Memorial University of Newfoundland

Faculty of Engineering & Applied Science

Probabilistic Methods in Engineering

ENGI 9411

PROJECT TITLE

A Statistical Regression Methodology for Improving Robustness of Rotor Bar Fault Detection of Induction Motor Using Three- phase Current Estimation under Random Loading Conditions

Submitted by (Group 3):

Shafi Md Kawsar Zaman (ID: 201790485)

Hla- U- May Marma (ID: 201891620)

Abdullah Al Baki Sifat (ID: 201891725)

Date of Submission**:** November 13, 2018

Table Of Contents **Page**

List of Figures **Page**

List of Tables **Page**

EXECUTIVE SUMMARY

Condition Monitoring and Fault Detection of Induction Motors are of paramount importance in increasing substantial operational efficiency and reducing consequential damages and failures in industries. Early detection of faults are necessary to maintain operational continuity of Induction Motors. Recently, various machine learning algorithms are being used to train induction motors to detect incipient faults and localize the faulty motor before it causes the total industrial process to halt. Features extracted from three- phase current signatures help these algorithms to train up the machines. Only a few load conditions can be considered to collect the current data as considering all the load cases experimentally is both time consuming and economically unfeasible. Using three- phase currents for broader loading range would surely improve the robustness of this method as the motor would then be trained for detecting incipient faults for any load conditions. In this Project, a Statistical Regression Model is proposed for estimating three- phase currents for any random load between no load (0%) and full load (100%) cases, when a three-phase induction motor is subjected to faulty rotor bar conditions. Experimental test data of three- phase currents were collected in an experiment by testing an induction motor with 1 broken rotor bar faults under different loading conditions. Normality Test of collected data is carried out using Kolmogorov- Smirnov (K-S) test in IBM- SPSS Statistics Software. Then exact distributions of the three- phase currents' data are analyzed using MATLAB's Distribution Fitter Toolbox. After a detailed case study, it is found that the proposed methodology can be used to predict three- phase currents' features in faulty rotor conditions for any arbitrary load between 0% and 100%. Finally, predicted and experimental data are compared to compute the accuracy of the proposed methodology.

Chapter I
INTRODUCTION

1.1.Introduction:

Induction motors are the foremost necessary electrical machineries in modern industrial world. Induction motor operation has become more automatic, precise and efficient nowadays. Any types of faults in motor reduces its operational efficiency, leading to significant economic losses, rapid motor failure, and possible human casualties. Therefore fault diagnosis and condition monitoring of induction motor has become of great concern in order to maintain reliability and continuity of operation. Condition monitoring is a technique of observing the operating characteristics of machine and constructing a reliable mechanism for fault detection in order to avoid any hazardous situations.

Reference [1] reveals that bearing, winding and broken rotor bar faults contributes to 88% of the faults in motors where 10% faults are rotor bar faults [2]. Therefore, early diagnosis of broken rotor bar fault is very important in squirrel cage induction motor. A large current can flow through the bars at the starting, running, load changing, voltage fluctuation, or torque oscillation times. Large current causes overheating in the rotor bars and this thermal stress cracks or breaks the rotor bars. This type of faults are known as broken rotor bar faults.

The fault diagnosis methods can be classified into four categories: 1) model-based, 2) signal-based, 3) knowledge-based, and 4) hybrid/active methods [3]. The knowledge-based methods are known as data-driven methods which require an enormous number of historic data for establishing the fault modes about the systems without priori known models or signal patterns [4]. Hence, this method is very suitable for explicit models or signal symptoms.

Recent stream of fault diagnosis of induction motor is based on different types of advanced signal processing methods, such as the Hilbert transform, Wigner–Ville distribution, Stochastic resonance and Wavelet transform. One of the most powerful online methods for induction motor faults diagnosis is current signature analysis of induction motor. With the increasing storage capability of fault pattern records of condition monitoring, fault diagnosis study based on induction motor current signature analysis through statistical features has attracted considerable research attention in recent years. Using these features, various machine learning algorithms are being used to train up the induction motor to detect early incipient faults. Presently, mostly used machine learning method for three-phase induction motor's rotor fault detection is artificial neural networks (ANNs), whose inputs are statistical features of three- phase current signatures in the time domain [5]. Machine learning methods for induction motor fault diagnosis have already been developed but the robustness of fault diagnosis by analyzing current signatures is still a topic of research. In those already developed methods, the current signature data for only a few load conditions are considered as considering all load cases experimentally is time consuming as well as economically unfeasible.

The project report is arranged as follows: in Chapter II, collected experimental test data of three- phase currents of induction motor during rotor bar fault condition is processed using MATLAB 2018b Software; a regression methodology for improving robustness of rotor bar fault detection of induction motor is proposed in Chapter III; in Chapter IV, goodness-of-fit for the proposed regression model is analyzed; the proposed regression model is validated through case study in Chapter V; and conclusions are drawn in Chapter VI.

1.2.Objectives of the project

- To develop a regression model for predicting the features (mean and standard deviation) of three-phase current signatures for any arbitrary loading of an induction motor subjected to 1 broken rotor bar fault.
- To optimize the predicted data of current mean and current standard deviation by comparison with experimental ones.
- To improve the robustness of Rotor Bar Fault Detection of Induction Motor.

Chapter II
DATA PROCESSING

2.1. Collection of Data

Experimental Test data of three- phase currents I_1, I_2, I_3 of a **Leeson 101649.00, Standard Eff., 0.25 HP, 1725 RPM, 208-230/460V, S56, TENV, Rigid** Induction Motor under Rotor Bar Fault with 1 Broken Rotor Bar is collected[1]. The Loading cases were considered as 10%, 30%, 50%, 70%, 85%, and 100% of the full load, during collection of the three- phase current data from experiment. As there are some fault transients in the initial phase of data collection which damps out within about 5 seconds, all of these currents' data within the time span of 5 seconds to 60 seconds are taken into account for this project. We have picked the means and standard deviations of each phase currents for 50% loading among the various statistical features that are generally used for training Induction Motors using machine learning algorithms. Comparison of predicted best means and standard deviations of all the three- phase currents with experimental ones for 50% loading depicted that this model can be used for predicting statistical features for any arbitrary loading which will make the machine learning more robust.

2.2. Processing of Data

Means and Standard Deviations of the collected three- phase current data under 1 broken rotor bar fault are calculated using MATLAB 2018b Software. Normality of these data is tested using IBM- SPSS Statistics 25 Software by Kolmogorov- Smirnov (K- S) Test. Results from this test depict that current data of each phase is not normally distributed. Actual Distributions of the Current data (I_1, I_2, I_3) are then tested using MATLAB 2018b Software's Distribution Fitter Toolbox which show us that each set of phase current data are Bi- modally (two different peaks) distributed. From this analysis, we have come to the conclusion that there are two normally distributed data sets within each set of phase current data. As the three- phase currents propagate in a sinusoidal manner with respect to time (with some unbalanced transients due to fault), there should be two different normally distributed current data (for positive and negative current modes) in data set of each phase current. Table I shows the values of statistical features

(Means and Standard Deviations) of Phase Current I_1, I_2, I_3. These two normal distributions would have been identical if the three- phase currents were balanced. We have developed the Regression Model using the following load percentage cases: 10%, 30%, 70%, 85%, and 100%. Statistical Features of three- phase currents from experimental 50% load percentage case is left for carrying out comparison with the predicted features in the Result Analysis section. Table I shows values of features (means and standard deviations) of experimentally collected three phase currents under different loading percentages.

TABLE I

VALUES OF FEATURES (MEANS AND STANDARD DEVIATIONS) OF EXPERIMENTALLY COLLECTED THREE PHASE CURRENTS UNDER DIFFERENT LOADING PERCENTAGES

Phase Current	Statistical Feature	10% Loading	30% Loading	70% Loading	85% Loading	100% Loading
I_1	Mean	0.00218885 81084187	0.00213455 45271559	0.00175868 36856555	0.00149859 55136808	0.00146471 85170065
	Standard Deviation	0.51341789 55465929	0.54853631 25356849	0.69350026 45976272	0.75717809 61699230	0.83353508 50032065
I_2	Mean	0.00169651 05099675	0.00163447 80314069	0.00125057 09264967	0.00136276 30455443	0.00103808 08275156
	Standard Deviation	0.49228477 66539311	0.53704650 56796397	0.70178628 76521644	0.76975075 42587756	0.84745997 06353779
I_3	Mean	0.00267763 13973116	0.00296529 26860029	0.00423722 67270223	0.00404947 79156283	0.00462135 01662619
	Standard Deviation	0.53909496 39738512	0.58555404 15378424	0.74324596 45459280	0.81337538 29521263	0.88823359 33836403

Figs. 2.1, 2.2, and 2.3 show the Histograms of currents of phases 1, 2, and 3 respectively for loading cases of 10%, 30%, 70%, 85%, and 100% of Full- load. From the Histograms, it is evident that all the experimentally collected phase currents have two different peaks thus the distributions are Bi- modal.

