

# Machine Learning Algorithms applied to the Classification of Robotic Soccer Formations and Opponent Teams

Brígida Mónica Faria\*, Luís Paulo Reis<sup>§</sup>, Nuno Lau<sup>&</sup> and Gladys Castillo<sup>#</sup>

\*Escola Superior Tecn. Saúde do Porto (ESTSP/IPP), Inst. Eng. Electrónica e Telemática de Aveiro (IEETA)  
Universidade do Porto and Universidade de Aveiro, Portugal

Email: btf@estsp.ipp.pt

<sup>§</sup>Lab. Intel. Artificial e Ciência de Computadores (LIACC), Dep. Eng. Informática Fac. Eng. Univ. Porto (DEI/FEUP)  
Universidade do Porto, Portugal

Email: lpreis@fe.up.pt

<sup>&</sup>Inst. Eng. Electrónica e Telemática de Aveiro (IEETA), Dep. Elect., Telecomunicações e Informática (DETI/UA)  
Universidade de Aveiro, Portugal

Email: nunolau@ua.pt

<sup>#</sup>Departamento de Matemática (DMAT/UA), Universidade de Aveiro, Portugal  
Email: gladys@ua.pt

**Abstract**— Machine Learning (ML) and Knowledge Discovery (KD) are research areas with several different applications but that share a common objective of acquiring more and new information from data. This paper presents an application of several ML techniques in the identification of the opponent team and also on the classification of robotic soccer formations in the context of RoboCup international robotic soccer competition. RoboCup international project includes several distinct leagues where teams composed by different types of real or simulated robots play soccer games following a set of pre-established rules. The simulated 2D league uses simulated robots encouraging research on artificial intelligence methodologies like high-level coordination and machine learning techniques. The experimental tests performed, using four distinct datasets, enabled us to conclude that the Support Vector Machines (SVM) technique has higher accuracy than the k-Nearest Neighbor, Neural Networks and Kernel Naïve Bayes in terms of adaptation to a new kind of data. Also, the experimental results enable to conclude that using the Principal Component Analysis SVM achieves worse results than using simpler methods that have as primary assumption the distance between samples, like k-NN.

**Keywords**—Machine Learning, Principal Component Analysis, Support Vector Machines, RoboCup, Soccer Simulation

## I. INTRODUCTION

RoboCup is an international project to promote research in (Distributed) Artificial Intelligence, (Intelligent) Robotics and associated areas. It is an attempt to promote artificial intelligence and robotics research by providing a standard problem (playing a game of soccer with pre-established rules) where a wide range of technologies can be integrated and examined [1]. RoboCup Simulation League has been one of the competitions integrated on the RoboCup international project since its beginning in 1997. The main objectives of this league are concerned with developing the high-level decision and coordination modules of teams of robots [2]. The 2D

simulation league uses the SoccerServer [3] which is a simulator that creates a 2D virtual soccer field enabling virtual players to play a realistic game of soccer. This simulator implements the physics of the simulated soccer world (including the movement of the players and the ball), player's perception (including vision, aural and body sensorial information, stamina and effort of the players, players actions (like kicking, dashing, turning, catching, among others) and the coaching and refereeing model of the virtual world. Another aspect that brings realism is the fact that the models in the simulator are taken both from real robots and from human like characteristics. FC Portugal is a joint project from the Universities of Porto and Aveiro in Portugal focusing on applying coordination methodologies in RoboCup. One of the subprojects of the FC Portugal team [4] focuses their investigation in the usability of formations as a suitable coordination methodology for robotic soccer [2,3,4]. In this context, the team proposed SBSP – Situation Based Strategic Positioning [2,3,4] as its main formation definition algorithm in 2000. After the release of the algorithm and its base source code, this algorithm was adapted by most of the simulation league teams since 2001 until the present. Besides being able of generating appropriate formations, a very important aspect is to be able to classify and predict the formations that are being used by opponent on the competition games [5]. Moreover it is very useful to identify the opponent team and its characteristics in the first moments of the game. In this paper a comparative study of four techniques for the classification of opponent teams and their formation is presented: Support Vector Machines (SVM) [6]; Artificial Neural Network (ANN), k-Nearest-Neighbor and Kernel Naïve Bayes (KNN). The tool used for developing the machine learning and data mining experiments was RapidMiner [7]. In order to perform the tasks of Principal Component Analysis (PCA) the Statistical Package for the Social Sciences (SPSS) was used.

