}^{n}\left[\frac{+\frac{2}{\sigma^2}k(e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)} * -exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}{(1+k)\ -2k(1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}}\right](-\frac{(\delta_i)}{\sigma})$$

$$\frac{\partial \log(s(y))}{\partial \sigma}$$

$$= (1-\delta_i)\sum_{i=1}^{n} \frac{\frac{1}{\sigma^2} * \left((1+k)\,(1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)} * exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right) + 2k\left[1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}\right] * -exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)\right)}{\left[1-(1+k)\,(1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}) + k\left[1-e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}\right]^2\right]}$$

Therefore, the model of Transmuted weibull in (12) can be written as linear log location-scale regression model as following:

$$log\mu(y_i) = \beta_0 + \beta_1 x + \sigma_i z_i \qquad \textbf{(13)}$$

where y_i is the dependent variable. x is the independent variable , β_0 is the intercept, β_1 is the slope in which the relation between dependent and the independent variables are determined, and $\sigma_i z_i$ is the error term.

3.4.Residual analysis:

Residual analysis is an important step of any regression analysis to check the sufficiency of the fitted model. If the fitted model is accurate for the data used, the residuals have to meet the distributional assumptions. Here, we used four kinds of residuals modified under T-W regression model including the Martingale Residual, Deviance Component Residual, Pearson Residuals and Cox and Snell Residual.

3.4.1 the Martingale Residual:

Is much used in the counting process These residuals are a symmetric and take maximum values (+1) and minimum values (-∞) **(Alamoudi, et al. 2017)**: .we defined the martingale residual as:

$$\boldsymbol{r_{Mi} = \delta_i + \left(\int_0^y h(u)\, du \right)}$$

where $\boldsymbol{\delta_i = \begin{bmatrix} 0 & the\ i^{th} observation\ is\ censored \\ 1 & the\ i^{th} observation\ is\ uncensored \end{bmatrix}}$

As we known $\int_0^y \boldsymbol{h(u)\ du} = \boldsymbol{log}\ [\boldsymbol{s(y_i)}]$

Then, the martingale residual can be reduced to:

$$\boldsymbol{r_{Mi} = \delta_i + log\ [s(y_i)]}$$

Here ,the martingale residual for the log T-W regression model takes the following form:

$$r_{Mi} = \begin{cases} 1 + \left[log \left[1 - (1+k)\left(1 - e^{-exp\left(\frac{y-\beta_0 - x\beta_1}{\sigma}\right)}\right) + k\left[1 - e^{-exp\left(\frac{y-\beta_0 - x\beta_1}{\sigma}\right)}\right]^2 \right] \right] & , \delta_i = 1 \\ log \left[1 - (1+k)\left(1 - e^{-exp\left(\frac{y-\beta_0 - x\beta_1}{\sigma}\right)}\right) + k\left[1 - e^{-exp\left(\frac{y-\beta_0 - x\beta_1}{\sigma}\right)}\right]^2 \right] & , \delta_i = 0 \end{cases}$$

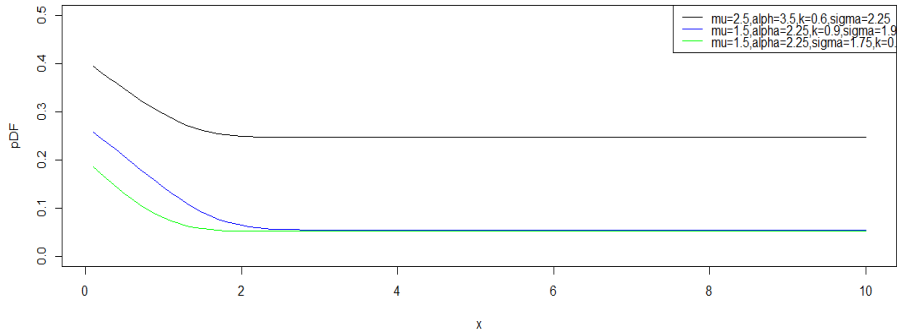


Figure (3.9)The martingale residual for transmuted weibull regression model

3.4.2.Deviance component Residual:

This residue was suggested to make the martingale residual more symmetric around zero **(Alamoudi, et al. 2017)**. The deviance component for the parametric regression model is given:

$$\hat{r}_{Di} = sign\,(\hat{r}_{Mi})\,\{-2\,[\hat{r}_{Mi} + log\,(1 - \hat{r}_{Mi})]\}^{1/2}$$

Where r_{Mi} is the martingale residual. * sign () function is a function that drives the (***+1***) values if the argument is positive and (***-1***) is negative. The deviance component residual for the Transmuted Weibull model is given by:

$$\hat{r}_{Di} = \begin{cases} \text{sign}\,(\hat{r}_{Mi}) \left\{ -2 \left[\begin{array}{c} 1 + \left[log \left[1 - (1+k)\left(1 - e^{-exp\left(\frac{y-\beta_0 - x\beta_1}{\sigma}\right)}\right) + k\left[1 - e^{-exp\left(\frac{y-\beta_0 - x\beta_1}{\sigma}\right)}\right]^2 \right] \right] \\ +log \left[1 - \left(1 - \left[log \left[1 - (1+k)\left(1 - e^{-exp\left(\frac{y-\beta_0 - x\beta_1}{\sigma}\right)}\right) + k\left[1 - e^{-exp\left(\frac{y-\beta_0 - x\beta_1}{\sigma}\right)}\right]^2 \right] \right] \right) \right] \end{array} \right\}^{1/2} & , \delta_i = 1 \\ \text{sign}\,(\hat{r}_{Mi}) \left\{ -2 \left[\begin{array}{c} 1 + \left[log \left[1 - (1+k)\left(1 - e^{-exp\left(\frac{y-\beta_0 - x\beta_1}{\sigma}\right)}\right) + k\left[1 - e^{-exp\left(\frac{y-\beta_0 - x\beta_1}{\sigma}\right)}\right]^2 \right] \right] \\ +\log \left(1 - 1 - \left[log \left[1 - (1+k)\left(1 - e^{-exp\left(\frac{y-\beta_0 - x\beta_1}{\sigma}\right)}\right) + k\left[1 - e^{-exp\left(\frac{y-\beta_0 - x\beta_1}{\sigma}\right)}\right]^2 \right] \right] \right) \end{array} \right\}^{1/2} & , \delta_i = 0 \end{cases}$$

3.4.3.Pearson Residuals:

are used to detect outliers. It depends on the idea of subtracting the mean and dividing by the standard deviation. The Pearson Residual knows the bounded regression of the inverse of the exponential distribution as follows:

$$r_i = \frac{y_i - \hat{\mu}}{\sqrt{\widehat{var(y)}}} \quad \textbf{(14)}$$

3.4.4.Cox and Snell Residual:

Cox and Snell (1968) residual defined as follows:

$$e_i = -lin[1 - F(y_i, \beta)] \quad \textbf{(15)}$$

Substituting in equation No.(15) for the value of the Cumulative Function, we get the Cox and Cox and Snell residual for Transmuted Weibull regression model as the following:

$$\boldsymbol{e_i = -lin\left[1 - \left[(1+k)\left(1 - e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}\right) - k\left[1 - e^{-exp\left(\frac{y-\beta_0-x\beta_1}{\sigma}\right)}\right]^2\right]\right]}$$

4. Numerical Experiment:

4.1. simulation study:

In this section, a simulation study is given to evaluate the performance of coefficients for the proposed regression model according to Transmuted Weibull regression model. All results were obtained from 1,000 Monte Carlo replications. The following steps is used in Monte Carlo simulation.

1. Generate the data from this model with $\beta_0 = 0.8, \beta_1 = 1.4, K = 1 \; and \; \sigma = 1$
2. Generate $y \sim T - W(\mu, \sigma)$ **where** $\boldsymbol{\mu = \beta_0 + \beta_1 x}$.
3. Generate $x \sim U(0,1)$.
4. Generate $Z \sim U(0,1)$.
5. The sample sizes are taken as n = 20, 100, 200.
6. Each sample size is replicated 1000 times.
7. For each generated sample sizes, the biases, (AS) and (MSE) are evaluates at level the three censoring rates (20%, 30%, 40%).

The simulation results are reported in Table (1),Table(2) and Table (3). As seen from the results, the estimated biases, average of estimates (AS) and mean square error (MSEs) are near the desired value, zero.

Table (1) Simulation Study Results of (T-W) Regression model at Censoring Rate=20%

Censoring rate= 0.20	n=20			n=50			n=100		
Parameter	**AE**	**Bias**	**MSE**	**AE**	**Bias**	**MSE**	**AE**	**Bias**	**MSE**
$\widehat{K}$	2.351	0.351	0.626	2.210	0.210	0.524	2.023	0.023	0.241
$\hat{\beta}_0$	0.976	0.176	0.334	0.675	0.125	0.210	0.741	0.112	0.110
$\hat{\beta}_1$	1.923	0.477	0.830	1.624	0.211	0.622	1.522	0.110	0.322
$\hat{\sigma}$	1.256	0.256	0.326	1.130	0.115	0.245	1.012	0.012	0006

Table (1); explains that **at censoring rate =0.20**, Average Estimation (AE), Bias, and Mean square error (MSE) are estimated at different sample size n=20, n=50, n=100 for different parameters ($\widehat{K}$, $\hat{\beta}_0$, $\hat{\beta}_1$, $\hat{\sigma}$). The previous table indicates that the estimated AE, Bias, MSE are tends to desired value (zero) as be shown in figure (1). When the sample size increases, the estimated AE, Bias, MSE decreases, this mean that the difference between true and estimated values for parameters declines up to desired value (zero).

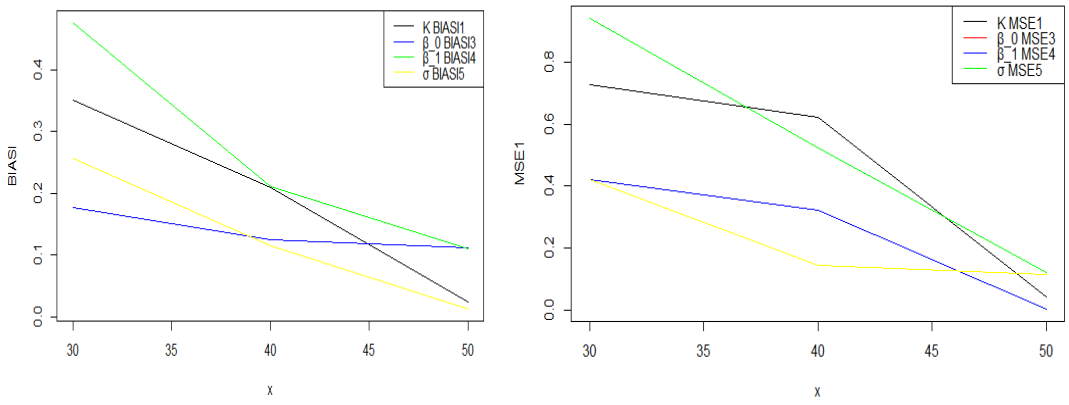


Figure (1)The Bias and MSE for different sample sizes at censoring rate (20%)

Table (2), Simulation Study Results of (T-W) Regression model at Censoring rate 30%

Censoring rate= 0.30	n=20			n=50			n=100		
Parameter	**AE**	**Bias**	**MSE**	**AE**	**Bias**	**MSE**	**AE**	**Bias**	**MSE**
$\widehat{K}$	2.634	0.634	0.816	2.423	0.423	0.629	2.042	0.042	0.016
$\hat{\beta}_0$	0.843	0.312	0.147	0.712	0.213	0.131	0.613	0.113	0.013
$\hat{\beta}_1$	1.842	0.441	0.921	1.552	0.141	0.625	1.343	0.131	0.016

$\hat{\sigma}$	1.239	0.239	0.412	1.220	0.020	0.312	1.125	0.021	0.022

Table (2), Average Estimation (AE), Bias, and Mean square error (MSE) are estimated but at **censoring rate =0.30** with different sample size n=20, n=50, n=100 for different parameters ($\widehat{K}$, $\hat{\beta}_0$, $\hat{\beta}_1$, $\hat{\sigma}$). Also as the sample size increases, the estimated AE, Bias, MSE decreases, this mean that the difference declines up to desired value (zero) and this can be explained by the figure (2) that shown the patterns of relationships between the different sample sizes and the estimated AE, Bias, MSE.

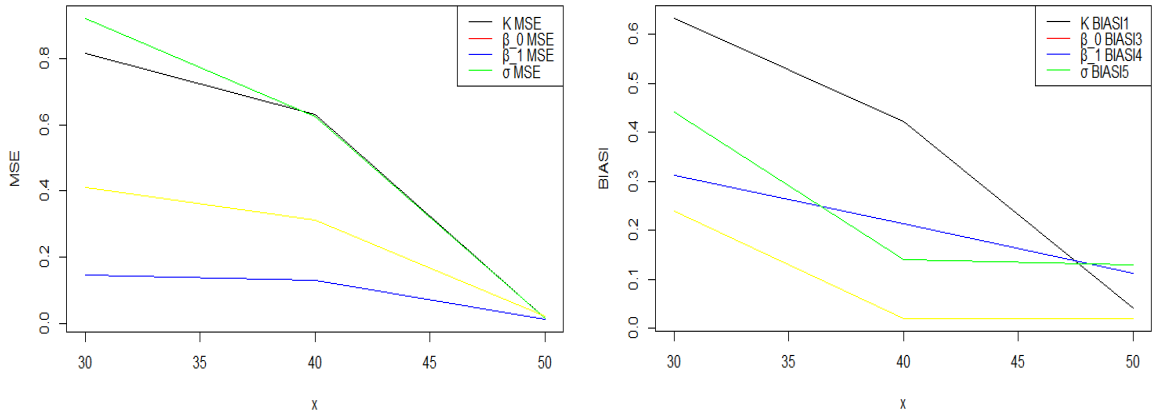


Figure (2) The MSE and Bias for different sample sizes at censoring rate (30%)

It is clear from the previous figure (2) that the mean square error and bias estimators goes to zero when the sample size increases.

Table (3) Simulation Study Results of (T-W) Regression model at censoring rate 40%

Censoring rate 0.40	n=20			n=50			n=100		
Parameter	**AE**	**Bias**	**MSE**	**AE**	**Bias**	**MSE**	**AE**	**Bias**	**MSE**
$\widehat{\boldsymbol{K}}$	2.456	0.456	0.562	2.342	0.342	0.413	2.023	0.023	0.004
$\widehat{\boldsymbol{\beta}}_{\boldsymbol{0}}$	0.842	0.242	0.423	0.624	0.024	0.462	0.321	0.321	0.023
$\widehat{\boldsymbol{\beta}}_{\boldsymbol{1}}$	0.795	0.195	0.423	0.675	0.075	0.049	0.657	0.057	0.054
$\widehat{\boldsymbol{\sigma}}$	1.423	0.423	0.162	1.125	0.125	0.625	1.025	0.025	0.004

Through Table (3), the Average Estimation (AE), Bias, and Mean square error (MSE) are estimated but at **censoring rate =0.40** with different sample sizes. Also the results of estimators ensure that there is negative relationship between the sample size and estimated AE, Bias, MSE.

Table (4) coefficient of regression at different sizes of sample with different rates

	Rate20%			Rate30%			Rate50%		
	n=20	n=50	n=100	n=20	n=50	n=100	n=20	n=50	n=100
$\widehat{\boldsymbol{\beta}}_{\boldsymbol{0}}$	0.976	0.675	0.741	0.843	0.712	0.613	0.242	0.624	0.321
$\widehat{\boldsymbol{\beta}}_{\boldsymbol{1}}$	1.923	1.624	1.522	1.842	1.552	1.343	0.195	0.675	0.657
AIC	125.24	123.85	109.25	108.56	99.96	80.54	78.39	66.97	52.65
BIC	124.37	122.56	115.86	114.67	97.35	83.96	82.42	78.54	53.54
R^2	0.623	0.692	0.763	0.793	0.832	0.893	0.932	0.972	0.983

-Table (4); illustrates the estimated values for coefficient of regression at different sample sizes 20, 50 and 100 under different censoring rates = 20%, 30% and 50%.