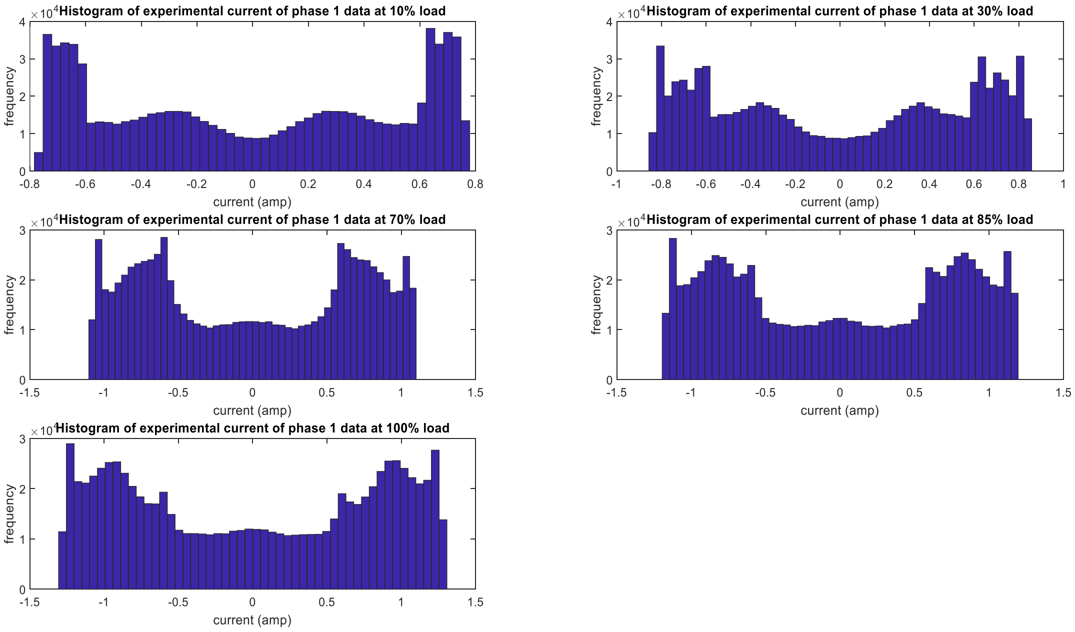

Fig. 2.1 Histograms of current of phase 1, I_1 for loading cases of 10%, 30%, 70%, 85%, and 100% of Full- load

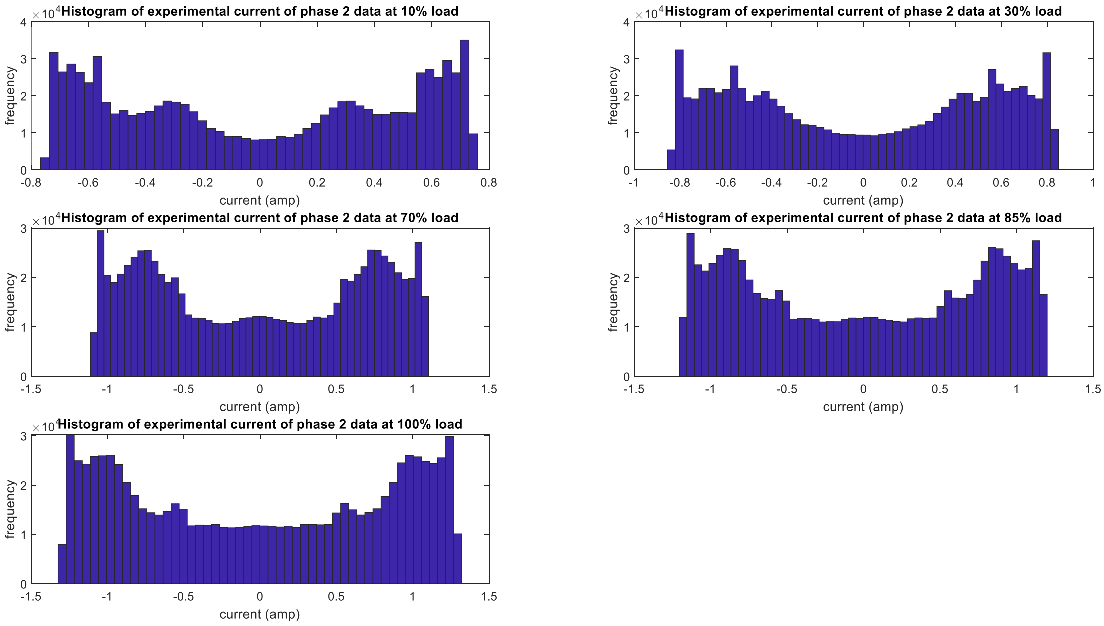

Fig. 2.2. Histograms of current of phase 2, I_2 for loading cases of 10%, 30%, 70%, 85%, and 100% of Full- load

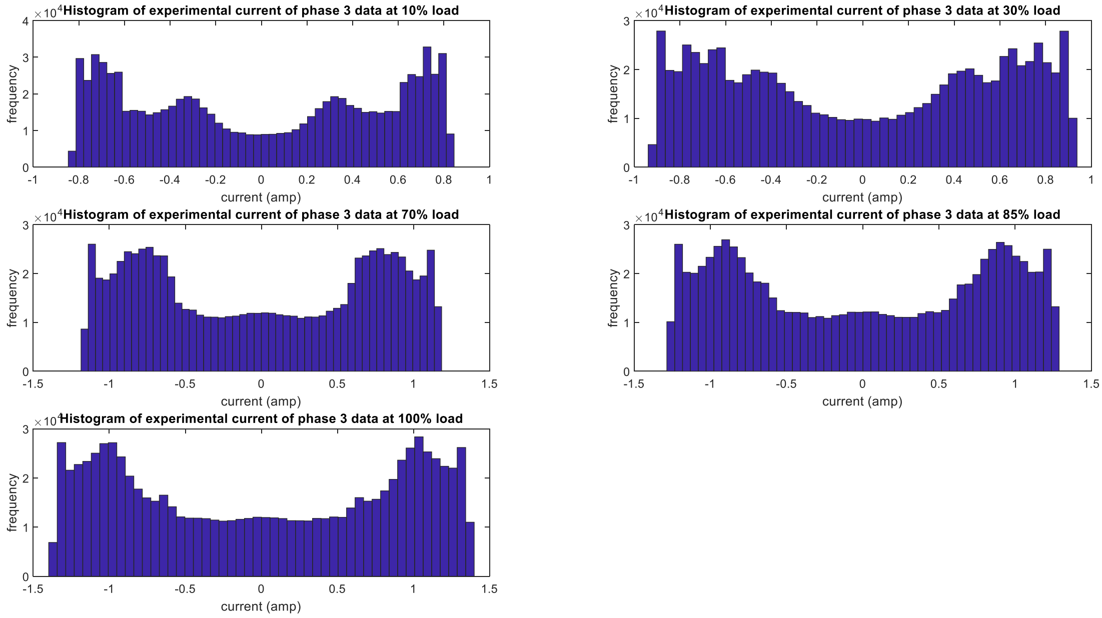

Fig. 2.3. Histograms of current of phase 3, I_3 for loading cases of 10%, 30%, 70%, 85%, and 100% of Full- load

2.3. Separation of Positive and Negative Sets from I_1, I_2, and I_3

As there are two different normal distributions (Bi- modal) within the same data set of each phase current, Positive and Negative current data from I_1, I_2, I_3 are separated for analysis. Mean and Standard Deviations of the positive and negative current data sets are calculated for each phase current under loading percentages of 10%, 30%, 70%, 85%, and 100% respectively.

Chapter III
REGRESSION MODEL

3.1. Regression Model

When the functional relationship between the response and the basic random variables is implicit or unknown, it can be developed by considering the available information on them. Since some or all of the variables are random, the relationship will be probabilistic in nature. Regression analysis is a statistical forecasting model that is concerned with describing and evaluating the relationship between a given variable (dependent variable) and one or more other variables (independent variables). Regression analysis models are used to help us predict the value of one variable from one or more other variables whose values can be predetermined.

The first phase of the process is to identify the variable that has to be predicted (dependent variable) and then to carry out multiple regression analyses focusing on the variables we want to use as predictors (explanatory variables). The multiple regression analysis would then identify the relationship between the dependent variable and the explanatory variables which finally can be presented as a model.

Two variable regression model has been considered in our work. Here *x* stands for load percentage and *y* stands for feature (mean or standard deviation) of that particular year. MATLAB 2018b software's Curve fitting toolbox has been used here to develop the models and to analyze the developed model's accuracy.

