A Comparative Analysis of K-NN and ANN Techniques in Machine Learning

Igiri Chinwe Peace
Department of Computer Science
University of Port Harcourt
Nigeria

Anyama Oscar Uzoma
Department of Computer Science
University of Port Harcourt
Nigeria

Silas Abbasiama Ita
Department of Computer Science
University of Port Harcourt
Nigeria

Sam Iibi
Department of Computer Science
University of Port Harcourt
Nigeria

Abstract— **Different machine learning algorithms have been applied in various domains and have yielded good results. The application of a preferred technique to a named field is determined by the type of datasets and target goal in question. Although some researches have shown different techniques resulting to the same prediction result. However, in this study, a critical analysis of the application of k- Nearest Neighbour (k-NN) and Artificial Neural Network (ANN) has been carried out. This comparative analysis was done using the same datasets (English Premiership League) on this same platform (Rapid Miner). K-NN classification showed a prediction success of 53.33% while that of ANN was 70%. This proved that ANN is a better technique than k-NN for a polynomial label.**

Keywords—ANN; K-NN; Machine Learning; Prediction

I. Introduction

The use of machine learning techniques in modelling and predictive analysis cannot be overstated. The application of these techniques in various domains has resulted in a better world. Today we can forecast weather, predict games outcome, perform diagnosis and improve various distance e-learning methods using Machine Learning techniques.

K-Nearest Neighbour (K-NN) and Artificial Neural Network (ANN) machine learning techniques have been developed and designed to model the human brain. Both techniques also offers good use especially in the area of sports analytics (sports mining). Hence given adequate and appropriate datasets, one can forecast sports outcome and in turn make money. With the popularity of these techniques over the years, academics have developed several ways to make good use of these techniques from either the solving of classification or regression problem to providing optimization strategies to make good use of. These optimization strategies are:

- Distance of neighbours (K-Nearest Neighbour)
- Momentum (Artificial Neural Network)
- Learning rate (Artificial Neural Network)

The role and adjustments of these strategies provides for good classification and prediction result hence the application in these areas can never be underscored.

1) What is a Machine Learning?

The term “Machine Learning” can be referred to as a scientific and systematic domain that critically explores the creation, study and overall application of a broad-based algorithms that can learn from datasets or model datasets. These algorithms operate by simulating a model from inputs datasets called example sets or test sets, then using the model results to further forecast, make predictions or varying forms of decisions in different application domains.

Machine Learning helps in eliminating the static, fixed and strict approach of a well-structured programming which usually provides for either poor optimization, non-efficient use of memory space and time-based factors.

In the field of Computer Science, the definition and role of machine learning overlaps with fields of mathematics, computational statistics and Artificial Intelligence. These overlap creates for a more thorough and rigorous application of the Machine learning. Machine learning can be divided into three forms of learning.

- Supervised Learning

In this form of learning, the computer is presented with test inputs and their possible anticipated outputs, given by a "trainer", and the underlying goal is to learn a general rule that maps inputs to outputs.

- Unsupervised learning

In unsupervised learning, there are no labels given to the learning algorithm or the trainer, leaving it on its own to find structure in its input in an unstructured manner. Unsupervised learning can be a goal in itself in terms of ascertaining hidden patterns in data or a possible means towards an end, which is in the prediction proper.

- Reinforcement learning

In reinforcement learning, a computer suite is designed to interact with an active environment outside its environment in which it must perform a certain without a trainer explicitly telling it whether it has come very close to its goal or not. Various application of reinforcement learning exists such as in learning how to drive, e-learning, team strategy to study an opponent playing pattern used by the managers in decision support systems.

2) *Neural Network Learning*

Learning is a very important module to every intelligent system. Looking at artificial neural network, learning typically happens during a precise training/classification phase. Once the neural network has been trained, it goes into a phase called the production phase where it produces results independently. In This phase, training can take on diverse forms, using a mixture of learning archetypes, learning guidelines, and learning algorithms. A network which has discrete learning and production phases is referred to as a static network. Networks that are able to continue learning during production phase are known as dynamical systems.

