Subject Specific Treatment to Neural Networks for Repeated Measures Analysis

Tanmay Kumar Maity and Asim Kumar Pal

Abstract—**Analysis of repeated measures data for the purpose of prediction is not an easy task particularly when the problem under consideration is highly nonlinear, number of subjects is large and the sample available to learn the model is small. The efficacy of the ANN for subject level treatment has been studied here empirically. Data were generated through a random coefficient model and a few nonlinear mixed effect models. For ANN feedforward backprop has been tried. Simulations have been conducted with varying number of covariates and parameters (both common and subject dependent), number of subjects and different sizes of repeated measures. ANN has demonstrated considerable promise.**

Index Terms—**ANN learning, longitudinal analysis, mixed effect model, panel data, random coefficient model.**

I. Introduction

REPEATED measures data refer to the data generated through observing a number of subjects repeatedly under various experimental conditions. A common type of repeated measures data are longitudinal data where each subject or individual is observed at different time points. The biggest advantage of analyzing repeated measures or longitudinal data is that each subject is observed for several occasions. For any given set of repeated measures the data for all the subjects together refer to a common problem. Therefore it is natural to assume that data for all the subjects share a common characteristic referred as *common effect* and the data for any one subject differ from that for another so far as the two subjects differ in their individual characteristics. For example, if we consider the effect of nutrition on health of any individual it is expected that each person would improve her or his health given additional nutritional inputs, but the growth rate would differ from person to person. Further for each person there may be a different point in the nutritional input level wherefrom there will be a gradual decline in health for any additional nutrition. This difference in growth rate, point of retardation and rate of decline between any two individuals is due to what is known as *subject specific effect*. For repeated measures or longitudinal data often the number of data points (measurements) for any given subject would be too inadequate to learn the underlying model using the data meant for that subject alone. Further, if one does succeed in achieving a model specific to each subject it may not serve any good purpose towards understanding the common characteristics that apply to the given problem. Continuing on the previous example, the effect of different nutritional inputs (carbohydrates, proteins and vitamins) on human health (body weight, height, resistance to diseases, etc.) would be captured differently. It will not be possible for doctors to suggest common approaches to health, nor would it be feasible to suggest different emphasis to individual patients from the common approach. Similar kind of things would occur while curing a disease by applying single or multiple doses of different drugs (particularly if some drugs are newly introduced). All these mean learning a separate model for each subject in general is not feasible given the amount of data, and even if it is feasible does not serve the purpose. Finally, the analysis of repeated measures data for the purpose of prediction is a challenging problem particularly when the problem under consideration is highly nonlinear, number of subjects is large and the sample available to learn the model is small.

One of the earliest methods proposed for analyzing repeated measures data or longitudinal data was a mixed-effect ANOVA or univariate repeated-measures ANOVA, with a single random subject effect. The linear mixed-effects model is the most widely used method for analyzing longitudinal data. The linear mixed-effects model proposed by Laird and Ware included the univariate repeated-measures ANOVA and growth curve models for longitudinal data as special cases [1]. When the longitudinal response is discrete, linear models are no longer appropriate for relating changes in the mean response to independent variables. Instead, extensions of *generalized linear models* have been developed [2]. Here three broad, but quite distinct, classes of regression models have been considered for longitudinal data: (i) *marginal* or *population averaged* model [15], [16] (ii) *random-effects* or *subject-specific* model and (iii) *transition* model [17]. Following the same basic ideas as in linear mixed-effects models, generalized linear models can be extended to longitudinal data by allowing a subset of the regression coefficients to vary randomly from one individual to another [10]-[13]. These models are known as *generalized linear mixed models* (GLMMs). In contrast to the marginal and generalized linear mixed models which allow for non-linearity in a restricted way, in non-linear mixed-effect models (NLME) the mean response is assumed to be non-linear in the regression parameters and the random effects [3]-[5], [14].

