PREDICTION ON ‘TENNIS - MATCH’ RESULT USING DIFFERENT CLASSIFIER MODELS

Abstract: **Tennis is one of the most popular game all over the world. Now a days a huge amount of people are attracted to it. That’s why there are many betting happen on those matches. Our project is for them who wanted to predict the result of those matches. He/she can predict using this models based on the previous stats of the players who are going to play or a match already started, by the on going stats he/she can also predict the result. This project mainly use for the betting purpose. But one can also use it to find out a specific players weak or strong points.**

1. INTRODUCTION:-

Tennis is basically a game to two persons or four persons playing individually or doubles in two different teams. The game is the one of the most popular game in every country. A match consists of number of sets, mostly 3 and 5 sets match are played. The winner is decided on the basis of the set score. To win a set, you must win at least six games. The games are scored at “love” (or zero) and go up to 40, but actually just four points. From love, the first point is 15, then 30, then 40, then the break point, which wins the game. If suppose, the two teams score is 40-40 then it called Deuce. The match continues to find the winner of the particular on going sub-set. At that time that player makes advantage who win the next point. But still the team will not call as a winner of the sub-set. If the team win the next point also then the team will called as a winner of the sub-set. If the team lose the next point then again the match will back in the deuce position. The match is conducted in four different types of surface like clay, grass, hard, carpet. Each surface has its own characteristics which affect the playing style of the game. The grass court is the fastest surface in the tennis.Female tennis match can last up to 3 hours as they only have to win 2 sets. But for men it can take 5 hours if playing 5 sets match but the average is about 3 hours.

In the world of technology, this technology used by everyone in every field, in every part of today’s life. So for getting any result before the match completes sort of prediction needed. And this prediction can be done by Machine Learning. By using the previous sets of data the machine can be trained for making the prediction. A huge set of data must be used for the proper training of machine so that the accuracy of the result increases. Machine is trained on basis of different attributes like winner name, looser name, surfaces, player’s hand, and no. of match won, player’s height and many other factors. A machine is trained with those factors which can improve the machine to predict the current match. The data is divided in 2 set, the sets are the training set and testing set. The maximum data are used to learn the machine .The main objective is to predict and particular towards the development of predictive models, the models will be typically used to take decisions, and make predictions as accurate as possible. With a predictive model this principal focus is no longer on the data but on the type of the theory about reality.

Few parts of data must be used for the testing purpose so to get a handle on the ability of a predictive model to perform on future data. Although the access to the future cannot be gained before it occurs so the currently available data is reserved and treat it as if were data from the future. A machine that will predict the match result can be used in different purpose. Such machines can be used for bating, such machine can be used for the team selection, and such machine can be used in improvement of the performance of any player.

Prediction can be done on different ways, one way is to predict the match result before the match starts by using the historical data and another way of prediction is to predict the match result during the match that is in-game prediction. In this type of prediction, comparison between the players made.

Not only in the field of tennis, in every field Machine Learning play an important role. Every field requires some prediction so that a future plan can be made on the basis of the result to improve the result.

2. PROPOSED METHEDOLOGY:-

Here 5 different model is used to predict the match result, on the other way it is also used for the comparison among this 5 models. By this data set the prediction of the match result by two ways. One depends on the historical and another one depends on the on-going match result. So here part by part discuss the models and their working method briefly.

i) In case of NAIVE BAYES it simply use the Bayes' Theorem .In this case find out the probability of event A when some other events like B,C,D is already happened.

$$\underline{P(A/B)=P(A).P(B/A)/P(B)} \quad \text{or} \quad \underline{P(C_i/X_1,X_2,X_3\ldots\ldots X_d)=P(X_1,X_2,X_3\ldots\ldots X_d/C_i)P(C_i)}$$

So here event A is the target column and others columns are the independent columns. Value of the probability of event B, C and D determines that what should be the value of the probability of happening the event A. It's a simple mathematics based methodology. So in scene of numeric methods the prediction is appropriate.

ii) Now the next model is KNN (K-Nearest Neighbours) model. It is a lazy learning process. Here the values of target columns are divided into their unique values. That means it divide them into classes. Now it find the centre for each class. Now at time of prediction it take k several point on the graph randomly and check at which class them fitted perfectly without changing their position.