- Also noticed that coefficient of determination R^2 increases when the censoring rates and sample size increases. At rate 50% with sample size 100, it is observed that R^2 =.983 at higher level than other rates and sample size as be shown in figure (3).

- also it is observed from the previous table that goodness of fit criteria AIC, BIC declining with increasing censoring rates and sample size as be shown in figure (3).

-it is concluded that the higher the censoring rates and sample size, the higher the quality of regression model for the real data sets.

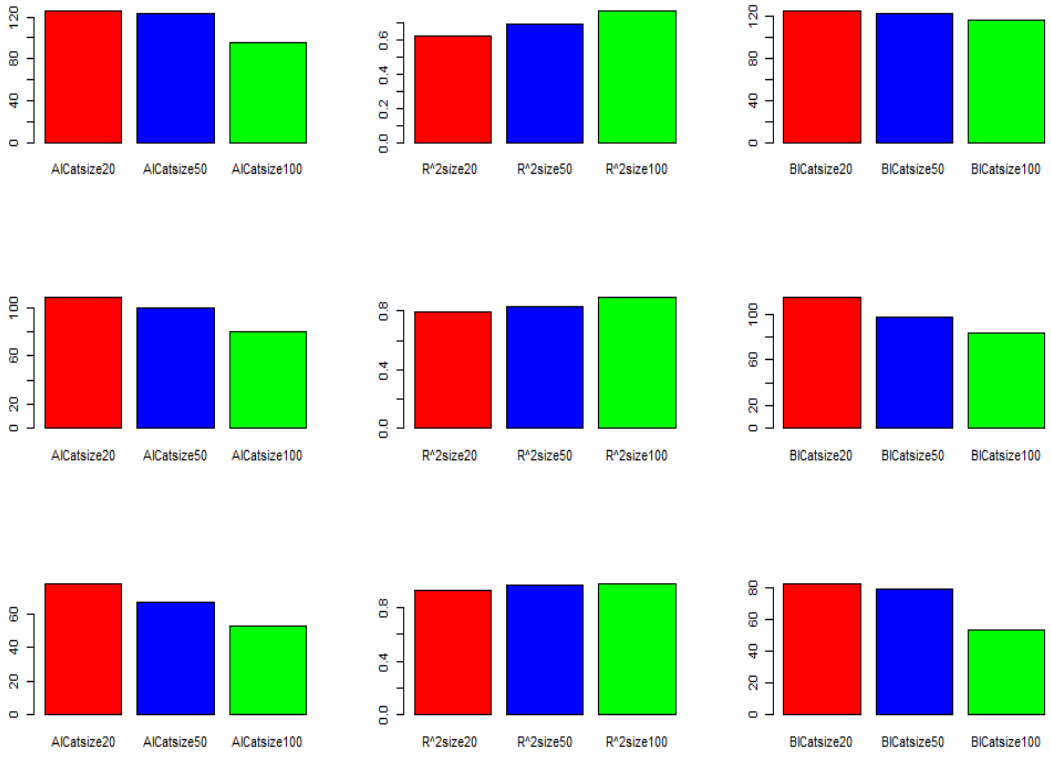


Figure (3) Some statistical criteria to judge the quality of the model

From the previous figure(3), we notice that the coefficient of determination reaches its maximum value when the sample size increases, reaching .983 when the sample size was large, and that the statistical criteria through which the model was evaluated decrease with the increase in sample size. Therefore, it is necessary to use the proposed regression model with large samples.

4.2 Applications To Real data:

This section provides the applications of Transmuted weibull regression model on real data sets by using some accounting items. The accounting data involving one dependent variable refer to bad debt rate (y), and three independent variables x_1 refer to the liquidity ratio, x_2 refer to the quick liquidity ratio, x_3 refer to Rate of loan maturities granted to the company.

4.2.1 Descriptive statistics:

Through the following table, we notice that the bad debt rate had a mean value of **.291429**, that the median had a value of **.22**, that the first quartile had a value of **.09**, and the third quartile had a value of **.46**.,and the liquidity ratio had a mean value **1.735831**, that the median had a value of **1.065453**, that the first quartile had a value of **.958781**, and the third quartile had a value of **1.582895**. and the quick liquidity had a mean value **.397745**, that the median had a value of **.104810**, that the first quartile had a value of **.054675**, and the third quartile had a value of **.216721**. and, the Rate of loan maturities had a mean value **.844688**, that the median had a value of **.657652**, that the first quartile had a value of **.068553**, and the third quartile had a value of **.899976.**

Table (5) Summary of real data set

statistics	y	X_1	X_2	X_3
mean	.291429	1.735831	.397745	.844688
median	.220000	1.065453	.104810	.657652
min	.0100	.0239	.0117	.0000
Q1	.090000	.958781	.054675	.068553
Q3	.460000	1.582895	.216721	.899976

-The Relationship between bad debt rate, the liquidity ratio, the quick liquidity ratio,and the Rate of loan maturities:

The following represent the Correlation matrix between bad debt rate (y), the liquidity ratio (x_1), the quick liquidity ratio(x_2), the Rate of loan maturities (x_3) in order to determine the explanatory variable that is more related to the bad debt rate.

$$\begin{bmatrix} 1.0000000 & -0.7841519 & -0.6049875 & -0.6484035 \\ -0.7841519 & 1.0000000 & 0.5753677 & 0.5796666 \\ -0.6049875 & 0.5753677 & 1.0000000 & 0.6705442 \\ -0.6484035 & 0.5796666 & 0.6705442 & 1.0000000 \end{bmatrix}$$

Through the correlation matrix, we notice that the maximum value of the correlation between the bad debt rate (y), the liquidity ratio (x_1) reached **-0.78**, which is an inverse relationship as be shown in figure (4).

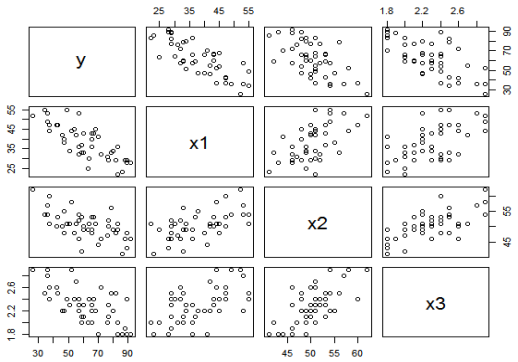


Figure (4) Correlation matrix between dependent and independent variables

4.3 Goodness- of – Fit Test of Real Data.

some statistical criteria, such as the Kolmogorov- Smirnov test, the Cramer- von statistic test, and Anderson- darling will be presented to test that the data follows Transmuted weibull distribution.

Table (4.6) Goodness- of – Fit Test of Real Data

Goodness of fit statistics	**Transmuted Weibull**	**Weibull**	**MG-Weibull**
Kolmogorov- Smirnov	**0.08707688**	0.0989061	0.164695
Cramer-Von statistic	**0.01712300**	0.0234912	0.097667
Anderson- darling	**0.14797663**	0.1734884	0.624825

From the previous table, we notice the following that the value of some statistical criteria, such as the Kolmogorov-Smirnov test, the Cramer- von statistic test, and Anderson- darling, for the Transmuted Weibull distribution is smaller than the rest of the other distributions, and thus the data represents the Transmuted Weibull distribution.

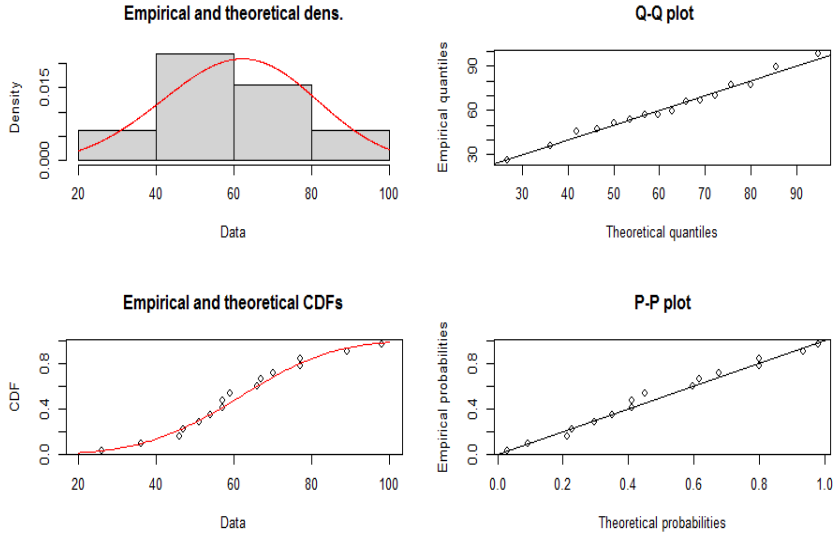


Figure (5) Goodness-of-fit

The previous figure show that the data follows the transmuted Weibull distribution, then, Fit of The Regression Model for Transmuted Weibull

4.4 Comparison between T-W With Kw-Lindley and MG Weibull regression model:

The comparison between T-W With Kw-Lindley and MG Weibull regression model by using R-program according to AIC, BIC and HQIC criteria as the following:

Table(7) The comparison between T-W With Kw-Lindley and MG Weibull regression

Regression model	$\widehat{\boldsymbol{\beta}}_{\mathbf{0}}$	$\widehat{\boldsymbol{\beta}}_{\mathbf{1}}$	HQIC	BIC	AIC	R^2
Transmuted Weibull	0.42352	0.6234	132.2815	125.3456	124.5423	0.8986
Kumaraswamy Lindley	0.03465	0.5234	134.8452	130.7256	128.4213	0.4239
MG Weibull	0.34214	0.4652	138.232	139.341	136.423	0.3492

Table (7) displays the results of previous criteria which show that the T-W regression model is more appropriate model compared to the KW-Lindley and MG Weibull regression model according to smaller values of HQIC AIC and BIC for T-W. Additionally, the higher coefficient of R-square for T-W compared to the KW-Lindley and MG Weibull regression model as be shown in Figure (6).

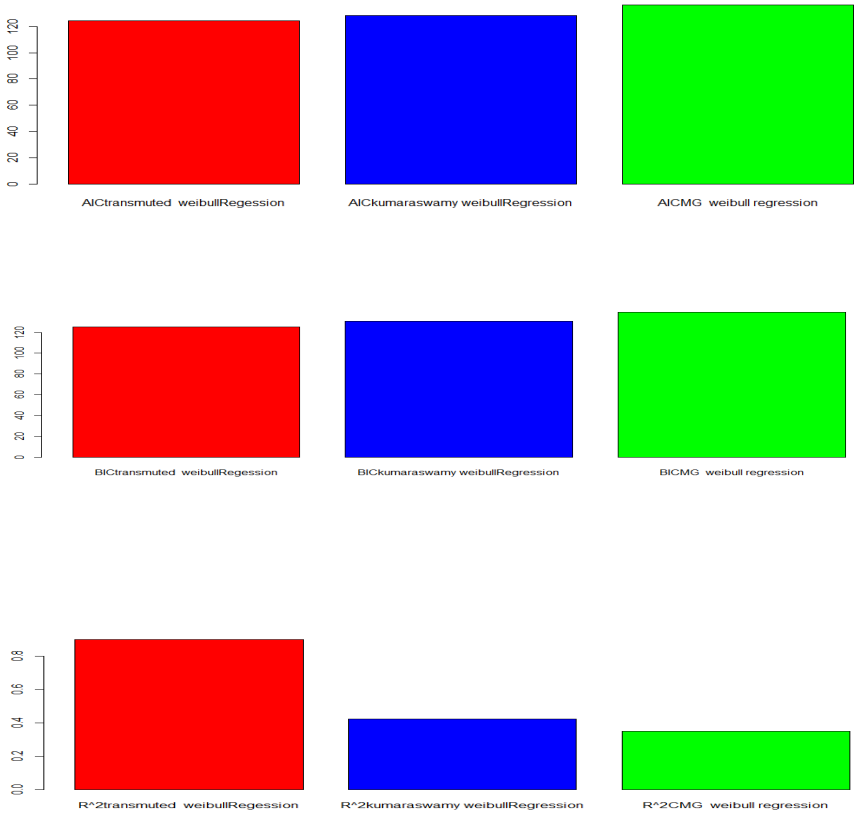


Figure (6) AIC, BIC and R^2 for transmuted weibull regression model, kumaraswamy weibull Regression and MG weibull regression Models

From Figures (6) we notice that each of the statistical criteria AIC,BIC,HQIC for the transmuted-Weibull regression is smaller than the rest of the same criteria for the other regressions. Thus, the transmuted Weibull regression is the best regression model. The same is true for the coefficient of determination, We note that the coefficient of determination for the transmuted Weibull regression is larger than the rest of the other regressions. Thus, the transmuted Weibull regression is the best regression model. According to equation (13) the regression model of T-W is:

$$log\mu(y) = \mathbf{0.42352} + \mathbf{0.6234}\, x\, (liquidity\ ratio)$$

4.5 Global Influence:

Is the diagnostic influence depending on case drop that represents one of the tools to perform sensitivity analysis introduced by Cook case drop is a famous method depending on the influence of deleting the i^{th}observation from the data on the parameter estimation. This method compares $\boldsymbol{\gamma_i}$ and $\boldsymbol{\gamma_{i-1}}$, where $\boldsymbol{\gamma_i}$ is MLE when the i^{th} observation is deleted from the original data. Then the ith case could be considered as influential observation if $\boldsymbol{\gamma_{i-1}}$ is far from $\boldsymbol{\gamma_i}$ (De Bastiani, F, et al 2015).

Table (8): Sensitive Analysis for regression

Case	$\widehat{\beta}_0$	$\widehat{\beta}_1$	AIC	BIC	HQIC	A Min -log (Likelihood)
Before	0.3211	0.6234	112. 842	115.432	114.9633	356.312
After	2.9364	0. 3544	137.574	138.542	135.213	324.624

Source: **Developed by the researcher.**

Cook =2 (A Min – log (Likelihood) before **-** A Min -log (Likelihood) After)

Cook =(356.312- 324.624) = 32.4974.

Noted that the cook is a positive value, meaning that the log-likelihood in small values is less than in large values. This has happened because the values of estimates become less.

5.Conclusion:

In this paper we introduce a new regression model for Transmuted Weibull distribution with application on real data set. The method of maximum likelihood was utilized to estimate the model parameters. As well as the residuals analysis for T-W regression model are provided. Monte Carlo simulation is conducted to assess the performance of parameters of T-W regression model. Based on goodness of fit criteria it is shown that T-W regression model is more fitted to real data set than other compared model the KW-Lindley and MG Weibull regression model. Finally the Global influence is conducted to assess the T-W regression model before and after the i^{th} observation is deleted shown that the cook is a positive value, meaning that the log-likelihood in small values is less than in large values.

6. Reference:

Afify, A. Z., & Mead, M. E. (2017). On five-parameter Burr XII distribution: properties and applications. *South African Statistical Journal*, *51*(1), 67-80. DOI:10.37920/sasj.2017.51.1.4

Afify, A. Z., Cordeiro, G. M., Yousof, H. M., Saboor, A., & Ortega, E. M. (2018). The Marshall-Olkin additive Weibull distribution with variable shapes for the hazard rate. *Hacettepe Journal of Mathematics and Statistics*, *47*(2), 365-381. DOI:10.1007/s00009-017-0955-1

Afify, A. Z., Nofal, Z. M., & Butt, N. S. (2014). Transmuted complementary Weibull geometric distribution. *Pakistan Journal of Statistics and Operation Research*, 435-454.

Alamoudi, H. H., Mousa, S. A., and Baharith, L. A. (2017). Estimation and application in log-Fréchet regression model using censored data. Int. J. Adv. Stat. Probab, 5(1), 23-31.

AL-Kadim, K. A., & Mohammed, M. H. (2017). The cubic transmuted Weibull distribution. *Journal of University of Babylon*, *3*, 862-876.