Artificial Neural Network (ANN) has been proved in many cases to be superior in terms of predictive capability

Manuscript received January 19, 2013; revised January 30, 2013.

Tanmay Kumar Maity is with Dept. of Statistics, Haldia Govt. College under Vidyasagar University, West Bengal, India (e-mail: tanmaymaity@gmail.com).

Asim Kumar Pal is with MIS group, Indian Institute of Management Calcutta, India (e-mail: asim@iimcal.ac.in).

compared to various statistical methods for modeling cross sectional or series data where the actual relationship between the variables cannot be specified or is not known a priori. It provides a flexible nonlinear modeling technique without requiring any or much domain knowledge about the inter-relationship between the variables. It provides a flexible class of models in an unrestricted manner and it learns automatically from the training data to estimate the parameters (weights) unlike statistical models where the actual form of the model has to be provided. However, not much work has been done for the analysis of repeated measures or longitudinal data using neural networks. Differently from conventional longitudinal or panel models a standard ANN does not include temporal correlation. Therefore the main question is how can ANN recognize and treat the time or subject correlation in the data? Two approaches were used in [6] and [7] to capture the time effect or subject effect in the model. The first approach uses time or subject dummy variables as inputs along with other covariates. The second one employs a variable that identifies – by means of a text (string) variable – the time points concerned. This approach is made possible by internally rescaling the text variable, e.g., year value is associated with a numerical value within (0, 1), therefore identifying year-specific intercepts for time. In these papers no specific architecture of ANN for longitudinal data had been considered. Tandon et al. [8] propose a neural network designed for longitudinal data called mixed effects neural network (MENN). Here the progression of Alzheimer's disease has been studied using MENN which generalize the mixed effect model by incorporating a general nonlinear function of the input variables. A modified likelihood function has been formed using some matrix transformation and a modified backpropagation algorithm and an iteration procedure has been used to maximize the likelihood function and thus the weights have been estimated. But here ANN hadn't been allowed to learn automatically from the data, only a specific form of the model (network having only the input layer and the output layer along with random subject effect) has been assumed. None of these works refer to purely repeated measures data. Here we have looked into the capability of ANN to model repeated measures data.

Contribution

There are two main objectives of the present work are to find out: a) the utility of ANN for modeling repeated measures data particularly for prediction, and b) how the subject effects can be incorporated in the ANN architecture. For this purpose the most popular ANN architecture, namely feed-forward multi-layer ANN with back propagation learning has been used. The method has been tried on four different reasonably difficult problems, one linear, and three non-linear (two rational functions and one posynomial function – polynomial with real powers), simulations based on Random Coefficient model and Nonlinear Mixed Effect model. We have tried the functions with varying number of input variables and varying number of parameters per subject. We have also tested the method to see the effectiveness of the model when the number of repeated observations available for a subject ranges from being 'very small' to 'small' compared to the problem complexity.

In the next section there is a brief discussion on Random coefficient model and Nonlinear mixed effect model. The experimental set up for the simulations are presented in sections III. Section IV talks about the ANN architectures used for analyzing repeated measures data, particularly the one which gives a special treatment to the subjects separately from the input variables. Section V discusses the performance measures for prediction. Finally, the last two sections contain the analysis of the experimental results and conclusions.

II. Random coefficient & NLME models

In Random Coefficient model the parameters are assumed to be drawn randomly from a given distribution:

$E(y_{ij}) = x_{ij}^{'}\beta_i, i = 1(1)s, j = 1(1)t, \beta_i \sim (\beta, \Sigma)$

Here β_i is randomly selected from a distribution with mean vector β and dispersion matrix Σ.

The Nonlinear Mixed Effect model (NLME) has been developed in two stages: a model for intra-subject variability and a model for inter-subject variability.