DISTANCE FORMULA EUCLIDEAN $\sqrt{\sum_{i=1}^{k}(x_i-y_i)^2}$

So for predicting the result of an individual it check the classes of those k points. Now there may be a tie. That's why to avoid the tie use always take k as odd. There is also two algorithm. One is uniform and another one is weighted. That means the vote of the closer point is more valuable than the vote of a fuhrer point.

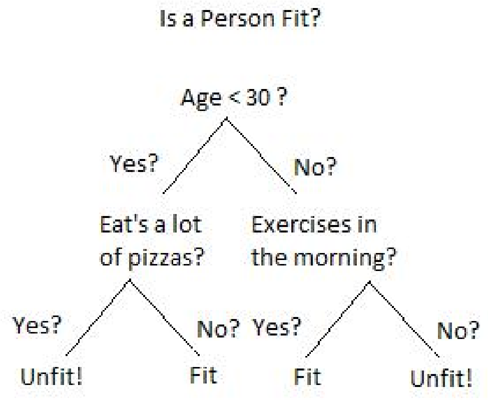
iii) Next Logistic Regression, which actually comes from the word 'logit'. There is a function known as logit function or sigmoid function for any value of X the value of F(X) or Y should be lies between 0 and 1.

A logistic function or logistic curve is a common "S" shape (sigmoid curve), with equation:

$$h_{\theta}(x) = g(\theta^{T}x) = \frac{1}{1+e^{-\theta Tx}}$$

It is more appropriate at the time of binary prediction. Where the prediction is in the form Yes/No. As in this work the same type of issue is getting that's why logistic regression may be one of the best solution.

iv) Now Decision Trees are a type of Supervised Machine Learning (that is you explain what the input is and what the corresponding output is in the training data) here the data is split according to a certain parameter. The tree have two entities, decision nodes and leaves. The leaves are the decisions or the final output. And the decision nodes are where the data is spited.



- **Entropy**

  Entropy, known as Shannon Entropy is denoted by H(S) for a finite set S, is the measure of the amount of uncertainty or better to say randomness in data.

$$H(S) = \sum_{x \in X} p(x) \log_{2} \frac{1}{p(x)}$$

Intuitively, it tells about the predictability of a certain event. Example, consider a coin toss whose probability of heads is 0.5 and probability of tails is 0.5. Here the entropy is the highest possible, since there's no way of determining what the outcome might be .In other words, this event has **no randomness,** hence its entropy is zero.

- **Information Gain**

  Information gain is also called as Callback-Libeler divergence denoted by IG(S, A) for a set S is the effective change in entropy after deciding on a particular attribute A. It measures the relative change in entropy with respect to the independent variables.

$$IG(S, A) = H(S) - H(S, A)$$

  Alternatively,

$$IG(S,A) = H(S) - \sum_{i=0}^{n} P(x) * H(x)$$

  where IG(S, A) is the information gain by applying feature A. H(S) is the Entropy of the entire set, while the second term calculates the Entropy after applying the feature A, where P(x) is the probability of event x.

v) Last model is MLPClassifier of Neural Network. A Neural Network or NN has 3 layers .Input layer, hidden layer and output layer.

The performance of the NN highly depends on the initial weight vector that has been supplied to it at beginning of training phase. Generally, the weight vector is generated randomly keeping its component values within a given range.

Here the output values are only two types but it randomly spread on the plane. That's why it is nearly impossible to separate these points using only one layer or surface. Here the concept is needed of Multi-Layer-Perceptron or MLP. It is nothing but the **feed forward** artificial neural network. Which learn using gradient descent technique. And this training method is known as **back propagation**.

Based on the above mentioned algorithms the models are designed and found out the confusion matrix as well as we calculated the accuracy, precision, recall & f-measure scores.

3. EXPERIMENTAL METHODOLOGY:-

A. ***PREPROCESSING*****:** =

Before Starting up the project it need processed the raw data. For this work the following steps are followed before Classification.

