Paper to be published in Artificial Intelligence in Medicine --- Elsevier…

**Title:** 10 to 13 words..

**Summary / Abstract 250/300 words:** Objective / Methods and materials / Results/ Conclusion…

**Introduction:** Apnea / Data prepossessing and feature selection with / Artificial Neural network / Evaluation technique / Literature review with gaps / What we are doing (problem definition) / What we are doing objectives and methods / How paper is organized..

(Check good introduction from other papers and see how it is written)….

**Previous Research:** literature review … Shekar to include the part here what I have told you..

Machine Learning over the last decade, has come leaps and bounds, with various advancements in solving numerous problems once considered intuitive, and hence inherently difficult for computers to solve: Face Recognition [9], Sentiment Analysis [10], Speech Recognition [11], Anomaly detection [12], Recommender systems [13] along with a myriad of others. As this field has matured, new opportunities have emerged for various applications in the Medical domain. Neural networks have proved to be highly powerful models in this domain, being used to solve various classification and regression problems, especially medical diagnosis.

A decision-reference system was proposed by Vanisree et al. [5] for diagnosis of Congenital Heart Diseases. The system used a Multilayer feed forward Neural network that was trained on a widely used, benchmarked dataset. Various physiological features of a patient were considered in this work, including signs, symptoms and medical test parameters. This system managed to achieve an accuracy of 90% in providing an accurate measure of the patient’s condition. In another work by Dangare et al. [7], a Heart Disease Prediction System was developed based on neural networks. The model predicts the probability of a patient developing a heart disease using 13 continuous medical parameters like blood pressure and cholesterol, discrete features such as obesity and smoking habits, as features. The study manages to diagnose heart disease with 99.25% accuracy by training a comparatively simple neural network with 1 hidden layer and 570 training examples. This study is a good example of how effectively neural networks even in their nascent form, assisted in diagnosis of medical diseases. Further, Shanthi et al. proposed a Thrombo-embolic stroke disease predictor system that used a feed-forward multilayer perceptron architecture with 3 layers: 20 inputs, 1 hidden layer with 10 nodes and the output layer with 10 nodes. From the 25 physiological parameters considered as features for the prediction of heart stroke, 20 features were selected using a Backward stepwise method based on their correlation importance with the provided class label. The model was then trained using the Backpropagation algorithm and the right model was selected by its performance over the validation set. The trained, optimized artificial neural network model provided an accuracy of 89% overall for heart stroke prediction.

Neural networks, since 2006, under the aegis of Deep Learning, have been applied to various other medical domains such as Biomedical Imaging, Medical transcription, Medical history analysis, among numerous others, with great success. Recently, a Natural Language processing technique was proposed to automate medical transcription and translation [8]. Automated Speech Recognition (ASR) system was developed by Wołk et al. [8]. This work utilized a sophisticated recurrent neural network with 750 hidden layers trained on sentences of Polish-to-English and vice-versa. Each sentence composed of 50 words or fewer and was preprocessed through various stages: tokenization, cleaning, factorization, and cleaning . The recurrent neural network was trained on the resulting vocabulary with 500,000 iterations and the model managed to achieve a working accuracy of up to 64.36% on the METEOR metric, under strict hardware constraints.

(try to put a sentence that links next para – if possible …)

With Neural networks providing encouraging results in several medical domains, we focus its application on the \*Problem statement here\*.

The neonatal period of a child is considered one of the most crucial phases in its physical development and future health. As per the WHO, India has the greatest number of pre-term births Blencowe et al. [21], with over 3.5 million babies are born prematurely and up to 40% of them are babies with low birth weights, highly prone to a multitude of diseases. Estimates by the UN Inter-Agency Group for Child Mortality Estimation, show Neonatal mortality rates in India, too are among the highest, with nearly 28 babies dying in their neonatal period, per 1000 live births [18].

Apnea of prematurity (AOP) is the major concern for caretakers of neonates in intensive care units.  Apnea is defined as a "pause in breathing of longer than 10 to 15 seconds, often associated with bradycardia, cyanosis, or both” [19]. In premature babies, apnea can result in failure of cerebral blood flow, causing ischemia and eventually even leukomalacia Further, as per the National Institutes of Child Health and Human Development (NICHD), crucial conditions that are responsible for the occurrence of AOP are poorly understood and aren’t integrated into the care facilities [20]. There is a growing need for an improved, efficient & automated Neonatal apnea diagnosis. This paper describes how artificial neural networks can be employed at \*Exact Problem statement desired here\* (we will write this ones the results are over…)

Table: Literature for application of neural network for neonatal domain.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Author | Methodology | Application/ Result | Advantages / Disadvantages of the approaches if any | Novelty /Reference |
| Daphné et al.  [15] | Maximum Likelihood and Single layer Gradient Descent MultiLayer Perceptron | Neonatal Risk estimation with upto 92.6% (AUC) mortality estimation | Maximum Likelihood Estimator (what is MLE) approach has been proved inconsistent. [16] (Why 16 here is it same author two papers. Sir, the [16] is a reference to my argument that MLE is inconsistent)  Simple single-hidden layer network is used. | Kolgomorov’s Superposition theorem used to justify no. of neurons in network. |
| Chowdhury et al.  [21] | Multi-Layer Perceptron with a BP algorithm | ANN based neonatal disease diagnosis with accuracy of 75%. | Simple single-hidden layer network.  Neural network without any hyper-parameter variations or network comparisons. | Utilization of Genetic Method for Feature Selection |
| Noguchi et. al.  [22] | Positive sinusoidal Fetal Heart Rate? fed into Multi -layer perceptron trained by Back Propagation algorithm | Neural Network Fetal Heart Rate Diagnosis with referenced accuracy of 86% | Simple single-hidden layer network lacking hyper-parameter variations. | Creation & normalization of FHR variations into categorical features for Neural Network training |
| Sermet et al.[23] | Trained 2 single-layer neural networks with Histogram representations of two sensor signals | Very high accuracy in recognition of apnea (nearly 100% with electrical impedance patterns and 93.6% with strain guage patterns) using neural networks | Electrical impedance inputs are unreliable; Type I error may be high despite using two sensors.  Simple single-hidden layer network used. | Usage of strain gauge & electrical impedance to reduce type II error, training two MLPs. Neural networks outperform Nearest neighbor models |
| Várady et al.  [24] | Four different artificial neural networks are  trained for recognition of respiration signals. Each ANN uses Sigmoid activation function and was trained using Linear Gradient Descent Backpropagation algorithm. The *targeted* MSE was 10^-3 (should it be here or in results. Actually sir, here it is part of the approach) | Creation of robust apnea detection system.  Achieved up to 97% specificity in detection of apnea, 91% for hypo-apnea, and 94% for normal breathing, through respiration signals. | Used four 2-hidden layer neural networks with  Respiration signals of apnea superimpose with other cardiovascular symptoms and is thus not very reliable. | Utilized effective Noise filtration of respiration signals, uses moderate computational power, allows portable applications. |