Stage 1: Intra-subject model: This specifies the mean and covariance structure for a given subject. Here it is assumed that the mean response can be expressed in terms of a non-linear regression function of covariates and regression parameters: $E(y_{ij}) = f(x_{ij}, \beta_i)$. Although the functional form, *f(.)*, is the same for all subjects, differences between subjects in their longitudinal response trajectories are accommodated by allowing for different β_i as well as differences in the covariates, x_{ij}.

Stage 2: Inter-subject model: The second-stage model characterizes inter-subject variation in regression parameters β_i given by, $\beta_i = g(A_i, \beta, b_i)$, where A_i's are the covariates. The random effects $b_i \sim (0, \Sigma)$, β is the fixed effect parameter and $g(\cdot)$ is a known vector-valued function.

III. Simulation Experiments

Experiments are conducted on repeated measures data reflecting different scenarios. For the parameter space both random coefficient model and NLME (see above) are considered. The data simulation also considers the notion that two main effects may drive the response variable of the repeated measure data: 1) *Common Effect*: This effect is common affecting all subjects in the same manner, 2) *Subject Specific Effect*: This effect depends on a particular subject and it varies from subject to subject. It is regarded as created by some unobserved factors. *Common effect* can be captured by taking the same parameter(s) across different subjects. The parameters are varied from one subject to another but kept same for different repeated measures for a particular subject in the case of *Subject specific effect*. Here the parameters have been drawn randomly from Normal distribution with a mean (common effect) and a standard deviation (subject specific effect). The data have been simulated using four different models: one linear and three non-linear (two rational functions and one posynomial):

1) LINEAR model: Here the simulation is performed based on random coefficient model with varying number independent variables (3, 4, 7 and 10):

$$E(y_{ij}) = \alpha + b_i' x_{ij}, i = 1(1)s, j = 1(1)t,$$
$$b_i = (b_{i1}, b_{i2}, ..., b_{ik})', x_{ij} = (x_{1ij}, x_{2ij}, ..., x_{kij})'$$
$$b_i \sim N(\beta, \Sigma), x_{ij} \sim Unif(-10,10)$$

Each independent variable has been drawn randomly from Uniform (-10, 10) distribution and α is the fixed parameter and N denotes the Normal or Gaussian distribution. Here, x's are input variables, y's are output, i is the subject index, j the observation index and b's stand for parameter values coming from normal distribution with parameters β and Σ. Similar notations have been followed throughout.

2) RATIONAL1 model: Here the simulation is based around a rational function $f(x) = \frac{(x-2)(2x+1)}{1+x^2}$ (used in many examples in [9]) and the concept of NLME model:

$$E(y_{ij}) = \frac{(x_{ij} - b_{i1})(b_{i2} x_{ij} + 1)}{b_{i3} + x_{ij}^2}, i = 1(1)s, j = 1(1)t,$$
$$b_i = (b_{i1}, b_{i2}, b_{i3})' \sim N(\beta, \Sigma), x_{ij} \sim Unif(-12,12)$$

There is only one independent variable which has been drawn randomly from Uniform (-12, 12) distribution.

3) RATIONAL2 model: This is same as RATIONAL1 except with an extra additive parameter (This is also a rational function, apparently more complex than Rational1.):

$$E(y_{ij}) = \frac{(x_{ij} - b_{i1})(b_{i2} x_{ij} + 1)}{b_{i3} + x_{ij}^2} + b_{i4}, i = 1(1)s, j = 1(1)t,$$
$$b_i = (b_{i1}, b_{i2}, b_{i3}, b_{i4})' \sim N(\beta, \Sigma), x_{ij} \sim Unif(-12,12)$$

4) POSYNOMIAL model: This simulation is based on a polynomial with non integer powers which are random variables:

$$E(y_{ij}) = \alpha_1 x_{1ij}^{b_{i1}} + \alpha_2 x_{2ij}^{b_{i2}} + \alpha_3 x_{3ij}^{b_{i3}}, i = 1(1)s, j = 1(1)t,$$
$$b_i = (b_{i1}, b_{i2}, b_{i3})' \sim N(\beta, \Sigma),$$
$$x_{kij} \sim Unif(0,4), k = 1(1)3$$

Here three independent variables have been used and each drawn randomly from Uniform (0, 4) distribution and $\alpha_1, \alpha_2, \alpha_3$ are three fixed parameters.