Aryal G.R, Tsokos C.P (2009) On the transmuted extreme value distribution with applications. Nonlinear Anal 71:1401–1407.

Aryal, G. R., & Tsokos, C. P. (2011). Transmuted Weibull distribution: A generalization of the Weibull probability distribution. *European Journal of pure and applied mathematics*, *4*(2), 89-102.

Ashour, S. K., & Eltehiwy, M. A. (2013). Transmuted exponentiated modified Weibull distribution. *International Journal of Basic and Applied Sciences*, *2*(3), 258.

Cox, D. R., & Snell, E. J. (1968). A general definition of residuals. *Journal of the Royal Statistical Society: Series B (Methodological)*, *30*(2), 248-265.

De Gusmao, F. R., Ortega, E. M., & Cordeiro, G. M. (2011). The generalized inverse Weibull distribution. *Statistical Papers*, *52*(3), 591-619.

Dey, S., Kumar, D., Anis, M. Z., Nadarajah, S., & Okorie, I. (2021). A review of transmuted distributions. *Journal of the Indian Society for Probability and Statistics*, *22*(1), 47-111.

Elbatal, I., & Aryal, G. (2013). On the transmuted additive weibull distribution. *Austrian Journal of statistics*, *42*(2), 117-132.

Khan, M. S. (2019). Transmuted Modified Inverse Weibull distribution: Properties and application. *Pakistan Journal of Statistics and Operation Research*, 667-677.

Louzada, F., & Granzotto, D. C. (2016). The transmuted log-logistic regression model: a new model for time up to first calving of cows. *Statistical Papers*, *57*, 623-640.

Merovci, F. (2013a). Transmuted lindley distribution. Int. J. Open Problems Compt. Math, 6(2), 63-72.

Merovci, F. (2013b). Transmuted rayleigh distribution. *Austrian Journal of statistics*, *42*(1), 21-31.

Merovci, F., Elbatal, I., & Ahmed, A. (2013). Transmuted generalized inverse Weibull distribution. *arXiv preprint arXiv:1309.3268*.

Nofal, Z. M., Afify, A. Z., Yousof, H. M., Granzotto, D. C. T., & Louzada, F. (2018). The transmuted exponentiated additive weibull distribution: properties and applications. *Journal of Modern Applied Statistical Methods*, *17*(1), 4.

Sarstedt, M., & Mooi, E. (2014). A Concise Guide to Market Research. The Process, Data, and, 12.(p-p. 193-233). DOI:10.1007/978-3-642-53965-77 .

Shaw, W. T., & Buckley, I. R. (2009). The alchemy of probability distributions: beyond Gram-Charlier expansions, and a skew-kurtotic-normal distribution from a rank transmutation map. arXiv preprint arXiv:0901.0434.

Comparing The Performance of Penalized Logistic Regression versus other Machine Learning Algorithms in Predicting Phenotypes of Ectodermal Dysplasia Patients.

Phoebe Magdy Abd El Massieh Metias

Department of Oro-dental Genetics, Human Genetics and Genome Research Institute,

National Research Centre, Cairo- Egypt.

Correspondence Email: pm.abdel-massih@nrc.sci.eg & phoffa_m@hotmail.com

Amany Mousa Mohamed

Department of Applied Statistics and Econometrics,

Faculty of Graduate Studies for Statistical Research, Cairo- Egypt.

Email: dramanymousa04@gmail.com

Mostafa Ibrahim Mostafa

Department of Oro-dental Genetics, Human Genetics and Genome Research Institute,

National Research Center, Cairo- Egypt.

Email: mostafanrc@yahoo.com

Shereen Hamdy Abdel Latif

Department of Applied Statistics and Econometrics,

Faculty of Graduate Studies for Statistical Research, Cairo- Egypt.

Email: shereen_hamdy_@cu.edu.eg

Abstract:

Background: Ectodermal dysplasia (ED) is a monogenic genetic disorder characterized by primary defects in two or more structures derived from the ectoderm, namely hair, nails, teeth, and sweat glands. These patients have a wide diversity of characteristic phenotypic features that range from mild to severe, consequently affecting their quality of life to varying degrees. To our knowledge, no study has compared the performance of different machine learning algorithms in predicting the phenotype severity of ED cases from their genotype profiles. Accurate prediction is vital for future prognoses and the early planning for future therapies and proper preparation for genetic counseling for parents and patients.

Methods: We applied six different modeling techniques, (Decision Trees (DT), Random Forests (RF), Extreme Gradient Boosting (XGBoost), K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and penalized Logistic Regression (LR)) to a total of 41 ED patients with 569,650 single nucleotide polymorphisms (SNPs), and compared their performances in phenotype prediction. We aimed to optimize the receiver operating characteristics area under the curve (ROC-AUC) as our primary metric of interest in model evaluation and algorithm choice.

Results: Ensemble techniques (RF and XGboost) together with KNN recorded comparable and the highest ROC-AUC values of 0.9 with a 95% confidence interval (0.723-0.9999), while penalized LR gave much lesser prediction ability to distinguish between different classes (ROC-AUC (95% CI)) = 0.6 (0.31 – 0.89).

Conclusion: We can predict the phenotype of ED patients with a respectable level of accuracy and that will help them improve their quality of life. Small-sized datasets that are obtained from local studies should not hamper researchers from implementing ML algorithms for

prognostication purposes. The privilege of using ML in prognostic modeling may be dependent on multiple factors like sample size, selected features, and the underlying disease of interest.

Keywords: Machine Learning, Ectodermal Dysplasia, Phenotype Prediction, Prognostic Modeling.

Introduction:

Ectodermal dysplasia is a monogenic genetic disorder that comprises a massive group of rare conditions [1]. It is caused by congenital defects affecting the development of two or more ectodermal-derived body structures specifically hair, teeth, nails, skin, and certain glandular structures, e.g. sweat glands [2]. ED patients have a wide diversity in their phenotypes. The characteristic phenotypic features of ED patients range from mild to severe. In severe cases, the large number of missing teeth (oligodontia), abnormally shaped teeth, consequent growth problems, and with the continually rising cost of dental management, all tend to affect these patients' quality of life [3]. Moreover, the reduced number or absent sweat glands increases the risk of elevated body temperatures. Also, the dryness of the respiratory airways causes recurrent chest infections leading to life-threatening conditions [4]. Being bullied by their peers, and always being called out as "Streeters" is another issue that causes ED patients to suffer from psychological distress. In a study made by Saltnes SS et al (2017) on ED patients, they found that half of the participants suffered from anxiety and a reduction in their mental health-related quality of life to such a degree that they may require evaluation, and treatment [5].

Machine learning algorithms are paradigms used for predicting future events from learning past experiences [6]. They are gaining considerable interest from clinicians because of

their wide predictability and applicability in medical diagnosis [7]. The primary aim of the current study is to apply ML algorithms and compare their performance in predicting ED patients' phenotypes from their genotype data. Achieving accurate prediction will largely help in forecasting future prognoses and the early planning of future therapies and proper preparation for genetic counseling with parents and patients, also accurate foretelling of carrier detection of at-risk family members as well as a prenatal diagnosis is of particular importance, especially in severe life-threatening cases. This might help parents in the decision-making of abortion, taking into consideration the religious background and the laws of the country they reside.

An accurate prediction can also derive parents to try new treatments as was discussed by Dr. Holm Schneider in his report [8] where prenatal treatment of 6 boys was carried out by injecting the replacement protein into the amniotic fluid which induced the development of functional sweat glands. Follow-up showed that the two oldest boys treated in utero have a normal ability to sweat and this has so far persisted for more than 5 years. Thus, proper timing of protein replacement appears to be a promising therapeutic modality for ED, and it might be employed in other congenital disorders. Thus, accurate prediction of ED patients' phenotypes is therefore of vital role.

Methods:

1. Study Population:

The study design was approved by the Medical Research Ethics Committee at the National Research Centre (NRC), Cairo, Egypt, with registration number (19/118). Data available for the study were obtained from patients attending the Oro-dental Genetics Clinic and

Genodermatosis Clinic, NRC. All patients have undergone extensive clinical and Oro-dental examination with emphasis on skin and related ectodermal structures. Panoramic radiographs were done whenever possible to evaluate the number of missing teeth of the recruited patients. Our study cohort consisted of 41 cases, 20 females and 21 males, with ages ranging from 1.5 to 25 years with an average age of 7 years. They were categorized as mild if they had only two ectodermal structures affected, moderate for three affected structures, and severe for more than 3 structures affected. All patients' phenotypes in our study were either moderate (19 patients) or severe (22 patients).

2. Data Collection and Integration:

Data for this study was supplied in three groups: **Group 1:** comprised 9 patients' Variant Call Format (VCF) files (version 4.1), generated by Pisces variant caller version 5.2.9.23 [9]. **Group 2**: contained 23 patients' VCF files (version 4.2) generated by the FreeBayes variant caller version 1.3.1 [10]. **Group 3:** came with 9 patients' VCF files (version 4.2) encompassing variants called using Genome Analysis Tool Kit (GATK) variant caller version 4.2.6.1 [11]. We applied quality control (QC) measures as directed by Naj AC et al. in [12] to ensure that we finally get a homogenous set of genotypes for all patients from the different callers and transform them into a tabular format that can be used for applying machine learning (ML) algorithms.

3. Data Preprocessing and Quality Control:

All data analyses in this study were done in a Linux operating system Ubuntu 22.04.1 LTS (Jammy Jellyfish), with a Package Manager: Miniconda3-latest-Linux-x86_64.sh, Conda version 22.9.0, and Python version 3.11.0.

Data preprocessing is a crucial step in any machine-learning application, and it always takes from 70 to 80% of the effort done in analyzing data. It has a great role in the quality of model performance. It is of special importance in the field of bioinformatics because files supplied from genomic facilities, such as VCF files used in our study, are not in a format that can be directly used to apply ML algorithms and need a lot of preprocessing and information extraction. Preprocessing steps explanation is often neglected in most literature, a simplified clarification will be outlined below to give guidance to the reader on how our work flowed from beginning to end.

A short notice on what a VCF file for an individual case looks like: It is a tab-separated text file, containing meta-information lines beginning with '##', a header line beginning with '#', and then data lines each containing information about a single SNP in the genome. Each of these data lines comprises 8 fixed fields plus a genotype information field, and a patient-specific information field. The 8 fixed fields are 'CHROM', 'POS', 'ID', 'REF', 'ALT', 'QUAL', 'FILTER', and 'INFO', the genotype information field is titled 'FORMAT', and the patient's specific field should be a unique ID for each patient [13]. VCF file manipulation steps:

1- First, we deleted the meta-information section from every file and read the rest of the VCF file in a tabular format using pandas data frames (df) version 1.5.1.

2- We extracted the required information e.g. SNP genotype (GT) and allele frequency (AF), from the 'INFO' and 'FORMAT' fields, into separate columns using Python regular expressions [REGEX] codes.

3- We applied QC measures using pandas df filtering options. First, we ensured that all cases were aligned against the same reference genome GRCh37/hg19 as stated in the meta information section of their files, and applied multiple filters to include only SNPs

that have: a 'FILTER' field equal to PASS and a 'QUAL' field of more than 30. SNPs with missing genotypes like '1/.', '0/.', or './.' and SNPs with minimum allele frequency less than 0.01, were excluded. Finally, we included SNPs only from chromosomes that are known to be associated with our disease of interest where the 'CHROM' field was adjusted to be equal to ['chr1', 'chr2', 'chr3', 'chr9', 'chr22', and 'chrX'] only.

4- We added a new column, named “SNP_ID”, to every file combining the information from both the “CHROM” and “POS” columns.

5- Two columns from each file were then extracted, these are “SNP_ID” and “GT”.

6- The “GT” column was renamed with a unique Sample_ID.

7- The resulting df was then transposed so that every patient df now consists of two rows, SNP_IDs, and the genotype of every SNP. Then we let the “SNP_ID” row, be the columns header so that every patient’s final df comprises only one row for all genotypes, and as many columns as every patient had in his or her VCF file.

8- We concatenated all patients' dfs so that our final df now has 41 rows indexed with the unique Sample_ID and 569,650 columns representing SNPs from all cases, some SNPs are common in all patients while others exist in some patients rather than others, so lot of Nan values appeared in this final df that should be dealt with.

9- Since some SNPs have Nan values for specific patients, this means that this exact SNP was like the reference genome and consequently was not called by the variant caller in that patient, and with that, we can replace Nan values with genotype 0/0.

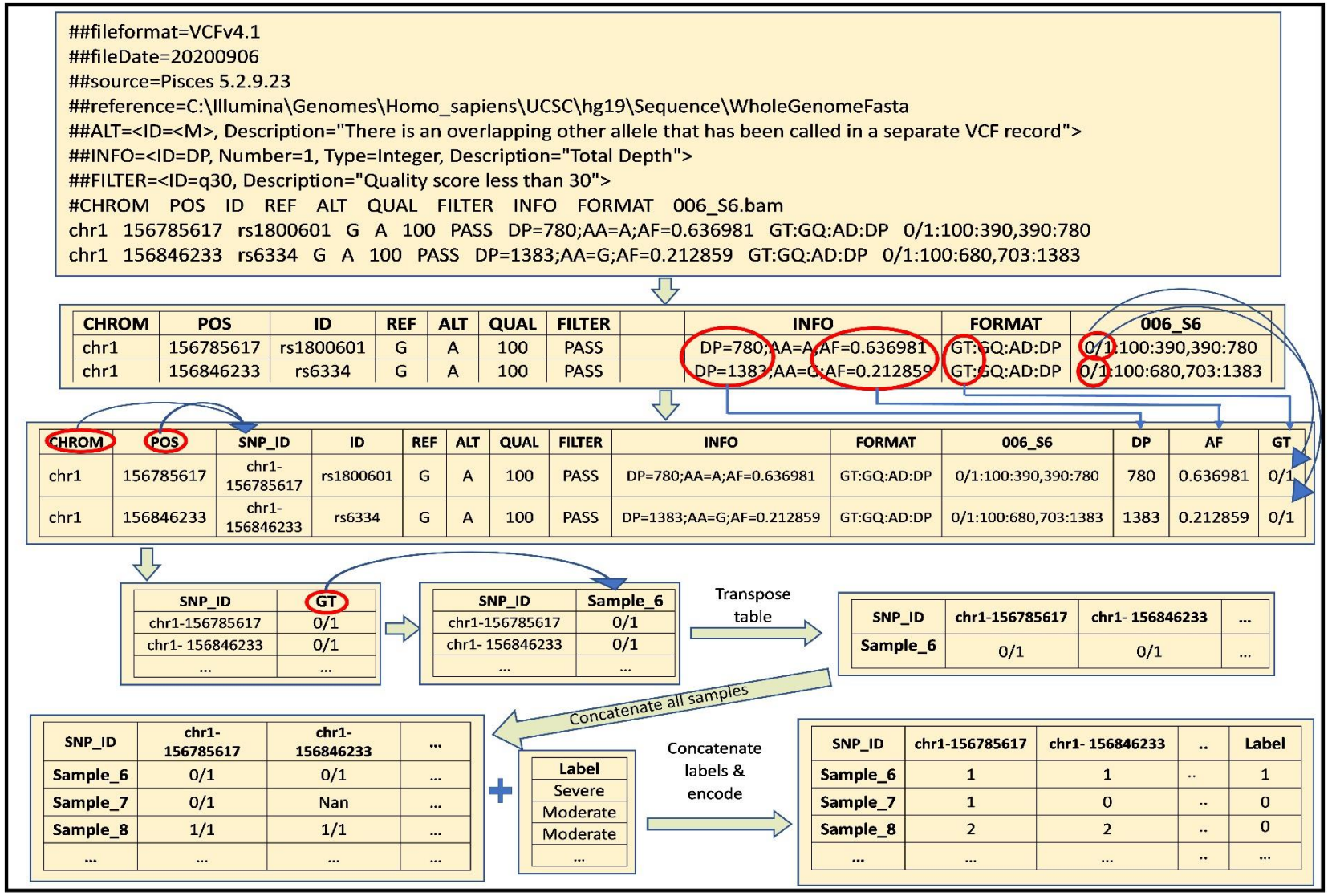
10- Our label variable which is a binary variable representing the severity of the ED condition, either moderate or severe is then added to the df.