3.2. Least-square (LS) Method

The LS method finds *f (x)* by minimizing the residual according to the following formula:

$$\frac{1}{n}\sum_{i=0}^{n-1} w_i \, (f\,(x_i) - y_i)$$

Where n is the number of data samples, w_i is the i^{th} element of the array of weights for the data samples, $f(x_i)$ is the i^{th} element of the array of y-values of the fitted model y_i is the i^{th} element of the data set (x_i, y_i).

The LS method calculates *f (x)* by minimizing the square error and processing data that has Gaussian-distributed noise. If the noise is not Gaussian-distributed, for example, if the data contains outliers, the LS method is not suitable.

3.3. Proposed Regression Model

Curve fitting Toolbox and Polynomial fitting feature of MATLAB 2018b is used to generate estimated polynomial equations representing relation between mean vs percentage loading and standard deviation vs percentage loading for the load cases of 10%, 30%, 70%, 85%, and 100%. This is done for all the phase currents I_1, I_2, and I_3. The simulation results are shown in the Analysis of Results Chapter.

3.4. Monte- Carlo Simulation

Using the selected means and standard deviations for 50% load case from the regression equations resulting from employing curve- fitting of separated positive and negative currents' experimental data, Monte- Carlo Simulation with 1000 repetitions (Sample Size= 1000) is carried out to find 1000 mean and standard deviation values of each phase current after combining the positive and negative current sets. These values of means and standard deviations for 50% loading case fit either above the primarily selected points of mean and standard deviation or below those points. The probabilities of these points being above or below the regression curve are calculated within 95% confidence interval.

3.5. Optimization of prediction

The 1000 mean values calculated for 50% loading case using Monte- Carlo Simulation are compared with the selected mean value from the regression equation resulting from experimental data (using 10%, 30%, 70%, 85%, and 100% loading). From this, 1000 difference of means data of I_1, I_2, and I_3 were produced. Minimum Absolute difference between the simulated means and the selected mean (from regression of selected experimental means vs. load) for 50% loading is calculated. Similar procedure is carried out for standard deviation case also. These means and standard deviations which corresponds to the minimum absolute differences are the closest predictions that can be fitted with the regression model developed from experimental data.

3.6. Solution of Uncertainty of the Model

There can be some cases during selection of the optimized mean and standard deviation values where the minimum absolute difference values between the simulated feature (mean and standard deviation) values lying above or below the regression curve and the primarily selected value, within the confidence bound, are the same. As a result, an uncertainty arises as to which data to be selected as the best prediction. For such cases, we have incorporated a technique in our model to select the best mean or standard deviation value for a specific loading. The technique picks the best feature value (mean or standard deviation) from the set between regression curve and upper bound of the confidence interval if the probability of feature values being in this region is greater than the probability of feature values being in the region between regression curve and lower bound of the confidence interval. If the calculated probabilities are reciprocated, the best feature value is selected by applying the vice- versa of the above logic.

3.7. Analysis of Results

General Regression Equations Resulting from Polynomial Curve fitting Toolbox of MATLAB 2018b are tabulated in Table II where L denotes the percentage loading as independent variable and μ and σ are termed as mean and standard deviation respectively. A, B, and C are the coefficient of second order term, coefficient of first order term and constant respectively of the regression equation for mean vs percentage loading. Again, D and E are the coefficients of first order term and constant respectively of the regression equation for standard deviation vs percentage loading. The Goodness of Fit of each polynomials are analyzed in Goodness of Fit Chapter.

TABLE II

EQUATIONS BY REGRESSION DENOTING RELATION BETWEEN MEAN AND STANDARD DEVIATION OF EXPERIMENTAL PHASE CURRENTS AND LOADING PERCENTAGE

Relation	Regression Equation Resulting from Curve- Fitting
Mean (μ) vs. Percentage Loading (L)	$\mu(L) = AL^2 + BL + C$; μ as a function of L (1)
Standard Deviation (σ) vs. Percentage Loading (L)	$\sigma(L) = DL + E$; σ as a function of L (2)

Table III shows the coefficients and constants of the regression equation corresponding the relation between mean and percentage loading for all the three- phase currents. The coefficient A in this case is zero as in the regression curves of mean vs loading for all three phase currents, the polynomials best fitted are of first order.

TABLE III

COEFFICIENTS OF REGRESSION EQUATION FOR MEAN OF EXPERIMENTAL PHASE CURRENTS VS. LOADING PERCENTAGE

Current \ Polynomial Coefficient	Coefficient, A	Coefficient, B	Constant, C
Mean Current of Phase 1, I_1	0	-0.0000088800114519	0.0023330027460455
Mean Current of Phase 2, I_2	0	-0.0000067796231810	0.0017964784358666
Mean Current of Phase 3, I_3	0	0.0000217087150801	0.0024293815887217

Table IV shows the coefficients and constants of the regression equation corresponding the relation between standard deviation and percentage loading for all three- phase currents. The polynomials best fitted are of first order.

TABLE IV

COEFFICIENTS OF REGRESSION EQUATION FOR STANDARD DEVIATION (σ) OF EXPERIMENTAL PHASE CURRENTS VS. LOADING PERCENTAGE

σ of Current of Phase 1, I_1		σ of Current of Phase 2, I_2		σ of Current of Phase 3, I_3	
Coefficient, D	Constant, E	Coefficient, D	Constant, E	Coefficient, D	Constant, E
0.0035708115298611	0.4585556505088027	0.0039829938549301	0.4346690215351010	0.0039063500797506	0.4834261345733908

Table V and VI shows the coefficients and constants of the regression equation corresponding the relation between mean, standard deviation and percentage loading respectively for all the three- phase currents when separated as sets of positive and negative currents. In these cases, all polynomials best fitting the means and percentage loadings relations are of second order. As a result, the coefficient A is not zero in any of these cases. For the standard deviations vs percentage loading cases, polynomials best fitted are still of first order as previous.

TABLE V

COEFFICIENTS OF REGRESSION EQUATION FOR MEAN OF PHASE CURRENTS VS. LOADING PERCENTAGE, SEPARATING POSITIVE AND NEGATIVE SEQUENCES

Polynomial Coefficient / Current	Coefficient, A	Coefficient, B	Constant, C
Mean of Positive I1	0.0000178613107915	0.0012907700040859	0.4478988542425746
Mean of Positive I2	0.0000158405117899	0.0018001655150320	0.4263418813275320
Mean of Positive I3	0.0000154187285184	0.0018416389292275	0.4674010995846712
Mean of Negative I1	-0.0000178722255528	-0.0012859541464588	-0.4445181620563557
Mean of Negative I2	-0.0000158947362797	-0.0018367384684631	-0.4230469722302160
Mean of Positive I3	-0.0000156477473302	-0.0018017199150833	-0.4640741371390887

TABLE VI

COEFFICIENTS OF REGRESSION EQUATION FOR STANDARD DEVIATION OF PHASE CURRENTS VS. LOADING PERCENTAGE, SEPARATING POSITIVE AND NEGATIVE SEQUENCES

Polynomial Coefficient / Phase Current	Coefficient, D	Constant, E
Standard Deviation of Positive Current of Phase 1	0.0015247693601749	0.1968401808298853
Standard Deviation of Positive Current of Phase 2	0.0018297383731645	0.1785216711026834
Standard Deviation of Positive Current of Phase 3	0.0017261970715102	0.2053203832027072
Standard Deviation of Negative Current of Phase 1	0.0015314247671600	0.1954264201037256
Standard Deviation of Negative Current of Phase 2	0.0018303301741044	0.1777509789007861
Standard Deviation of Positive Current of Phase 3	0.0017143422465670	0.2031318446070701

Figs. 3.1 (a), 3.2 (a), and 3.3 (a) depict the curves of experimental means vs percentage loadings, generated by curve fitting toolbox of MATLAB 2018b. The circled portions contain the selected mean values (Green points) and predicted best mean values (Red points) of I_1, I_2 and I_3 for 50% loading case.

Figs. 3.1 (b), 3.2 (b), and 3.3 (b) show the zoomed views of the circled portions of Figs. 3.1 (a), 3.2 (a), and 3.3 (a) respectively. From the zoomed views, it is evident that the predicted points that represent the best mean values are in very close proximity of the primarily selected mean values.