Summary & Gaps

*Thus, much* (change this a bit..find a better word for this phrase) of the work on automated diagnosis of neonatal apnea has found that neural networks outperform other methods, and other machine learning models [23]. Most study have used simple ANNs with a single hidden layer architecture, backpropagation algorithm using gradient descent for optimization, without optimized networks to prevent overfitting of their models on possibly skewed medical datasets.

\* The ending of the survey will require a little introduction to the proposed work, so that will be filled in the end too? \*

\*Will add a little more here to summarize the survey and its gaps. Ending with an introduction to the idea behind the current proposed work? \*

**Method:**

Data source: In order to perform the research reported in the manuscript, 367 observations of new born babies were collected from NICU, Kasturba Hospital. The mean Gestation age of all the babies was 29.56 diagnosed with…..Ethical approval has been obtained from Institutional Ethics committee of Manipal University. The anonym zed data set obtained consists of more than 100 predictor variables with outcome variables pertaining to number of apnea episodes from the time of admission to discharge. The variable which signifies the number of apnea episodes from Day 4 to Day 7 was chosen to be the final outcome variable and the rest were discarded. We chose this because we want to predict whether apnea persists till the 7th Day based on data collected during the first 3 days. This was made into a binary classification problem by transforming the outcome column where the number of episodes greater than 0 would indicate Presence of apnea and the number of episodes equal to 0 would indicate Absence of apnea. With the help of medical experts advice only 20 variables taken during the first 3 Days are been used to predict the presence or absence of apnea from Day 4 to Day 7. Table 1 and Table 2 show the list of predictor variables and the outcome variable of the final data set respectively. ( I feel we should include this part in data nalytics otherwise it doesn’t have any sence)

Table 1. List of Predictor Variables ***[ TO ADD REAL NAMES]***

|  |  |
| --- | --- |
| **Predictor Variables** | **Description** |
| Gestation Age | Numeric |
| AGASGA | Categorical |
| Surfactant | Dichotomous Categorical |
| Birth Weight | Numeric |
| Steroids | Categorical |
| Head circumference at birth | Numeric |
| Birth cry | Dichotomous Categorical |
| Apgar Score at 1st minute | Numeric |
| Apgar Score at 5th minute | Numeric |
| Need of Resuscitation | Categorical |
| Mode of Resuscitation | Categorical |
| Desaturation1\_3DAYS | Numeric |
| Brady1\_3DAYS | Numeric |
| BD1\_3DAYS | Numeric |
| BD4\_7DAYS | Numeric |
| IMVDAYS | Numeric |
| NIVDAYS | Numeric |
| ECHO | Categorical |
| PDA | Dichotomous Categorical |
| DAYS OF SUPPORT | Numeric |

Table 2. Outcome Variable

|  |  |
| --- | --- |
| **Outcome variable** | **Description** |
| Apnea | Dichotomous Categorical |

Data understanding and preparation: Data analysis and preprocessing concept include here…Genetic / PCA and other preprocessing include here…SMOT also include here method and why we need to use it. Mention about methods and how we have used them in detail….

The data set to be preprocessed consist of 21 variables and 367 observations. All the categorical variables were converted to numeric codes where each code represented a particular category. The missing values of numeric columns were handled by replacing the missing value with the mean of all values corresponding to the similar group particular number of apnea episodes from Day 4 to Day 7.(this should be in general and provide reference). Similarly, the missing values of categorical columns were handled by replacing the missing value with the mode of all values corresponding to the particular number of apnea episodes. For some categorical columns such as Mode of Resuscitation and Steroids, where there were many missing values, a new category or class representing missing values was created rather than replacing the missing value with the mode or dropping the rows [ ]. This was done because the reason so many values were missing could be semantic rather than bad record keeping and they could represent some information crucial for predictive analysis. (put it in better sentence no meaning). The observations consisting of higher number of missing values were removed. The final data set comprised of 364 observations. Representation of categorical variables as integers is not logically correct (WHY? Reference []). For example, the column Steroids is represented by 4 numeric codes shown in Table 3.

There were many missing values in the data set but after summarizing and analyzing the entire data set with the help of scatterplots and histograms, no outliers were found. (Doesn’t have any meaning please tell how?)

Table 3. Representation of “Steroids”

|  |  |
| --- | --- |
| **Numeric Code** | **Description** |
| 1 | Not Given |
| 2 | Partially Given |
| 3 | Completely Given |
| 4 | Not known |

In Table 3, the numeric codes for the categories are not logically correct as for machine learning models “Completely Given” having the code 3 would be 3 times that of “Not Given” having the code 1 and hence be logically greater in value than it. To have a logically correct representation of categorical variables, all of them were dummy coded with 1’s and 0’s as seen in Table 4.

Table 4. Dummy coding of “Steroids”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Numeric Code** | **Description** | **Dummy Code** | | |
| **Not Given** | **Partially Given** | **Completely Given** |
| 1 | Not Given | 1 | 0 | 0 |
| 2 | Partially Given | 0 | 1 | 0 |
| 3 | Completely Given | 0 | 0 | 1 |
| 4 | Not known | 0 | 0 | 0 |

In Table 4, the numeric codes for Steroids have been converted to Dummy codes which is a more logical representation. For any column with k categories, the codes can be represented with k-1 categories. In table 4, “Not known” can be represented as 000 rather than 0001 with an extra column which is more redundant. Having k-1 categories not only reduces redundancy but also has a lesser dimensionality. Similar dummy coding was done for all categorical variables and the total number of columns increased from 21 to 32. The final data set to be used for analysis had 364 observations and 32 columns. Out of the 32 columns, 31 were predictor variables and 1 was a binary outcome variable. Since the entire data set consists of only numbers, it could be successfully normalized to 0-1 scale. Normalization of this data set was really crucial because it had to be brought to the same scale for optimization when applied to the machine learning algorithms []. (Simple sentence why normalization is done in general0. Min-Max Normalization was used to bring the entire data set to the 0-1 scale.