II. RELATED LITERATURE

K- Nearest Neighbor (K-NN) and Artificial Neural Network (ANN) are both machine learning techniques that have being used to implement prediction or forecasting system as the case maybe. However, the choice of technique depends on the aim and objectives of the project in view. According to [1], K-NN can be a classification or regression model depending on the input data. The k-nearest-neighbor classifier is commonly based on the Euclidean distance between a test sample and the specified training samples, [2]. [3] pointed out that one of the difficulties that arises when utilizing this technique is that each of the labeled samples is given equal importance in deciding the class memberships of the pattern to be classified, regardless of their `typicalness'.

The application of MultiClass Classifier, Logitboost, Rotation Forest, BayesNet, and Naive Bayes techniques to Home and Away win as football dataset by [4] yielded 55% prediction accuracy. The highlights in [4] study is that the four different techniques yielded the same result on the same data set. He recommended that higher prediction could be obtained with more relevant dataset. Based on [4] recommendations, [5] developed improved system by applying ANN on eighteen (18) datasets among which are match streak, players' performance index, managers' performance index etc. [5]'s system yielded 70% prediction accuracy when applied to the 10th and 11th week of 2014/2015 season of English premier league match.

Hidden Markov Process Model fitted with Newton Raphson's method was applied to train World Chess Federation rating systems with 2000 chess players by [6]. A prediction of 55.64% was obtained. [6], however was not satisfied with result, hence, suggested that a different technique could improve the result. Although, other researches have proved [6]'s recommendation right. [7], for instance applied K-NN on National Football League data with 80% prediction success. An improved prediction of 90.32% was obtained when ANN techniques was applied to the same dataset in [7] by [8].

A research by [9], showed the application of Bayesian Network Model on psychological and non- psychological factors that affect Barcelona team in the 2008-2009 Spanish League. The success of the application of Bayesian Network model to the thirteen features was 92%. The detailed streamlined research focused on just Barcelona team yielded excellent prediction, proving that Bayesian Network is the appropriate model for the dataset used.

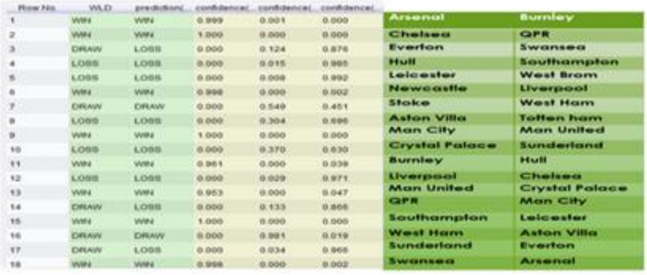
III. EXISTING SYSTEM

The existing system is the research work done by [5]. In their research work, an Improved Prediction System for Football a Match Result was developed using a straightforward application logistic regression and Artificial Neural network modeling in the prediction of English Premier League results of 2014/2015 season as shown in fig.1.

The following feature sets were used:

- number of goals
- attack and defense strength,
- players' performance index
- managers' index
- match streak, etc.

Result shows a prediction accuracy of 85 percent.



Row No	WLD	prediction(	confidence(	confidence(	confidence(		
1	WIN	WIN	0.999	0.001	0.000	Arsenal	Burnley
2	WIN	WIN	1.000	0.000	0.000	Chelsea	QPR
3	DRAW	LOSS	0.000	0.124	0.876	Everton	Swansea
4	LOSS	LOSS	0.000	0.015	0.985	Hull	Southampton
5	LOSS	LOSS	0.000	0.008	0.992	Leicester	West Brom
6	WIN	WIN	0.998	0.000	0.002	Newcastle	Liverpool
7	DRAW	DRAW	0.000	0.549	0.451	Stoke	West Ham
8	LOSS	LOSS	0.000	0.304	0.696	Aston Villa	Totten ham
9	WIN	WIN	1.000	0.000	0.000	Man City	Man United
10	LOSS	LOSS	0.000	0.370	0.630	Crystal Palace	Sunderland
11	WIN	WIN	0.961	0.000	0.039	Burnley	Hull
12	LOSS	LOSS	0.000	0.029	0.971	Liverpool	Chelsea
13	WIN	WIN	0.953	0.000	0.047	Man United	Crystal Palace
14	DRAW	LOSS	0.000	0.133	0.866	QPR	Man City
15	WIN	WIN	1.000	0.000	0.000	Southampton	Leicester
16	DRAW	DRAW	0.000	0.981	0.019	West Ham	Aston Villa
17	DRAW	LOSS	0.000	0.034	0.966	Sunderland	Everton
18	WIN	WIN	0.998	0.000	0.002	Swansea	Arsenal