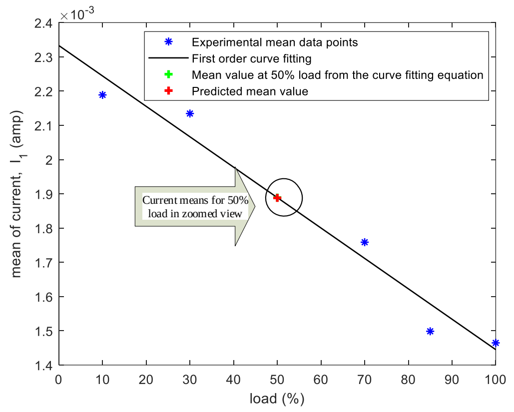

Fig. 3.1 (a). First Order Polynomial Fitting of Means of I_1 vs Load Percentage

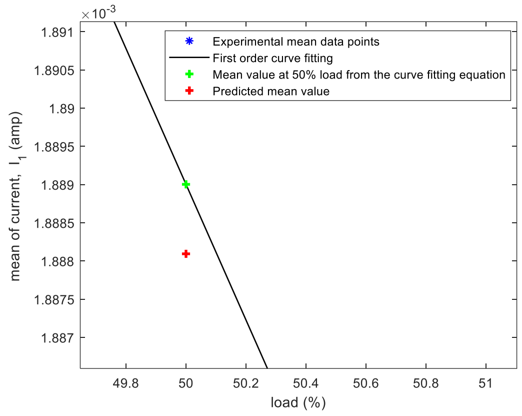

Fig. 3.1 (b). Zoomed view within circle to show the predicted best mean I_1 for 50% load

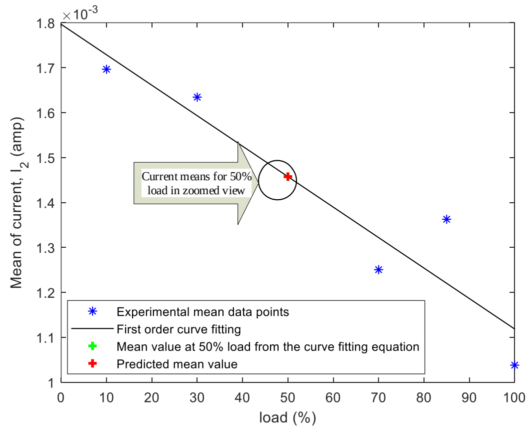

Fig. 3.2 (a). First Order Polynomial Fitting of Means of I_2 vs Load Percentage

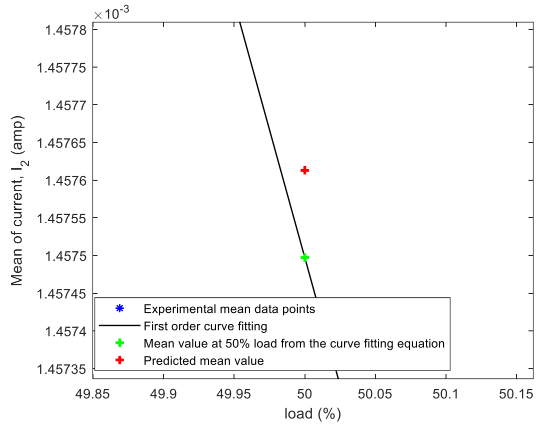

Fig. 3.2 (b). Zoomed view within circle to show the predicted best mean I_2 for 50% load

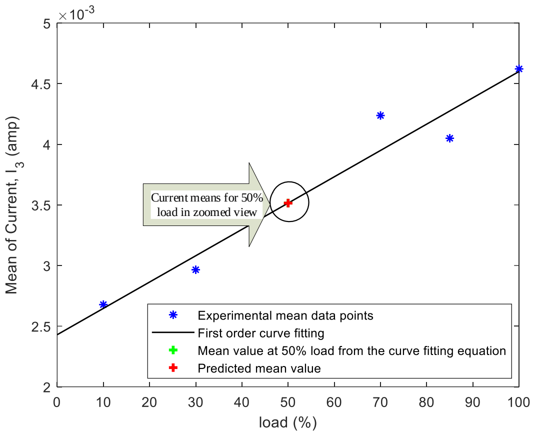

Fig. 3.3 (a). First Order Polynomial Fitting of Means of I_3 vs Load Percentage

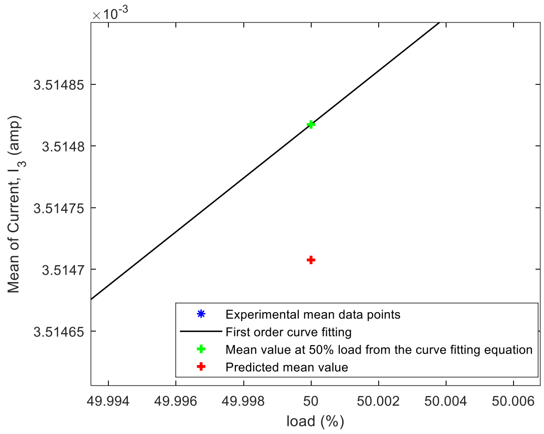

Fig. 3.3 (b). Zoomed view within circle to show the predicted best mean I_3 for 50% load

Figs. 3.4, 3.5, and 3.6 depict the curves of experimental standard deviations vs percentage loadings, generated by curve fitting toolbox of MATLAB 2018b. From the figures, it is evident that the predicted points that represent the best standard deviation values (Red points) are in very close proximity of the primarily selected standard deviation values (Green points).

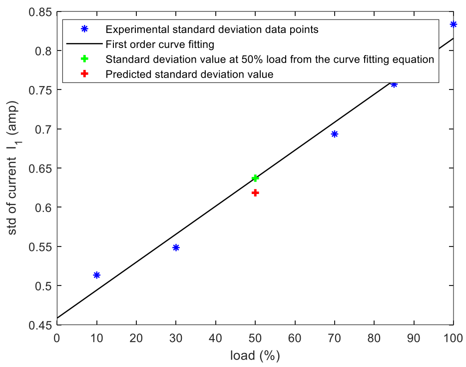

Fig. 3.4. First Order Polynomial Fitting of Standard Deviations of I_1 vs Load Percentage with predicted best Standard Deviation for 50% Load

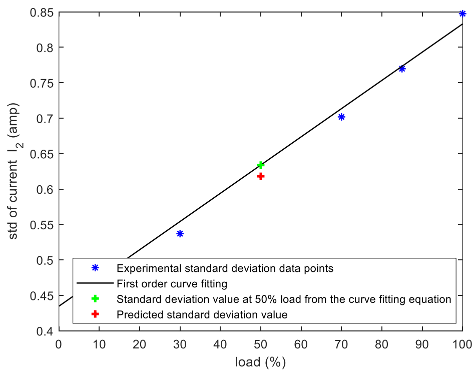

Fig. 3.5. First Order Polynomial Fitting of Standard Deviations of I_2 vs Load Percentage with predicted best Standard Deviation for 50% Load

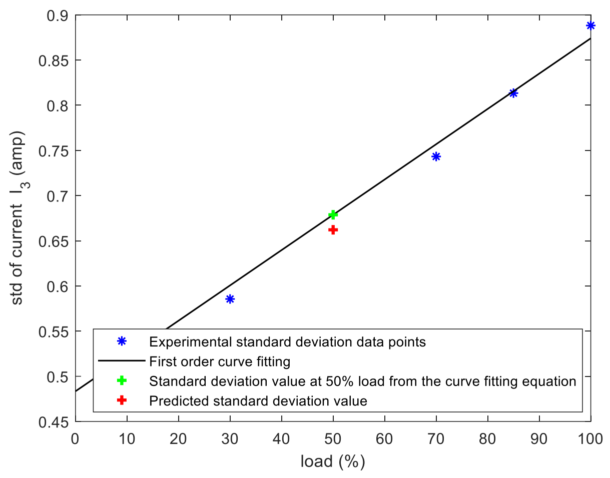

Fig. 3.6. First Order Polynomial Fitting of Standard Deviations of I_3 vs Load Percentage with predicted best Standard Deviation for 50% Load

Chapter IV

GOODNESS OF FIT ANALYSIS

4.1. Goodness of the Proposed Regression Models

Goodness of fit analysis of the proposed model is a vital part in regression analysis, implies how well do the actual data set correspond to the fitted developed model. Goodness of fit analysis is studied underneath the model using some fit statistic, or discrepancy measure, such as residuals, Chi-square or deviance. Here, Goodness of fit is studied through residual analysis in order to validate the proposed model. Some key points of the goodness of fit analysis are stated below.

4.2. Goodness of Fit Analysis

4.2.1. Residuals

The residuals from a fitted model are defined as the differences between the response data and the fit to the response data at each predictor value.

$Residual = data - fit$

$r = y - \hat{y}$

Where, r is the residual for a specific predictor value, y is the response value and $\hat{y}$ is the predicted response value. Assuming the model fit to the data is correct, the residuals approximate the random errors. Therefore, if the residuals appear to behave randomly, it suggests that the model fits the data well. However, if the residuals display a systematic pattern, it is a clear sign that the model fits the data poorly. The curve Fitting Toolbox supports these goodness of fit statistics for parametric models**:** R-square, Adjusted R-square, Root mean squared error (RMSE), Sum of Squares Due to Error (SSSE), these statistic measures the total deviation of the response values from the fit.