To assess (or to check) the medical importance of the variables chosen for prediction of apnea, and to shortlist the variables out of the entire list, analysis of data was done for both numeric and categorical variables (Confusion is there first we are doing it through medical expert in first para the are we going to check them with statistically?). Some (which one) visualizations were used to spot trends in the data set. No outliers and redundant data were found in the list and the analysis was done after filling in the missing values as done in the preprocessing step. To analyze the medical importance of each of the variables used, bar charts were used to represent categorical variables and scatterplots were used to represent numeric variables.

The relationship between each of the numeric variables was checked with the outcome variable. The scatterplots with smoothing curves shown from Fig.(a) to Fig. (l) represent this relationship of the variables chosen in the final data set. The smoothing curves of the following figure show that the variables used for analysis are distinctive in nature. They are plotted against the probability of the occurrence of apnea. The numeric predictors are both indicative of presence as well as absence of apnea. Variables like Apgar score at 1st Minute and 5th Minute are more densely populated in the region that has low probability of presence of apnea or a high probability of absence of apnea. Variables like Desaturation from Day 1 to Day 3, Days Of Support and NIV Days seem to be good predictors of presence of apnea. When the Desaturation value is between 5 and 15, presence of apnea has very high probability, making it a good predictor of apnea. Similarly, when the Days Of Support value is between 40 and 60, there is a good chance of presence of apnea making it a good predictor too.

|  |  |  |
| --- | --- | --- |
| (a) | (b) | (c) |
| (d) | (e) | (f) |
| (g) | (h) | (i) |
| (j) | (k) | (l) |

The analysis of numeric data as shown previously was done using scatterplots and variables like Days of Support, Desaturation from Day 1 to Day 3, NIV Days were considered as good predictors of apnea. The analysis of categorical variables in the data set was done using filled bar charts where each category was analyzed to visualize the ratio of presence and absence of apnea as shown from Fig.(a) to Fig(h). From these visualizations, these categorical variables seem to be good differentiators of presence and absence of apnea. Need of Resuscitation, Mode of Resuscitation, Steroids and Echo seem to be good predictors of presence of apnea.

(How we are using this concept ie we got some thing but now we should relate this is actual research work)

(We can make this together with medical expert to select variables from 100 what we have said na before) or else we can say to verify the selected variables usefull ness in research)

(Put this diagrams in a proper format with proper tag line explaining what it is)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |

(Put this diagrams in a proper format with proper tag line explaining what it is)

Feature selection deals with the selection of features that are important and relevant to effectively predict classes is very crucial before applying any machine learning algorithm []. It is very important to note that by discarding some of the features in the data set, no valuable information should be lost. The goal of feature selection is to reduce the over-fitting in the data set or reduce the variance with reduction of training time of model. Add references og genetic algorithm [] []….Genetic Algorithm which is a feature selection algorithm was applied to the data set. This algorithm is mimetic to the way darwin’s evolution theory works, in which a pair of chromosomes share some genetic information and then separate. This process in Genetic Algorithm is called Crossover as genetic information crosses over from a chromosome to another. Evolution uses the concept of survival of the fittest which is also used by this algorithm using a fitness value. In the algorithm, a population of individuals is created represented by chromosomes. These chromosomes are represented using binary digits similar to the way our DNA is encoded but by some other method. Genetic Algorithm looks for good and robust solutions that avoid local minimas and search for the global minimum. It searches for solutions with higher fitness values and discards the the ones with low fitness values. First, a population size is defined and a population is created with individuals. Next, a fitness value for every individual is calculated and evaluated. Next, a new population is created of the selected individuals having high fitness values using Crossover between individuals and Mutation of bits. The old population is then discarded and this whole iteration is repeated multiple times. One iteration of this process is called a Generation. The first generation is applied on a randomly selected population and later improved based on the fitness function. Crossover specified earlier is basically the process of selecting a pair of parents and crossing over each other’s genes or bits in the case of this algorithm. The child that is created, shares the bits of it’s parents. This is majorly done by specifying a cross over point in the parents where the bits are swapped beyond that point. Mutation that was specified earlier is used for the maintenance of genetic diversity among the individuals. It involves inverting a random bit of an individual. Both Crossover and Mutation have a corresponding probability of them being applied. Mutation probability should usually be kept low in order to not convert the algorithm into a simple random search. The most fit features or the most important features are chosen to be applied to a classification problem using Genetic Algorithms. It is shown in Fig.

(I feel rather than writing so much detail write in short and with a proper own block diagram / flow chart of working )

start

Randomly initialize population

Fitness of

population

All

Generations over?

Perform mutation and crossover

Select parents from population

Fitness of population

Best Individual

End

Fig. Flowchart of Genetic Algorithm

In this paper, Genetic Algorithm was implemented with 10-fold cross validation. The algorithm returns a subset of data with the selected features. Out of 31 predictors or features,25 features were included in the subset with 290 training examples. The maximum generations of the algorithm was 20 and population per generation was 10. Low values were used, as this algorithm is computationally very expensive. The mutation probability was set to 0.1, which is low, as described earlier and crossover probability was set to 0.8. The Top 5 selected features out of 31 features are, BD4\_7DAYS, Desaturation1\_3Days, RESUSNEED\_yes, ECHO\_asd and NIVDAYS.

For SMOTE as explained in the next section, 29 features were included for training and the 5 selected features were, APG1MIN, Gestation, RESUSNEED\_yes, STEROIDS\_partial and APG5MIN.