➢**Feature Selection-**

Model Based On Historical Data: From the given data filtered out only the columns like winner name, winner age, winner hand, winner height, winner rank, winner rank points & all the respective columns for loser also including the surface column.

Model Based On In-game Result: Here in this model only those columns are considered which reflects the in-game results of each match. This kind of model is helpful for betting odds. Only those columns are taken which actually hold the in match result for both player like ace_point, break_point, match_save_point with surface.

Two different records for winner & loser are created. In the winner record all the winner attributes were labeled as **‘player’** & all the loser attributes were labeled as **‘opponent’** & a new attribute was added as **‘win’** containing all entries as **‘1’** . And for the loser record all the winner attributes were labeled as ‘**opponent’** & all the loser attributes were labeled as **‘player’** & a new attribute was added as **‘win’** containing all entries as **‘0’** .

Then both the winner & loser record were **merged** together to form our new data-set. Here our target variable is the **‘win’** attribute.

➢**Data Cleaning-**

The data might contain missing values or noise. It is important to remove noise and fill up empty entries by suitable data by means of statistical analysis.

➢**Data Normalization -** May be the data-set holds the value for a very large range. That's why it needs to be normalized before classification task which reduce distance between attribute values. It is generally happen when we keep the range in between -1 to +1.

B. <u>TRAIN-TEST SPLIT: =</u>

After process the data is need to split it. Because it Trained the MODEL(S) by training data and tested by testing data.

1. After pre-processing the data-sets are divided into train and test. One is known as **training** and the other as **testing** data-set. In this project for creating the model four fifth (**80%**) of the data is used as training data and rest (**20%**) as testing data.

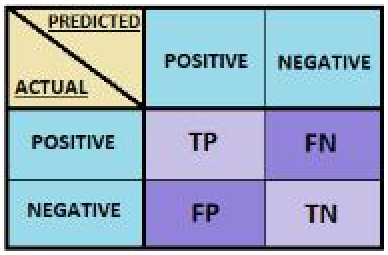
3. But here K-Fold Cross-Validation technique is used. The whole data-set is divided into k no. of subsets. Now it trained by (k-1) no. of subsets and tested by left subset. In this way all possible combination of training subset and testing subset are taken out. Applying those k no. of results gotten. So the actual result will be the average of those.

2. In this prediction process training data-set is send to creating the models, and for getting the values of co-efficient. And by the rest of the data we tested our models, and find out its accuracy.

In the testing phase the classification models obtained from the training phase is employed to test the accuracy of the model.

After we obtain the experimental results it is time to find out the performance of the algorithms which are employed to perform the task. To measure the performance and to compare the performances we use several statistical performance measures like correlation coefficient, accuracy, Kappa statistic, Root mean squared error (RMSE), Mean absolute error, True positive rate (TP rate), and F-measure. The performance measuring parameters are defined as follows;

C. <u>CONFUSION MATRIX: =</u> A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix. This actually shows the models confusion while taking the decision..It shows us not only the errors being made by a classification model but also the types of errors that are being made.



PREDICTED ACTUAL	POSITIVE	NEGATIVE
POSITIVE	TP	FN
NEGATIVE	FP	TN

Definition of the Terms:

- Positive (P): Observation is positive (for example: is an apple).
- Negative (N): Observation is not positive (for example: is not an apple).
- True Positive (TP): Observation is positive, and is predicted as positive.
- False Negative (FN): Observation is positive, but is predicted as negative.
- True Negative (TN): Observation is negative, and is predicted as negative.
- False Positive (FP): Observation is negative, but is predicted as positive.

***Classification Rate/Accuracy*:**

Classification Rate or Accuracy score is calculated by the relation:

$$\boldsymbol{Accuracy = \frac{tp+tn}{tp+fp+fn+tn}}$$

However, there are problems with accuracy. It assumes equal costs for both kinds of errors. A classification model which have high accuracy score can be excellent, good, mediocre, poor or terrible depending upon the problem definition or target requirement.