Fig. 1: Diagram showing results of existing system

- Advantages of existing system

  Good prediction accuracy

  Unique features

  Large datasets

- Disadvantages of existing system

  Limited capabilities

  Cost intensive

IV. COMPARATIVE ANALYSIS OF K-NN AND ANN TECHNIQUES

1) *k-NN technique*

The k-Nearest Neighbor algorithm is based on learning by analogy, that is, by comparing a given test example with training examples that are similar to it. The training examples are described by n attributes. Each example represents a point in an n-dimensional space. In this way, all of the training examples are stored in an n-dimensional pattern space. When given an unknown example, a k-nearest neighbor algorithm searches the pattern space for the k training examples that are closest to the unknown example. These k training examples are the k "nearest neighbors" of the unknown example. "Closeness" is defined in terms of a distance metric, such as the Euclidean distance.

The k-nearest neighbor algorithm is amongst the simplest of all machine learning algorithms: an example is classified by a majority vote of its neighbors, with the example being assigned to the class most common amongst its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the example is simply assigned to the class of its nearest neighbour. The same method can be used for regression, by simply assigning the label value for the example to be the average of the values of its k nearest neighbors. It can be useful to weight the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones.

The neighbors are taken from a set of examples for which the correct classification (or, in the case of regression, the value of the label) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required.

The basic k-Nearest Neighbor algorithm is composed of two steps: Find the k training examples that are closest to the unseen example. Take the most commonly occurring classification for these k examples (or, in the case of regression, take the average of these k label values).

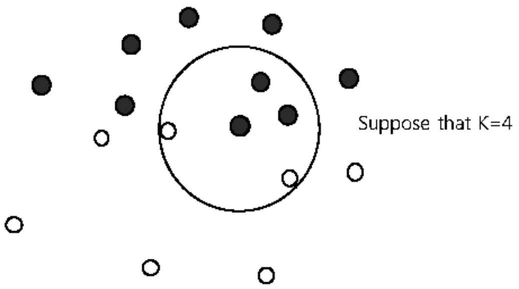


Fig 2: Illustration of the kNN search problem for k = 4

As seen in fig. 2, the white points correspond to the reference points and the black points correspond to the query point. The circle gives the distance between the query point and the closest reference point.

The basic k-Nearest Neighbor algorithm is composed of two steps: 1. the algorithm found the k training sports data that are closest to the unseen dataset. 2. Then took the most commonly occurring classification for these k datasets. The k value found in this research is 5. This was verified by adjusting the k-value between 1 and 15 to obtain the best possible prediction of 53.33%.

a) Input

Sports data as training set

This input was transformed by normalization and fed to the k-NN model process for training to find the k- value.

b) Output

Model

The k-Nearest neighbour was delivered from the output port which was applied on the prediction data sets to generate the label attributes.