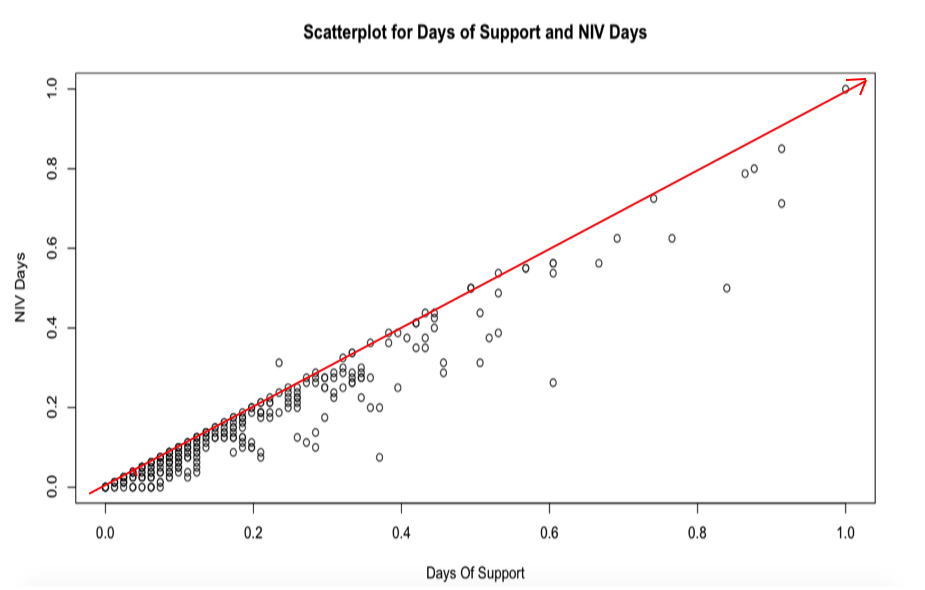
(Don’t include this here - add this after SMOTE section) …

Synthetic Minority Over-Sampling Technique is used for what…. The outcome classes of the training data set fed to the model for training were imbalanced. There were 204 classes that indicated absence of apnea and only 86 classes that indicated its presence. This could be a problem when training the data set as the goal of the classifier is to detect the presence of apnea, and the class required to do so was a minority class. A way to improve model performance is to increase the number of minority class observations. This can be done synthetically by creating new examples corresponding to the minority class. By increasing the number of minority class examples so that it can be in the same ratio as the majority class examples, the model will be able to distinguish the classes better as SMOTE makes the decision boundary more general. NV Chawla et al.[] proposed a Synthetic Minority Over-Sampling Technique that created synthetic examples rather than over-sampling with replacement [2]. Extra training data were created by performing some operations on the real data. All of this was operated in the feature space rather than the data space (means what... ) Each of the minority class observation is over-sampled by the introduction of synthesized observations along the line joining any or all of the K nearest neighbors of the nearest minority class. K nearest neighbors less than or equal to 5 of the minority class are randomly chosen depending upon the over-sampling required. If the requirement is to increase the minority class observations K times, then K nearest neighbors will be chosen for each minority class observation and one sample will be created in each direction. Algorithmically, SMOTE creates synthetic observations by subtracting the nearest neighbor from the feature vector that is taken under consideration, multiplying this result by any random value from 0 to 1, and then adding this result to the feature vector under consideration. SMOTE also allows to under sample the majority class observations by randomly removing them. In this paper, the amount of over-sampling done of the minority class observations was 100% or the amount was doubled.(Not porper) Initially, there were 86 observations, and another 86 were synthetically created. 237% of 86 observations of the majority class were used, i.e. 203 observations out of 204 were retained in the training set and only 1 majority class observation was under-sampled to avoid loss of meaningful information. Finally, the training set comprised of a total of 375 observations out of which 203 were majority class observations and 172 were minority class observations. The training set was relatively balanced in the ratio of 55:45.

Here you can put that part…see where you can fit this…

For SMOTE as explained in the next section, 29 features were included for training and the 5 selected features were, APG1MIN, Gestation, RESUSNEED\_yes, STEROIDS\_partial and APG5MIN.

Feature Extraction process is similar to Feature selection and branches out from feature engineering. Feature Extraction is different from Feature selection because a new set of features is created from the original data set which is equal to or less than the size of the original data set, whereas a subset of the original data set is selected in Feature Selection. Feature Extraction leads to reduction in the dimensionality of data where informative and non-redundant features are used for analysis that are also interpretable by humans. Feature Extraction also reduces the over fitting in the data set caused by the high dimensionality of the data just like Feature Selection. The Feature Extraction technique used in this paper is called Principal Component Analysis. Principal Component Analysis is an algorithm to capture the maximum amount of variance in the data. It does this by converting a data set with highly correlated features to linearly correlated variables. These linearly correlated variables are called the principal components. [] [] (provide atleast one or two reference) This process of forming linearly correlated variables is called orthogonal transformation. Mathematically, PCA finds a surface, which is a vector, to project the data orthogonally. PCA then tends to minimize the sum of squares of distances from the projected surface to the data points. If, a N-Dimensional data set exists which needs to be reduced to a K-Dimensional data set by PCA, then K surfaces or vectors are computed for the data to be projected. PCA will then minimize the sum of squares of distances for each of the K vectors. PCA is implemented by standardizing the data set using mean normalization and then creating a covariance matrix. From this covariance matrix, Eigenvectors are computed which are the vectors to which the data is projected. Eigenvectors are non-zero vectors that do not change direction after applying linear transformation to them. Corresponding to Eigenvectors are scalar quantities called Eigenvalues that determine the Principal Components. The Eigenvector with the highest Eigenvalue is the first Principal Component. When a N-Dimensional data set is reduced to a k-Dimensional data set, K- Eigenvectors with the highest Eigenvalues are selected. The value of K is determined by the amount of variance to be captured in the data set which is generally defined by a particular threshold. As an example, Fig. 1 shows the scatterplot of two highly correlated variables in our data set, Days of Support plotted on the X-axis and NIV Days plotted on the Y-axis. The red line is the vector or the Principal Component to which the data points will be projected, and PCA will minimize the sum of squares of distances from the data points to this vector. The highly correlated variables earlier in 2 dimensions can be represented only using 1 dimension which is the red line. In this paper, PCA was applied as a Feature Extraction method before feeding the training data to the **Neural Network/Model Averaged Neural Network** model. For each principal component, the cumulative percentage of variance was computed. A total of 23 principal components out of 31 initial variables were retained to capture 99% of the variance in the predictor variables. A Neural Network model using these 23 principal components was created and trained. While predicting the observations on the test set, the test data was transformed similarly using the information from the Principal Component Analysis on the training data.