The above models are simulated without using any additive noise to the response variable. Also the Σ matrix is taken to be a diagonal matrix i.e., the subject specific parameters b_i 's are assumed to be independently distributed.

IV. ANN ARCHITECTURE

The simulated data have been fitted by different feedforward architectures with backprop training and the error function is optimized using Levenverg-Marquardt method (for computation NN toolbox of MATLAB 2012A has been used). Sigmoid and linear activation functions are considered for the hidden layers and the output layer respectively. Different sizes of training sets (alongwith cross validation sets for early stopping) are experimented with. We have mainly tried to see the efficacy of ANN when the sample size is very small or small with respect to the complexity of the problem attempted. Testing data sizes have also been varied.

Subjects are fed to the network as dummy variable. For example, if we have 4 subjects, subject 1 is taken as [1 0 0 0], subject 2 as [0 1 0 0], subject 3 as [0 0 1 0] and subject 4 as [0 0 0 1], i.e. for any data row (or, example) only the corresponding subject node gives positive response, other subject nodes or subject lines remain inactive.

Fig. 1 gives ANN1 which is the traditional feedforward ANN architecture with a single hidden layer for repeated measures data, where both the covariates and subjects inputs are connected in the same way to the only hidden layer:

$$y_i = \xi_0' + \sum_{j=1}^{k_2} \xi_j' h(u_j'); u_j' = \sum_{i=1}^{k_1} w_{ji}' x_i + w_{j0}' + s_j'$$

where, s_j' be the j[th] component of the subject level connection, x_i is the i[th] input, y_i is the output, w' , ξ' are the input layer to 1st hidden layer and 1[st] hidden layer to output layer weights respectively, ξ_0' and w_{j0}' are bias terms, h(.) is the transfer function of the hidden layer, k_1 be the number of independent variables and k_2 be the number of nodes in the hidden layer. Here the common effect and the subject specific effect are learned simultaneously.

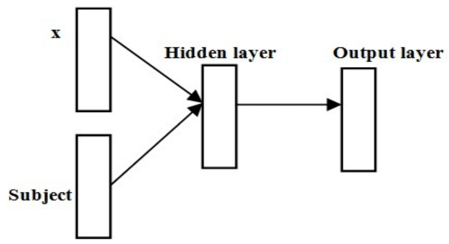

Fig. 1: ANN1 - Identical treatment to common effect and subject specific effects.

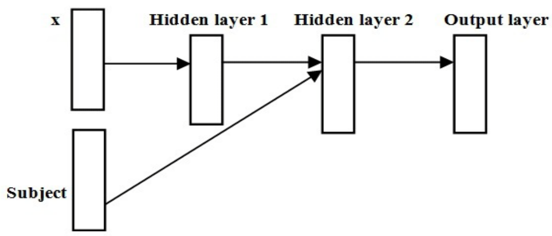

Fig. 2: ANN2 - Treatment differs from common effect to subject specific effects.

In ANN2 subject specific effects are treated differently from as compared to the common effect. Here independent variables are connected from input layer to the first hidden layer but the subject inputs are connected directly to the 2[nd] hidden layer, which is connected to the output layer (see Fig. 2):

$$y_i = \xi_0 + \sum_{k=1}^{l_2} \xi_k h_2(z_k);$$

$$z_k = \sum_{j=1}^{l_1} \psi_{kj} h_1(u_j) + \psi_{k0} + s_k; u_j = \sum_{i=1}^{k_1} w_{ji} x_i + w_{j0}$$

where, s_k be the k[th] component for the subject level connection, w , ψ , ξ are input layer to 1[st] hidden layer, 1[st] hidden layer to 2[nd] hidden layer and 2[nd] layer to output layer weights respectively, ξ_0 , ψ_{k0} , w_{j0} are bias terms, h_1(.) and h_2(.) be transfer functions and l_1 and l_2 be the number of nodes of the 1[st] and 2[nd] hidden layer respectively. Here the

ISBN: 978-988-19251-8-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

common effect is first learned through 1st hidden layer and then the subject specific effect is learned and these two effects are superimposed at 2nd hidden layer.

