Available online www.ejaet.com

European Journal of Advances in Engineering and Technology, 2019, 6(6):30-38

Research Article ISSN: 2394 - 658X

Effect of Multicollinearity in Predicting Diabetes Mellitus Using Statistical Neural Network

S. Suleiman and S. Badamsi

Department of Mathematics Usmanu Danfodiyo University, Sokoto
Email: suleman.shamsuddeen@udusok.edu.ng, shehubadamasi300@gmail.com

---

ABSTRACT

Diabetes mellitus (DM) is a diverse group of metabolic disorders that is frequently associated with a high disease burden in developing countries such as Nigeria. It also needs continuous blood glucose monitoring and self-management. Multicollinearity is one of the problems usually encountered by Economists and Statisticians when predicting a dependent variable from the set of independent variables that are significantly or highly correlated especially when the traditional method of regression analysis is used. This research is aimed to determine the effect of multicollinearity in predicting diabetes mellitus using statistical neural network. In this research, 100 patients were considered from Ahmadu Bello University Teaching Hospital who have undergone diabetes screening test and 29 risk factors were used. Variance inflation factor (VIF) detected multicollinearity among some risk factors and Principal component technique (PCA) was employed to remove it. Levenberg-Marquardt (LM) algorithm was used to train the statistical neural network for the original and principal components data. The results show that when five (5) hidden neurons architecture is used, the model achieved 99.0% and 93.9% accuracy for training the original and reduced data respectively. Similarly, when the number of hidden neurons is increased to ten (10), the model achieved 98.7% and 94.4% accuracy for training the original and reduced data respectively. The research therefore concludes that unlike traditional econometrics and statistical models, statistical neural networks estimation process is not negatively affected by the presence of multicollinearity in the data but get better when more information are utilised because it gives better estimates when the whole data is considered as inputs than when the reduced data is used.

Key words: Multicollinearity, Diabetes Mellitus, Statistical Neural Network

---

INTRODUCTION

The World Health Organization [1] defines diabetes as a chronic disease that occurs either when the pancreas does not produce enough insulin or when the body cannot effectively use the insulin it produces. Diabetes cases are on the increase all over the world and countries are struggling to fight the disease. The misconception that diabetes is "a disease of the wealthy" is still held by some people; but the evidence published in the Diabetes Atlas of the International Diabetes Federation [2] disproves that delusion: a staggering 80% of people with diabetes live in low and middle-income countries, and the socially disadvantaged in any country are the most vulnerable to that illness.

Today's emerging diabetes hotspots include countries in the Middle East, Western Pacific, sub-Saharan Africa and South-East Asia where economic development has transformed lifestyles. These rapid transitions are bringing previously unheard rates of obesity and diabetes; developing countries are facing a firestorm of ill health with inadequate resources to protect their population. Thus, it is necessary to increase awareness of the importance of a healthful diet and physical activity, especially for children and adolescents. Crucially though, environments have to be created that lay the foundations for healthy living [2].

Nigeria has the largest population in Africa (about 170 million); and of this the adult population (aged 20–79 years), is approximately 79 million. One third of all the cases of diabetes are in the rural communities, while the rest are in the urban centres. About two million of the cases of diabetes in Nigeria are undiagnosed. Deaths related to diabetes in Nigeria in 2013 were estimated to be 105,091 cases [3].