4.2.2. Sum of Square Error (SSE)

SSE is the sum of the squared differences between each observation and its group's mean. It can be used as a measure of variation within a cluster. If all cases within a cluster are identical the SSE would then be equal to 0.

SSE = $\sum_{i=1}^{n} w_i(y_i - \hat{y}_i)^2$

Where n is the number of data samples w_i is the i^{th} element of the array of weights for the data samples

y_i is the i^{th} element of the data set (x_i, y_i)

SSE is a network performance function, measures performance according to the sum of squared errors. A value closer to 0 indicates that the model has a smaller random error component, and that the fit will be more useful for prediction.

4.2.3. R-Square

R-square is the square of the correlation between the response values and the predicted response values. It is also called the square of the multiple correlation coefficient and the coefficient of multiple determination. R-square is defined as the ratio of the sum of squares of the regression (SSR) and the total sum of squares (SST). SSR is defined as,

SSR = $\sum_{i=1}^{n} w_i(\hat{y}_i - \bar{y})^2$

SST is also called the sum of squares about the mean, and is defined as

$$SST = \sum_{i=1}^{n} w_i(y_i - \bar{y})^2$$

Where SST = SSR + SSE. Given these definitions, R-square is expressed as

$$\text{R-square} = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

R-square can take on any value between 0 and 1, with a value closer to 1 indicating a better fit. Sometimes an increase the number of fitted coefficients in a model might increase the R-square although the fit may not improve in a practical sense. It is possible to get a negative R-square for equations that do not contain a constant term. If R-square is defined as the proportion of variance explained by the fit, and if the fit is actually worse than just fitting a horizontal line, then R-square is negative. In this case, R-square cannot be interpreted as the square of a correlation.

4.2.4. Degrees of Freedom Adjusted R-Square

This statistic uses the R-square statistic defined above, and adjusts it based on the residual degrees of freedom. The residual degrees of freedom is defined as the number of response values n minus the number of fitted coefficients m estimated from the response values.

$$v = n - m$$

Where, v indicates the number of independent pieces of information involving the n data points that are required to calculate the sum of squares. Note that if parameters are bounded and one or more of the estimates are at their bounds, then those estimates are regarded as fixed. The degrees of freedom is increased by the number of such parameters.

The adjusted R-square statistic is generally the best indicator of the fit quality when additional coefficients are added to a model.

$$\text{Adjusted R-square} = 1 - \frac{SSE\ (n-1)}{SST(v-1)}$$

The adjusted R-square statistic can take on any value less than or equal to 1, with a value closer to 1 indicating a better fit.

4.2.5. Root Mean Squared Error

This statistic is also known as the fit standard error and the standard error of the regression

RMSE = s = $\sqrt{MSE}$

Where MSE is the mean square error or the residual mean square.

MSE = $\frac{SSE}{v}$

A RMSE value closer to 0 indicates a better fit.

accuracy of the developed models are analysed below.

4.3 Goodness of Fit Evaluation

Residuals from the regression model of means and standard deviations of three- phase currents are shown in Figs. 4.1, 4.2, 4.3, 4.4, 4.5, and 4.6 respectively. It is clear from the figures that the residuals are not following any order in any of those cases. From the analysis, it can be concluded that the regression models properly fit with the data sets.