Recall:

Recall can be defined as the ratio of the total number of correctly classified positive examples divide to the total number of positive examples. High Recall indicates the recognized class is correct or wrong (small number of FN).

Recall is given by the relation:

$$\boldsymbol{Recall = \frac{tp}{tp+fn}}$$

Precision:

To get the value of precision we divide the total number of correctly classified positive examples by the total number of predicted positive examples. High Precision indicates an example labelled as positive is indeed positive (small number of FP). Precision is given by the relation:

$$Precision = \frac{tp}{tp+fp}$$

High recall, low precision: This means that most of the positive examples are correctly recognized (low FN) but there are a lot of false positives(high FP).

Low recall, high precision: This shows that we miss a lot of positive examples (high FN) but those we predict as positive are indeed positive (low FP)

F-measure:

Since we have two measures (Precision and Recall) it helps to have a measurement that represents both of them. We calculate an F-measure which uses Harmonic Mean in place of Arithmetic Mean as it punishes the extreme values more.

$$F-measure = 2 * \frac{Precision * Recall}{Precision+Recall}$$

The F-Measure will always be nearer to the smaller value of Precision or Recall.

4. RESULT & DISCUSSION:

The outcome of the match result before the final result with good accuracy with the uses of different models and serve the result is the main objective to be predicted. Here, the Accuracy Score in 'historical data' and also 'on in game result' are predicted. Five methods to predict the Accuracy Score.

1. NAIVE BAYES 2. LOGISTIC REGRESSION 3. KNN CLASSIFIER 4. DECISION TREE 5. NEURAL NETWORK

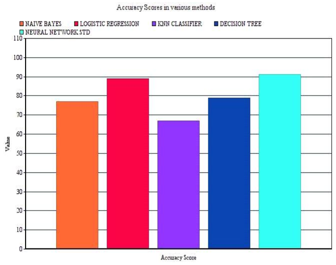
The Precision Score, Recall score, F1 Score, AUC Score using these methods also predicted. The Bar-Graphs are shown below.

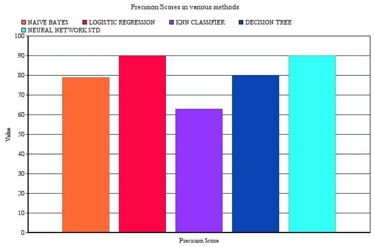
There is a possibility that the scores are vary 5% more or less by machine to machine.

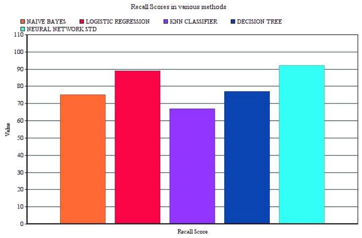
***SCORES BASED* ON IN GAME RESULT:**

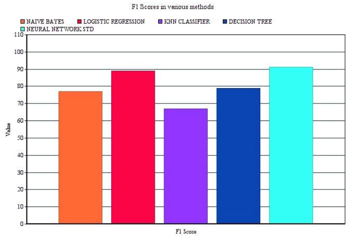
MODEL NAME	**NAIVE BAYES**	**LOGISTIC REGRESSION**	**KNN CLASSIFIER**	**DECISION TREE**	**NEURAL NETWORK**
Confusion Matrix	**[[527 133] [171 515]]**	**[[590 70] [75 611]]**	**[[438 222] [228 458]]**	**[[527 133] [155 531]]**	**[[590 70] [54 632]]**
Accuracy Score:	***0.7741***	***0.8922***	***0.6656***	***0.7860***	***0.9078***
Precision Score:	***0.7947***	***0.8972***	***0.6735***	***0.7996***	***0.9002***
Recall Score:	***0.7507***	***0.8906***	***0.6676***	***0.7740***	***0.9212***
F1-Score:	***0.7721***	***0.8939***	***0.6705***	***0.7866***	***0.9106***
AUC Score:	***0.7746***	***0.8923***	***0.6656***	***0.7862***	***0.9076***