Principal Component Analysis (PCA) is a dimension-reduction tool that can be used to reduce a large set of variables to a small set that still contains most of the information in the large set. It is a mathematical procedure that transforms a number of (possibly) correlated variables into a (smaller) number of uncorrelated variables called principal components.
Principal component analysis (PCA) is one of the most popular methods used for variable reduction, which can overcome the disturbance of the multicollinearity of the risk factors and has been used in social sciences, health service, and health sciences [4].
PCA is an appropriate multivariate technique to reduce the dimension of a data set consisting of a large number of interrelated variables, while retaining as much as possible of the variation present in the data set [5]. This is achieved by transforming set of original variables to a new set of variables, the principal components (PCs), and which are ordered so that the first few retain most of the variation present in all of the original variables [6].
Principal Component Analysis is a tool that allows the size of enormous databases to be reduced, while at the same time maintaining control over loss of information. In addition, it enables visualization of observations. The representation of a sample in the reduced space permits one to establish relationships between variables. PCA is one of the data mining methods that allow one to discover connections hidden in the data and better their understanding. On the other hand, it can be used as a preliminary method when the final statistical tests require analyzing independent variables. For example, it is used as a first step in the analysis of regression.
Statistical Neural Network is a non-parametric method that can be use in the medical field to classify subjects based on input variables into sick or healthy. Classification and prediction of the patient's condition based on risk factors are an application of artificial neural networks [7].
Statistical neural networks mimic mixed structure of the human brain. Billion nerve cells (neurons) through the communication that with each other (synapses) creates a biological neural network in the human brain that is dedicated to human activities such as reading, comprehension, speaking, breathing, movement, voice recognition, face detection, also resolve issues and data storage. Artificial neural networks, in fact, simulate a part of brain functions [7-8].
Rahimloo and Jafarian [9] predicted diabetes using artificial neural network, logistic regression statistical model and combined them where the accuracy and efficiency of the methods were investigated and acceptable results compared to the neural and logistic regression methods were obtained. They used some criteria to minimize the error function In neural network training using a neural network in a hybrid model which eventually came to the conclusion that the error function of the neural network is equal to 0.1 and the error function of the combined neural network model is equal to 0.0002.
Mishra *et al.,* [10] predicted the onset of diabetes using machine learning. The paper uses classification techniques, like logistic regression to predict the disease in its early stages. The result shows that the simple anthropometric variables like waist circumference are better predictors, than the three month average blood glucose level.
WU *et al.,* [11] establish an appropriate prediction model based on data mining techniques for predicting type 2 diabetes mellitus (t2dm).were the the accuracy of the prediction model were improved to make the model adaptive to more than one dataset. The model comprised of two parts, the improved k-means algorithm and the logistic regression algorithm. It utilized Pima Indians diabetes dataset and the Waikato environment for knowledge analysis toolkit to compare the results with the results from other researchers. It shows that the model attained a 3.04% higher accuracy of prediction than those of other researcher. The model ensures that the dataset quality is sufficient. The model was applied to two other diabetes data set, with shows that the model is useful for realistic health management of diabetes.
Selvakumar *et al.,* [12] predicteddiabetesdiagnosisusingclassificationbaseddataminingtechniques, they used binary logistic regression, multilayer perceptron and k-nearest neighbor as classification for diabetes data. They compared classification accuracy for classifying data.. They fund that the binary logistic regression accuracy is 0.69, multilayer perceptron accuracy is 0.71 and KNN gives the accuracy of 0.80 which shows k-nearest neighbour accuracy is higher than that of binary logistic regression and multilayer perceptron.
Zou et al., [13] analysed risk factors and their interactions in type 2 diabetes mellitus using cross-sectional survey in Guilin, China. The study aimed to not only analyzes the influence of a single factor for type 2 diabetes, but also to investigate the interaction effects between risk factors. It showed that type 2 diabetes resulted from the interactions of many factors; the interactions among age, triglycerides and non-alcoholic fatty liver disease. These are important risk factors for type 2 diabetes.
Wang *et al.,* [4] improved the ability of the binary logistic regression analysis to predict Diabetes Mellitus using Principal Components Analysis (PCA). The studies overcome the disturbance of the multicollinearity of the risk factors and examine the associations of these factors with diabetes using the principal component analysis (PCA) and regression analysis. PCA was utilized to deal with multicollinearity of the risk factors. Weighted univariate and multiple logistic regression analyses were used to estimate the associations of potential factors and PCS with diabetes. It concludes that the PCA can be used to reduce the indicators in complex survey data. The PCs of nutrition factors and physical activities were associated with diabetes.
However, none of these researches attempted to explore the effect of the correlation of the input variables (multicollinearity) in training their Neural Network models.
The aim of this research is to examine the effect of multicollinearity in predicting diabetes mellitus using statistical neural network.

MULTILAYER NETWORKS

Multilayer networks are universal approximators; the training of such networks means determining a procedure for selecting the network parameters (weights and biases) which will best approximate a given function. The procedure for selecting the parameters for a given problem is called *training* the network. In this research a training procedure called *Backpropagation*, which is based on gradient descent is used.

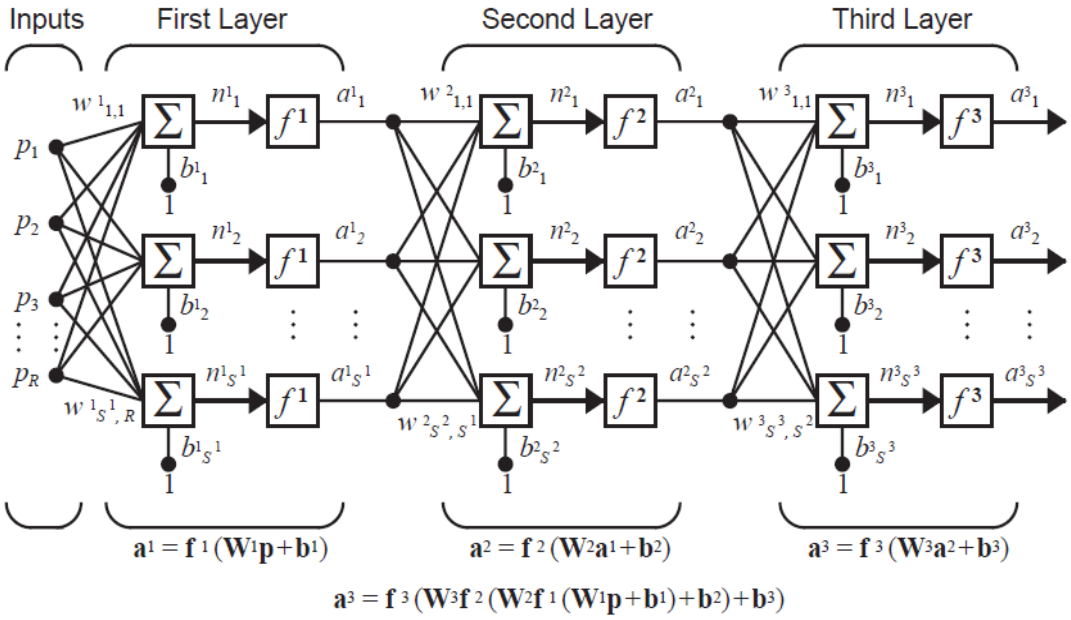

Fig. 1 Three layer Network

In multilayer networks the output of one layer becomes the input to the following layer. The equations that describe this operation are as follows:

$$a^{m+1} = f^{m+1}(w^{m+1}a^m + b^{m+1}) \quad (1)$$

For m= 0, 1, ----, m-1

Where M is the number of layers in the network. The neurons in the first layer receive external inputs:

$$a^0 = p \quad (2)$$

The outputs of the neurons in the last layer are considered the network outputs:

$$a = a^m \quad (3)$$

The algorithm should adjust the network parameters in order to minimize the sum squared error:

$$F(x) = \sum_{q=1}^{Q} e_q^2 = \sum_{q=1}^{Q}(t_q - a_q)^2 \quad (4)$$

Where x is a vector containing all of network weights and biases if the network has multiple outputs.

Since the performance index in (4) is sum of squares of nonlinear function, the numerical optimization techniques for nonlinear least squares can be used to minimize this cost function. The Levenberg-Marquardt algorithm, which is an approximation to the Newton's method is said to be more efficient in comparison to other methods for convergence of the Backpropagation algorithm for training a moderate-sized feed forward neural network [14]. As the cost function is a sum of squares of nonlinear function, the Hessian matrix required for updating the weights and biases need not be calculated and can be approximated as

$$H = J^T(x)J(x) \quad (5)$$

The updated weights and biases are given by

$$x_{k+1} = x_k - [J^T(x)J(x) + \mu I]^{-1}J^T(x)e(x) \quad (6)$$

Where μ is a scalar and I is the identity matrix.

PRINCIPAL COMPONENT ANALYSIS (PCA)

The essence of Principal Component Analysis is to convert collections of the P variables $X_1, X_2, X_3, \dots, X_p$ into a system of orthogonal variables$Z_1, Z_2, Z_3, \dots, Z_q$ where $q \leq p$ as follows:

$$\begin{aligned} Z_1 &= a_{11}X_1 + a_{12}X_2 + \cdots + a_{1P}X_P \\ Z_2 &= a_{21}X_1 + a_{22}X_2 + \cdots + a_{2P}X_P \\ &\vdots \\ Z_q &= a_{p1}X_1 + a_{P2}X_2 + \cdots + a_{PP}X_P \end{aligned} \quad (7)$$

The PCA algorithm assumes that eigenvalues $(\lambda_1, \lambda_2, \ldots, \lambda_P)$ and their corresponding eigenvectors of appropriate matrix should be calculated. The variable Z_1 is determined as the component that corresponds to the greatest eigenvalue. For a univocal solution, Z_1 is established in such a way that the following conditions should be executed:

$$\sum_{i=1}^{P} a_{1i}^2 = 1 \quad (8)$$

The selection of the components can be based on the following criterion:

a) Criterion of sufficient quality of representation – it allows one to take into consideration such initial components that the sum of the variances corresponding to them determines a majority of the total observed variability of data (so it is greater than a certain, predetermined level e.g. 70%),

b) Keiser criterion – it allows one to select components whose variance is larger than 1,

c) Criterion based on the scree plot – on a linear graph, which presents the eigenvalues, there is a chosen and marked point to the right of which a mild decrease in values occurs. According to this criterion, only components whose variances are on the left of that point are taken into consideration.

PROPOSED HYBRID MODEL

Two types of models were trained for diabetic detection. The first one is an SNN fed with patients attributes of diabetic data obtained from ABUTH. The second type is a hybrid model consists of a PCA and SNN classifier. PCA is fed with the patients attributes of diabetic data obtained from ABUTH while SNN is fed with a set of PCs as shown in fig 2.

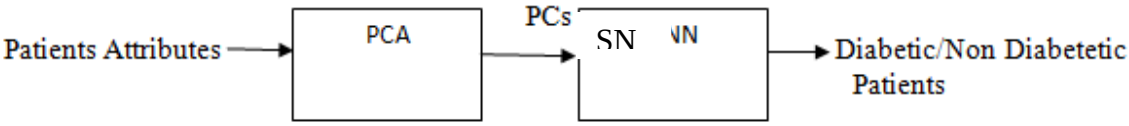

Fig. 2 PCA-SNN model for diabetes detection

For each attribute in the dataset, the enclosed values were normalized in the range [-1,1] to prevent the SNN from being dominated by the input attributes with large values (in case of using SNN only). If the attributes will be fed to PCA, normalization guarantees that PCs will be independent as PCA is sensitive to the relative scaling of the original attributes. Normalization is carried out by removing the feature's average and then dividing by its standard deviation using equation

$$X^1 = \frac{X - mean}{standarddeviation} \quad (9)$$

Where X and X^1 are the old and new value of each feature in the data set respectively.