V. Performance metrics

Results obtained from the ANN models have been verified using well known performance metrics such as RMSE (Root mean square error), MAPE (Mean absolute percentage error) and Normalized Mean Square error (NMSE). The measures are given below:

i) RMSE: $\sqrt{\frac{1}{n}\sum_i (y_i - \hat{y}_i)^2}$

ii) MAPE: $\sqrt{\frac{1}{n}\sum_i \left|\frac{y_i - \hat{y}_i}{y_i}\right|} \times 100$

iii) NMSE: $\frac{\sqrt{\sum_i (y_i - \hat{y}_i)^2}}{\sqrt{\sum_i (y_i - \bar{y})^2}}$

$y_i, \hat{y}_i$ are the ith observed and predicted variable respectively.

As the response variables from various simulated models are in different scale, it is necessary to use a scale free performance measure for comparing the results. RMSE is highly dependent on the range of the response variable. Though MAPE doesn't depend on the magnitude of the response variable but it could be severely affected by response variable values which are close to zero. NMSE which is scale free and is proportional to RMSE has been preferred for comparison here. NMSE=1 indicates all the predictions have converged to the mean prediction, and hence NMSE << 1 is preferable.

VI. Analysis and Findings

Two ANN models (given in Fig. 1 and Fig. 2) have been attempted on the four simulated problems (see above): LINEAR, RATIONAL1, RATIONAL2 and POSYNOMIAL (Refer to Table I through Table III). A large number of simulations have been performed considering complexity of the models and learning issues involved. Yet one has to admit that in this limited scope experiment only some selected simulations were carried out to have a primary understanding of the issues involved. In the experimental set up presented, each case represents one problem (e.g. LINEAR) with a given function complexity and sample complexity (discussed below). Thus, a total of thirteen cases (problem variants) have been attempted: LINEAR - 7, RATIONAL1 - 2, RATIONAL2 - 2 and POSYNOMIAL – 2. For each case both ANN1 and ANN2 have been experimented and the best architecture obtained is presented. For each ANN model, a set of architectures have been tried. For each architecture, the ANN learning has occurred based on training sample along with the cross validation sample for early stopping. Moreover, for each architecture tried for any case, the ANN solution has been obtained fifty times, every time starting with a new random initial weight vector as selected by the ANN software. The performance for any case for any given architecture is thus based on 50 runs, and the median performance (NMSE) has been used here.

The problem complexity in a repeated measures analysis depends on the function or problem such as linear or non-linear along with its number of arguments chosen, i.e. number of input variables, number of subject independent parameters, number of subjects and number of subject specific parameters. The total function complexity can be split into three things: generic function complexity, size complexity (input variables and subject independent parameters) and subject complexity (number of subjects and subject specific parameters). In an ANN experiment the learning capability arises from three sources, one is called the modeling complexity which is basically ANN architecture with sizes (correspondingly implying the number of free parameters in the model – the weight vector size), the second is the training sample complexity (this includes the size of the training set as well as that of the cross validation set used for early stopping the training to avoid over training) and last is the generalization sample complexity which is nothing but size of the test set. Bigger the test set more reliable is the performance measure obtained.