This paper is organized as follows. Firstly a clarification of the distinct RoboCup soccer leagues and its characteristics is presented. Next a theoretical description of the four algorithms used in the study and the measures used to compare the classifiers is presented. Finally the experimental results are described along with some conclusions and future work.

## II. ROBOCUP SOCCER LEAGUE

In the Robocup project there are different leagues divided in two main groups: robotics and simulation. The first group involves physical robots with different sizes and different rules based on the competition that they integrate. The second one has the goal of, without the necessity to maintain any robot hardware, being able to research on artificial intelligence, coordination methodologies and team strategy. There are three fields of simulation: 2D and 3D Simulation League and Mixed Reality with Eco-Be Citizen Robots. In the 2D Simulation League two teams of eleven autonomous agents (software programs) each play soccer in a two-dimensional virtual soccer stadium implemented by a central server, called SoccerServer [3]. This server knows everything about the game, the current position of all players and the ball. The game further relies on the communication between the server and each agent. Each player receives from the server the relative and noisy input of his virtual sensors (visual, acoustic and physical perceptions), and may perform some basic commands like dashing, turning or kicking, in order to influence its environment [3].

Adding an extra dimension and more complex physics increases the simulation realism. In fact, the 3D Simulation League began with a spherical robot model. Nowadays it implements a humanoid model identical to the Nao Robot from Aldebaran [8]. Fig. 1 shows examples of these two categories. The first picture is a frame of a 2D football game and the second an example of humanoids on a 3D field.

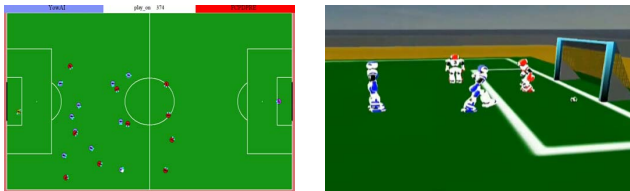


Figure 1. 2D and 3D simulation leagues

Finally the mixed reality was introduced in 2007 using the Eco-Be Robots from Citizen as a standard platform. Here the soccer game is on top of a virtual field with a virtual ball, using the concept of augmented reality as can be observed in Fig. 2.



Figure 2. Eco-Be Robots from Citizen [Adapted from [2]]

FC Portugal team (joint project from Universities of Aveiro and Porto) was created in 2000 and entered in that year at its

first competition, the European championship at Amsterdam, The Netherlands. There are some characteristics only available in this team, such as its flexible strategy, capacity to play with different formations and dynamic positioning and role exchange.

Preliminary work on knowledge discovery using RoboCup simulation league data [5] enabled to apply, to this type of data, several learning algorithms available on WEKA [9]. Nevertheless this study only considered the data obtained by simplified simulation 2D games using a simplified version of the FC Portugal team as the object of study. The main simplification was the removal of the ability to perform dynamic positioning and role exchange from the team. One proposal of this work is to study the main differences between the data base inspired by [5] and a new, more challenging, data base with dynamic positioning and role exchange active. The work is also focused on the detection of the team opponent in the first moments of the game. The work has also another upgrade made since the first version base on the application of Principal Component Analysis (PCA).

## III. MACHINE LEARNING ALGORITHMS

Machine Learning may be described as the union of statistics and Artificial Intelligence (AI). Machine Learning can be considered as an evolution of most AI algorithms, because it blends AI heuristics with advanced statistical analysis. Machine learning attempts to let computer programs learn about the data they study, such that programs make different decisions based on the qualities of the studied data, using statistics for primary concepts, and adding more advanced AI heuristics and algorithms to achieve its objectives [10]. In this work a special attention was given to PCA in order to choose the best and simple attributes for classification. Evaluation of the model and interpretation of the results should help the phase of decision by observing the extracted knowledge. Four supervised machine learning algorithms are discussed here. The classes of interest for the classification process are known in advance and the objective is to classify each new observation to the respective class using a specific function.