Afterwards, the data will be divided into 70% for training the model, 15% for validation, and 15% for testing the performance of the trained model.

RESULTS AND DISCUSSION

Table 1 shows variance inflation factor (VIF) of the predictor variables. When correlation exists among predictor's the standard error of predictors coefficients will increase and consequently the variance of predictor's coefficients is inflated. The VIF is a tool to measure and quantify how much the variance is inflated. The VIF value indicates if there is multicollinearity among predictor variables or not; the VIF value of 1 means no correlation, the VIF value between 1 and 5 means moderately correlated and the VIF value of greater than 5 means highly correlated. Using these collinearity statistics we conclude that the data almost certainly indicates a serious collinearity problem.

Table 2 shows total variance explained by each component. An eigenvalue is essentially a ratio of the shared variance to the unique variance accounted for in the construct of interest by each "factor" yielded from the extraction of principal components. An eigenvalue of **1.0 or greater** is the arbitrary criterion accepted in the current literature for deciding if a factor should be further interpreted. The logic underlying the criterion of 1.0 comes from the belief that the amount of shared variance explained by a "factor" should at least be equal to the unique variance the "factor" accounts for in the overall construct.

Figure 3 is a scree plot which provides a visual aid in deciding how many "factors" should be interpreted from the principal components extraction. In a scree plot, the eigenvalues are plotted against the order of "factors" extracted from the data. Because the first "factors" extracted from the principal components analysis often have the highest inter-correlations amongst their individual survey items, and will thus account for more overall variance in your construct of interest, they tend to be extracted first. As other "factors" are extracted, the inter-correlations will become weaker and have smaller eigenvalues. One can look at a scree plot and see a visually significant decrease at one point in time as eigenvalues decrease. This "**elbow**" or factor at which the screen plot has a significant reduction in eigenvalue and then level's off is often considered the criterion for selecting the number of "factors" to interpret. So, based on the two statistical calculations above, the eigenvalues and scree plot make a decision on how many "factors" should be extracted. In this research, it is observed that nine (9) factors should be retained.

Table -1 Detection of Multicollinearity based on collinearity statistics

Explanatory variables	Unstandardized Coefficient		standardized Coefficient	t-statistic	p-value	Collinearity statistics	
	β	S.E	β			Tolerance	VIF
Intercept	0.815	1.051		0.776	0.438		
Age	-0.002	0.002	-0.041	-1.086	0.278	0.569	1.757
Sex	-0.047	0.039	-0.047	-1.224	0.222	0.551	1.814
Occupation	-0.007	0.011	-0.018	-0.597	0.551	0.900	1.112
Educational status	0.024	0.013	0.061	1.868	0.063	0.762	1.312
Ethnicity	-0.039	0.010	-0.128	-3.790	0.000	0.714	1.400
Marital status	-0.016	0.022	-0.023	-0.711	0.477	0.744	1.344
Duration	0.054	0.018	0.093	2.998	0.003	0.845	1.184
Hist_ HP	0.116	0.037	0.123	3.100	0.002	0.511	1.958
F_Hist_ DM	0.206	0.033	0.217	6.317	0.000	0.687	1.456
Chronic smoking	0.132	0.087	0.049	1.522	0.129	0.791	1.264
Alcohol	0.107	0.092	0.038	1.161	0.246	0.767	1.304
DM treatment	0.067	0.028	0.072	2.439	0.015	0.918	1.089
Height	0.130	0.611	0.023	0.213	0.832	0.070	14.339
Weight	-0.003	0.007	-0.099	-0.430	0.667	0.015	64.775
BMI	0.003	0.019	0.037	0.159	0.874	0.015	66.076
West circumference	-0.002	0.002	-0.050	-0.789	0.431	0.198	5.055
SBP	0.002	0.001	0.103	2.325	0.021	0.412	2.427
DBP	-0.001	0.002	-0.032	-0.763	0.446	0.456	2.195
DMR	-0.074	0.042	-0.068	-1.761	0.079	0.535	1.867
FBG	0.029	0.005	0.225	6.373	0.000	0.650	1.539
RBG_2HRP	-0.024	0.005	-0.237	-5.243	0.000	0.398	2.512
HBAIC	-0.067	0.008	-0.356	-8.704	0.000	0.483	2.070
UACR	4.263E-005	0.000	0.017	0.520	0.603	0.787	1.271
NSS	0.006	0.005	0.050	1.200	0.231	0.470	2.130
NDS	-0.013	0.011	-0.098	-1.224	0.222	0.126	7.931
UKST	-0.045	0.081	-0.045	0.556	0.579	0.121	8.260
VPT_HIGH	0.000	0.003	0.005	0.070	0.944	0.160	6.236
VPT	0.106	0.114	0.104	0.925	0.356	0.064	15.641
Combined	-0.067	0.117	-0.065	-0.576	0.565	0.063	15.769