Table I
PERFORMANCE FOR LINEAR MODEL

Case	#Inputs[a], #Subjects[b], #Subject specific parameters[c]	#Obs. per Subject[d]- #Training, #Validation, #Testing	ANN1		ANN2	
			NMSE[e] (%) (two cases)	Network Size[f]	NMSE (%) (two cases)	Network Size[g]
1	3, 25, 3	25, 10, 200	21.50	3	26.17	3, 3
			0.34	4	0.19	3, 4
2	3, 100, 3	6, 2, 2	20.86	3	26.23	6, 3
			0.24	4	0.37	6, 4
3	4, 25, 4	60, 20, 20	10.23	4	11.70	4, 4
			0.17	5	0.12	4, 5
4	7, 25, 7	60, 20, 20	2.12	7	3.50	7, 7
			0.14	8	0.15	7, 8
5	7, 25, 7	25, 10, 200	21.38	6	1.22	7, 7
			0.40	7	0.25	7, 8
6	10, 25, 10	25, 10, 200	13.27	9	51.38	9, 10
			0.20	10	51.13	9, 9
7	10, 25, 10	60, 20, 20	12.88	9	15.74	10, 9
			0.14	10	0.19	10, 10

[a] No. of independent variables/ covariates, [b] No. of subjects, [c] No. of subject specific parameters (here for each case no. of fixed parameters is 1), [d] No. of repeated measurements for each subject for training, validation and testing set, [e] Median NMSE over 50 runs for two best architectures upto the point where a drastic improvement in performance (<1%) occurs, [f] No. of nodes in hidden layer, [g] No. of nodes in 1st and 2nd hidden layer respectively.

Now let's analyze the performance of two ANN models, ANN1 (single hidden layer with equal emphasis to variables and subjects) and ANN2 (two hidden layers with special emphasis to differentiate subject specific effects from common effects) for predicting from repeated measures data for four simulated functions.

LINEAR is a linear random coefficient model. For this simulation the number of independent variables (inputs) has varied: 3, 4, 7 and 10 and a total of seven variants have been tried (see Table 1). For all these 7 cases, as expected, it has been observed that as the number of nodes in a hidden layer increases the performance improves for both ANN1 and ANN2, i.e. NMSE decreases. Also a drastic fall in NMSE has been observed after a certain number of hidden nodes in

ISBN: 978-988-19251-8-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

each case (except case 6 for ANN2), e.g. for Case 1 the ANN1 with 3 hidden nodes gives NMSE of 21.50% and with 4 nodes it gives 0.34% and with ANN2 having (3, 3) architecture – 3 in the first hidden layer and 3 in the second hidden layer – the NMSE is 26.17% but with (3, 4) it reduces drastically to 0.19%. Note that in this case there are 3 input variables, 1 subject independent parameter, 3 subject level parameters, 25 subjects, training set and cross validation set sizes per subject are 25 and 10 respectively and test set size is 200 (i.e. overall 625, 250 and 5000 training, cross validation and testing set sizes). The performance obtained in LINEAR model is very good throughout irrespective of ANN1 or ANN2, NMSE ranges from 0.12% to 0.40%, except case 6 all are 0.40% or less. Performance for ANN1 and ANN2 are comparable, though ANN1 complexity is less. This seems possible because the linear model is relatively simple. But it is to be noted that ANN2 has not given satisfactory result for Case 6. For this case ANN1 also required a large size network. The Case 6 is the most complex linear problem in terms of function size and subject complexity and also (training) sample complexity. ANN2 possibly requires a sufficiently big network for which the sample is too small. This observation is amply supported by good performance demonstrated by ANN2 for Case 5 (with less function size and subject complexity, but same sample complexity) as well as for Case 7 (with same function size and subject complexity, but less sample complexity).

Finally, it can be seen as the problem complexity increases the ANN model (ANN1 or ANN2) needs to be bigger. However, more experiments are required to draw further conclusions regarding effects of subject complexity and sample complexity on the ANN modeling capability.