Prediction models: Neural network and all its variant explain here which we are using…..Ensemble approach / Holdout method / K fold Cross validation / Evaluating model performance measure /

**Artificial Neural Networks: [please provide at least 3 to 4 references]** Artificial Neural Networks are models that are inspired by the way the central nervous system works in the human body. ANNs are dependent on the inputs they receive and are a network of interconnected neurons that process information and calculate an output. These interconnections between the neurons have weights associated to them that can be tuned and are responsible for the learning process of the Neural Network model. Computationally, Neural Networks are large distributed parallel processors comprising of simple processing units that have a proclivity of storing experiential information. Just like the human brain, knowledge or information is gathered by Neural Networks by the environment through a learning phase. The interconnections and weights are used to store this gained knowledge by the Neural Network.

Activation functions and activation values are the processing units of a Neural Network and they consist of a Weight space which is the set of all the weights. Modification or adjustment of these weights is called the learning process and finally the updation of these weights describes a Learning Algorithm. According to McCulloch-Pitts model of a neuron, the activation value is given by the weighted sum of the M inputs to the neuron and a bias term .The activation value is then fed to an activation function which gives the output of the neuron. The activation value is given by the formula, ,where is the weight associated with each node and is the input of every node. The value is fed to the activation function . The output of the neuron is then given by the output of this activation function. The most commonly used activation function is called the sigmoid or logistic function. This function is an s-shaped function that maintains a balance between linear and nonlinear behavior. It is given by the formula, and is graphically represented in Fig. 4 where is given by the activation value. The reason it is widely used is because it is computationally very easy to perform as it satisfies a property between its derivative and itself which is widely used in learning algorithms. This property is, . Many input and output neurons can be used in ANNs and are paralleled using input and output layers. Apart from input and output layers, there exist hidden layers between them that are used for expensive computations. Hidden layers consist of hidden neurons that also form interconnections with the input and output neurons respectively. Feed forward networks are those networks that consists of input and output layers and may or may not contain hidden layers. Their interconnections do not form a cycle and the information moves only in the forward direction. It is shown in Fig. 5.

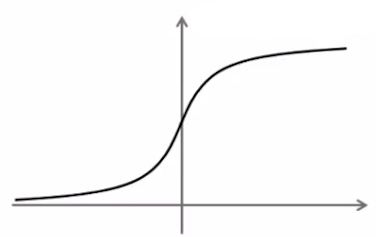
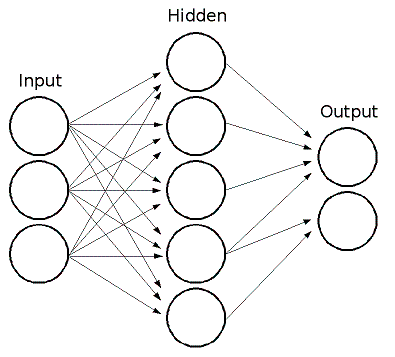
 

Fig. 4 Sigmoid Function Fig. 5 Feedforward Neural Network

Fig. 5 consists of a feedforward neural network with one hidden layer comprising of 6 neurons or 6 units. Such a network with 1 or more than 1 hidden layer is called a Multilayer Perceptron. In a Multilayer Perceptron, the hidden units have interconnections with both the input and output units. The hidden layer consists of a bias unit whose input is always equal to 1. ANNs can be used to solve classification problems where the number of output nodes is equal to the number of classes to classify. The number of input parameters required to classify will be equal to the number of input nodes in the input layer. The algorithm that is used for this computation is called the backpropagation algorithm which uses a Multi Layer Perceptron as the Neural Network model. Backpropagation algorithm minimizes the cost function using the gradient descent algorithm just like other machine learning algorithm like linear regression and logistic regression. The first step to implement the backpropagation algorithm is to initialize the weights and normalize the entire data set using min-max normalization or z-score normalization. After dividing the entire data set into training and test set, present the network with the training set. The next step is to perform forward computation on the training set. Let the input vector be and the desired output vector be . The input vector is applied to the input layer of the Multilayer Perceptron. The information is forwarded layer by layer and is calculated for neuron in layer . It is given by the equation,

Where, is the output function of neuron at previous layer for iteration and is the synaptic weight of neuron in layer that is fed from previous neuron in layer For value of output function is and the value of synaptic weight is where is the bias unit as described previously. The sigmoid function is applied to the output function The output function of the neuron after applying the sigmoid activation function at neuron j and layer is given by if the node is the input layer itself, then the value of output function is equal to the input value. If the neuron is in the last layer or the output layer, then the value of output function is . As the information is propagated in the forward direction, error function is calculated at the last layer which is given by, The next step in this algorithm is backward computation which is done by computing the local gradients . If the neuron is in the output layer ,

If the neuron is in the hidden layer ,

.

The weights in the interconnections are adjusted by the formula,

Here, is the learning rate constant and is the momentum constant.

The next step is to iterate the forward and backward computation for all the training set examples until the stopping criteria is met. The stopping criteria is met when the cost function in the gradient descent algorithm reaches the global minimum or a sufficiently small threshold. The demerits of this algorithm are, to reach the small threshold of the cost function, the algorithm may do a lot of iterations making it very computationally expensive and that this method is a black box method which cannot be visualized by the human eye.

In this paper, a feedforward neural network was used and the backpropagation algorithm was applied for binary classification. There were 2 nodes in the output layer each signifying the binary classes of prediction. Only one hidden layer was used for implementing the backpropagation algorithm but it comprised of many hidden nodes. Logistic or Sigmoid ? function was used as the activation function for the feedforward neural network and weights were randomly assigned. After applying 5-fold cross validation to the training sets, many different models were created with varied number of hidden nodes and weight decay values. The model with best Accuracy/Kappa or AUC value was chosen for the test set and, the number of hidden layers and weight decay values corresponding to that model were reported. Increasing the number of hidden layers may increase model performance. The weight decay parameter was the regularization parameter which reduces over-fitting and is used for optimization. Regularization modifies the cost function and reduces over fitting when there are too many features in the data set.