c) Parameters

K = 5

Measure type = mix measure

Mix measure = Mixed Euclidean Distance

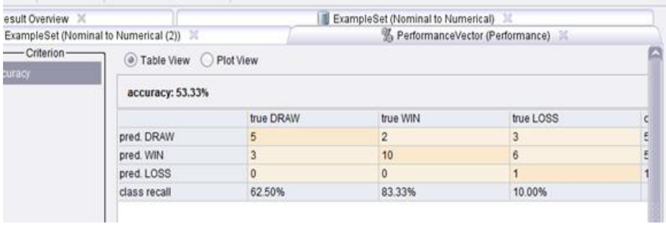
Training set examples = 80

Prediction data sets = 30

Number of attributes = 9

Number of class label = 3

Prediction = 53.33% see fig.3


esult Overview
ExampleSet (Nominal to Numerical)
ExampleSet (Nominal to Numerical (2))
PerformanceVector (Performance)
Criterion
curacy
◉ Table View
○ Plot View

accuracy: 53.33%

	true DRAW	true WIN	true LOSS	c
pred. DRAW	5	2	3	5
pred. WIN	3	10	6	5
pred. LOSS	0	0	1	1
class recall	62.50%	83.33%	10.00%	

Fig. 3: Screenshot of simulation result of k-NN prediction result

2) ANN (Artificial Neural Network)

ANN (Artificial Neural Network) can be referred to as both the natural and artificial alternatives, though classically this term is used to refer to artificial and external systems only. Mathematically, neural nets are regarded as nonlinear objects with each layer representing non-linear combination/variations of non-linear functions from the prior layers. Each neuron in the network is a multiple-input, multiple-output (MIMO) scheme that receives pointers from the inputs, produces a subsequent signal, and communicates that signal to all possible outputs.

ANN has algorithms in different forms to help address the problem domains that Artificial Neural Network (ANN) solves. Some of the algorithms include:

a) Back propagation with single modified neuron

b) Back propagation with linear neuron

c) Matrix approach

d) Gradient Descent

The back propagation algorithm has been the most popular approach for neural networks training/classification due to its flexibility and robustness. This method has been used to solve used to solve various real life problems.

The network is a multiple-input, multiple-output (MIMO) scheme that receives pointers from the inputs, produces a subsequent signal, and communicates that signal to all possible outputs.

Basically, neurons in an Artificial Neural Network (ANN) are arranged into different discrete layers. The first and topmost layer is the one that interacts with the surroundings to receive various combinations of possible input is known as the input layer.

The last and final layer that interacts with the output to present the final processed data is known as the output layer.

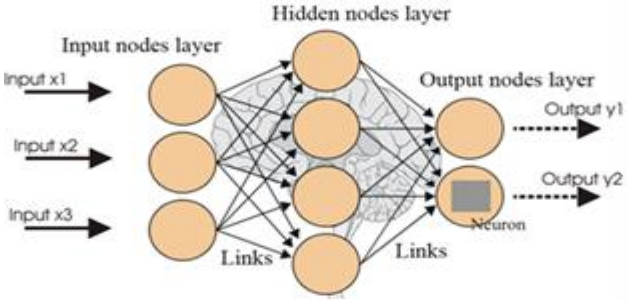


Fig. 4: Diagram showing Artificial Neural Network

While the layers that are between the input and the output layer that do not have any real communication with the environment are known as hidden layers. Hence increasing the complexity of an Artificial Neural Network (ANN), and its computational ability, requires the additions of a lot of more hidden layers and neurons per layer, as seen in fig. 4.

e) Input

Sports data as training set

The input port was fed with transformed sports data in neural net model process.

f) Output

Model

The Neural Net model was delivered from the model output port which was applied on unseen data sets for prediction of the label attribute.

Class 'WIN' (Sigmoid)

Node 1: 1.535

Node 2: -0.250

Node 3: -0.052

Node 4: -0.466

Node 5: -0.767

Node 6: -1.914

Node 7: -0.240

Node 8: 1.115

Node 9: 0.469

Node 10: -0.402

Node 11: -0.207

Node 12: 1.784

Node 13: -0.146

Node 14: -0.152

Node 15: -0.366

Node 16: 2.435

Node 17: -1.110

Node 18: -0.615

Node 19: -0.204

Node 20: -0.970

Node 21: -0.516

Node 22: -0.225

Node 23: 0.870

Node 24: -0.241

Node 25: -0.999

Node 26: -0.492

Node 27: 1.680

Node 28: 0.036

Node 29: -0.644

Node 30: -0.436

Node 31: -0.483

Node 32: 0.247

Node 33: -0.643

Threshold: -2.179

Class 'DRAW' (Sigmoid)