### A. Artificial Neural Networks

Artificial Neural Network is a mathematical/computational model that attempts to simulate the structure of biological neural systems. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase. The neurons are typically identical units that are connected by links. The interconnections are used to send signals from one neuron to the other [11]. The concept of weights between nodes is also present since it is used for establishes the importance from one connection to the other. The network may contain several intermediary layers between its input and outputs layers. The intermediary layers called hidden layers and the nodes embedded in these layers are called hidden nodes. In a feed-forward neural network the nodes in one layer are connected only to the nodes in the next layer. The Perceptron is the simplest model since do not use any hidden layers. One of the most used models for classification using NNs is the Multilayer Perceptron using the backpropagation

algorithm in which ANNs have 3 or 4 layers. The ANN model has several characteristics like the capability of handling redundant features since the weights are automatically learned during the learning phase. The weights for redundant features tend to be very small. The method called gradient descent [12] is used for learning the weights which often converge to some local minimum; however one way to overpass the local minimum is to add a momentum term [12] to the weight update formula. Another known characteristic is the consuming time for training an NN, especially when the number of hidden nodes is large.

#### B. Kernel Naïve Bayes

The Naïve Bayes is a classifier which presents the probability of the accuracy of an object and then applies the Bayes Theorem to produce the classification. The Naïve Bayes classifier assumes that the presence of a particular feature of a class is unrelated to the presence of any other feature. Because independent variables are assumed, only the variances of the variables for each class need to be determined and not the entire covariance matrix [13]. The kernel distribution is appropriate for features that have a continuous distribution. For each feature modeled with a kernel distribution, the Naïve Bayes classifier computes a separate kernel density estimate for each class based on the training data for that class. The most used kernel is the normal kernel but it is possible to specify different kernels for each feature and different widths for each feature or class [14].

#### C. K-Nearest Neighbor

A Nearest Neighbor classifier represents each example as a data in a d-dimensional space, where d is the number of attributes. Given a test example it is computed the proximity to the rest data points in the training set, using a measure of similarity or dissimilarity, such as Euclidian measure or its generalization, the Minkowski distance metric, the Jaccard Coefficient or Cosine Similarity [12]. The k-nearest neighbor (k-NN) of a given example refers to the k points that are closest to the example. Fig. 3 shows representations of the nearest neighbors for k=1, 2, 3.

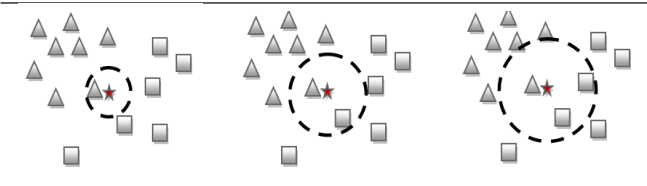


Figure 3. Representations of 1, 2 and 3 NN [Adapted from [12]]

Some of the main points that characterized k-NN are the insertion in the category of lazy learners, since they do not require building a model and only make their predictions on local information. However classifying a test example is an expensive task because it is necessary to compute individually the proximity values between the test and training examples. An important decision about the proximity measure it is also necessary since the wrong choice can produce wrong predictions [12].

#### D. Support Vector Machines

Support Vector Machine is a technique based on statistical learning theory which works very well with high-dimensional data and avoids the curse of dimensionality problem [12]. The objective is to find the optimal separating hyperplane between two classes by maximizing the margin between the classes' closest points. There are several cases which should be study. Fig. 4 presents the simplest one, since it is the case of linearly separable classes with class +1 and class -1.

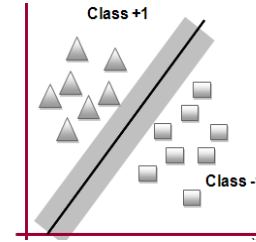


Figure 4. Linear decision boundary – separable case

The problem can be interpreted as an optimization where  $w \cdot x + b = 0$  is the hyperplane of a linear classifier which maximizes the margins and  $y_i$  represents the class (+1 or -1), and  $x_i$  represents the input vector. The points on the boundaries are called support vectors. There are solutions for multiclass problems and for the cases that are not linearly separable. These explanations can be found at [12] [15] [16]. To obtain the parameters most of the times it is used cross-validation or another scheme of experimental parameters [12].