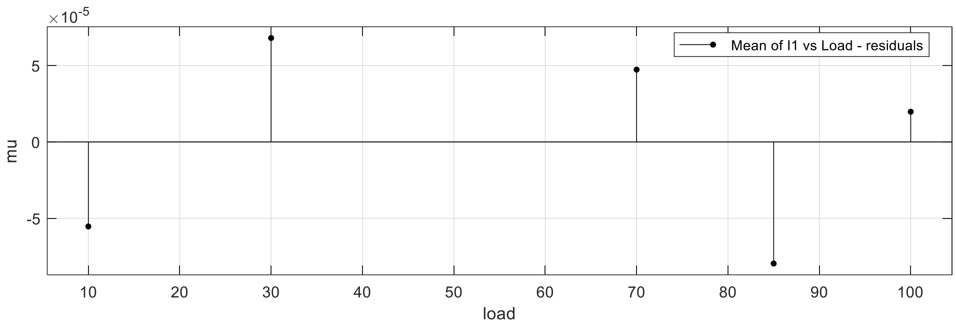

Fig. 4.1. Residuals from regression model of mean of I_1

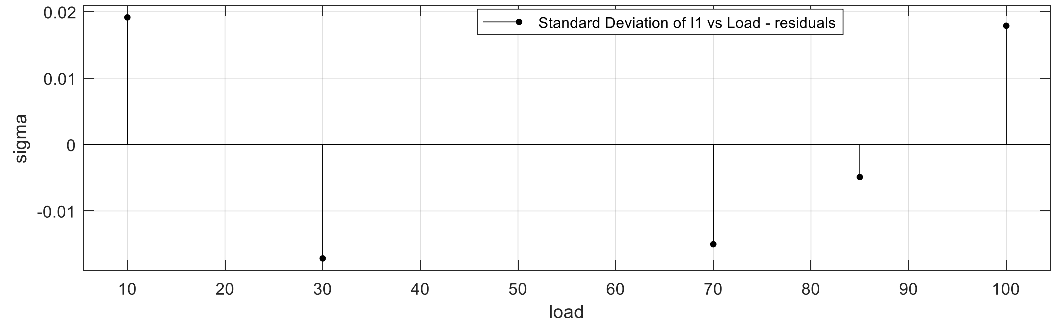

Fig. 4.2. Residuals from regression model of standard deviation of I_1

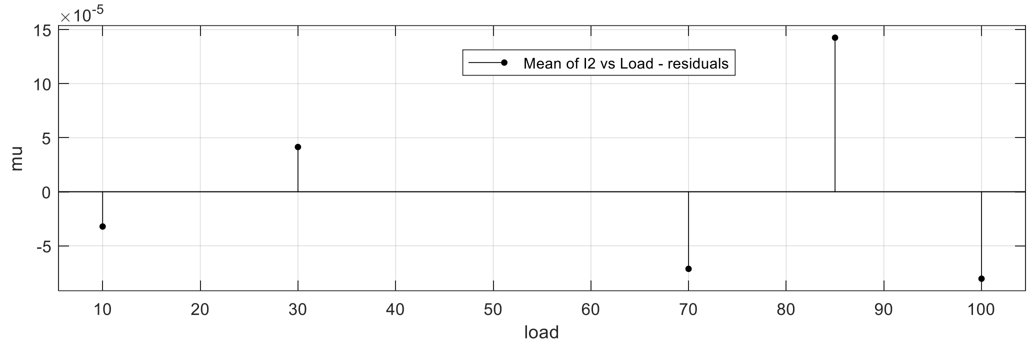

Fig. 4.3. Residuals from regression model of mean of I_2

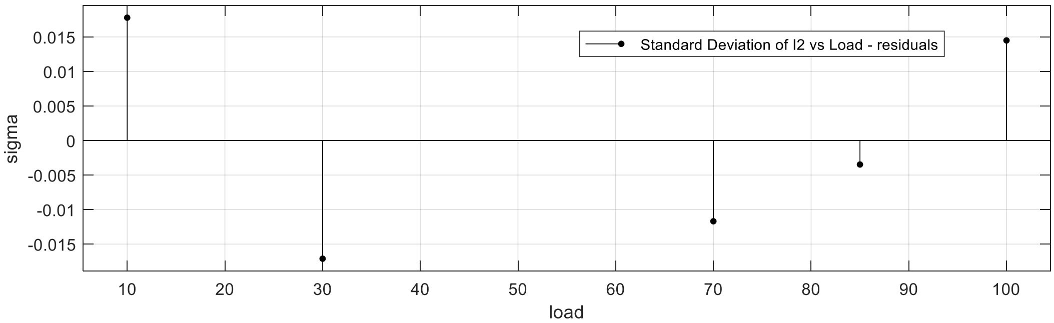

Fig. 4.4. Residuals from regression model of standard deviation of I_2

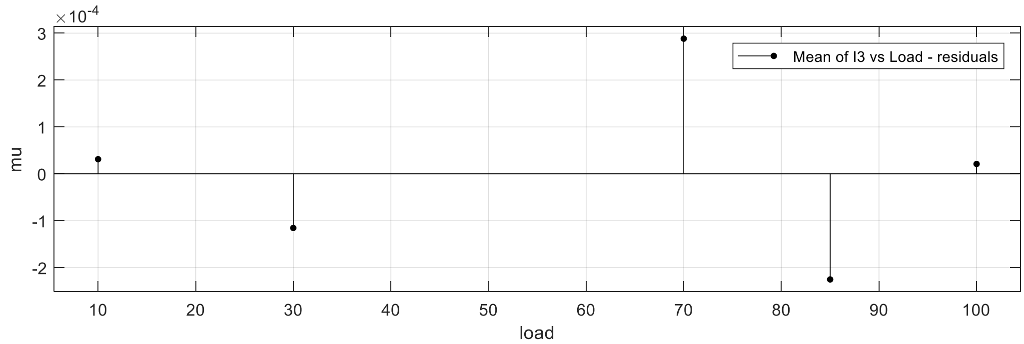

Fig. 4.5. Residuals from regression model of mean of I_3

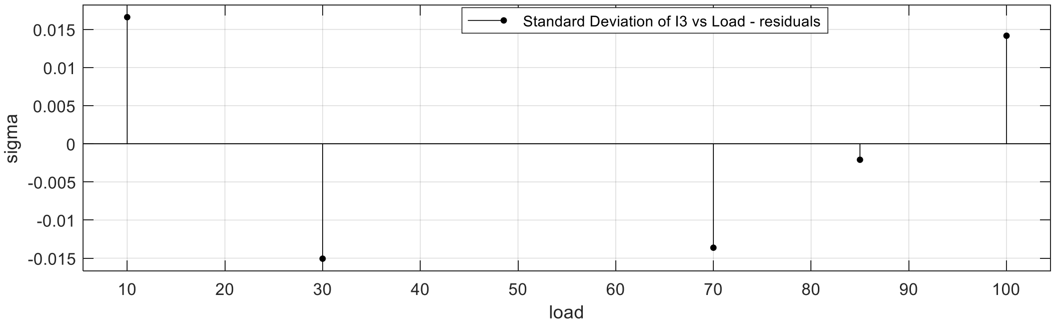

Fig. 4.6. Residuals from regression model of standard deviation of I_3

R-square and adjusted R-square can take on any value less than or equal to 1, with a value closer to 1 indicating a better fit. SSE and RMSE value closer to 0 indicates a better fit. Table VII shows the values of goodness of fit parameters corresponding to the regression of statistical features (mean, μ and standard deviation, σ) of three- phase currents with decision of fitness.

TABLE VII

GOODNESS OF FIT PARAMETER VALUES OF THREE PHASE CURRENTS' FEATURES

Regression Features	SSE (Acceptable closer to 0)	R- Square (Acceptable ≤ 1)	Adjusted R- Square (Acceptable ≤ 1)	RMSE (Acceptable closer to 0)	Remarks
μ of I_1	1.664×10^{-8}	0.9644	0.9526	7.448×10^{-5}	Good fit
μ of I_2	3.463×10^{-8}	0.8836	0.8448	0.0001074	Good fit
μ of I_3	1.485×10^{-7}	0.9478	0.9304	0.0002225	Good fit
σ of I_1	0.00123	0.9834	0.9779	0.02025	Good fit
σ of I_2	0.0009679	0.9894	0.9859	0.01796	Good fit
σ of I_3	0.0008935	0.9899	0.9865	0.01726	Good fit

Chapter V

REGRESSION MODEL VALIDATION

The proposed regression model is validated through comparison of the predicted best three- phase current data features (mean and standard deviation) resulting from regression curves of our proposed model and the calculated experimental features for a 50% loading case. Figs. 5.1 (a), 5.2 (a), and 5.3 (a) show the distributions of I_1, I_2, and I_3 from the primarily selected means and standard deviations for 50% loading case from the experimental data's regression. Again, Figs. 5.1 (b), 5.2 (b), and 5.3 (b) show the distributions of I_1, I_2, and I_3 from the predicted best means and standard deviations for 50% loading case from our proposed model. It is evident from those figures that our predicted current data for 50% loading also follow Bi- Modal (Two peaks) Distribution.

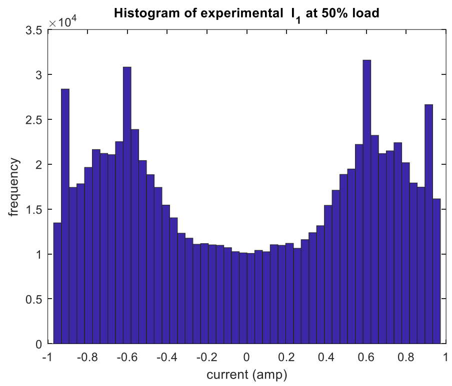

Fig. 5.1 (a). Distribution of Experimental I_1, 50% load

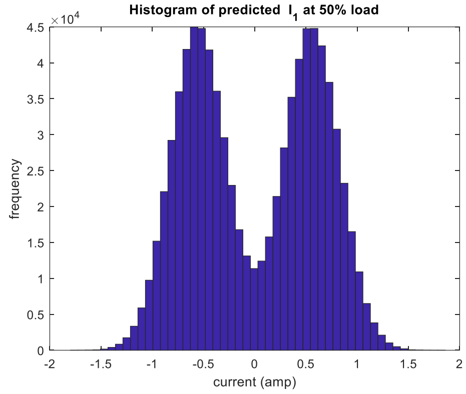

Fig. 5.1 (b). Distribution of Predicted I_1, at 50% load

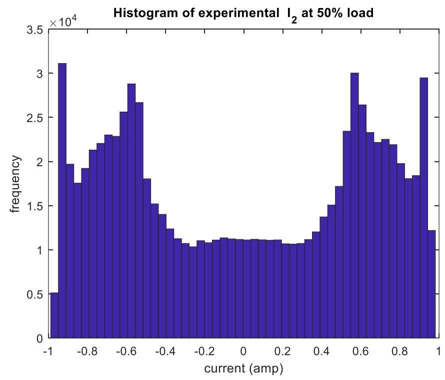

Fig. 5.2 (a). Distribution of Experimental I_2, 50% load

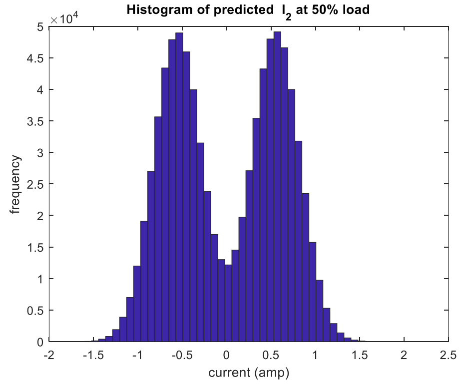

Fig. 5.2 (b). Distribution of Experimental I_2, 50% load

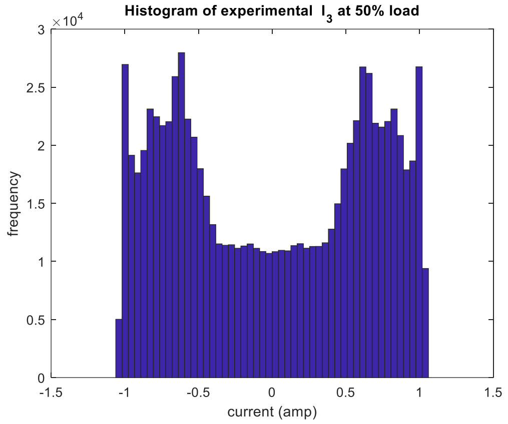

Fig. 5.3 (a). Distribution of Experimental I_3, 50% load

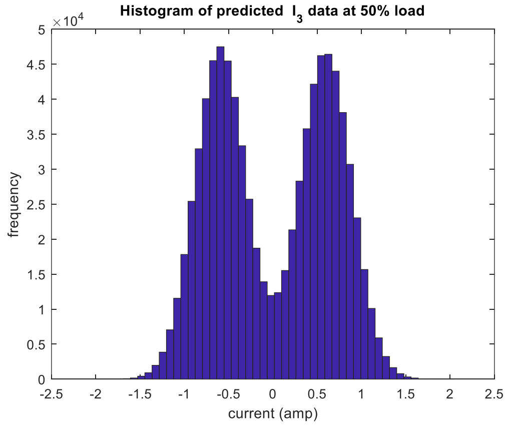

Fig. 5.3 (b). Distribution of Experimental I_3, 50% load

Table VII shows the means and standard deviations from the experimental data and the predicted best means and standard deviations from our proposed model for Currents I_1, I_2, and I_3 for 50% loading. From the prediction error percentages it can be concluded that our proposed model performs well in predicting the best statistical features for any arbitrary loading condition.