An Ensemble approach called bagging or bootstrap aggregation was used in this paper to improve the model performance. Bagging is used in Machine Learning to reduce the over fitting. This usually happens when the model performs well on the training data but not so well on the testing data as observed in the result obtained from the Neural Network. Many sample sets are created with replacement from the original data set where the length of each of the sample sets is less than or equal to the length of the original data set. Sampling with replacement means that the one observation may be present in different samples. Each of the models in bagging is built independently and is fitted using one of the sample sets. The models’ predictions are combined using the process of voting for classification problems and averaging the output for regression problems. [1] In this paper, a Model Averaged Neural Network was used to implement the concept of bagging and improve the performance of the prior base model. The same Neural Network model was fit using different seeds of random numbers and all of the models that were created using Bagging were used, and their outputs were averaged, and then translated to the outcome classes. (more references from the papers are required)

The holdout method is a technique of dividing the entire data set into training and testing set. The training set is used to generate or train a model which is then evaluated by the test set. The training and test sets should be mutually exclusive and should be randomly sampled. The test set in no way should be allowed to influence the model generated by the training set. [1]. Model performance can be evaluated using the predictions of the test set and the labels of the test set for classification problems. In this paper, 80% of the data was used for training the classifier and 20% of the data was used for testing purposes. Out of the 364 observations in hand, 290 belonged to the training set and 64 were in the test set. The 80% of the training data also went through K-Fold Cross Validation as explained in the next section. (more references from the papers are required)

K-Fold Cross Validation is a technique that improves the holdout method. It reduces the variance of the resulting estimate. The training data set is divided into K random subsets that hold equal amount of data. The holdout method is repeated K times for each of these K subsets. Every time, one of the K subsets is used as a test set and the remaining K-1 subsets are used to train the model. Measures of accuracy like, ROC AUC, Accuracy and Kappa are used to report the validity of Cross Validation. After the process of training and validating the data is done K times, the average performance of all k folds is reported. K-Fold Cross Validation approach is a little computationally expensive if the value of K is large. In this paper, 5- Fold Cross Validation was applied on the training data. 5 different folds were created at random by the process of sampling from the training data set that consisted of 20% of the training data each. Every observation in the training set got to be in the test set K times or 5 times and got to be in the training set K-1 times or 4 times. The performance was averaged for all K Folds and reported in the form of Accuracy and Kappa value for all models other than the one that implemented SMOTE. ROC AUC value was used in case of the model whose training data was oversampled by SMOTE. The 5-Fold CV technique was implemented for different parameters of the Neural Network classifier. For example, different numbers of hidden units and different regularization constants were validated, and the model that gave the best result was forwarded to the test set. (This part you should add after the **Evaluating Model Performance)**

**Evaluating Model Performance**

The models had been evaluated both on the training and test sets. To choose the best model after applying 5-Fold Cross Validation on the training set, Accuracy and Kappa had been reported, and the model with the best values was chosen to be validated on the test set. Accuracy and the Kappa value are defined as follows:

**Accuracy**: The number of correctly classified observations upon the sum of correctly classified observations and the incorrectly classified observations.

**Kappa Value**: This value is similar to Accuracy, but it modifies Accuracy by accounting for a possibility that the correct prediction on the test set could be by chance alone. The value of Kappa is between 0 and 1, where a value between 0.8 and 1 indicates very good agreement between the actual outcome and the predicted outcome, a value between 0.6 and 0.8 indicates good agreement, a value between 0.4 and 0.6 indicates fair agreement, a value between 0.2 and 0.4 indicates fair agreement, and any value less than 0.2 refers to a poor agreement between the actual outcome and the predicted outcome. The formula to calculate the value of kappa is:

To fully understand the formula of Kappa, It is very important to understand the Confusion Matrix as shown in Fig. 2

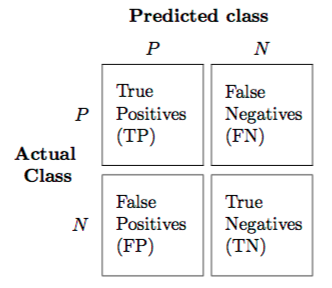


Fig. 2 Confusion Matrix

The confusion matrix is plotted between the actual class and the predicted class. True positives are those observations that are correctly classified as positive classes by the model. True negatives are those observations that correctly classified as negative classes by the model. False Negatives are those observations that are actually positive classes but are incorrectly classified as negative classes by the model. Similarly, False Positives are those observations that are actually negative classes but incorrectly classified as positive classes by the model. Coming back to Kappa value, is defined as the sum of the proportions of True Positives and True Negatives which is very similar to the concept of Accuracy. Kappa value modifies the accuracy of the model relative to the expected agreement. is the probability that chance alone would get the actual and predicted outcomes to match. This probability is based on the probability theorem, assuming two events to be independent, the probability that both of them occur is equal to the product of the two events. is defined as the sum of the probabilities that either the actual outcome and predicted outcome agree that the class is Positive or they agree that the class is Negative. It is given by the formula referring to Fig. 2 as, where,

and,

To evaluate model performance on the test set, many performance metrics were used such as, Receiver Operating Characteristics(ROC) curve Area Under Curve(AUC) which requires sensitivity and specificity of a model, Kappa value as described before, Precision, Recall and F-Measure. The preceding terms are described as follows:

**Sensitivity**: From Fig. 2, sensitivity of a model which is also called the True Positive Rate, is defined as the proportion of the positive class observations that were classified correctly by the model. It is given by the formulae,

**Specificity**: specificity of a model which is also called the True Negative Rate, is defined as the proportion of negative class observations that were classified correctly by the model. It is given by the formulae,

**ROC curves:** These curves are used to examine the tradeoff between the True Positives, while they avoid the False Positives. The proportion of True Positives or the True Positive Rate is plotted on the Y-axis and the proportion of the False Positives or the False Positive Rate is plotted on the X-axis. These values are equivalent to the sensitivity and (1-specificity) respectively. The curve is traced vertically for correct predictions and are traced horizontally for incorrect predictions. In the ROC plot, if a straight line passes coincides with the line, then the predictive model has no predictive value as it detects the true positives and the false positives at the exactly same rate and hence, cannot differentiate between the two. However, if a straight line coincides with the Y-axis of the ROC plot, then it means that the predictive model is perfect as it classifies all the True Positives correctly with a 0% False Positive Rate. These are the two extreme cases of ROC plots and classifiers with any predictive value lie between these two extremes. The classification can also be measures by a value called the Area Under the Curve or AUC. For a classifier with no predictive value whose ROC plot is a straight line coinciding with the line, and for a perfect classifier, the . An example of an ROC curve is shown in Fig.3.