11- A final step to prepare the data for ML algorithms is to encode the genotypes, additive encoding was applied in our case so that genotypes, with zero minor alleles '0/0', one minor allele '0/1', and 2 minor alleles '1/1', were encoded as 0, 1, and 2 respectively. The target variable is also encoded as 0, and 1 representing moderate and severe cases respectively.

All VCF file preprocessing steps are illustrated in Figure (1).

12- X-y split: we then split our data frame into X and y where X represents the samples with all 569,651 features and y represents the binary target (label) variable.

13- Train-test split: the dataset was then furtherly split into training and testing groups with a 75/25 percentage, to use the latter set in testing and comparing the performance of the different Algorithms.



171 **Fig.1** Diagram to illustrate the preprocessing steps needed to convert a Variant call file (VCF) to a format suitable for machine

172 learning applications.

4. Feature Selection:

Since our data comprises only 41 cases and 569,650 features (SNPs), here emerges the ghost that waves with the dimensionality curse problem that bothers statisticians because of all the calamity it brings with it, such as data sparsity, feature dependence (multicollinearity), multiple testing, and overfitting, where their impact is much amplified by the small sample size which is, unfortunately, a typical case in most genomic data analysis as well as in ours [14]. At this moment where the umbrella of problems dims the analysis space, there shines the role of interdisciplinary sciences where the knowledge from multiple disciplines that are statistics, programming, and machine learning together with the strong genetic background are sewed together to aid in the dissipation of this dimness and also provide explainability of the results which is, in most of the scenarios, of great importance to the clinician.

Hence, dimensionality reduction was one of our major concerns to remove irrelevant and redundant features aiming to improve our mining performance and increase learning accuracy. We were in the face of two major ways for dimensionality reduction either feature selection or feature extraction. Broadly speaking, feature extraction is a technique used to obtain new features from existing ones by applying transformations on original features to reduce their complexity and to give a simple representation of data representing each variable in the feature space as a linear combination of input variables. There are several feature extraction techniques available the most known of which is the Principal Component Analysis (PCA) [15]. However, PCA has some limitations as it assumes that the relationships between variables are linear [16], in feature extraction techniques, much information will typically be lost, thus leading to difficulty in interpretability, for these reasons, we directed our effort to reduce our data

dimensions by using several feature selection techniques rather than using feature extraction techniques.

Selecting only a subset of features that are most “relevant” to the label variable and discarding noisy “irrelevant” and “redundant” ones help researchers to understand the biological process(es) that underlie the disease, also aid in the generalizability of machine learning algorithms to unseen datasets, reduce model overfitting, and significantly decrease the computational time and resources needed for the analysis [17].

Contrary to feature extraction approaches, feature selection strategies simply choose a subset of the available features without altering their original form. Thus, they keep the original semantics of the features, hence, offering the advantage of interpretability and explainability by a domain expert [18]. Feature selection techniques, in the context of classification modeling, can be arranged into three main categories, depending on how they merge the feature selection search methods with the classification model. These categories are filter methods, wrapper methods, and embedded methods [15,17]. Other emerging strategies for feature selection are a) Integrated methods, which include the use of external knowledge to limit the search space; b) Hybrid methods, which include the combined use of different successive feature selection procedures and; c) Ensemble methods, which include combining the outcome features selected from multiple feature selection methods.

In our study, we used multiple successive feature selection techniques to reduce the total number of features in a step-by-step manner by removing irrelevant (those features that are noninformative and must be excluded) [19] and redundant (those that are correlated with other features and not relevant in the sense that they do not improve the discriminatory ability of a set of features) features [20].

First, the integrative method for feature selection was used depending on the biological background known of the ectodermal dysplasia patients (disease of interest), only SNPs from chromosomes/ genes that are known to be associated with the disease were selected.

Second, certain features were deemed necessary to be removed from consideration. These included constant, quasi-constant, and duplicate features. A constant feature was defined as any feature that had the same value for all patients. Meanwhile, a quasi-constant feature was identified, in this study, as any feature that had only one patient in the training set with a different value, while all other cases had the same value for that feature. Additionally, any duplicate features were considered redundant and would not aid in distinguishing between the two outcome classes. This narrowed our feature space significantly from having 569,650 to only 63089 SNPs.

Third, we then used a univariate filtering technique by applying a univariate Decision Tree algorithm to evaluate the receiver operator characteristics area under the curve (ROC-AUC) for every single feature with the target variable and choose those features with ROC-AUC values greater than or equal to 0.8.

Lastly, a correlation matrix was then calculated for the chosen set of features, and groups of correlated features were created, then we used a Random Forest algorithm to choose the most important feature as a representative feature for every correlated group thus further reducing dimensionality and removing redundant information.

We then compared the effect of using these aforementioned methods for feature selection against using the penalized method in Logistic regression for feature selection, in terms of model performance and generalizability [21].

Penalized feature selection belongs to the family of embedded feature selection methods [22]. Examples of regularization models for feature selection are LASSO and elastic net. They usually work by penalizing or shrinking the coefficient of features that do not contribute to the improvement of the model performance in a meaningful way. The regularization methods return a ranked list of features. The ranking of features is given by the magnitude of the feature coefficients which is forced to be small or to be exactly zero. [17,23]. Penalized approaches can eliminate redundant features, but they lack the inherent capacity to recognize feature relationships. Instead, the analysis must explicitly take interaction terms into account. While this method may work well for low-dimensional data, it may be erroneous and computationally expensive when dealing with high-dimensional data.

To avoid information leakage from the training set to the testing set, all our feature selection techniques were applied to the dataset used in training ML algorithms only, and then the discarded features were dropped from both the train and test sets. [21]

ML Methods:

Our problem is one of the supervised learning problems where our target variable is a binary target, so all ML algorithms used are supervised ML classifiers. All codes used in the analysis for this study, are available in the public GitHub repository (https://github.com/PhoebeMagdy/Ectodermal-Dysplasia-Data-Analysis).

K- Nearest Neighbours (KNN):

KNN is one of the simplest and non-parametric ML algorithms, it does not make any assumption about the data used. Classification is accomplished by identifying the nearest

neighbors to a query example and using those neighbors to decide the class of the query based on measuring distances between them [24].

Decision Trees (DT)

DT classifies a population into inverted tree-like branches that begin with a root node, subdivide into internal nodes, and end with leaf nodes. It is a non-parametric algorithm that can skillfully deal with large, complicated datasets without setting distributional assumptions. It is a powerful tool for classification and prediction, also it is easy to interpret and has several applications in recent medical research. However, it has a common drawback as it is highly prone to overfitting, which limits the generalizability and robustness of the resultant models [25].

Ensemble Techniques:

Building a single ML estimator may fail to obtain good enough performances when dealing with complex, noisy, and high-dimensional data leading to some technical challenges like 1) Model overfitting (high variance) where the testing error is much bigger than the training error (almost equal to zero) and that in turn affects the model generalizability. 2) Model underfitting (high bias) where the model fails to capture the pattern in the trained dataset resulting in both high training and test errors. The ensemble learning method can find solutions to such problems by creating multiple instances of traditional ML algorithms and combining them to give a single optimal solution to a problem. This technique is capable of producing better predictive and more robust models compared to the traditional approach. The instances created are called base learners. Each base learner works independently as a traditional ML method, and

the eventual results are combined to produce a single robust output. The combination could be done using majority voting for classification problems.

Extensive research has been conducted to explore the impact of employing ensemble techniques in improving the overall performance of the ML classifiers especially in the prediction of disease prognosis. In this regard, a study was conducted to investigate the prognosis of cervical cancer using a combination of the genetic algorithm for feature selection and the adaptive boosting algorithm, both implemented in Support Vector Machine and Decision Tree models. The performance of these models was compared and validated under three scenarios: (a) using all attributes, (b) using the genetic algorithm as a feature selector without adaptive boosting, and (c) using both the genetic algorithm and adaptive boosting. The performance of the classifiers was evaluated based on accuracy, sensitivity, specificity, and precision. The results indicated that when only the genetic algorithm was employed as the feature selector, without the inclusion of adaptive boosting, there was no significant improvement in performance compared to using all attributes. However, when the integrated approach of the genetic algorithm and adaptive boosting was utilized, all classifiers demonstrated an enhancement in accuracy, sensitivity, specificity, and precision. In conclusion, this study highlighted the importance of utilizing advanced algorithms, such as the genetic algorithm and adaptive boosting, in the field of cervical cancer prognosis. Its results demonstrated the potential of the ensemble technique to enhance the overall performance of the classifiers, ultimately leading to improved patient care [26]. Another study was conducted to delve into the immense potential of pre-trained convolutional neural networks (CNNs) such as EfficientNetV2L, ResNet152V2, and DenseNet201. These networks were examined both individually and as an ensemble to assess their effectiveness in classifying Invasive Ductal

Carcinoma Breast Cancer (IDC-BC) grades, utilizing the DataBiox dataset. The study also explored the influence of sample size and various training epochs on the models' performances. The findings of this study revealed that the ensemble of CNNs consistently outperformed the individual models, showcasing remarkable stability and robustness [27]. For these reasons, we used two ensemble techniques in our study:

A) Random Forest (RF)

It is an ensemble method that combines several tree predictors, trees are fitted in parallel each tree uses a bootstrap sample from the training dataset and a random sub-selection of the available features, and the final decision is taken by the majority vote on all trees [28]. RF greatly reduces variance and overfitting, which the individual DT estimator is prone to, as more trees are fitted [29].

B) Xtreme Gradient Boosting (XGBoost):

In boosting, trees are sequentially fitted, in a manner that we fit a weak classifier to a version of the data set then we calculate the weighted misclassification rate of that classifier, update the weightings measured in the next round iteratively, then evaluate the model performance [28]. Here we used the XGBoost classifier, which is a highly scalable tree-boosting system. It is more than ten times faster than other existing solutions, utilizes out-of-core computation, and enables researchers to process millions of examples on a desktop single machine, billions of examples can be managed in distributed or limited memory settings. Quicker model evaluation can be achieved by making parallel and distributed computing which fastens the learning process [30].

Support Vector Machine (SVM):

The SVM is an algorithm based on the idea of finding and optimizing a separating hyperplane between classes, the hyperplane has to be settled such that it is at a maximum distance from the different classes, thus helping the model to generalize better. This is essential because training is done on a sample of the population, whereas prediction is to be performed on unseen datasets that may have a distribution that is slightly different from that of the trained set [31].

Logistic Regression (LR):

LR is used for solving classification problems, it predicts the outcome of a categorical discrete variable. It can be either diseased or not diseased, True or False, Yes or No, or 0 or 1, it gives the probabilistic values that lie between 0 and 1, instead of giving the exact value as 0 and 1. It uses the sigmoid function to map the predicted values to probabilities. LR is an important ML algorithm because it can predict probabilities and classify new data points after being trained using continuous and discrete datasets.

Model Tuning and Hyperparameter Optimization:

The main goal of tuning is to reduce misclassifications, assess the model generalizability, and compare the performance of different classifiers. To achieve these goals, we divided the data set into a 75% train and validate set and a 25% test set that is used only for external testing after the tunning process was done. We used a nested cross-validation (CV) approach to tune the hyperparameter and validate the model [32]. In the outer loop of the nested

CV, we used a 10-fold stratified CV for model generalizability assessment, whilst, in the inner loop we used a 5-fold stratified CV and the GridSearch method to tune hyperparameters [33]. We aimed to optimize the area under the receiver operator curve (ROC-AUC). The best hyperparameters chosen were used to fit a final model on the whole 75% training set and then the external test set was used to evaluate and compare the performance of different algorithms. All optimal hyperparameters chosen for each of the ML algorithms are recorded in Table (1), and all the final training and external testing evaluation metrics are reported in Table (2).

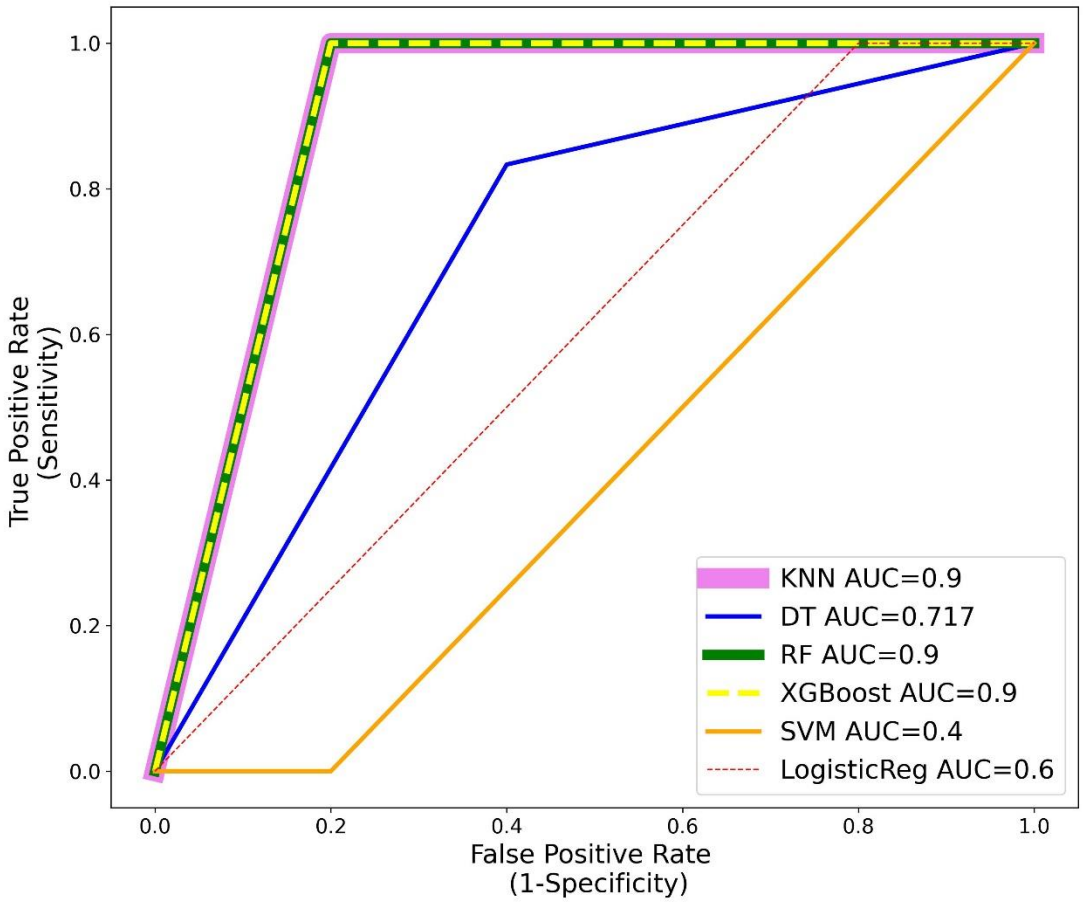
We assessed models' performances using different classification metrics but the one of interest was the (ROC-AUC). Its values range from 0 to 1, with a value of 1 when the model can efficiently distinguish between severe and moderate cases, 0.5 when the model cannot distinguish the different classes in the target variable, and 0 when the model is entirely incorrect in its classification. Other classification metrics of interest to us were Recall (sensitivity), precision, and specificity. True Positive Rate (Recall) and Predictive Positive Values (Precision) were of special importance because accurate detection of the positive (severe) cases was our main concern for the reasons mentioned earlier in the introduction section. All model training, hyperparameters tunning, and models' evaluation were implemented using scikit-learn libraries version 1.1.3 [34].