Table -2 PCA total variance explained

	Initial eigenvalues			Extraction Sums of squared loadings		
Component	Total	% of variance	Cumulative %	Total	% of variance	Cumulative %
1	6.608	22.787	22.787	6.608	22.787	22.787
2	3.269	11.273	34.060	3.269	11.273	34.060
3	2.019	6.961	41.021	2.019	6.961	41.021
4	1.984	6.842	47.863	1.984	6.842	47.863
5	1.550	5.346	53.209	1.550	5.346	53.209
6	1.367	4.715	57.924	1.367	4.715	57.924
7	1.219	4.203	62.127	1.219	4.203	62.127
8	1.139	3.926	66.053	1.139	3.926	66.053
9	1.035	3.569	69.622	1.035	3.569	69.622
10	0.970	3.345	72.967			
11	0.948	3.270	76.237			
12	0.823	2.838	79.075			
13	0.795	2.742	81.817			
14	0.675	2.327	84.144			
15	0.652	2.250	86.393			
16	0.556	1.917	88.310			

17	0.518	1.786	90.097			
18	0.493	1.701	91.797			
19	0.460	1.585	93.383			
20	0.408	1.406	94.789			
21	0.349	1.205	95.993			
22	0.279	.961	96.955			
23	0.260	.895	97.850			
24	0.251	.866	98.717			
25	0.134	.464	99.180			
26	0.123	.425	99.605			
27	0.072	.250	99.855			
28	0.035	.121	99.976			
29	0.007	.024	100.000			

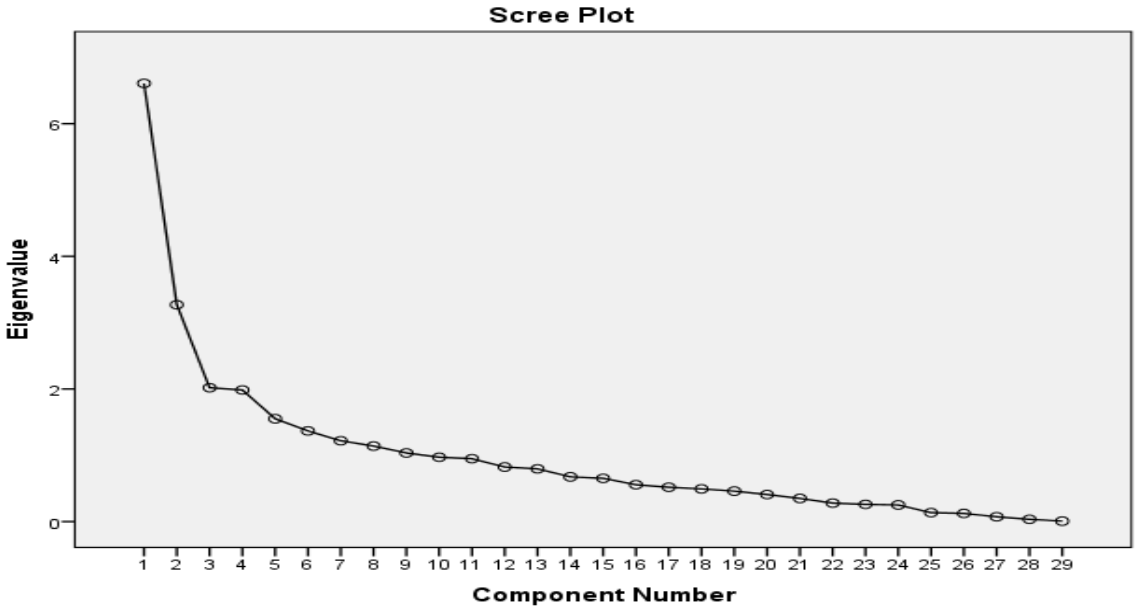

Fig. 3 Scree Plot of principal component analysis

The neural network model was trained using Levenberg-Marquardt (LM) training algorithms. An Intel (R) Core (TM) i3-2310M CPU @ 2.10GHz processor was used to train the neural network model. Figure 4.2 is the example of how neural network is trained using two different architectural designs in figure 4 and figure 5 for the original and reduced data respectively.

In order to train, validate and test the neural networks developed using the LM algorithms, we have divided the data set in the following way: 70% of it for the training process, 15% for the validation process and the remaining 15% for the testing process. In all the cases, the samples have been randomly chosen as to cover the specified percentages. In order to train the neural networks, we have used the mean square error (MSE) as an objective function. When training a network with this function, if there are multiple outputs having different ranges of values, the accuracy is optimized for the output element that has a wider range of values and is less optimized relative to the output element with a smaller range of values. Thus, the network will learn to fit the first output element very well, while the second output element is not fit as accurate as the first. In order to solve this issue, we have normalized the errors, by setting the normalization performance parameter to its 'standard' value. By using this method, the errors have been computed as if both of the output elements had values ranging from -1 to 1 and consequently, the two output elements have been fitted very well.