Node 1: -2.108

Node 2: -0.140

Node 3: -1.121

Node 4: -2.322

Node 5: 2.170

Node 6: 2.250

Node 7: -0.101

Node 8: -1.357

Node 9: 2.193

Node 10: -0.297

Node 11: -0.145

Node 12: -2.628

Node 13: -0.196

Node 14: -1.305

Node 15: 0.307

Node 16: -2.856

Node 17: -0.062

Node 18: -2.079

Node 19: 0.592

Node 20: -1.482

Node 21: -0.047

Node 22: -1.598

Node 23: -1.320

Node 24: -0.049

Node 25: -0.934

Node 26: 0.077

Node 27: -2.659

Node 28: -0.310

Node 29: 0.889

Node 30: 0.304

Node 31: 0.845

Node 32: 0.634

Node 33: -1.803

Threshold: 1.052

Class 'LOSS' (Sigmoid)

Node 1: -1.505

Node 2: 0.130

Node 3: 0.759

Node 4: 1.830

Node 5: -2.479

Node 6: -0.821

Node 7: -0.063

Node 8: -1.572

Node 9: -3.037

Node 10: 0.469

Node 11: -0.038

Node 12: -1.223

Node 13: 0.063

Node 14: 1.190

Node 15: -0.186

Node 16: -1.636

Node 17: 0.848

Node 18: 1.930

Node 19: -1.176

Node 20: 1.778

Node 21: 0.178

Node 22: 1.441

Node 23: -1.176

Node 24: -0.101

Node 25: 1.337

Node 26: -0.036

Node 27: -0.980

Node 28: -0.197

Node 29: -0.458

Node 30: -0.253

Node 31: -0.886

Node 32: -1.857

Node 33: 1.711

Threshold: -0.125

Parameters

Hidden layers = 33 nodes

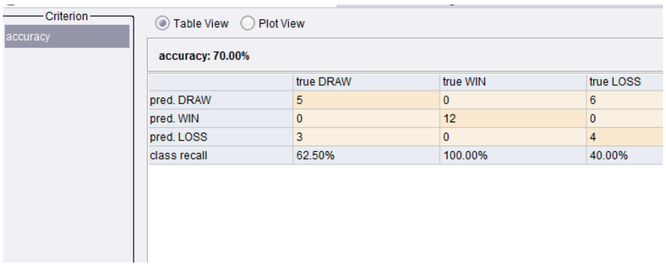
Training cycle = 500

Learning rate = 0.9

Momentum = 0.2

Prediction accuracy = 70% see fig. 6

Number of label = 3



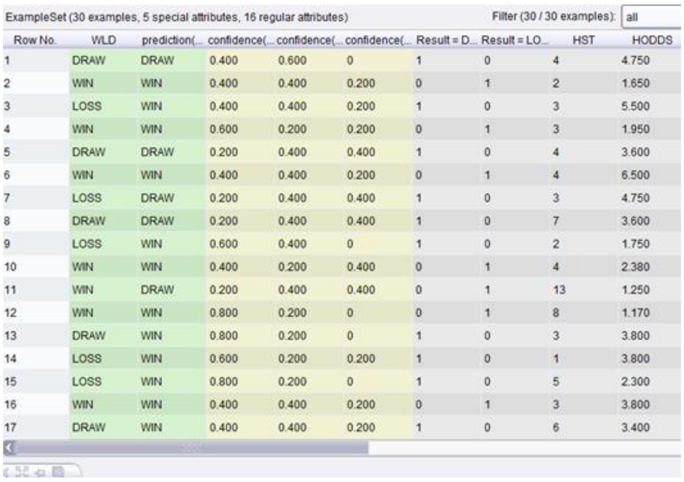
accuracy: 70.00%

	true DRAW	true WIN	true LOSS
pred. DRAW	5	0	6
pred. WIN	0	12	0
pred. LOSS	3	0	4
class recall	62.50%	100.00%	40.00%