Results:

KNN, RF, and XGBoost all gave similar and the highest results, on external testing, for all metrics used in the evaluation process of the different ML algorithms. Their ROC-AUC results were (0.9), with a 95% confidence interval (CI) (0.723-0.9999). Despite DT giving a fairly good ROC-AUC score of 0.717, still, its 95% CI (0.45-0.983) is very wide and the lower

limit indicates nonsignificance in the class determination being less than 0.5. SVM recorded the worst values in almost all metrics used, with ROC-AUC values and 95% CIs of 0.4(0.11 – 0.69). A graph declaring the ROC-AUC values for all algorithms used is shown in Figure 2. In terms of test accuracy KNN, RF, and XGBoost, all recorded 0.909 with 95% CI (0.739 – 0.999), DT test accuracy with 95% CI was 0.727 (0.464 – 0.99), SVM revealed the worst test accuracy values with 95% CI of 0.364 (0.079 – 0.648). For precision, KNN, RF, and XGBoost predicted 85.7% of all the severely predicted cases, with 95% CI (0.65 – 0.999). DT showed a slightly lower precision value of 71.4% and a wide 95% CI (0.447 – 0.981). SVM recorded 0% precision. The sensitivity of KNN, RF, XGBoost algorithms was 1 indicating that these algorithms were excellent at predicting severe cases among all truly severe cases. SVM recorded 0 sensitivity. Finally, KNN, RF, and XGBoost recorded high test capability to distinguish truly moderate cases (Specificity) of 0.8 with 95% CI (0.564 – 0.999). SVM recorded 0.8 in terms of specificity as shown in Figure 3.

On the contrary, the result of the penalized Logistic Regression (LR) algorithm was deemed satisfactory, with results as follows: ROC-AUC values of 0.741 and 0.6, for train and test data respectively. Sensitivity values of 0.625 and 1.0, respectively. Precision values for train and test data of 0.833 and 0.6, respectively. Specificity values of 0.857 and 0.2, respectively.



397

398 **Fig.2** Receiver operating characteristics area under the curve for the different ML algorithms
399 used. Lines are color-coded as represented in the figure legend in the bottom right corner of the
400 graph. KNN: K-nearest neighbors, AUC: area under the curve, DT: decision tree, RF: random
401 forest, XGBoost: extreme gradient boosting, SVM: support vector machine, LR: logistic
402 regression.

403

404

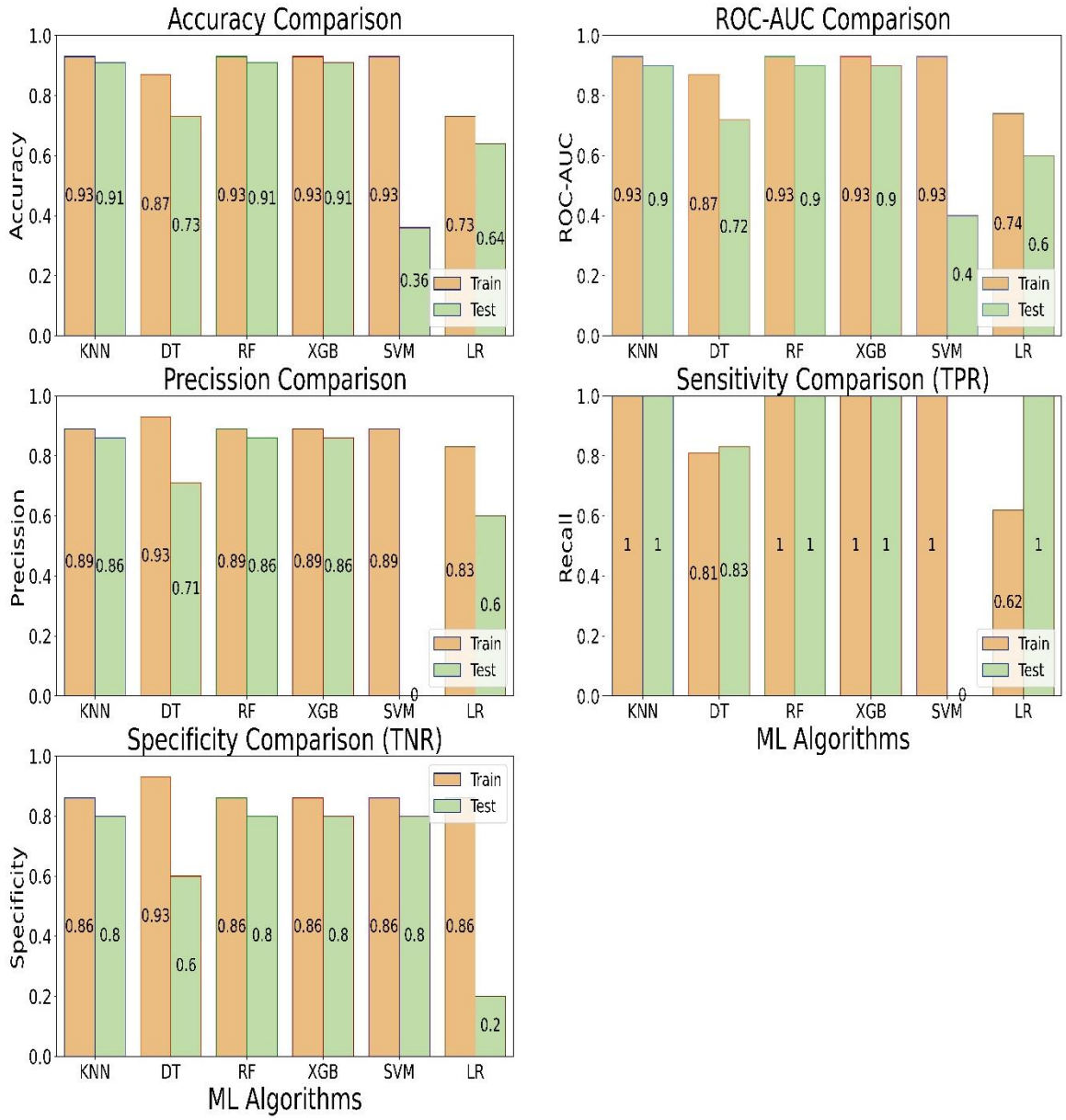


Fig.3 Performance metrics of the six machine learning algorithms used. ROC-AUC: receiver operating characteristics area under the curve, DT: decision tree, KNN: K-nearest neighbors, RF: random forest, XGB: extreme gradient boosting, SVM: support vector machine, LR: logistic regression

Discussion:

Though our sample size is considered very small, in comparison to studies made on databases from open-source repositories, ML algorithms can still be applied to local databases and its derived predictions can take into consideration some unique features of the local population, thus providing the local clinicians with valuable clinical insights [35]. Machine learning is increasingly being applied to a lot of healthcare and medical prognostic problems. We have successfully compared different ML algorithms for predicting the phenotype of ED patients from their genotypes. Tree-based algorithms showed relatively high results with the ensemble techniques naming RF and XGBoost giving even much better performances than DT due to their ability to reduce overfitting by continuous resampling off the training dataset. LR displayed poorer performance in comparison to other ML algorithms, this could be attributed to the nonlinearity between features and the outcome variable, whereas in such cases, the logistic regression can't predict targets with good accuracy. Also, LR needs a large dataset for training to perform well, while our dataset is quietly small and this could justify its poor performance. KNN is known to be very sensitive to irrelevant and/or redundant features [24], so it performed very well and gave similar results to the ensemble techniques used when applied to our carefully selected set of features, which we claim to be the cause of the high testing performance. SVM is known to work well with a clear margin of separation between classes which might not be the case in our study thus SVM reported a ROC-AUC value of 0.4 where the model was not able to distinguish the different classes in the target variable. Although some algorithms performed very well and gave high predictions on the external testing data sets, still we were confronted with the wide CIs that could be due to the small-sized data points used for validation which increased the validation variability.

Limitations:

Although we achieved satisfactory predictive results, our sample size is considered very small. Nevertheless, this relative success could be accredited to the nature of the data at hand and the proper selection of features with the highest relevance to the target variable that improved the performance to a great extent.

Conclusion:

Small-sized datasets that are obtained from local studies should not hamper researchers from implementing ML algorithms for prognostication purposes. Although our study was purely academic, it reflects the potential for ML to predict different classes of patients and help in the early administration of a suitable treatment plan at a local level. We also would like to clarify that the benefits obtained from using ML in prognostic modeling depend on several factors as the sample size used in model training, the choice of a proper set of features of high relevance to the target variable, and the nature of the underlying disease of interest.

Future Recommendations:

Suggestions for future work include the following: (1) Applying the same analysis techniques on a larger dataset whenever available. (2) Comparing different feature selection techniques and monitoring their effect on models' performance. (3) Doing further investigations for the set of features in hand, as using Ensemble Vep (variant effect predictor) [36] and various other tools to predict their actual relevance to the disease phenotype and investigate SNP interactions that allow for a better understanding of the biological process that underlies the disease.

List of Abbreviations:

ED: Ectodermal Dysplasia.

SNP: Single Nucleotide Polymorphism.

Indel: Insertions/ Deletions.

ML: Machine Learning.

QC: Quality control.

VCF: Variant Call Format.

GATK: Genome Analysis Tool Kit.

DT: Decision Tree.

RF: Random Forest.

SVM: Support Vector Machine.

XGBoost: Xtreme Gradient Boosting.

LR: Logistic Regression.

Declarations:

Ethics approval and consent to participate

This study complies with the approval of the Medical Research Ethics Committee at the National Research Centre (NRC), Cairo, Egypt, with registration number: (19/118)

Consent for publication

Not applicable.

Availability of data and materials

The raw data that support the findings of this study are available from the Department of Medical Molecular Genetics, at the National Research Center Cairo- Egypt, but restrictions apply to the availability of these data, which were used under license for the current study. Qualified researchers may request these data directly from the Medical Molecular Genetics Department. However, all codes used in the analysis for this study, are available in the public GitHub repository (https://github.com/PhoebeMagdy/Ectodermal-Dysplasia-Data-Analysis).

Competing interests

The authors declare that they have no competing interests.

Funding

This study is supported by the Science and Technology Development Fund (STDF), grant no. 33494

Authors' contributions

Ph.M. Abel Massieh conducted the statistical analysis and ML algorithms implementation and drafted the manuscript; A.M.Mohamed developed the ideas presented in this paper; Sh.H.Abdel Latif sketched the design of the study and revised the manuscript; M.I.Mostafa conducted the orodental examination of patients and case selection.
All the authors read and approved the final manuscript.

Acknowledgments

The authors would like to thank Inas S. Mostafa, Associate Professor of Oro-Dental Genetics, NRC, for her efforts in collecting and organizing the orodental clinical data of the patients and for her constructive comments which significantly improved the quality of the manuscript; Khalda S. Amr, Professor of Medical Molecular Genetics, NRC, for the conduction of the molecular investigations for the patients; and Ghada.Y. EL-Kamah, Professor of Clinical Human Genetics, NRC, for her efforts in the general clinical examination of patients with emphasis on skin and related ectodermal structures; and Mennat I. Mehrez, Associate Professor of Oro-Dental Genetics, NRC, for her diligent proofreading of this paper.

References:

1. Pagnan NAB, Visinoni ÁF. Update on ectodermal dysplasias clinical classification. Am J Med Genet A. 2014;164:2415–23.

2. Wright JT, Fete M, Schneider H, Zinser M, Koster MI, Clarke AJ, et al. Ectodermal dysplasias: Classification and organization by phenotype, genotype and molecular pathway. Am J Med Genet A. 2019;179:442–7.

3. Murdock S, Lee JY, Guckes A, Wright JT. A costs analysis of dental treatment for ectodermal dysplasia. Journal of the American Dental Association. 2005;136:1273–6.

4. Wohlfart S, Meiller R, Hammersen J, Park J, Menzel-Severing J, Melichar VO, et al. Natural history of X-linked hypohidrotic ectodermal dysplasia: A 5-year follow-up study. Orphanet J Rare Dis. 2020;15.

5. Saltnes SS, Jensen JL, Sæves R, Nordgarden H, Geirdal AO. Associations between ectodermal dysplasia, psychological distress and quality of life in a group of adults with oligodontia. Acta Odontol Scand. 2017;75:564–72.

6. Pavithra D. A study on machine learning algorithm in medical diagnosis. International Journal of Advanced Research in Computer Science. 2018;9:42–6.

7. Sidey-Gibbons JAM, Sidey-Gibbons CJ. Machine learning in medicine: a practical introduction. BMC Med Res Methodol. 2019;19.

8. Schneider H. Ectodermal dysplasias: New perspectives on the treatment of so far immedicable genetic disorders. Front Genet. 2022;13:1–7.

9. Dunn T, Berry G, Emig-Agius D, Jiang Y, Lei S, Iyer A, et al. Pisces: An accurate and versatile variant caller for somatic and germline next-generation sequencing data. Bioinformatics. 2019;35:1579–81.

10. Garrison E, Marth G. Haplotype-based variant detection from short-read sequencing. 2012; Available from: http://arxiv.org/abs/1207.3907

11. Geraldine A. Van der Auwera, Brian D. O'Connor. Genomics in the Cloud: Using Docker, GATK, and WDL in Terra. First Edition. O'Reilly Media, Inc.; 2020.

12. Naj AC, Lin H, Vardarajan BN, White S, Lancour D, Ma Y, et al. Quality control and integration of genotypes from two calling pipelines for whole genome sequence data in the Alzheimer's disease sequencing project. Genomics. 2019;111:808–18.

13. The Variant Call Format (VCF) Version 4.2 Specification. 2022;

14. Altman N, Krzywinski M. The curse(s) of dimensionality this-month. Nat Methods. Nature Publishing Group; 2018. p. 399–400.

15. Jindal P, Kumar D. A Review on Dimensionality Reduction Techniques. Int J Comput Appl. 2017;173:975–8887.

16. Ringnér M. What is principal component analysis? Nat Biotechnol. 2008;26:303–4.

17. Pudjihartono N, Fadason T, Kempa-Liehr AW, O'Sullivan JM. A Review of Feature Selection Methods for Machine Learning-Based Disease Risk Prediction. Frontiers in Bioinformatics. 2022;2.

18. Saeys Y, Inza I, Larrañaga P. A review of feature selection techniques in bioinformatics. Bioinformatics. 2007;23:2507–17.

19. Yu L, Liu H. Efficient Feature Selection via Analysis of Relevance and Redundancy. Journal of Machine Learning Research. 2004;5:1205–24.

20. Osl M, Dreiseitl S, Cerqueira F, Netzer M, Pfeifer B, Baumgartner C. Demoting redundant features to improve the discriminatory ability in cancer data. J Biomed Inform. 2009;42:721–5.

21. Feature Selection for Machine Learning | Udemy [Internet]. [cited 2023 Jan 22]. Available from: https://www.udemy.com/course/feature-selection-for-machine-learning/

22. Ma S, Huang J. Penalized feature selection and classification in bioinformatics. Brief Bioinform. 2008;9:392–403.

23. Jović A, Brkić K, Bogunović N. A review of feature selection methods with applications. 38th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2015 - Proceedings. Institute of Electrical and Electronics Engineers Inc.; 2015. p. 1200–5.

24. Cunningham P, Delany SJ. K-Nearest Neighbour Classifiers-A Tutorial. ACM Comput Surv. Association for Computing Machinery; 2021.

25. Song YY, Lu Y. Decision tree methods: applications for classification and prediction. Shanghai Arch Psychiatry. 2015;27:130–5.

26. Sharma M. Cervical cancer prognosis using genetic algorithm and adaptive boosting approach. Health Technol (Berl). 2019;9:877–86.

27. Kumaraswamy E, Kumar S, Sharma M. An Invasive Ductal Carcinomas Breast Cancer Grade Classification Using an Ensemble of Convolutional Neural Networks. Diagnostics. 2023;13.

28. Wyner AJ, Olson M, Bleich J, Mease D. Explaining the Success of AdaBoost and Random Forests as Interpolating Classifiers [Internet]. Journal of Machine Learning Research. 2017. Available from: http://jmlr.org/papers/v18/15-240.html.