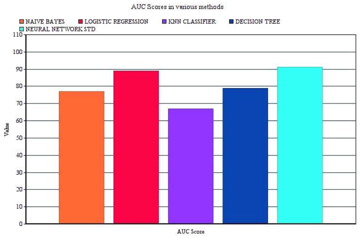
GRAPHS











🟧 NAIVE BAYES'

🟥 LOGISTIC REGRESSION

🟪 K-NN CLASSIFIER

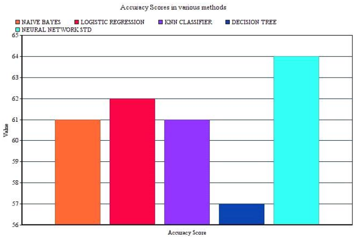
🟦 DICISION TREE

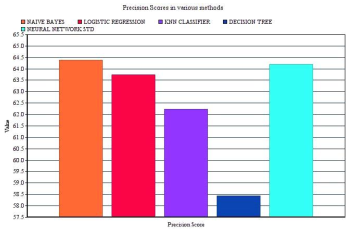
🟦 NEURAL NETWORK

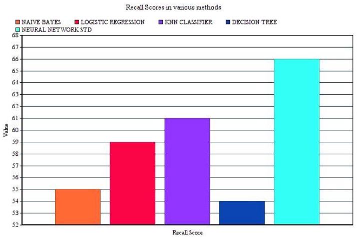
SCORES BASED ON **HISTORICAL DATA :**

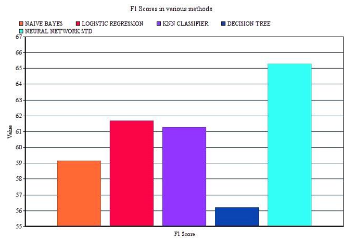
MODEL NAME	**NAIVE BAYES**	**LOGISTIC REGRESSION**	**KNN CLASSIFIER**	**DECISION TREE**	**NEURAL NETWORK**
Confusion Matrix	**[[446 209] [313 378]]**	**[[420 235] [278 413]]**	**[[402 253] [274 417]]**	**[[389 266] [317 374]]**	**[[399 256] [232 459]]**
Accuracy Score:	***0.6121***	***0.6188***	***0.6084***	***0.5668***	***0.6374***
Precision Score:	***0.6439***	***0.6373***	***0.6223***	***0.5843***	***0.6419***
Recall Score:	***0.5470***	***0.5976***	***0.6034***	***0.5412***	***0.6642***
F1-Score:	***0.5915***	***0.6168***	***0.6127***	***0.5619***	***0.6529***
AUC Score:	***0.6139***	***0.6194***	***0.6086***	***0.5675***	***0.6367***

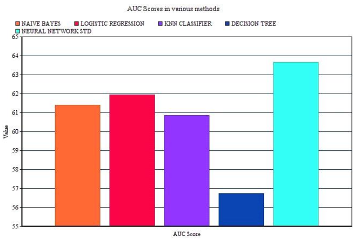
GRAPHS

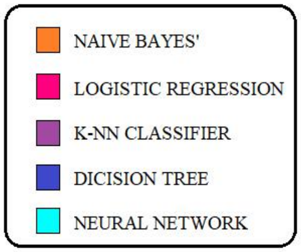












After observing the Scores and the Graphs, in the number 1 way that is based on 'on in game result', the better Accuracy Score is find from by using NEURAL NETWORK method. And in 'Historical Data', we find better Accuracy Score also by using NEURAL NETWORK.

-: CONCLUSSION:-

In this paper ' A Tennis Match Prediction ' depends on past data and on-match data is proposed. Using the data-set we can predict a stats of a player. Possible other approaches that could produce better results are using only the top tournaments for training data, experimenting with more kernels and larger ranges of parameters, as well as using data from farther in the past. Given the good performance of our baselines, it may also be feasible to actually use a boosting algorithm using our baselines as the weak learners to obtain better results. The data here we get is actually not so large. It only consist the information of the tennis matches played in one single year. There is hardly one match between two players. So for getting the higher result we need more data. If the data is more large and hold the records of a long time period then I am sure that our model we able to predict better result.