### IV. EXPERIMENTAL DEVELOPMENT

The four machine learning algorithms were applied to a dataset produced by the positions of the players of the FC Portugal in 2D Simulation league. The performance measures are described in this section together with the experimental settings and results.

#### A. Data set description

The dataset was produced with the x, y positions of eleven players of FC Portugal in 2D Simulation League in six distinct games using dynamic positioning and role exchange for the players. This ability enables the team players to exchange their positionings in the team formation making the classification process a lot more difficult. FC Portugal played two games against some known robotic soccer teams: Hellios, Brainstormers and NCL [8]. The attributes used for this study are the ball and players' positions and the class is the formation (multi-class problem) that the team was playing with. Table I displays the 10 possible formations that the team could play. For example a formation 433 means that the team it is playing with 4 defenders, 3 midfielders and 3 forwards players.

TABLE I. FORMATIONS OF FC PORTUGAL - MULTI-CLASS PROBLEM

Classes	One	Two	Three	Four	Five	Six	Seven	Eight	Nine	Ten
Formation	433	442	343	352	541	532	361	451	334	325

As can be observed in Fig. 5 [4] the coordinate x has the range of -52,5 and 52,5 and the coordinate y varies between -34,0 and 34,0 (corresponding to a typical real soccer field of

105x68m). The center of the field is the origin of the referential and the axes orientation are also represent at Fig. 5.

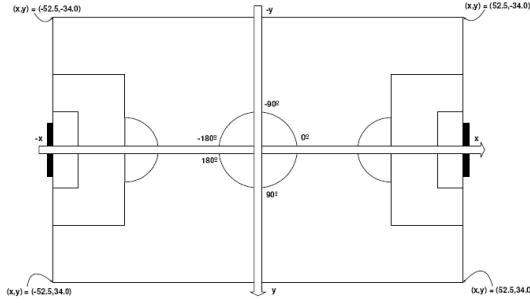


Figure 5. Coordinates and angles of the field game [Adapted from [4]]

The games were executed in Linux and the logs files are converted in text files with a simple application *getWState* [4] written in C++ for this purpose. There are several variables that can be acquired from the games such the position and velocity of the ball, the eleven players of the two teams and others characteristics as can be seen in Fig. 6. In a previous work [5] it was discovered that the database with the center of mass of the FC Portugal team produces better results. Since the primordial objective of this work is to compare three different classifiers and obtain the best model, the variables corresponding to the center of mass were included on the databases. The mathematical formulation (1) of the center of the mass of the team is:

$$(cx, cy) = \left( \frac{\sum_{i=1}^{11} x_i}{11}, \frac{\sum_{i=1}^{11} y_i}{11} \right) \quad (1)$$

where  $x_i$  and  $y_i$  represents the coordinates of the positions of the player<sub>i</sub> of the FC Portugal team.



Figure 6. Interface of *getWState* [Adapted from [4]]

Therefore the final data set had the positions of the players, the position of the ball, the center of mass and the formation that FC Portugal was playing. Thus, the data base has 26 numerical and continuous attributes ( $\mathbf{R}^{26}$ ) and one nominal attribute (10 formations options of FC Portugal). The first dataset (Database A) has 37943 examples with approximately 6000 cycles by game. There are differences on the number of examples since there is the possibility to have periods in the game that are stopped or others that are not counted but in which players are still moving and thus are included in the database. After an initial exploratory study it was removed equal cycles obtaining a second database (Database B) with 35763 examples. In order to simplify the number of variables

used to identify the player's positions it was applied Principal Components analysis (PCA). This is a multivariate and an exploratory technique with the objective to transform a set of correlated variables into a set with less independent variables (linear combination of the original variables) [17]. So a third database (Database C) was created with four components extracted from players and ball position and center of mass. These four components explain 71,92% of total variance.