Fig.5: Neural network prediction accuracy

V. RESULT DISCUSSION

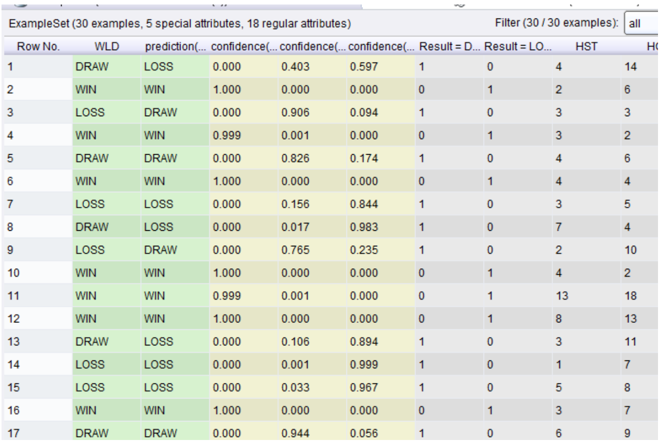
Looking at the total games that were predicted, a prediction accuracy of 53.3 percent was obtained using Euclidean distance of 5 as shown in fig. 2. The predicted result of 9th – 11th week of 2014/2015 EPL season is shown in fig. 3. Most of the incorrectly predicted games by the K-NN (see fig. 6) were predicted correctly by ANN as shown in fig. 7, this confirm the performance of the ANN over the KNN with respect to the given datasets. The actual result of predicted result of 30 matches played in 9th -11th week of 2014/2015 EPL season is shown in fig 6. Also, some of the games incorrectly predicted by the ANN can be referred to as upsets.



ExampleSet (30 examples, 5 special attributes, 16 regular attributes) Filter (30 / 30 examples): all

Row No.	WLD	prediction(...	confidence(...	confidence(...	confidence(...	Result = D...	Result = LO...	HST	HODDS
1	DRAW	DRAW	0.400	0.600	0	1	0	4	4.750
2	WIN	WIN	0.400	0.400	0.200	0	1	2	1.650
3	LOSS	WIN	0.400	0.400	0.200	1	0	3	5.500
4	WIN	WIN	0.600	0.200	0.200	0	1	3	1.950
5	DRAW	DRAW	0.200	0.400	0.400	1	0	4	3.600
6	WIN	WIN	0.400	0.400	0.200	0	1	4	6.500
7	LOSS	DRAW	0.200	0.400	0.400	1	0	3	4.750
8	DRAW	DRAW	0.200	0.400	0.400	1	0	7	3.600
9	LOSS	WIN	0.600	0.400	0	1	0	2	1.750
10	WIN	WIN	0.400	0.200	0.400	0	1	4	2.380
11	WIN	DRAW	0.200	0.400	0.400	0	1	13	1.250
12	WIN	WIN	0.800	0.200	0	0	1	8	1.170
13	DRAW	WIN	0.800	0.200	0	1	0	3	3.800
14	LOSS	WIN	0.600	0.200	0.200	1	0	1	3.800
15	LOSS	WIN	0.800	0.200	0	1	0	5	2.300
16	WIN	WIN	0.400	0.400	0.200	0	1	3	3.800
17	DRAW	WIN	0.400	0.400	0.200	1	0	6	3.400

Fig. 6: Screenshots of 30 predicted sports result using k-NN



ExampleSet (30 examples, 5 special attributes, 18 regular attributes) Filter (30 / 30 examples): all