29. Breiman L. Random forests. Mach Learn [Internet]. 2001 [cited 2023 Jan 10];45:5–32. Available from: https://link.springer.com/article/10.1023/A:1010933404324

30. Chen T, Guestrin C. XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, NY, USA: ACM; 2016.

31. Awad M, Khanna R. Support Vector Machines for Classification. Efficient Learning Machines [Internet]. 2015 [cited 2023 Jan 11];39–66. Available from: https://link.springer.com/chapter/10.1007/978-1-4302-5990-9_3

32. Raschka S. Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning. 2018;

33. Hyperparameter Optimization for Machine Learning | Udemy [Internet]. [cited 2023 Feb 7]. Available from: https://www.udemy.com/course/hyperparameter-optimization-for-machine-learning/

34. Pedregosa F, Michel V, Grisel O, Blondel M, Prettenhofer P, Weiss R, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 2011;12:2825–30.

35. Panesar SS, D'Souza RN, Yeh FC, Fernandez-Miranda JC. Machine Learning Versus Logistic Regression Methods for 2-Year Mortality Prognostication in a Small, Heterogeneous Glioma Database. World Neurosurg X. 2019;2:100012.

36. McLaren W, Gil L, Hunt SE, Riat HS, Ritchie GRS, Thormann A, et al. The Ensembl Variant Effect Predictor. Genome Biol. 2016;17:1–14.

List of Tables:

Table 1 is placed in the manuscript at the end of the section "Model Tuning and Hyperparameter Optimization"

Table (1): Hyperparameters chosen for each of the ML algorithms and their values		
ML algorithm	**Tuned Hyperparameter**	**Values chosen**
K-Nearest Neighbors	'leaf_size': Leaf size passed to BallTree or KDTree.algorithms	1
	, 'metric': Metric to use for distance computation.	'manhattan'
	, 'n_neighbors': Number of neighbors to use.	5
	'weights': Weight function used in prediction.	'distance'
Decision Tree	'criterion': The function to measure the quality of a split.	Gini
	'max_depth': The maximum depth of the tree.	5
	'min_samples_leaf': The minimum number of samples required to be at a leaf node.	3
	'min_samples_split': The minimum number of samples required to split an internal node.	2
Random Forests	'n_estimators': The number of trees in the forest.	50
	'criterion': The function to measure the quality of a split.	Gini
	'max_depth': The maximum depth of the tree.	2
	'min_samples_leaf': The minimum number of samples required to be at a leaf node.	3
	'min_samples_split': The minimum number of samples required to split an internal node.	2
XGBoost	'max_depth': The maximum depth of the tree.	3

	'learning_rate': Step size shrinkage used in the update to prevent overfitting.	1
	'min_child_weight': Minimum sum of instance weight needed in a child.	0.5
	'n_estimators': The number of estimators used for boosting.	100
Support Vector Machine	'C': Regularization parameter.	0.0001
	'gamma': Kernel coefficient for ‘rbf’, ‘poly’, and ‘sigmoid’.	0.1
	'kernel': Specifies the kernel type to be used in the algorithm.	Rbf
Logistic Regression	'penalty': Specify the norm of the penalty.	Elastic-net
	'l1_ratio': The Elastic-Net mixing parameter, with 0 <= l1_ratio <= 1.	0.3
	'solver': Algorithm to be used in the optimization problem.	Saga
	'C': Inverse of regularization strength.	0.1

618

Table 2 is placed in the manuscript at the end of the section "Results"

Table (2): Final training and external testing Performance Metrics for All ML Algorithms												
	DT				**KNN**				**RF**			
	Trian[a]	**Test[b]**	**95% CI**		**Trian[a]**	**Test[b]**	**95% CI**		**Trian[a]**	**Test[b]**	**95% CI**	
			lower	**upper**			**lower**	**upper**			**lower**	**upper**
Accuracy	0.867	0.727	0.464	0.99	0.933	0.909	0.739	0.999	0.933	0.909	0.739	0.999
ROC-AUC	0.871	0.717	0.45	0.983	0.929	0.9	0.723	0.999	0.929	0.9	0.723	0.999
Precision	0.929	0.714	0.447	0.981	0.889	0.857	0.65	0.999	0.889	0.857	0.65	0.999
Recall (Sensitivity)	0.812	0.833	0.613	0.999	1	1	1	1	1	1	1	1
Specificity	0.929	0.6	0.31	0.89	0.857	0.8	0.564	0.999	0.857	0.8	0.564	0.999
	XGBoost				**SVM**				**LR**			
	Trian[a]	**Test[b]**	**95% CI**		**Trian[a]**	**Test[b]**	**95% CI**		**Trian[a]**	**Test[b]**	**95% CI**	
			lower	**upper**			**lower**	**upper**			**lower**	**upper**
Accuracy	0.933	0.909	0.739	0.999	0.933	0.364	0.079	0.648	0.733	0.636	0.352	0.921
ROC-AUC	0.929	0.9	0.723	0.999	0.929	0.4	0.11	0.69	0.741	0.6	0.31	0.89

Precision	0.889	0.857	0.65	0.999	0.889	0	0	0	0.833	0.6	0.31	0.89
Recall (Sensitivity)	1	1	1	1	1	0	0	0	0.625	1.0	1.0	1
Specificity	0.857	0.8	0.564	0.999	0.857	0.8	0.564	1	0.857	0.2	0	0.436

[a]: whole 75% training dataset, [b]: 25% external testing set, CI: confidence interval, ROC-AUC: receiver operating characteristics area under the curve, DT: decision tree, KNN: K-nearest neighbors, RF: random forest, XGBoost: extreme gradient boosting, SVM: support vector machine, LR: logistic regression.

Performance of multiple quantile regression with heteroscedasticity

Marwa R. Saad - Ahmed H. Youssef - Shereen H. Abdel Latif

Abstract

Since the seminal work of Koenker and Bassett (1978), quantile regression has gradually emerged as a comprehensive approach to the statistical analysis of linear and nonlinear response models, especially, when the least-square estimator has several drawbacks when dealing with Heteroscedasticity; this estimate will not be a Best Linear Unbiased Estimator (BLUE). As a result, Quantile Regression (QR) has been used instead of linear regression. We designed the Monte Carlo experiment in small, moderate, and large samples to compare the performance of the quantile regression model algorithms (simplex, interior point, and smoothing) to the linear regression model with heteroskedastic data through some information criteria such as Akaike Information Criterion corrected for small samples (AICc), Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC) for large samples and Pseudo R squared. According to the simulation study, multiple quantile regression beats multiple linear regression in the presence of heterogeneity.

Keywords: Information Criterions, Heteroscedasticity, Linear Programming, Multiple Linear Regression, Multiple Quantile Regression.

1- Introduction

The least-squares estimator has several disadvantages when dealing with heteroscedasticity. Hence, this estimate will not be a Best Linear Unbiased Estimator (BLUE) which leads to data asymmetry. Consequently, Quantile Regression (QR) has been used instead of linear regression. Quantile regression, by nature, is an extension of linear regression used when the conditions of linear regression are not met. Nevertheless, quantile regression has some substantial computational problems as its estimation algorithms are based on Linear Programming (LP), where three algorithms are mainly used: Simplex, Interior Point, and Smoothing. Quantile regression is an alternative to linear regression when dealing with heteroscedastic data. Quantile regression is a type of regression analysis used in statistics and econometrics. Whereas the method of least squares estimates the conditional mean of the response variable across values of the predictor variables, quantile regression estimates the conditional median (or other quantiles) of the response variable. Quantile regression is an extension of linear regression that is used when the conditions of linear regression are not met. Therefore, quantile regression provides the complete picture of understanding the relationship between independent variables and all parts of the dependent variable.

The main motive for quantile regression is that the standard linear regression technique summarizes the average relationship between independent variables and the response variable based on the conditional mean function. This provides only a narrow and partial scope for this relationship. Quantile regression provides the capability of describing the relationship at different points in the conditional distribution of Y.

Quantile regression has several unique features. It can be utilized to characterize a dependent variable's whole conditional distribution. The estimation is performed by linear programming (LP), which can be considered to be relatively easier than other conventional estimation algorithms. It provides a robust measure of location for non-symmetrically distributed datasets. Also, quantile regression is relatively able to grant a more efficient estimator than the OLS method when the error term normality assumption is violated (Ahmed et al., 2018).

Quantile regression differs from ordinary least squares (OLS) in some aspects:

1. Given a collection of explanatory variables, quantile regression may be used to define the full conditional distribution of a dependent variable.
2. QR provides a more comprehensive view of the independent factors' influence on the dependent variable.
3. QR has a linear programming model that allows for straightforward estimates to provide a robust measure of location.
4. When the error term is non-normal, the quantile regression estimator outperforms the least squares estimator.
5. Modeling flexibility for data with diverse conditional distributions.
6. The median regression is less sensitive to outliers than the OLS regression, (Ibrahim & Yahaya, 2015) and (Rodriguez & Yao, 2017).

2- The Essentials of the Quantile Regression Model

Basically, quantile Regression introduced by Koenker and Bassett in 1978, is a good alternative to ordinary least squares regression. While simple least squares regression minimizes the sum of squared errors, the median regression estimator minimized the total of absolute errors. By minimizing an asymmetrically weighted sum of absolute errors, quantile regression model specifies changes in the conditional quantiles (Koenker & Bassett, 1978).

A linear regression model for the τ -the conditional quantile of y_i can be expressed as

$$Q_\tau(y_i|x_i) = \sum_{i=1}^{n} \beta_{0\tau} + \beta_{i\tau}\, x_i + \varepsilon_{i\tau} \qquad (1)$$

where y_i is the i^{th} dependent variable, $\beta_{0\tau}$ is the constant term at τ^{th} conditional quantile, $\beta_{i\tau}$ is the i^{th} coefficient at τ^{th} conditional quantile, x_i is the i^{th} independent variable conditional quantile, $\varepsilon_{i\tau}$ is the random error at the τ^{th} conditional quantile, and τ is the conditional quantile of interest for $\tau \in (0,1)$.

An estimate for the τ^{th} quantile of (y) can be obtained by minimizing the next objective function:

$$\hat{\beta}_\tau = \min_{\beta} \sum_{i=1}^{n} \rho_\tau(y_i - x_i'\beta_\tau) \qquad i = 1 \ldots . n \qquad (2)$$

where the loss function ρ_τ (see Figure 1) is defined to be (τ) for i: $y_i \geq x_i'\beta$ and $(1-\tau)$ for i : $y_i < x_i'\beta$.

$\hat{\beta}_\tau$ is the estimator parameter at a particular rank of τ, so the eq (2) can be presented in an alternative form:

$$\hat{\beta}_\tau = \min_{\beta} \sum_{\substack{i=1 \\ y_i < x_i'\beta}}^{n} (1-\tau)|y_i - x_i'\beta| + \sum_{\substack{i=1 \\ y_i \geq x_i'\beta}}^{n} (\tau)|y_i - x_i'\beta| \qquad (3)$$

Since the symmetry of absolute value yields the median (by setting τ = 0.5), minimizing a sum of asymmetrically weighted absolute residuals (giving different weights to positive and negative residuals) yields the quantiles (Koenker & Hallock, 2001).

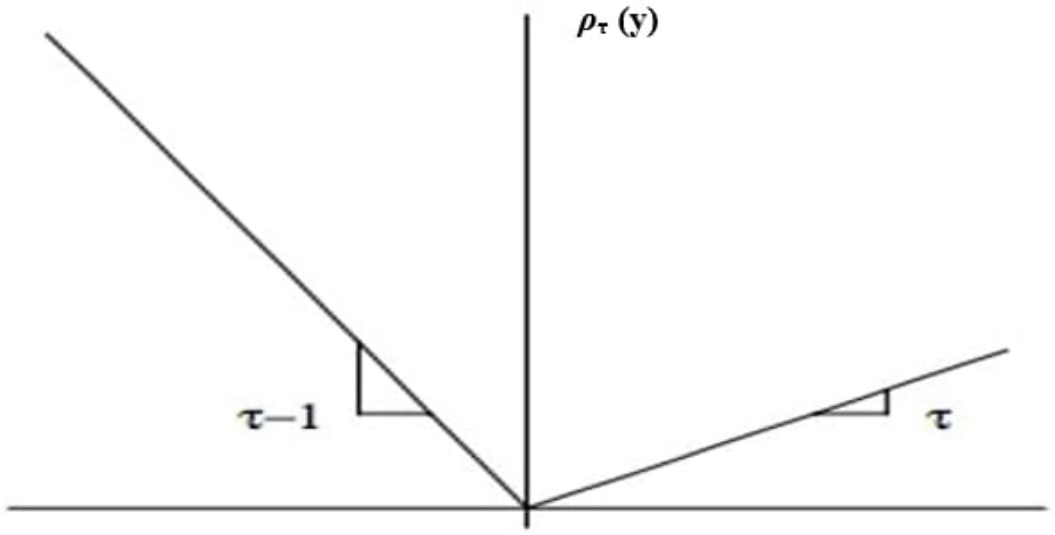


Figure (1): The Loss Function $\rho_\tau(y)$ for the τ^{th} conditional quantile.

3- Computing Aspects of Quantile Regression Model

Let $\mu = [y - X\beta]_+$, $v = [X\beta - y]_+$, $\emptyset = [\beta]_+$, and $\varphi = [-\beta]_+$, where $y = [y_1, \ldots . y_n]$,

$X = [x_1, \ldots . x_n]$, and $[z]_+$ is the non negative part of z .

Let $D_{LAR}(\beta) = \sum_{i=1}^{n} |y_i - x_i'\beta|$

$$D_{\rho_\tau}(\beta) = \sum_{i=1}^{n} \rho_\tau(y_i - x_i'\beta) \quad (4)$$

The l_1 problem $\min_\theta D_{LAR}(\theta)$ can be reformulated as:

$$\min_{\beta} \{e'\mu + e'v | y = X\beta + \mu - v, (\mu, v) \in \mathbf{R}_+^n\} \quad (5)$$

Where e denotes as n vector of ones.

Let $B = [X, -X, I, -I]$, $\theta = [\emptyset', \varphi', \mu', v']'$, and $d = [0' \; 0' \; e' \; e']'$

Where $0' = [0 \; 0 \ldots .]_p$

The reformulation presents a standard LP problem

$$\text{Subjcted to} \quad \begin{matrix} \min\limits_{\theta} & [d'\theta] \\ & B\theta = y \\ & \theta \geq 0 . \end{matrix} \qquad \text{(Primal Problem)(P)}$$

The primal problem has the dual formulation

$$\text{Subjcted to} \quad \begin{matrix} \max\limits_{d} & [y'z] \\ & B'z \leq d \end{matrix} \qquad \text{(Dual Problem)(D)}$$

This can be simplified as:

$$\max_z \{y'z | X'z = 0, z \in [-1, 1]^n\}.$$

By setting $\gamma = (\frac{1}{2}z + \frac{1}{2}e)$, $b = (\frac{1}{2}X'e)$, it becomes

$$\max_\gamma \{y'\gamma | X'\gamma = b, \gamma \in [0, 1]^n\}$$

For quantile regression, the minimization problem is $\min_\beta \rho_\tau \left(y_i - x_i'\beta\right)$ with the final dual formulation:

$$\max_z \{y'z | X'z = (1-\tau)X'e, \; z \in [0,1]^n\} \quad (6)$$

(Davino et al., 2013) (Chen &Wei, 2005)

3.1 Simplex Algorithm

Since the 1950s, median regression (regression) has been defined as linear programming problems that can be addressed quickly using a simplex approach (Chen, 2004). Barrodale and Roberts (1973) devised an efficient variant of the simplex algorithm that uses the special structure of the coefficient matrix B to solve the primary LP problem (P) in two stages. In the first stage, only pick the columns in X or -X as pivotal columns. In the second stage, only interchange the columns in (I) or (-I) as a basis or non-basic columns (Chen, 2004). The simplex algorithm is suitable for datasets with fewer than 5,000 observations and 50 variables (Chen & Wei, 2005). The algorithm obtains an optimal solution by executing these two stages interactively. By using the simplex algorithm, solutions are focused on the vertices of the constraint set of a feasible region, where the solution can be obtained in two stages:

In the first stage: We have to find an initial feasible vertex, and in the second stage, we switch from one vertex to another until optimality is achieved (Ahmed et al., 2018).