TABLE VIII

ERROR IN PREDICTION OF STATISTICAL FEATURES BETWEEN EXPERIMENTAL AND PROPOSED METHOD FOR 50% LOADING

Phase Currents	Statistical Features for 50% Loading	Features Calculated from Experimental Data	Predicted Best Features using Our Model	Prediction Error
I_1	Mean	0.0018890021734506	0.0018888806796832	6.504 %
	Standard Deviation	0.6112174211474820	0.6193588371971661	1.332 %
I_2	Mean	0.0014574972768154	0.0014576131025047	2.996 %
	Standard Deviation	0.6167324336211801	0.6171456443517061	0.067 %
I_3	Mean	0.0035148173427248	0.0035147076795838	4.858 %
	Standard Deviation	0.6575901541369190	0.6623313791482460	0.721 %

Chapter VI
DISCUSSION AND CONCLUSION

A Statistical Regression Model for predicting statistical features (Mean and Standard Deviation) of three-phase current data for any arbitrary loading when the rotor of the Induction Motor is subjected to 1 broken rotor bar fault is developed in this project. Accuracy of this prediction was verified by comparing the predicted best means and standard deviations with experimental means and standard deviations of three-phase currents calculated for 50% loading case. The percentage error between means and standard deviations remain within a reasonable range for all the current cases. We also removed the uncertainty which may occur during prediction of the features. This regression model is developed using only 5 data sets of each phase current during 1 broken rotor bar fault and can be used for predicting other statistical features like skewness, kurtosis, mean absolute deviation, median absolute deviation, maximum norm etc. for any arbitrary loading between 0% to 100% of full load. This model can also be used with substantial accuracy to predict three phase currents' statistical features for any arbitrary loading when the motor is subjected to other faults like 2 or 3 broken rotor bars; if experimental data can be acquired for a few loading percentages. As previously mentioned, these statistical features are used extensively to train machines using machine learning algorithms to detect early incipient faults. Due to limitation of financial aspects as well as resources, It is not feasible to perform experiment for collecting fault current data for all loading percentages between 0% to 100% during different faults. Use of our proposed model to predict the statistical features from current data using only a few experimental data will reduce the financial and time constraints substantially. Lastly, availability of features for any random loading case will surely make the training up process of induction motor using machine learning more robust.

ACKNOWLEDGEMENT

[1] We would like to show our sincere gratitude to Dr. Xiaodong Liang, Associate Professor, Memorial University for sharing the fault currents' data with us during the course of this project, and we thank Mohammad Zawad Ali, Graduate Research Assistant, Memorial University for his insights on these data as the data were collected by him.

APPENDIX

I1: current of phase 1
I2: current of phase 2
I3: current of phase 3
L10 = current data for 10 % loading
L30 = current data for 30 % loading
L50 = current data for 50 % loading
L70 = current data for 70 % loading
L85 = current data for 85 % loading
L100 = current data for 100 % loading
L10_pos = positive current data for 10 % loading
L10_neg = negative current data for 10 % loading
L30_pos = positive current data for 30 % loading
L30_neg = negative current data for 30 % loading
L50_pos = positive current data for 50 % loading
L50_neg = negative current data for 50 % loading
L70_pos = positive current data for 70 % loading
L70_neg = negative current data for 70 % loading
L85_pos = positive current data for 85 % loading
L85_neg = negative current data for 85 % loading
L100_pos = positive current data for 100 % loading
L100_neg = negative current data for 100 % loading
mean_pos = mean of experimental positive current data
mean_neg = mean of experimental negative current data
mu = mean of experimental current data
std_pos = standard deviation of experimental positive current data
std_neg = standard deviation of experimental negative current data
sigma = standard deviation of experimental current data
load = different load
mean_eqn_pos = mean equation of experimental positive current data
mean_eqn_neg = mean equation of experimental negative current data
mean_eqn = mean equation of experimental current data
std_eqn_pos = standard deviation equation of experimental positive current data
std_eqn_neg = standard deviation equation of experimental negative current data
std_eqn = standard deviation equation of experimental current data
meanX_pos = calculated mean of positive current for 50% loading
meanX_neg = calculated mean of negative current for 50% loading
meanX = calculated mean of current for 50% loading
stdX_pos = calculated standard deviation of positive current for 50% loading
stdX_neg = calculated standard deviation of negative current for 50% loading

stdX = calculated standard deviation of current for 50% loading
mean_std_pred = array containing predicted mean and standard deviation of current for 50% loading
X_pos = generated new normal random distribution (monte carlo) of positive current for 50% loading
X_neg = generated new normal random distribution (monte carlo) of negative current for 50% loading
X = combination of positive and negative current random generated new distribution (monte carlo) for 50% loading
hi_m = number of mean values which will be in the region between the regression curve and the upper bound of the confidence interval
lo_m = number of mean values which will be in the region between the regression curve and the lower bound of the confidence interval
eql_m = number of mean values which will be in the regression curve
hi_s = number of standard deviation values which will be in the region between the regression curve and the upper bound of the confidence interval
lo_s = number of standard deviation values which will be in the region between the regression curve and the lower bound of the confidence interval
eql_s = number of standard deviation values which will be in the regression curve
prob_mean_hi = probability of mean values which will be in the region between the regression curve and the upper bound of the confidence interval
prob_mean_lo = probability of mean values which will be in the region between the regression curve and the lower bound of the confidence interval
prob_mean_eql = probability of mean values which will be in the regression curve
prob_std_hi = probability of standard deviation values which will be in the region between the regression curve and the upper bound of the confidence interval
prob_std_lo = probability of standard deviation values which will be in the region between the regression curve and the lower bound of the confidence interval
prob_std_eql = probability of standard deviation values which will be in the regression curve
min_mean_diff = minimum mean value current for 50% loading
min_std_diff = minimum standard deviation value current for 50% loading
actual_mean = experimental (actual) mean of current for 50% loading
actual_std = experimental (actual) standard deviation of current for 50% loading
error_in_mean = error between actual and predicted mean of current for 50% loading
error_in_std = error between actual and predicted standard deviation of current for 50% loading