Row No.	WLD	prediction(...	confidence(...	confidence(...	confidence(...	Result = D...	Result = LO...	HST	H
1	DRAW	LOSS	0.000	0.403	0.597	1	0	4	14
2	WIN	WIN	1.000	0.000	0.000	0	1	2	6
3	LOSS	DRAW	0.000	0.906	0.094	1	0	3	3
4	WIN	WIN	0.999	0.001	0.000	0	1	3	2
5	DRAW	DRAW	0.000	0.826	0.174	1	0	4	6
6	WIN	WIN	1.000	0.000	0.000	0	1	4	4
7	LOSS	LOSS	0.000	0.156	0.844	1	0	3	5
8	DRAW	LOSS	0.000	0.017	0.983	1	0	7	4
9	LOSS	DRAW	0.000	0.765	0.235	1	0	2	10
10	WIN	WIN	1.000	0.000	0.000	0	1	4	2
11	WIN	WIN	0.999	0.001	0.000	0	1	13	18
12	WIN	WIN	1.000	0.000	0.000	0	1	8	13
13	DRAW	LOSS	0.000	0.106	0.894	1	0	3	11
14	LOSS	LOSS	0.000	0.001	0.999	1	0	1	7
15	LOSS	LOSS	0.000	0.033	0.967	1	0	5	8
16	WIN	WIN	1.000	0.000	0.000	0	1	3	7
17	DRAW	DRAW	0.000	0.944	0.056	1	0	6	9

Fig. 7: Screenshots of 30 predicted sports result using ANN

Table 1: Summary of both techniques using sports data.

	KNN	ANN
PRED ACCURACY	**53.3 PERCENT**	**70 PERCENT**
OPTIMIZATION	**LEARNING RATE = 0.9**	**K=5**
DATASETS	**ENGLISH LEAGUE**	**ENGLISH LEAGUE**
EXECUTION TIME	**2 SECS**	**10 SECS**

VI. CONCLUSION

This comparative analysis has shown the effectiveness of using machine learning techniques (K-NN and ANN) in the development of models used in sports prediction. At the end of the experiment, ANN yielded to be a better technique using same datasets and same platform. This has further shown the effectiveness of the use data mining techniques in sports mining.

A) *Research Highlights*

The research highlights of this paper are:

- This paper compares the effect of K-Nearest Neighbour and Artificial Neural Network (ANN) on sports data.
- The approach uses K-Nearest Neighbour for implementation.
- The results shows prediction accuracies of 53.3 percent for the K-NN and 70% percent for the ANN respectively.

REFERENCES

[1] Akhtar, F. and Hahne, C. Rapid Miner 5 Operator Reference. Rapid-I GmbH, 2012, Retrieved 13:15, February 13, 2015 from: http://rapidminer.com/wpcontent/uploads/2013/10/RapidMiner_OperatorReference_en.pdf

[2] Peterson, Leif E. "K-nearest neighbor." Scholarpedia 4(2), 1883, 2009.

[3] Keller, J. M., Gray, M.R. and Givens, J.A.”A fuzzy K-nearest neighbor algorithm”. Systems, Man and Cybernetics, IEEE Transactions 15(4), 580 – 585, 1985.

[4] Buursma, D. Predicting sports event from past result: Towards effective betting on football matches. Preceding 14th Twente Student Conference on IT. University of Twente, Faculty Electrical Engineering, Mathematics and Computer Science, Netherlands. Conference paper 7226, 2011.

[5] Igiri, C. P. and Nwachukwu, E. O. An Improved Prediction System for Football a Match Result. IOSR Journal of Engineering (IOSRJEN. 04(1), 12-20, 2014.

[6] Zheyuan F., Yuming K., Xiaolin, L. Chess game results prediction system. Stanford University CS Machine Learning Project Report, 2013, Retrieved from:http://cs229.stanford.edu/proj2013/FanKuangLinChessGameResultPredictionSystem.pdf.

[7] Anyama O. U. and Nwachukwu E. O. A Hybrid Prediction System for American NFL Results. International Journal of Computer Applications Technology and Research (IJCATR), 4(1) 42-47, 2015.

[8] Anyama, O. U. and Igiri, C. P. An Application of Linear Regression & Artificial Neural Network Model in the NFL Result Prediction. International Journal of Engineering Research & Technology (IJERT), 4(1), 457-461, 2015.

[9] Farzin, O., Parinaz, E., and Faezeh, S. M. Football result prediction with Bayesian network in Spanish league-Barcelona team. International Journal of Computer Theory and Engineering, 5(5), 812-815, 2013.