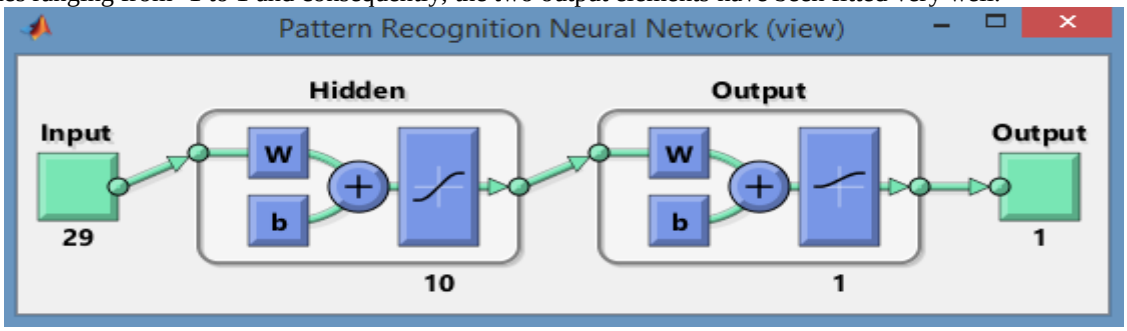

Fig. 4 Pattern Recognition Neural Network (view)

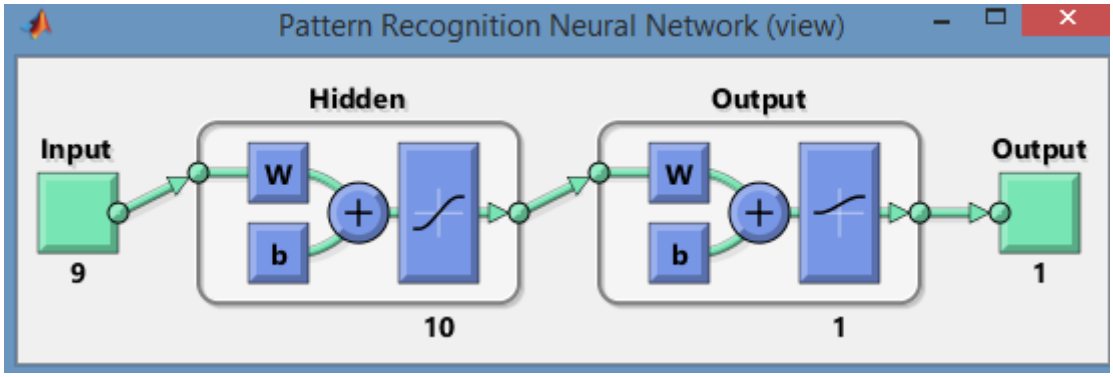

Fig. 5 Pattern Recognition Neural Network (view)

Table -3 Cofusion Matrix at 5 hiddeen neurons for the original data

	Actual Classifiaction		
Predicted classifaction	**Positive (Diabetic patients)**	**Negative (Non-diabetic patients)**	**Total**
Positive (Diabetic patients)	TP=265	FP=2	267
Negative(Non-diabetic patients)	FN=2	TN=124	126
Total	267	126	393

Table -4 Cofusion Matrix at 10 hiddeen neurons for the original data

	Actual Classifiaction		
Predicted classifaction	**Positive (Diabetic patients)**	**Negative (Non-diabetic patients)**	**Total**
Positive (Diabetic patients)	TP=262	FP=0	262
Negative(Non-diabetic patients)	FN=5	TN=126	131
Total	267	126	393

Table -5 Cofusion Matrix at 5 hiddeen neurons for the reduced data

	Actual Classifiaction		
Predicted classifaction	**Positive (Diabetic patients)**	**Negative (Non-diabetic patients)**	**Total**
Positive (Diabetic patients)	TP=256	FP=13	269
Negative(Non-diabetic patients)	FN=11	TN=113	124
Total	267	126	393

Table -6 Cofusion Matrix s at 10 hiddeen neurons for the reduced data

	Actual Classifiaction		
Predicted classifaction	**Positive (Diabetic patients)**	**Negative (Non-diabetic patients)**	**Total**
Positive (Diabetic patients)	TP=255	FP=10	265
Negative(Non-diabetic patients)	FN=12	TN=116	128
Total	267	126	393

Table -7 Comparison of Neural Network Classification Performance at original and reduced data

Architecture	**Indices**	**Original data**	**Reduced data**
Five (5) hidden Neurons	Accuracy	99.0%	93.9%
	Sensitivity	99.3%	95.9%
	Specificity	98.4%	89.7%
	Positive Predictive Value	99.3%	95.2%
	Negative Predictive Value	98.4%	91.1%
Ten (10) hidden Neurons	Accuracy	98.7%	94.4%
	Sensitivity	98.1%	95.5%
	Specificity	100%	92.1%
	Positive Predictive Value	100%	96.2%
	Negative Predictive Value	96.2%	90.6%

The *accuracy* (*AC*) is the proportion of the total number of predictions that were correct. Neural network classifier achieved 99.0% and 98.7% accuracies for 5 and 10 hidden neurons architectural design respectively when original data is used. Similarly, 93.9% and 94.4% accuracies were obtained respectively for 5 and 10 hidden neurons architectural designs when the reduced is used.