Coming to RATIONAL1 and RATIONAL2, ANN2 gives significantly better result than ANN1. This is important, because it tells subject specific treatment is required when the problem becomes little complex. Further, for the rational functions (apparently more difficult to learn), the training set size plays a significant role– compare case 1 and 2 for RATIONAL1, NMSE is 18.34% (training size 6) and 7.31% (training size 60). Similarly also for RATIONAL2 – 11.56% vs. 4.66%.

Table II
PERFORMANCE FOR
RATIONAL1 (R1) and RATIONAL2 (R2) MODEL

Case	#Inputs, #Subjects, #Subject specific parameters[a]	#Obs. per Subject - #Training, #Validation, #Testing	ANN1 Best NMSE[b] (%)	ANN1 Network Size	ANN2 Best NMSE (%)	ANN2 Network Size
1 (R1)	1, 100, 3	6, 2, 2	81.51	6	18.34	7, 2
2 (R1)	1, 100, 3	60, 20, 20	11.11	9	7.31	7, 5
3 (R2)	1, 100, 4	6, 2, 2	59.97	6	11.56	8, 2
4 (R2)	1, 100, 4	60, 20, 20	7.10	9	4.66	7, 5

[a] No fixed parameter, [b] Best performance based on median NMSE over 50 runs (for ANN1 upto size 9 and for ANN2 upto size 8, 5 have been tried). This set-up is also followed for POSYNOMIAL.

In POSYNOMIAL the result however is a bit mixed, in one case ANN1 (0.37%) is slightly better than ANN2 (0.49%) when training size is 60, in the other case ANN2

Table III
PERFORMANCE FOR POSYNOMIAL MODEL

Case	#Inputs, #Subjects, #Subject specific parameters[a]	#Obs. per Subject - #Training, #Validation, #Testing	ANN1 Best NMSE (%)	ANN1 Network Size	ANN2 Best NMSE (%)	ANN2 Network Size
1	3, 25, 3	60, 20, 20	0.37	9	0.49	8, 5
2	3, 25, 3	25, 10, 200	2.45	9	1.55	8, 5

[a] No. fixed parameter

(1.55%) beats well ANN1 (2.45%) when training size is 25 – which is a harder problem because of smaller training set. Thus on the whole here also it looks like that ANN2 outperforms ANN1, but more evidence needs to be collected.

The efficiency of ANN for the rational functions is the worst, possibly indicating that these are harder functions to learn. In general we can say that ANN can be used to solve a reasonably complex problem in repeated measures domain. The solution complexity depends on all aspects of the function complexity as well as sample complexity. Subject level treatments are necessary for prediction purpose. It further appears that specialized treatment to subjects different from that to independent variables for ANN modeling of repeated measures data give additional power to ANN in at least some problem areas if not all.

VII. Conclusions and Future directions

The objective of this work is to provide effective ANN model for predicting repeated measures data. There are currently huge literatures where ANN technique has been successfully used to predict cross-sectional or time series data. But there is a shortage of study regarding the use of ANN in analyzing longitudinal or repeated measures data. ANN has been studied here empirically for its effectiveness for learning in general and for predicting capability in particular by trying on several functions of varying complexity (both functional and subject level) under the constraint availability of adequate samples to learn from. Functions have been simulated based on statistical models such as random coefficient model and nonlinear mixed effect model. For ANN, the most popular feedforward architecture with backpropagation learning has been considered. Further, the study has been directed towards introducing the subject level connections in the network.

In general, results obtained are quite satisfactory. It has been empirically established that special attention to catch the subject effects separated from the common effects through network connections has considerable scope of improvement. This study could be an important first step towards determining latent subject effects. The ANN learning also has been found to respond logically towards the function complexity, subject complexity and sample complexity.