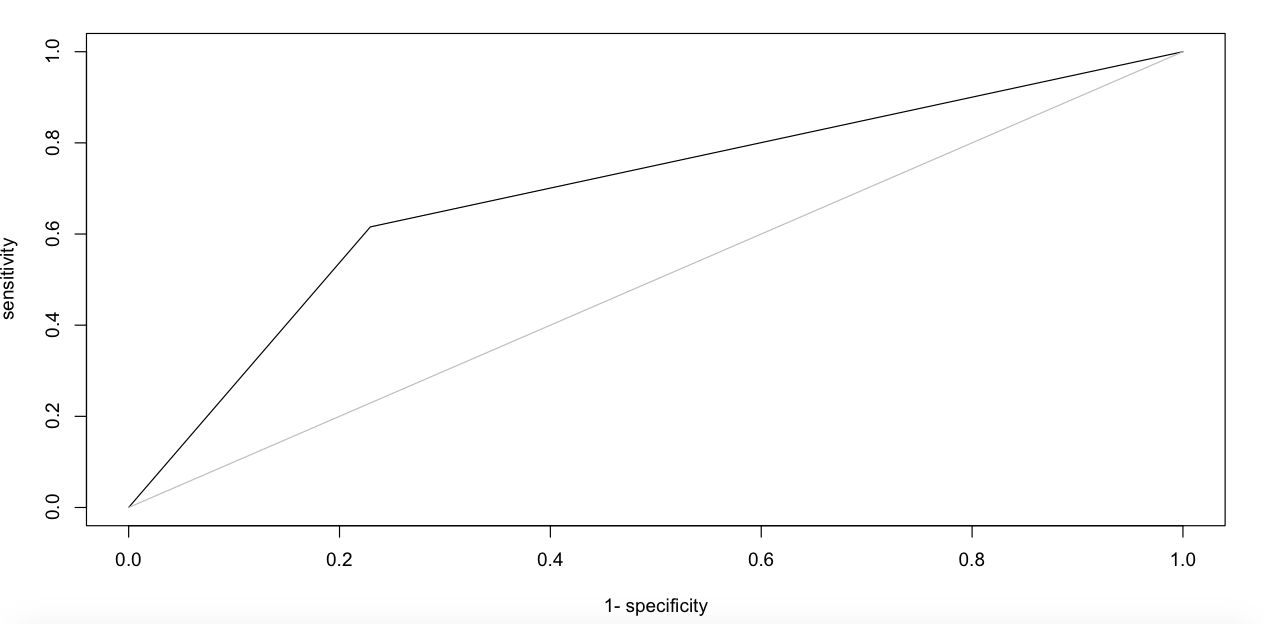


Fig. 3 ROC curve with AUC=0.70

Two other performance measures that are similar to sensitivity and specificity are called Precision and Recall. They closely relate to the compromises that are made by the model during classification. They show the meaningfulness of the outcome of the classifier and dilute the meaningless noise. These measures are defined as follows:

**Precision:** It takes into account the proportion of positive class observations that are actually positive. It is defined by the number of True Positives divided by the sum of True Positives and False Positives. Formula being,

**Recall:** It is a measure which tells us how complete the results are. It is defined as the number of True Positives divided by the the total number of Positives. The formula being,

**F-measure:** This is a measure which takes Precision and Recall both into consideration. It is the combination of Precision and Recall by expressing the arithmetic mean and is given by the formula, . This measure provides a simple way to model both Precision and Recall using a single value. It makes testing the the efficacy of the classifier really important.

There is another measure which I had send you paper...please include that also….

Which measure?

The measures of the method performance are: sensitivity [SN = TP/(TP + FN)]; specificity [SP = TN/(TN + FP)]; accuracy [AC = (TP + TN)/(TP + FN + TN + FP)]; and precision [PR = TP/(TP + FP)]. In addition, we employed a Matthews correlation coefficient (MCC) MCC= TP∗TN−FP∗FN (TP+FN)(TP+FP)(TN+FP)(TN+FN) (4) Here, TP, TN, FP and FN refer to true positives, true negatives, false positives and false negatives, respectively

Mentioned in paper:

**Results: Put the result in tables with all evaluating performance measure we have defined above…Tables graphs…atleast one or two ROC curves..**

Table. 5 Results Table on the Test set

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Approach** | **Feature Extraction/Feature Selection Technique** | **AUC** | **Precision** | **Recall** | **F-Measure** | **MCC** |
| Neural Network | Na | 0.62 | 0.52 | 0.50 | 0.51 | 0.25 |
| Neural Network with Bagging | Principal Component Analysis | 0.66 | 0.63 | 0.46 | 0.53 | 0.34 |
| Neural Network with Bagging and SMOTE | Principal Component Analysis | 0.69 | 0.54 | 0.69 | 0.61 | 0.36 |
| Neural Network with Bagging | Genetic Algorithm | 0.65 | 0.65 | 0.42 | 0.51 | 0.34 |
| Neural Network with Bagging and SMOTE | Genetic Algorithm | 0.63 | 0.52 | 0.54 | 0.53 | 0.26 |

Table 6. Final models chosen after training

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Approach** | **Feature Extraction/Feature Selection Technique** | **Accuracy** | **Kappa** | **AUC** | **Hidden Nodes** | **Decay Parameter** |
| Neural Network | Na | 0.74 | 0.35 | Na | 10 | 0.1 |
| Neural Network with Bagging | Principal Component Analysis | 0.77 | 0.38 | Na | 6 | 0.5 |
| Neural Network with Bagging and SMOTE | Principal Component Analysis | Na | Na | 0.91 | 12 | 0.001 |
| Neural Network with Bagging | Genetic Algorithm | 0.77 | 0.39 | Na | 4 | 0.1 |
| Neural Network with Bagging and SMOTE | Genetic Algorithm | Na | Na | 0.89 | 6 | 0.001 |