Sensitivity is the probability that a test will indicate 'disease' among those with the disease. Neural network classifier achieved 95.3% and 94.9% sensitivities for 5 and 10 hidden neurons architectural design respectively when original data is used. Similarly, 91.4% and 85.3% sensitivities were obtained respectively for 5 and 10 hidden neurons architectural designs when the reduced is used.

Specificity is the fraction of those without disease who will have a negative test result. Neural network classifier achieved 98.4% and 100% specificity for 5 and 10 hidden neurons architectural design respectively when original data is used. Similarly, 89.7% and 92.1% specificity were obtained respectively for 5 and 10 hidden neurons architectural designs when the reduced is used.

The positive predictive value (PPV) of a test is defined as the proportion of people with a positive test result who actually have the disease. Neural network classifier achieved 99.3% and 100% PPV for 5 and 10 hidden neurons architectural design respectively when original data is used. Similarly, 95.2% and 96.2% PPV were obtained respectively for 5 and 10 hidden neurons architectural designs when the reduced is used.

The Negative Predictive Value (NPV) of a test is the proportion of people with a negative test result who do not have disease. Neural network classifier achieved 98.4% and 96.2% NPV for 5 and 10 hidden neurons architectural design respectively when original data is used. Similarly, 95.2% and 96.6% NPV were obtained respectively for 5 and 10 hidden neurons architectural designs when the reduced is used.

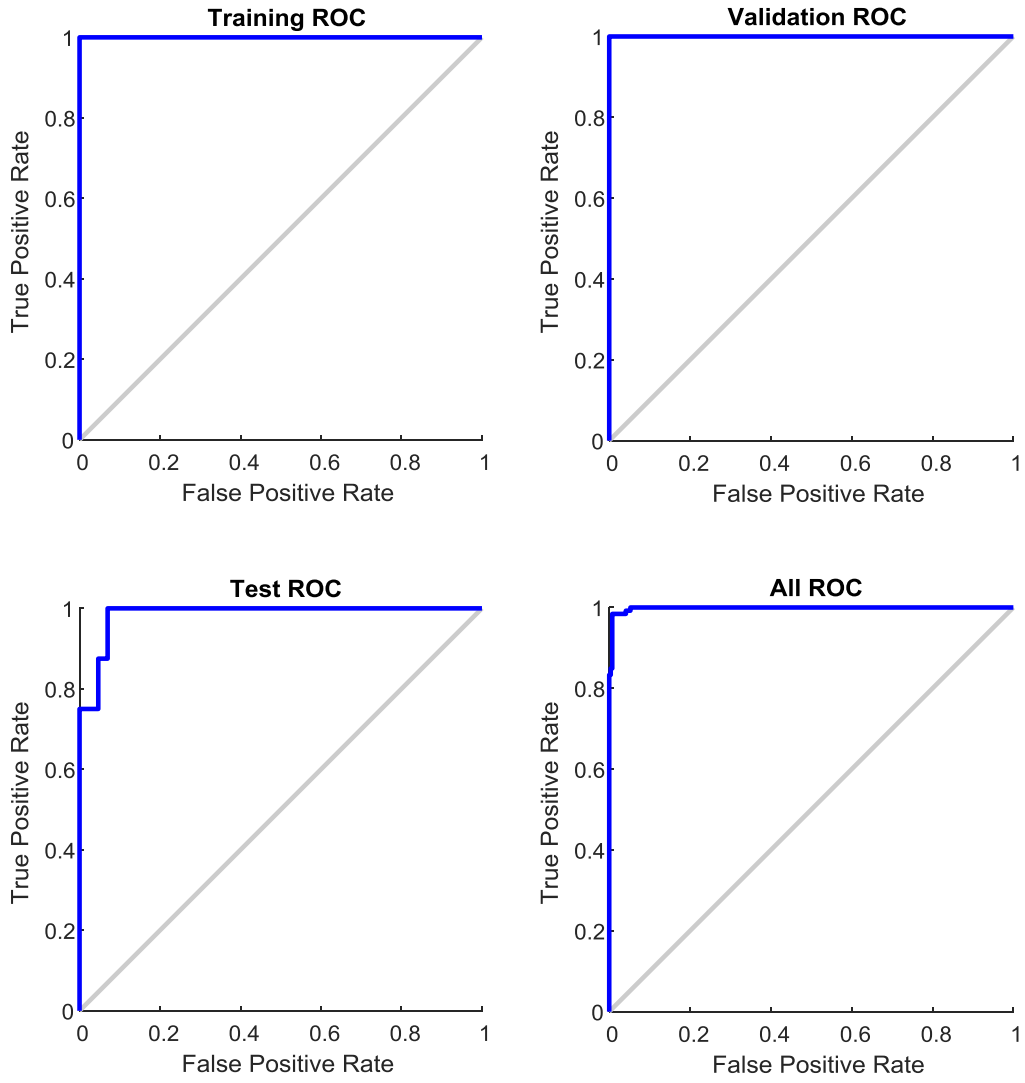

Fig. 6 Receiver Operating Curve (ROC) of the original data at five (5) hidden neurons