The study will be more complete if further experiments are conducted to look into the issues more deeply, particularly to find improvement along the line of ANN2, especially when the sample size is too limited. Attempts should be made to develop confidence bounds over the performance of ANN models using methods like Bootstrap. Going beyond simulations one should look for opportunities to apply these concepts to real life problems. Also in the present only the balanced case, where each subject has same number of samples to learn from, has been attempted. One needs to investigate the unbalanced case which could be a tricky job. Also special attention may be required to increase the accuracy of the ANN model for small samples or highly complex functions for repeated measures data. In the present study noise has been avoided in the simulated data which is a restriction. One more important issue of study would be the performance metric which is especially suitable for repeated measures analysis. The simulated models include random parameters but ANN parameters (weights) come out as fixed. So there is a scope to consider the ANN parameters as random one similar to the mixed effect model. Finally, the ANN methods for repeated data should give hints for longitudinal data analysis.

REFERENCES

[1] N. M. Laird and J. H. Ware, "Random effects models for longitudinal data," *Biometrics*, 1982, vol. 38, pp. 963–974.
[2] J. A. Nelder and R. W. M. Wedderburn, "Generalized linear models," *Journal of the Royal Statistical Society*, 1972, Series A, vol. 135, pp. 370-384.
[3] E. F. Vonesh and V. M. Chinchilli, *Linear and Nonlinear Models for the Analysis of Repeated Measurements*, CRC press, 1996.
[4] M. Davidian and D. M. Giltinan, *Nonlinear Models for Repeated Measurement Data*, Chapman & Hall, 1995.
[5] M. Davidian and D. M. Giltinan, "Non-linear models for repeated measurement data: An overview and update," *Journal of Agricultural, Biological, and Environmental Statistics*, 2003, vol. 8, pp. 387-419.
[6] S. Longhi, P. Nijkamp and E. Maierhofer, "Neural network modeling as a tool for forecasting regional employment patterns," *International Regional Science Review*, 2005b, vol. 28 (3), pp. 330-346.
[7] R. Patuelli, A. Reggiani, P. Nijkamp and U. Blien, "New neural network methods for forecasting regional employment: an analysis of German labour markets," *Spatial Economic Analysis*, 2006, vol. 1, no. 1, pp. 7-30.
[8] R. Tandon, S. Adak and J.A. Kaye, "Neural networks for longitudinal studies in Alzheimer's disease," *Artificial Intelligence in Medicine*, 2006, vol. 36, pp. 245-255.
[9] M. H. Hassoun, *Fundamentals of Artificial Neural Networks*, PHI Learning Private Limited, 2009, pp. 222-223.
[10] J. R. Ashford, and R. R. Sowden, "Multivariate probit analysis," *Biometrics*, 1970, vol. 26, pp. 535–546.
[11] E. L. Korn and A. S. Whittemore, "Methods for analyzing panel studies of acute health effects of air pollution," *Biometrics*, 1979, vol. 35, pp. 795–802.
[12] R. Stiratelli, N. M. Laird and J. H. Ware, "Random effects models for serial observations with binary response," *Biometrics*, 1984, vol. 40, pp. 961–971.
[13] D. A. Pierce and B. R. Sands, "Extra-Bernoulli variation in binary data," *Technical Report 46*, 1975, Department of Statistics, Oregon State University.
[14] M. J. Lindstrom and D. M. Bates, "Nonlinear Mixed Effects Models for Repeated Measures Data," *Biometrics*, 1990, vol. 46, no. 3, pp. 673-687.
[15] G. M. Fitzmaurice, N. M. Laird and A. G. Rotnitzky, "Regression models for discrete longitudinal responses (with discussion)," *Statistical Science*, 1993, vol. 8, pp. 248–309.
[16] M. S. Pepe and G. L. Anderson, "A cautionary note on inference for marginal regression models with longitudinal data and general correlated response data," *Communications in Statistics – Simulation and Computation*, 1994, vol. 23, pp. 939–951.
[17] G. Molenberghs and G. Verbeke, *Models for Discrete Longitudinal Data*, Springer, 2005, pp. 225-238.

ISBN: 978-988-19251-8-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)