After preprocessing and analyzing the data set, the holdout method was applied and 80% of the data or 290 observations were used as the training data set and the remaining 20% or 74 observations were used as the test set. Many combinations of neural network models with different numbers of hidden units and decay parameters were applied on the training set with 5-fold cross validation and the model with the best values of accuracy and kappa was chosen for the evaluation of model accuracy. For the very first approach, a basic neural network without any feature selection or optimization technique was trained with 10 hidden nodes in the hidden layer and 0.1 as the regularization or decay parameter. The training accuracy and kappa value is shown in Table 6. This basic model reported an AUC of 0.62 when tested on the test set as shown in Table 5. For the next two models, PCA was applied as the Feature Extraction technique which captured 99% of the variance in the data set. A Model Averaged Neural Network or a bagged neural network with 6 hidden units and 0.5 as the decay parameter was trained, and it reported an increased AUC of 0.66. SMOTE was then applied to this training data whose features had been extracted by PCA and was bagged. The training data increased to 375 observations as the minority class observations were doubled. ROC curve AUC was used to evaluate the training model performance and a network with 12 hidden nodes and decay parameter equal to 0.001 was chosen for training and reported an AUC of 0.69 on the test set which increased slightly from the earlier model. For the next two models, Genetic Algorithm was applied as the Feature Selection technique on the training data. In the Neural network model with Bagging, 25 predictor variables out of the 31 were chosen to build the training model with 4 hidden units and the decay parameter equal to 0.1. This model, when tested, gave an AUC equal to 0.65 that was slightly lesser than the preceding model. SMOTE was applied in the next approach and the training data again increased as earlier. A bagged neural network was created where only genetic algorithm selected 29 predictor variables out the 31 with 6 hidden nodes and decay parameter equal to 0.001. However, this model gave a relatively low AUC of 0.63.

**If possible put all ROC curves in one graph with AUC measure**

**Discussion:** Discuss the results and algorithm, slowly come to conclusion why and which algorithm is better and its drawbacks if any Check out in papers what discussion contains and how it is different from results and conclusion….

**Conclusion with future work:** We have to conclude the paper with something novel interesting and with future work…

References:

[1] Reducing the Dimensionality of Data with Neural Networks G. E. Hinton\* and R. R. Salakhutdinov Science 2006

[2] Comparison of Artificial Neural Networks and Human Brains on Solving Number Series Report of the practical part of the lecture Cognitive Modelling (WS11/12) Tina Kammerer Ioulia Kalpakoula et. al.

[3] Artificial neural networks for diagnosis and survival prediction in colon cancer Farid E Ahmed BioMed 2005

[4] BACKPROPOGATION NEURAL NETWORK FOR PREDICTION OF HEART DISEASE NABEEL AL-MILLIISSN: **ISSN: 1992-8645 JTAIT 2013**

[5] Vanisree K, Jyothi Singaraju, Decision Support System for Congenital Heart Disease Diagnosis based on Signs and Symptoms using Neural Networks, International Journal of Computer Applications (0975 8887) Volume 19 No.6, April 2011.

[6] E. Pietka and J. Kawa (Eds.): Information Tech. in Biomedicine, ASC 47, pp. 299–306, 2008.springerlink.com c*\_* Springer-Verlag Berlin Heidelberg 2008

[7] Miss. Chaitrali S. Dangare**,** Dr. Mrs. Sulabha S. Apte International Journal of Computer Engineering and Technology (IJCET), ISSN 0976 –6367(Print), ISSN 0976 – 6375, October-December (2012), IAEME

[8] Neural-based machine translation for medical text domain. Based on European Medicines Agency leaflet texts. [**arXiv:1509.08644**](http://arxiv.org/abs/1509.08644)

[9] Parkhi, Omkar M., Andrea Vedaldi, and Andrew Zisserman. "Deep face recognition." *British Machine Vision Conference*. Vol. 1. No. 3. 2015.

[10] Pak, Alexander, and Patrick Paroubek. "Twitter as a Corpus for Sentiment Analysis and Opinion Mining." *LREc*. Vol. 10. 2010.

[11] Sak, Haşim, et al. "Fast and accurate recurrent neural network acoustic models for speech recognition." arXiv preprint arXiv:1507.06947 (2015).

[12] Guha, Sudipto, et al. "Robust Random Cut Forest Based Anomaly Detection On Streams." *Proceedings of The 33rd International Conference on Machine Learning*. 2016.

[13] Wang, Hao, Naiyan Wang, and Dit-Yan Yeung. "Collaborative deep learning for recommender systems." *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015.

[14] Shanthi, D., G. Sahoo, and N. Saravanan. "Designing an artificial neural network model for the prediction of thrombo-embolic stroke." *International Journals of Biometric and Bioinformatics (IJBB)* 3.1 (2009): 10-18.

[15] Townsend, Daphne, and Monique Frize. "Complimentary artificial neural network approaches for prediction of events in the neonatal intensive care unit." *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2008.

[16] Basu, D. "An inconsistency of the method of maximum likelihood." *The Annals of Mathematical Statistics* 26.1 (1955): 144-145.

[17] “An Artificial Neural Network Model for Neonatal Disease Diagnosis”, International Journal of Artificial Intelligence and Expert Systems (IJAE)**,** Vol. 2, Issue 3, pp. 96-106, August, 2011.

[18] UN Inter-agency group for Child Mortality estimation <http://data.worldbank.org/indicator/SH.DYN.NMRT>

[19] Martin RJ, Miller MJ, Carlo WA. Pathogenesis of apnea in preterm infants. J Pediatr 1986;109:733-741.

[20] D.J. Nimavat et. al. <http://emedicine.medscape.com/article/974971-overview>

[21] Chowdhury, Dilip Roy, Mridula Chatterjee, and R. K. Samanta. "An artificial neural network model for neonatal disease diagnosis." *International Journal of Artificial Intelligence and Expert Systems (IJAE)* 2.3 (2011): 96-106.

[22] Noguchi, Yasuaki, et al. "Neural network analysis and evaluation of the fetal heart rate." *Algorithms* 2.1 (2009): 19-30.

[23] Anagun, A. SERMET. "A neural network based approach for apnea recognition." *ASME Intell. Eng. Syst. Artific. Neural Networks* 10 (2000): 683-688.

[24] Várady, Péter, et al. "A novel method for the detection of apnea and hypopnea events in respiration signals." *IEEE Transactions on Biomedical Engineering* 49.9 (2002): 936-942.