For I:
MATLAB codes:

```
% ENGI9411
% Project
clear all
close all
clc
format long

% open and scan files to import data
L10 = fscanf (fopen('L10.txt' , 'r') , '%f') ;
L30 = fscanf (fopen('L30.txt' , 'r') , '%f') ;
L50 = fscanf (fopen('L50.txt' , 'r') , '%f') ;
L70 = fscanf (fopen('L70.txt' , 'r') , '%f') ;
L85 = fscanf (fopen('L85.txt' , 'r') , '%f') ;
L100 = fscanf (fopen('L100.txt' , 'r') , '%f') ;

fclose('all') ;
```

```
% split positive and negative data
L10_pos = L10 ;
L10_neg = L10 ;
L10_pos(L10<0) = [] ;
L10_neg(L10>0) = [] ;

L30_pos = L30 ;
L30_neg = L30 ;
L30_pos(L30<0) = [] ;
L30_neg(L30>0) = [] ;

L50_pos = L50 ;
L50_neg = L50 ;
L50_pos(L50<0) = [] ;
L50_neg(L50>0) = [] ;

L70_pos = L70 ;
L70_neg = L70 ;
L70_pos(L70<0) = [] ;
L70_neg(L70>0) = [] ;

L85_pos = L85 ;
L85_neg = L85 ;
L85_pos(L85<0) = [] ;
L85_neg(L85>0) = [] ;

L100_pos = L100 ;
L100_neg = L100 ;
L100_pos(L100<0) = [] ;
L100_neg(L100>0) = [] ;

% calculate mean of experimental data
mean_pos =
[(mean(L10_pos));(mean(L30_pos));(mean(L70_pos));(mean(L85_pos));(mean
(L100_pos))] ;
mean_neg =
[(mean(L10_neg));(mean(L30_neg));(mean(L70_neg));(mean(L85_neg));(mean
(L100_neg))] ;
mu = [(mean(L10));(mean(L30));(mean(L70));(mean(L85));(mean(L100))] ;

% calculate standard deviation of experimental data
std_pos =
[(std(L10_pos));(std(L30_pos));(std(L70_pos));(std(L85_pos));(std(L100
_pos))] ;
std_neg =
[(std(L10_neg));(std(L30_neg));(std(L70_neg));(std(L85_neg));(std(L100
_neg))] ;
sigma = [(std(L10));(std(L30));(std(L70));(std(L85));(std(L100))] ;

load = [10;30;70;85;100] ;

% find equation of mean
```

```
mean_eqn_pos = polyfit (load,mean_pos,2) ;
mean_eqn_neg = polyfit (load,mean_neg,2) ;
mean_eqn = polyfit (load,mu,1) ;

% find equation of standard deviation
std_eqn_pos = polyfit (load,std_pos,1) ;
std_eqn_neg = polyfit (load,std_neg,1) ;
std_eqn = polyfit (load,sigma,1) ;

% calculate mean from the equation
meanX_pos = polyval (mean_eqn_pos,50) ;
meanX_neg = polyval (mean_eqn_neg,50) ;
meanX = polyval (mean_eqn , 50) ;

% calculate standard deviation from the equation
stdX_pos = polyval (std_eqn_pos,50) ;
stdX_neg = polyval (std_eqn_neg,50) ;
stdX = polyval (std_eqn,50) ;

n_pos = length (L50_pos) ;
n_neg = length (L50_neg) ;
n = n_pos + n_neg ;
N = 1000 ;
X = zeros (n,N) ;
mean_std_pred = zeros (N,2) ;

for i = 1:N

% generate new distribution (monte carlo)
X_pos = random ('normal' , meanX_pos , stdX_pos , [n_pos,1]) ;
X_neg = random ('normal' , meanX_neg , stdX_neg , [n_neg,1]) ;

% combine positive and negative data
X(:,i) = [X_pos;X_neg] ;

mean_std_pred(i,1) = mean (X(:,i)) ;
mean_std_pred(i,2) = std (X(:,i)) ;

end

hi_m = 0 ;
lo_m = 0 ;
eql_m = 0 ;
hi_s = 0 ;
lo_s = 0 ;
eql_s = 0 ;

% calculate probability
for i = 1:N
    if mean_std_pred(i,1) > meanX
        hi_m = hi_m + 1 ;
    elseif mean_std_pred(i,1) < meanX
```

```
        lo_m = lo_m + 1 ;
    else
        eql_m = eql_m + q ;
    end
    if mean_std_pred(i,2) > meanX
        hi_s = hi_s + 1 ;
    elseif mean_std_pred(i,2) < meanX
        lo_s = lo_s + 1 ;
    else
        eql_s = eql_s + q ;
    end
end
prob_mean_hi = hi_m/N ;
prob_mean_lo = lo_m/N ;
prob_mean_eql = eql_m/N ;
prob_std_hi = hi_s/N ;
prob_std_lo = lo_s/N ;
prob_std_eql = eql_s/N ;

% calculate difference between actual and predicted
dfrnc = zeros (N,2) ;
for i = 1:N
    dfrnc(i,1) = abs(meanX - mean_std_pred(i,1)) ;
    dfrnc(i,2) = abs(stdX - mean_std_pred(i,2)) ;
end

min_mean_diff = min (dfrnc(:,1)) ;
[r1,~] = find (dfrnc(:,1) == min_mean_diff) ;

min_std_diff = min (dfrnc(:,2)) ;
[r2,c] = find (dfrnc(:,2) == min_std_diff) ;

% final result
sets = [(mean_std_pred (r1,1)),meanX,(mean_std_pred
(r2,1));(mean_std_pred (r1,2)),stdX,(mean_std_pred (r2,2))] ;
pred_mean_1 = mean_std_pred (r1,1) ;
pred_std_1 = mean_std_pred (r1,2) ;
pred_mean_2 = mean_std_pred (r2,1) ;
pred_std_2 = mean_std_pred (r2,2) ;
data_1 = X(:,r1) ;
data_2 = X(:,r2) ;

eqn = polyfit (mu,sigma,1) ;
val = polyval (eqn , meanX) ;
v1 = polyval (eqn , pred_mean_1) ;
v2 = polyval (eqn , pred_mean_2) ;
d1 = abs(val - v1) ;
d2 = abs(val - v2) ;

if d1 < d2
    final_data = data_1 ;
    final_mean = mean (data_1) ;
```

```
    final_std = std (data_1) ;
    disp('data set from mean') ;
else
    final_data = data_2 ;
    final_mean = mean (data_2) ;
    final_std = std (data_2) ;
    disp('data set from standard deviation') ;
end

% show error between actual and predicted
actual_mean = mean (L50) ;
actual_std = std (L50) ;
error_in_mean = (abs((actual_mean - final_mean)/actual_mean)) * 100 ;
error_in_std = (abs((actual_std - final_std)/actual_std)) * 100 ;

fig_1 = figure ('position' , [50 , -50 , 1600 , 1000]) ;
subplot (3,2,1)
hist (L10 , 50)
title ('Histogram of experimental current of phase 1 data at 10%
load') ;
ylabel ('frequency') ;
xlabel ('current (amp)') ;
subplot (3,2,2)
hist (L30 , 50)
title ('Histogram of experimental current of phase 1 data at 30%
load') ;
ylabel ('frequency') ;
xlabel ('current (amp)') ;
subplot (3,2,3)
hist (L70 , 50)
title ('Histogram of experimental current of phase 1 data at 70%
load') ;
ylabel ('frequency') ;
xlabel ('current (amp)') ;
subplot (3,2,4)
hist (L85 , 50)
title ('Histogram of experimental current of phase 1 data at 85%
load') ;
ylabel ('frequency') ;
xlabel ('current (amp)') ;
subplot (3,2,5)
hist (L100 , 50)
title ('Histogram of experimental current of phase 1 data at 100%
load') ;
ylabel ('frequency') ;
xlabel ('current (amp)') ;
fig_2 = figure ;
hist (final_data , 50) ;
title ('Histogram of predicted current of phase 1 data at 50% load') ;
ylabel ('frequency') ;
xlabel ('current (amp)') ;
```

```
fig_3 = figure ;
hist (L50 , 50) ;
title ('Histogram of experimental current of phase 1 data at 50%
load') ;
ylabel ('frequency') ;
xlabel ('current (amp)') ;

xval1 = 0:100 ;
yval1 = (mean_eqn(1,1))*xval1 + mean_eqn(1,2) ;
fig_4 = figure ;
plot (load , mu , 'b*' , 'linewidth' , 1) ;
title ('Plot of mean of experimental data of current phase 1 vs
percentage load') ;
xlabel ('load (%)') ;
ylabel ('current (amp)') ;
hold on
plot (xval1 , yval1 , 'k-' , 'linewidth' , 1) ;
plot (50 , meanX , 'g+' , 'linewidth' , 2) ;
hold on
plot (50 , final_mean , 'r+' , 'linewidth' , 2) ;
legend ('Experimental mean data points' , 'First order curve fitting'
, 'Mean value at 50% load from the curve fitting equation' ,
'Predicted mean value' , 'location' , 'best') ;
fid = fopen ('textfile.txt' , 'w') ;
fprintf (fid , 'Experimental Data:\n\t\tMean\t\t\t\tStandard
Deviation\n') ;
for i = 1:length (mu)
    fprintf (fid , '%.16f\t\t\t%.16f\n' , mu(i,1) , sigma(i,1)) ;
end
fprintf (fid, '\n\nError in mean = %.3f\n\nError in standard deviation
= %.3f\n\n\n' , error_in_mean , error_in_std) ;
fprintf (fid, 'Final Predicted mean = %.16f\n\nFinal Predicted
standard deviation = %.16f\n\n' , final_mean , final_std) ;
fprintf (fid, 'Predicted 50%% load mean = %.16f\n\n' , meanX) ;
fprintf (fid, 'Predicted mean of positive data 50%% load =
%.16f\n\nPredicted mean of negative data 50%% load = %.16f\n\n\n' ,
meanX_pos , meanX_neg) ;
fprintf (fid, 'Equation of experimental data mean:
%.16f\t\t\t%.16f\n\n' , mean_eqn) ;
fprintf (fid, 'Equation of positive experimental data mean:
%.16f\t\t\t%.16f\t\t\t%.16f\n\n' , mean_eqn_pos) ;
fprintf (fid, 'Equation of negative experimental data mean:
%.16f\t\t\t%.16f\t\t\t%.16f\n\n\n' , mean_eqn_neg) ;
fprintf (fid, 'Equation of experimental data standard deviation:
%.16f\t\t\t%.16f\n\n' , std_eqn) ;
fprintf (fid, 'Equation of positive experimental data standard
deviation: %.16f\t\t\t%.16f\n\n' , std_eqn_pos) ;
fprintf (fid, 'Equation of negative experimental data standard
deviation: %.16f\t\t\t%.16f\n\n' , std_eqn_neg) ;
```

Similar coding procedures are maintained for the cases of I_2 and I_3.

REFERENCE

[1] Gugaliya, Agam, et al. "Effective Combination of Motor Fault Diagnosis Techniques." *2018 International Conference on Power, Instrumentation, Control and Computing (PICC)*, 2018.

[2] Liang, Xiaodong, and Kenneth Edomwandekhoe. "Condition Monitoring Techniques for Induction Motors." *2017 IEEE Industry Applications Society Annual Meeting*, 2017.

[3] Wen, Long, et al. "A New Convolutional Neural Network-Based Data-Driven Fault Diagnosis Method." *IEEE Transactions on Industrial Electronics*, no. 7, 2018.

[4] Ting Yang, Haibo Pen, Zhaoxia Wang, and Che Sau Chang, "Feature Knowledge Based Fault Detection of Induction Motors through the Analysis of Stator Current Data," *IEEE Transactions on Instrumentation and Measurement, vol. 65, no. 3, March 2016.*

[5] Martin-Diaz, Ignacio, et al. "An Experimental Comparative Evaluation of Machine Learning Techniques for Motor Fault Diagnosis under Various Operating Conditions." *IEEE Transactions on Industry Applications*, no. 3, 2018.