The diagonal joining the point (0, 0) to (1, 1) divides the square in two equal parts and each has an area equal to 0.5. When ROC is this line, overall there is 50-50 chances that test will correctly discriminate the diabetic and non-diabetic subjects. The minimum value of AUC should be considered 0.5 instead of 0 because AUC = 0 means test incorrectly classified all subjects with disease as negative and all non-disease subjects as positive. If the test results are reversed then area = 0 is transformed to area= 1; thus a perfectly inaccurate test can be transformed into a perfectly accurate test. It is very clear that all the ROCs in figure 8 have shown that the area covered are far greater than 0.5 because the lines of the curvesare at the top left of the graphs. This therefore suggests that the model has been able to excellently discriminate between the diabetic and non diabetic patients.

CONCLUSION

The automatic diagnosis of diabetes is an important real-world medical problem. Detection of diabetes in its early stages is the key for treatment. This study shows how principal component analysis (PCA) and artificial neural network are used to predict actual diagnosis of diabetes patients visiting Ahmadu Bello University Teaching Hospital (ABUTH) Zaria for local and systematic treatment, along with presenting related work in the field. The experimental results show that there exists multicollinearity among the predictors of the diabetes in patients as variance inflation factors (VIF) of some of these variables exceeds five (5).PCA was used in the pre-processing stage to extract the important factors, which were then included in the ANN model.

Artificial Neural Networks (ANNs) trained using the Levenberg-Marquardt algorithm, were later designed for both the original (unreduced data) and reduced data in order to predict diabetes in patients who visit ABUTH Zaria. The results of a five (5) hidden neurons Artificial Neural Network classifier have achieved 99.0% accuracy when the original data was used as against 93.9% accuracy when the data was reduced by the principal components analysis. Similarly, the results of a 10 hidden neurons Artificial Neural Network classifier have achieved 98.7% accuracy when the original data was used as against 94.4% accuracy when the data was reduced by the principal components analysis. This has therefore demonstrated the ability of Artificial Neural Network algorithm to predict diabetes in Patients of ABUTH, Zaria is not affected by the effects of multicollinearity as it gives better prediction when all the variables are used despite that the correlations exist among them.

REFERENCE

[1]. https://www.who.int/news-room/fact-sheets/detail/diabetes

[2]. Aguiree, F., Brown, A., Cho, N. H., Dahlquist, G., Dodd, S., Dunning, T., Hirst, M., Hwang, C., Magliano, D., Patterson, C. & Scott, C. (2013). IDF diabetes atlas.

[3]. Oputa R.N. and Chinenye S., (2015). Diabetes in Nigeria – a translational medicine approach. *African Journal of Diabetes Medicine*, 23 (1), 7-10.

[4]. Wang K.S., Liu Y., Xie X., Gong S., Xu C., and Sha S., (2017). Component Regression Analysis of Nutrition Factors and Physical Activities with Diabetes, *Journal of Biometrics & Biostatistics*, 8(4), 1-7.

[5]. Özkan M.M. and Mendes M. (2004). Empirical Table Values for Different Variable Numbers and Sample Size Combinations, *Pakistan Journal of Biological Sciences*, 7(6), 870-878.

[6]. Joliffe I.T. (2002). *Principal Component Analysis*, Springer.

[7]. Livingstone, D., Totowa, NJ (2008). *Artificial Neural Networks Methods and Application*. 1st ed.: Hummana Press

[8]. Dunne, RA., Wiley, J., Inc, S. (2007) "*A Statistical Approach to Neural Networks for Pattern Recognition*", New Jersey: John Wiley & Sons Inc.

[9]. Rahimloo P., and Jafarian A., (2016). Prediction of Diabetes by Using Artificial Neural Network, Logistic Regression Statistical Model and Combination of Them, *Bulletin de la Société Royale des Sciences de Liège*, 85, 1148 – 1164.

[10]. Mishra V., Samuel C. and Sharma S. K. (2015). Use of Machine Learning to Predict the Onset of Diabetes. *International Journal of Recent advances in Mechanical Engineering*, 4(2), 9-14.

[11]. Wu H., Yang S., Huang Z., He J., and Wang X. (2018). Type 2 diabetes mellitus prediction model based on data mining, *Informatics in Medicine Unlocked*, 10, 100-107.

[12]. Selvakumar S., Senthamarai K.K and Gothai Nachiyar S., (2017) Prediction of Diabetes Diagnosis Using Classification Based Data Mining Techniques. *International journal of Statistics and Systems*, 12(2), 183-188.

[13]. Zou D, Ye Y., Zou N. and Yu J (2017). Analysis of risk factors and their interactions in type 2 diabetes mellitus: A cross- sectional survey in Guilin, China, *Journal of Diabetes Investigation*, 8,188–194.

[14]. Hagan, M.T., & Menhaj, M., (1994) "Training feed-forward networks with the Marquardt algorithm", IEEE Trans. *Neural Networks*, Vol. 5, No. 6, pp989-993.