3.2 Interior Point Algorithm

The basic idea of the interior-point algorithm for solving linear programming problems is not to visit every vertex. We need only find an intelligent way of passing from one vertex to the next. The convexity of the constraint set and the linearity of the objective function assures that starting at any vertex there is a path through adjacent vertices along which the objective function decreases on each edge (Roos et al., 2005).

Alternative methods have been developed for addressing huge LP problems. Karmarkar's (1984) interior point algorithm solves a series of quadratic issues by using an ellipsoid to approximate the relative interior of the constraint set rather than proceeding from vertex to vertex around the constraint set's outer surface as the Simplex requires (Chen &Wei, 2005). Karmarkar's method was a polynomial-time algorithm with polynomial computational complexity in linear programming. Despite the fact that a single iteration of the Karmarkar algorithm is expensive, optimality is achieved after just a few iterations, making the technique computationally appealing (Zhao & Yu, 2020). The interior point algorithm is shown to perform better than the simplex algorithm in the worst-case scenario (Chen, 2004).

The interior point algorithm emerges in a variety of shapes and sizes. The primal-dual with predictor-corrector method is the most often used regression or quantile regression technique (Chen, 2004).

Let $c = y, b = (1-\tau)X'e$ and $A = X'$, the dual problem, eq (6), with a general upper bound (u) is

$$\max \quad c'z$$

Subjcted to

$$Az = b$$
$$0 \le z \le u$$

To solve this LP problem, $0 \le z \le u$ is split in to $z \ge 0$ and $z \le u$. Let (v) be the primal slack that $z + v = u$, and associated dual variable (w) with these constraints. The interior point algorithm solves the system of equations to satisfy the "Karush-Kuhn-Tuker (KKT) conditions for optimality:

(KKT)

$$\begin{cases} Az = b \\ z + v = u \\ A't + s - w = c \\ ZSe = 0 \\ VWe = 0 \\ z, s, v, w \ge 0, \end{cases} \qquad (7)$$

Where $Z = \text{diag}(z)$, (that is, $Z_{i,j} = z_i$ if $i = j$, $Z_{i,j} = 0$ otherwise), $S = \text{diag}(s), W = \text{dig}(w), V = \text{diag}(v)$ these are the conditions for feasibility, with the *complementarity* conditions $ZSe = 0$ and $VWe = 0$ added. $c'z = b't - u'w$ must occur at the optimum. Complementarity conditions force the optimal objectives of the primal and dual to be equal, $c'z_{opt} = b't_{opt} - u'w_{opt}$ (Chen & Wei, 2005).

The interior point algorithm functions by iteratively using Newton's method to locate a decent direction $(\Delta z^k, \Delta t^k, \Delta s^k, \Delta v^k, \Delta w^k)$ to move from the current solution $(z^k, t^k, s^k, v^k, w^k)$ towards a better solution (Chen & Wei, 2005).

This is accomplished in two steps. The first step is known as *an affine step*, which involves solving a linear problem using Newton's method in order to determine a direction $(\Delta z^k_{aff}, \Delta t^k_{aff}, \Delta s^k_{aff}, \Delta v^k_{aff}, \Delta w^k_{aff})$ for reducing complementarity toward zero. The second step is known as *the centering step*. It involves solving another linear system to find a centering vector

$(\Delta z_c^k, \Delta t_c^k, \Delta s_c^k, \Delta v_c^k, \Delta w_c^k)$ in order to reduce complementarity even further. The centering phase may not significantly reduce complementarity, but it does strengthen *the central path* and makes significant progress toward the optimal in the next iteration. With these two steps, then

$$(\Delta z^k, \Delta t^k, \Delta s^k, \Delta v^k, \Delta w^k) = (\Delta z_{aff}+, \Delta t_{aff}, \Delta s_{aff}, \Delta v_{aff}, \Delta w_{aff}) + (\Delta z_c, \Delta t_c, \Delta s_c, \Delta v_c, \Delta w_c) \quad (8)$$

$$(z^{k+1}, t^{k+1}, s^{k+1}, v^{k+1}, w^{k+1}) = (z^k, t^k, s^k, v^k, w^k) + \alpha\left(\Delta z^k, \Delta t^k, \Delta s^k, \Delta v^k, \Delta w^k\right), \quad (9)$$

Where, α is the *step length* assigned a value as large as possible but not so large that a z_i^{k+1}, s_i^{k+1}, v_i^{k+1}, or w_i^{k+1} is "too close" to zero.

The Predictor-Corrector variant usually takes less iteration to reach the optimum, although requires solving two linear equations instead of one. In the affine step and the centering step, factorization of the $(X'[\emptyset^k]^{-1}X)$ matrix, where $\emptyset^k$ is a diagonal matrix computed in k^{th} iteration, takes the majority of computing time when solving the linear systems. However, the additional overhead of calculating the second linear system is small, as the factorization of the $(X'[\emptyset^k]^{-1}X)$ matrix has already been performed to solve the first linear system (Chen & Wei, 2005).

3.3 Smoothing Algorithm

The finite smoothing algorithm was introduced by Clark and Osborne (1986), Madsen and Nielsen (1993) for the l_1 regression. It can be naturally extended to compute regression quantiles as follow:

The non-differentiable function (3) can be approximated by the smooth function

$$D_{\gamma,\tau}(\beta) = \sum_{i=1}^{n} H_{\gamma,\tau}\left(r_i(\beta)\right), \quad (10)$$

where $r_i(\beta) = y_i - x_i'\beta$ and

$$H_{\gamma,\tau}(t)\begin{cases} t\,(\tau-1) - \frac{1}{2}(\tau-1)^2\gamma & \text{if } t \le (\tau-1)\gamma \\ \frac{t^2}{2}\gamma & \text{if } (\tau-1)\gamma \le t \le \tau\gamma \\ t\tau - \frac{1}{2}\tau^2\gamma & \text{if } t \ge \tau\gamma \end{cases} \quad (11)$$

See Figure 2 for a plot of $H_{\gamma,\tau}$ and ρ_τ

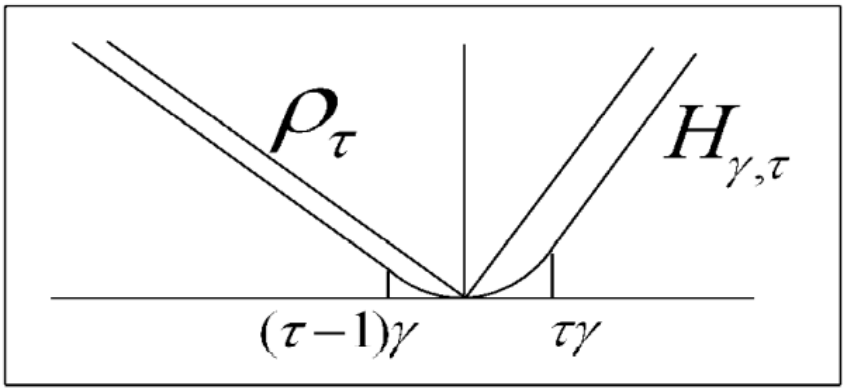


Figure 2: Objective Functions $H_{\gamma,\tau}$ and ρ_τ

The function $H_{\gamma,\tau}$ is determined by whether $r_i(\beta) \le (\tau-1)\gamma, r_i(\beta) \ge \tau\gamma$, or $(\tau-1)\gamma \le r_i(\beta) \le \tau\gamma$. These inequalities divide R^P into subregions separated by the parallel hyperplanes $r_i(\beta) = (\tau-1)\gamma$ and $r_i(\beta) = \tau\gamma$. The set of all such hyperplanes is denoted by $B_{\gamma,\tau}$:

$$B_{\gamma,\tau} = \{\beta \in R^P | \exists i : r_i(\beta) = (\tau-1)\gamma \text{ or } r_i(\beta) = \tau\gamma\} \quad (12)$$

Define the sign vector $s_{\gamma,\tau}(\beta) = (s_1(\beta), s_2(\beta), \dots\dots\dots s_n(\beta))'$by

$$s_i = s_i(\beta)\begin{cases} -1 & \text{if } r_i(\beta) \le (\tau-1)\gamma \\ 0 & \text{if } (\tau-1)\gamma \le r_i(\beta) \le \tau\gamma \\ 1 & \text{if } r_i(\beta) \ge \tau\gamma \end{cases}$$

And introduce $w_i(\beta) = 1 - s_i^2(\beta)$. Thus,

$$D_{\gamma,\tau}(\beta) = \frac{1}{2}\gamma r' W_{\gamma,\tau} r + v'(s) r + c(s), \quad (13)$$

where $W_{\gamma,\tau}$ is the diagonal n by n matrix with diagonal elements $w_i(\beta)$,

$$v'(s) = (s_1(2\tau-1)s_1 + 1)/2, \dots\dots s_n((2\tau-1)s_n + 1)/2),$$

$$c(s) = \sum[\tfrac{1}{4}(1-2\tau)\gamma s_i - 1/4 s_i^2(1-2\tau+2\tau^2)\gamma], \text{and } r(\beta) = (r_1(\beta), \dots\dots, r_n(\beta))'.$$

The gradient of $D_{\gamma,\tau}$ is given by

$$D^{(1)}_{\gamma,\tau}(\beta) = -X'\left[\frac{1}{\gamma}W_{\gamma,\tau}(\beta)r(\beta) + g(s)\right] \quad (14)$$

And for $\beta \in R^p$ / $B_{\gamma,\tau}$ the Hessian exists and is given by

$$D^{(2)}_{\gamma,\tau}(\beta) = \frac{1}{\gamma}X'W_{\gamma,\tau}(\beta)X. \quad (15)$$

The gradient is a continuous function in R^p , where the Hessian is pricewise constant.

The smoothing algorithm for minimize $D_{\rho\tau}$ is based on minimizing $D_{\gamma,\tau}$ for a set of decreasing γ. The main advantage of the smoothing algorithm is that the solution $\beta_{0,\tau}$ can be detected when $\gamma > 0$ is small enough. It is not necessary to let γ converge to zero in order to find a minimizer of $D_{\rho\tau}$. For every new value of γ, information from the previous solution is utilized. The algorithm stops before going through the whole sequence of γ, which is generated by the algorithm itself. The convergence is indicated by no change of the status while γ goes through this sequence. Therefore, the algorithm is usually called the finite smoothing algorithm. For a given threshold γ, a modified Newton iteration is used. Starting from an initial estimator $\beta_\tau(\gamma)$., the search direction **h** is found by solving the equation.

$$D^{(2)}_{\gamma,\tau}\ \left(\beta_\tau(\gamma)\right)h = D^{(1)}_{\gamma,\tau}\ \left(\beta_\tau(\gamma)\right) \quad (16)$$

If $D^{(2)}_{\gamma,\tau}$ is full rank, **h** is the solution to equation (16). Otherwise, if the system (16) is consistent, a consistent basic solution is used. If the system (16) is not consistent, a Marqurated perturbation is used to $D^{(2)}_{\gamma,\tau}$ .After **h** is computed, if $s_\gamma\left(\beta_\tau(\gamma)\right) + \mathbf{h} = s_\gamma\left(\beta_\tau(\gamma)\right)$, then $\beta_\tau(\gamma)$ + **h** minimizes$D_{\gamma,\tau}$, otherwise a line search is performed.

Another advantage of the smoothing algorithm is that when solving (16) for a new threshold(γ') , the factorization only needs a partial update instead of a full update as in the Interior Point algorithm. This saves time with large *p*.

Each of the previous three algorithms has its own advantages. None of them can fully dominate the others. Although the simplex algorithm is slow with a large number of

observations, it is the most stable of the algorithms. For various kinds of data, especially data with a large portion of outliers and leverage points, the Simplex algorithm always find a solution, while the other two algorithms might fail with floating point errors. The interior point algorithm is very fast for *slender data* sets, which have a large number of observations and a small number of covariates. The finite smoothing algorithm is simple in theory for computational issues for quantile regression and has the advantage of computing speed with a large number of covariates (Chen, 2007).

4- Methodology of the Simulation

The performance of multiple linear regression and multiple quantile regression with the three algorithms mentioned above is compared. some information measures are used (model selection criteria) to select the suitable model for a given data set such as AICc for small samples, AIC, and BIC for large samples, in addition to measures of goodness such as adjusted R^2 for multiple linear regression and Pseudo R^2 for multiple quantile regression. For this purpose, in this paper, the Monte Carlo experiment has been designed in small, moderate, and large samples to compare the performance of the quantile regression model algorithms (simplex, interior point, and smoothing) to the linear regression model with heteroskedastic data. The simulated models are generated with the settings of the following steps:

1. Generate the parameters (β_0 , $\beta_1$andβ_2)
2. Generate the independent variable (x_1) from a Uniform distribution ranging from 1 to 6
3. Generate the independent variable (x_2) from exponential distribution with rate (3)
4. Generate the response variable was defined as $y_i = 14 + 0.4x_1 + 0.2\ x_2 + \varepsilon_i$.
5. Generate the error term from a Normal distribution with mean (0) and variance is $(0.5 + 0.03 * x_1^2 + 0.1 * x_2^2)$.
6. The values of the sample size (n) were chosen to be; n=15, 25, 30, 50, 200,500, and 700.
7. The value of conditional quantiles were be chosen to be τ =0.1, 0.25, 0.5, 0.75, and 0.9.
8. Estimate the generated model by OLS.
9. Estimate the generated quantile regression model by the simplex algorithm, interior point algorithm, and smoothing algorithm at different conditional quantiles.
10. Estimating the statistical metrics of interest for each model.

11. The replication of the Monte Carlo is 1000 times

Four measures are used to measure the performance of the three algorithms of multiple quantile regression compared to multiple linear regression as follows:

4.1 Akaike Information Criterion

The Akaike Information Criterion (AIC) is an estimator of prediction error and thereby the relative quality of statistical models for a given set of data. Given a collection of models for the data, AIC estimates the quality of each model, relative to each of the other models. Thus, AIC provides a means for model selection. The Akaike information criterion (AIC) is a mathematical method for evaluating how well a model fits the data it was generated. In statistics, AIC is used to compare different possible models and determine which one is the best fit for the data.

$$\mathrm{AIC} = -2\,\mathrm{l}(\hat{\theta}|\mathrm{y}) + 2K$$

where

- K: The number of model parameters (the number of variables in the model plus the intercept)
- $\mathrm{l}(\hat{\theta}|\mathrm{y})$: The log-likelihood at its maximum point of the estimated model.

Akaike Information Criterion for Small Sample

Akaike's Information Criterion (AICc): The information score of the model (the lower-case "c") indicates that the value has been calculated from the AIC test corrected for small sample sizes). The smaller the AICc value, the better the model fit, the smaller the value of AIC is, the better the model is.

For small sample sizes ($\mathrm{n} \leq 30$), use AICc:

$$\mathrm{AICc} = \mathrm{AIC} + (2\mathrm{K}(\mathrm{K}+1)/(\mathrm{n}-\mathrm{K}-1))$$

Where n is the sample size. For more details see (Pho et al., 2019).

4.2 Bayesian Information Criterion

The Bayesian Information Criterion, often abbreviated BIC, is a metric used to compare the goodness of fit of different regression models. In practice, we fit several regression models to

the same dataset and choose the model with the lowest BIC value as the model that best fits the data. The Bayesian information criterion (BIC) or Schwarz information criterion is a criterion for model selection among a finite set of models. Models with lower BIC are generally preferred. It is based partly on the likelihood function and is closely related to the Akaike Information Criterion (AIC).

The BIC is formally defined as

$$\mathrm{BIC} = -2\,\mathrm{l}(\hat{\theta}|\mathrm{y}) + \mathrm{kln(n)}$$

where

- n: The sample size
- k : The number of parameters estimated by the model including the intercept
- $\mathrm{l}(\hat{\theta}|\mathrm{y})$: The log-likelihood at its maximum point of the estimated model.

The rule of selection: the smaller the value of BIC is, the better the model is (Pho et al., 2019).

4.3 Adjusted R squared

Adjusted R squared (R^2) is similar to R squared. It is a statistical measure that represents the proportion of the variance for the dependent variable that is explained by independent variables in the regression model.

$$\text{Adjusted } R^2 = 1 - \frac{(1 - R^2)(n - 1)}{n - p - 1}$$

where

R^2: Sample R squared

n: Sample size

p: Number of independent variable (Miles, 2005)

4.4 Pseudo R-Squared

The pseudo R square of QR differs from OLS regression because it is based on the minimization of an absolute weighted sum (not an unweighted sum of squares as in OLS). In addition, this

approach aims to provide an estimation of the quantile conditional distribution of the dependent variable (not an expected conditional distribution as in OLS). It follows that the equivalent of the residual sum of squares is, for each considered quantile τ, the residual absolute sum of weighted differences between the observed dependent variable and the estimated quantile conditional distribution. The pseudo R^2 can be considered as an index comparing the residual absolute sum of weighted differences using the selected model with the residual absolute sum of weighted differences using a model with only the intercept. The pseudo R^2 can be computed as follows:

$$\text{pseudo } R_\tau^2 = 1 - \frac{RASW_\tau}{TASW_\tau}$$

As $RASW_\tau$ is always less than $TASW_\tau$, the pseudo R_τ^2 ranges between 0 and 1 (Davino et al.,2013).

Table 1 (a): Comparison of Multiple Linear Regression and Multiple Quantile Regression with Three Algorithms in Case of Heteroscedasticity at Sample Sizes (N=15, 25 and 30).

Sample size	Criteria	Multiple Linear Regression	The Conditional Quantiles and the Three Algorithms for Multiple Quantile Regression														
			$\tau = 0.1$			$\tau = 0.25$			$\tau = 0.5$			$\tau = 0.75$			$\tau = 0.9$		
			Simplex	Interior point	Smoothing	Simplex	Interior point	Smoothing	Simplex	Interior point	Smoothing	Simplex	Interior point	Smoothing	Simplex	Interior point	Smoothing
١٥	AICc	١٢٠.٦٨	١٣٢.٧٣	١٣٥.١٠	١٣٥.١٠	١١٧.٥٣	١١7.90	١١7.90	١١٤	١١٥.18	١١٥.18	1٢٤.٣٥	١2٧.٧٢	١2٥	١٢0..٨١	١٢6.١٨	١٢٥
	Adjusted R^2 for MLR	٠.٥٢٩٨٥															
	Pseudo R^2 for MQR	----------	٠.٧٢٦٦٠	0.71656	0.71656	٠.٥٣٩٩٩٧	٠.٤٣٩٩٩٧	٠.٤٣٩٩٩٧	0. ٧٩٧٢٥٧	0. ٧٩٧٢٥٧	0. ٧٩٧٢٥٧	٠.٧٥٣٣٥	٠.7٠٧٥٣	٠.73٨٧٥	0.78212	٠.٧٧٦٩٥	٠.٧٧٩٩٥
25	AICc	١٢٢.٥٤٥٥٧٧٩	١٣٥.٦٠	١٣٩.٠٢	١٣٩.٠٢	١٢٩.٨٣	١٣٠.٢٤	١٣٠.٢٤	١١٦.٧٧	١١٦.٧٧	١١٦.٧٧	116	117.939	116.353	12٨	129.326	12٨.741
	Adjusted R^2 for MLR	٠.٦٧٠٠٧٤															
	Pseudo R^2 for MQR		٠.٤٤٧٤٢٢	٠.٤١٧٤٢٢	٠.٤١٧٤٢٢	٠.٦8 ٨٩٤٦	٠.٦١٨٩٤٦	٠.٦١٨٩٤٦	٠.78٢٧٤٤٥	٠.78٢٧٤٤٥	٠.78٢٧٤٤٥	٠.٦٣٨٩٨٤٣	٠.٦٢٨٩٨٤٣	0.625821	٠.٦٩٧٠٧٣	0.٥829808	0.685245
30	AICc	١٩٢.٢٧٥	٢١٧.٨٤٧	٢٣٣.٤٨٣	٢٣٣.٤٨٣	١٩٢.١٨٨	١٩٢.٣٢٥	١٩٢.٣٢٥	١٨٠.٤٣٧	١٨٠.٤٣٧	١٨٠.٤٣٧	١٣٩.٢٤٦	٢٠٠.٨٨٣	٢٠٠	٢٠٠	204.680	٢٠٠.٠٤٣
	Adjusted R^2 for MLR	٠.٥٩٢٨٦٥															
	Pseudo R^2 for MQR		٠.٦٤٢٨٥١٤	٠.٦٤125١٤	٠.٦٤0258	٠.٧٩٨٣٢	٠.٧٧٩٣٢	٠.٧٧٩٣٢	٠.٧٨١٠٤٣	٠.٧٨١٠٤٣	٠.٧٨١٠٤٣	٠.٧٧٧٤٥٨	٠.٧٢٧٤٥٨	0.752145	٠.٦69٦١	0.٦6804	0.669251

Table 1(b): Comparison of Multiple Linear Regression and Multiple Quantile Regression with Three Algorithms in Case of Heteroscedasticity at Sample Sizes N= (50, 200, 500 and700)

Sample size	Criteria	Multiple Linear Regression	The Conditional Quantiles and the Three Algorithms for Multiple Quantile Regression													
			$\tau = 0.1$			$\tau = 0.25$			$\tau = 0.5$			$\tau = 0.75$			$\tau = 0.9$		
			Simplex	Interior point	Smoothing	Simplex	Interior point	Smoothing	Simplex	Interior point	Smoothing	Simplex	Interior point	Smoothing	Simplex	Interior point	Smoothing
50	AIC	٢٤٦.١٩٧	٢٥٨.٤٤٩	٢٥٨.٤٤٩	٢٥٨.٤٤٩	٢٣9.٦٦٢	٢٣9.٦٦٢	٢٣9.٦٦٢	٢٣٩.672	٢٣٩.672	٢٣٩.672	٢٦٠.١١٩	٢٦٨.٥١٩	٢٦٠	٢٨١.٩٣٩	٢٨٦.٩٣٩	٢٨٠
	BIC	٢٧٧.٠٥٤٣	٢٦2.٠٥٤	٢٦2.٠٥٤	٢٦2.٠٥٤	٢٦٠.٠٥٤	٢٦٠.٠٥٤	٢٦٠.٠٥٤	٢٦٠.٠٢٥	٢٦٠.٠٢٥	٢٦٠.٠٢٥	٢٦٠.٠٥٥	٢٦٦.٢٥٨	٢٦٠.12	٢٦٠.٠٥٤	٢٦٨.١٢٠	٢٦٠
	Adjusted R^2 for MLR	٠.٦٢٢٠٠٩٦															
	Pseudo R^2 for MQR		٠.٦٨٨٠٤٣٤	٠.٦٨٨٠٤٣٤	٠.٦٨٨٠٤٣٤	٠.٧٤٨١٤٤٢	٠.٦٦٨١٤٤٢	٠.٦٦٨١٤٤٢	٠.٩٣٩٤٠٢٢	٠.٩٣٩٤٠٢٢	٠.٩٣٩٤٠٢٢	٠.٨٣٩٠٢٧٧	٠.٨٢٩٠٢٧٧	٠.٨٥٢٤٨١	٠.٩١٨٦١٧١	٠.٩٠٩٠٩٥	٠.٩٢٥١٤٣
200	AIC	57٧.18٥	680.863	680.863	680.863	595.847	595.847	595.847	571.102	571.102	571.102	601.985	605.985	٥9٢.١٢	64٤.799	646.799	٦٤٠.١٢
	BIC	5٩٠.682	590.682	590.682	590.682	589.682	589.682	589.682	558.682	558.682	558.682	58٠.682	58٢.682	٥٨٠	58٢.682	58٦.٩٥٥٢	٥٨٠
	Adjusted R^2 for MLR	0.٦34483															
	Pseudo R^2 for MQR		0.٥٦49549	0.٥649549	0.٥649549	0.٨٨2492	0.٨32492	0.٨32492	0.٩٩377	0.٩٩377	0.٩٩377	0.٩٢7836	0.٩17836	٠.٩٥٣٦٢٤	0.٧43341	0.٦65439	٠.٧٥٨١٢٣
500	AIC	4٨8.443	452.251	452.251	452.251	441.251	441.251	441.251	4٢2.2٢٢	4٢2.2٢٢	4٢2.2٢٢	50٠.١٤0	505.560	٥٠٠	655.88٧	657.٣٧٥	٦٥٢.١٢٥
	BIC	521.451	665.450	665.450	665.450	652.312	652.312	652.312	325.451	325.451	325.451	46٤.194	465.306	٤٦٠.١٢٣	465.202	480.451	٤٦٠
	Adjusted R^2 for MLR	0.870373															
	Pseudo R^2 for MQR		0.878542	0.776842	0.878542	0.851111	0.851111	0.851111	0. 959444	0. 959444	0. 959444	0.95151	0.951415	٠.٩٥٢١٤٥	0.8347431	0.821758	٠.٨٥٦٤1١
700	AIC	495.966	914.415	914.415	914.415	718.654	718.654	718.654	304.709	304.709	304.709	405.0918	410.0918	٤٠٠.٠٠	550.4182	550.4189	٥٠٠.١٢٣٥
	BIC	859.421	958.421	958.421	958.421	958.419	958.419	958.419	851.201	851.201	851.201	958.420	958.427	٩١٢.١٢٥	958.419	958.422	٩١٠.٢٤٥
	Adjusted R^2 for MLR	0.9731518															
	Pseudo R^2 for MQR		0.9802944	0.9802944	0.9802944	0.8875212	0.8875212	0.8875212	0.990824	0.990824	0.990824	0.843109	0.843109	٠.٨٥٤٢١٠	0.9646452	0.957741	٠.٩٧٥٢١٥

5- The Simulation Study's Results

Based on the results, table 1(a) shows that with n ≤ 30, the value of AICc with the simplex algorithm is lower than the value of the same measure with the other algorithms, at all conditional quantiles of the distribution which means that the simplex algorithm outperforms both interior-point and smoothing algorithms. At the upper conditional quantiles starting from (τ = 0.75 and 0.9), the results showed that the value of AICc with the smoothing algorithm is able to compete with the simplex algorithm to some extent but the simplex algorithm is still the best in the upper quantiles.

According to Table 1 (b) with n >30, it is noticeable that each AIC and BIC are equal in (τ = 0.1, 0.25, and 0.5), this ensures the performance for the three algorithms is equal at lower quantiles. Conversely, at the higher conditional quantiles (τ = 0.75 and 0.9) the value of both AIC and BIC is lowest in the smoothing algorithm followed by the simplex algorithm and then the interior point algorithm which means the smoothing algorithm performs excellently at the upper conditional quantiles and outperforms.

Overall, in multiple quantile regression, to estimate the model with a small sample size, the simplex algorithm is the best. However, the smoothing algorithm outperforms with a large sample size, especially at the upper conditional quantiles (τ = 0.75 and 0.9).

In addition, the coefficient of determination " R^2" for multiple linear regression/ pseudo R^2 for multiple quantile regression" is an approach to assessing which model can explain the variation of the response variable due to covariates. From scenario three, it is noticed that the multiple quantile regression is able to explain the variation of the dependent variable more than the multiple linear regression the pseudo R^2 at conditional quantile 0.5 is higher than the adjusted R^2 ratio for multiple linear regression.

On the other hand, to compare the multiple quantile regression algorithms: In the small sample sizes n ≤ 30, in the extreme quantiles, the simplex algorithm can explain the variation in the model more than each interior point and smoothing algorithm. As the sample size increases n =50, it is noted that the variation of the response variable can be explained clearly by both the simplex algorithm and the smoothing algorithm. As the sample size increases n = 500, and 700, the smoothing algorithm outperforms, especially at extreme conditional quantile (0.9).

Besides, according to Table 1(a) and Table 1 (b) Across all sample sizes, the AICc, AIC, and BIC in the multiple quantile regression at the conditional ($\tau = 0.5$) are smaller than their value in the multiple linear regression, which means that the multiple quantile regression model at ($\tau = 0.5$) is capable of representing the data more than the multiple linear regression model. Moreover, at the median with the other conditional quantiles, across sample sizes with the three algorithms used above, it is noted that the values of AICc, AIC, BIC, and pseudo R^2 are more accurate than their value at other conditional quantiles. This ensured that the data in the median is more homogeneity than in the extreme quantiles.

6- The Simulation Study's Conclusions

Finally; each AICc, AIC, and BIC provide more perfect results in the multiple quantile model than in the multiple linear model for all sample sizes. That is the multiple quantile regression is the suitable model to fit the data with the heteroscedasticity problem. For details about algorithms, in small sample sizes, the simplex algorithm provides the most accurate results of other algorithms at all conditional quantiles. The accuracy for the three algorithms is relatively equal at middle quantiles (τ =0.25, 0.5, and 0.75) in large sample sizes. In addition, in large sample sizes, the smoothing algorithm outperforms at the upper conditional quantiles (τ =0.75, and 0.9).

7- References

1- Ahmed, H., Rady, E., & Sheikh, A. (2018). Estimating Quantile Regression for Dynamic Panel Data Using Two-Stage Approach. *Far East Journal of Mathematical Sciences* ,103 (1),451-465

2- Barrodale, I., & Roberts, F. D. (1973). An improved algorithm for discrete $l_1$linear approximation. *SIAM Journal on Numerical Analysis*, *10*(5), 839-848.

3- Chen, C. (2004). An adaptive algorithm for quantile regression. In *Theory and applications of recent robust methods*, 39-48

4- Chen, C. (2007). A finite smoothing algorithm for quantile regression. *Journal of Computational and Graphical Statistics*, *16*(1), 136-164.

5- Chen, C., & Wei, Y. (2005). Computational issues for quantile regression. *Sankhyā: The Indian Journal of Statistics*, 399-417.

6- Clark, D. I., & Osborne, M. R. (1986). Finite algorithms for Huber's M-estimator. SIAM journal on scientific and statistical computing, 7(1), 72-85.

7- Davino, C., Furno, M., & Vistocco, D. (2013). *Quantile regression: theory and applications* (Vol. 988).

8- Ibrahim, A., & Yahaya, A. (2015). Analysis of quantile regression as alternative to ordinary least squares. *International Journal of Advanced Statistics and Probability*, *3*(2), 138-145.

9- Karmarkar, N. (1984). A new polynomial-time algorithm for linear programming. In Proceedings of the sixteenth annual ACM symposium on Theory of computing (pp. 302-311).

10- Koenker, R., & Bassett Jr, G. (1978). Regression quantiles. *Econometrica: journal of the Econometric Society*, 33-50

11- Koenker, R., & Hallock, K. F. (2001). Quantile regression. *Journal of economic perspectives*, *15*(4), 143-156.

12- Madsen, K., & Nielsen, H. B. (1993). A finite smoothing algorithm for linear $l_1$estimation. SIAM Journal on Optimization, 3(2), 223-235.

13- Miles, J. (2005). R-squared, adjusted R-squared. *Encyclopedia of statistics in behavioral science*.

14- Pho, K. H., Ly, S., Ly, S., & Lukusa, T. M. (2019). Comparison among Akaike information criterion, Bayesian information criterion and Vuong's test in model selection: A case study of violated speed regulation in Taiwan. Journal of Advanced Engineering and Computation, 3(1), 293-303.

15- Rodriguez, R. N., & Yao, Y. (2017), five things you should know about quantile regression. In Proc. The SAS global forum 2017 conference, 2-5.
16- Roos, C., Terlaky, T., & Vial, J. P. (2005). Interior point methods for linear optimization.
17- Zhao, P., & Yu, S. (2020). An Improved Interior Point Algorithm for Quantile Regression. *IEEE Access*, *8*, 139647-139657