#### B. Performance measures and configuration parameters

To be able to quantify and compare the performance of the four algorithms several measures were analyzed. Performance evaluation of a classification model is based on the counts of test records correctly and incorrectly predicted by the model. These counts are written in the confusion matrix. Although this matrix provides the information necessary to determine how well a classification model performs, it is significant to compare the performance of different models with just one single number. For example this can be done using accuracy. Besides that there are several methods for estimating the generalized error of a model during training, for finding a model with a manageable complexity and not susceptible to overfitting. The methods used for evaluate the performance of a classifier are: holdout method; random subsampling, cross-validation and bootstrap [12]. Another line of study is answering the question of the required dimension of the training set using different classifiers. The construction of the learning curve is an appropriate method and the procedure firstly divides the dataset into two parts and with an incremental ratio of 5% iteratively adds this percentage of examples into the training subsets and then calculates the performance values on the fixed test set. The study also used the 10-fold cross-validation to compare the performance of the three classifiers. The procedure initially begins with the division of the dataset into 10 equal-sized partitions. Then each classifier is applied to construct a model from 9 of the portions and test it on the remaining partition. This step is repeated 10 times, each time using a different partition as the test set. Since the objective is to compare the three classifiers it was followed the recommendation to acquire a bigger sample by implementing a scheme of 3x10 cross-validation in order to obtained a dimension sample of 30 examples and by the Central Limit Theorem perform a parametric test. The ANOVA and paired samples t test were performed with a significance level of 5%. By applying the ANOVA and t tests the p values obtained were near to zero so less than 0,05. Thus, there are statistical evidences to affirm that the means of the results produced by the four classifiers are different. It is important to refer the settings and parameters used for applying the classifiers. Since the classification is a multiclass problem the RapidMiner incorporates the LibSVM developed by Chih Chang et Chih-Jen Lin [7] which supports multiclass learning and probability estimation based on Platt scaling for proper confidence values after applying the learned model on a classification data set. The operator supports the SVM types C-SVC and nu-SVC for classification tasks. The differences are basically over the parameters. The range of C is from zero to infinity but nu is always between [0, 1]. For experimental results the type of SVM used was C-SVC which applies one against one approach and the kernel used was RBF. The parameters introduced are registered in the table II:



TABLE II. PARAMETERS USED WITH LIBSVM

Parameters	Values
Kernel Type	RBF
Gamma	0
Tolerance of termination criterion	0,001
Cost parameter for C-SVC	0

To apply Artificial Neural Networks the W-Multilayer Perceptron operator, also available on WEKA, was chosen. This operator is characterized for producing a classifier that uses backpropagation to classify instances and were the network can be built by hand, created by an algorithm or both. The network can also be monitored and modified during training time. The parameters used for the neural network are displayed in table III. The default parameter for the hidden nodes was calculated using information about the number of attributes and classes by the formula hidden nodes =  $[(\text{attributes} + \text{classes}) / 2]$ . The simplest classifier Nearest Neighbors was performed with  $K=3$  and the type of measure used were mixed Euclidean distance and the Kernel Naïve Bayes it was applied with a greedy kernel density estimation mode.

TABLE III. PARAMETERS USED WITH MULTILAYER PERCEPTRON

Parameters	Values
Learning rate	0,3
Momentum rate	0,2
Number of epochs of training	500
Hidden layers in the network	1
Hidden nodes in the network	18

## V. EXPERIMENTAL RESULTS

This section describes the results of several experimental tests. Applying support vector machine, neural network, k-nearest neighbor and kernel naïve Bayes and compare the results conducted on the dataset with information about FC Portugal team. The experimental results were achieved using RapidMiner Version 4.6 [7] and SPSS Version 17.0 in a Pentium dual-core processor T2330 (1.60 GHz, 533 MHz FSB L2 Cache) and 2 GB DDR2. The processing time was also measured and compared in the different experiences. The first experience was a simple project using the first dataset without the data pre-processing and using the method of 10-cross-validation for evaluating the accuracy. This experience was also repeated with the second dataset. The results of this experience are shown in table IV. It can be observed that the time consumed by the project that involves Neural Networks is the most expensive and the best results about accuracy are demonstrated by 3-Nearest Neighbor.

TABLE IV. ACCURACY AND TIME OF EXPERIENCE (DATABASES A / B)

	Classifier	SVM	NN	3-NN	KNB
Data Base A	Accuracy (%)	95,77	80,99	99,78	78,0
	Time	48'21''	5h18'16''	1h7'3''	1'14''
Data Base B	Accuracy (%)	96,07	84,72	99,79	77,04
	Time	28'03''	3h25'17''	9'26''	1'13''

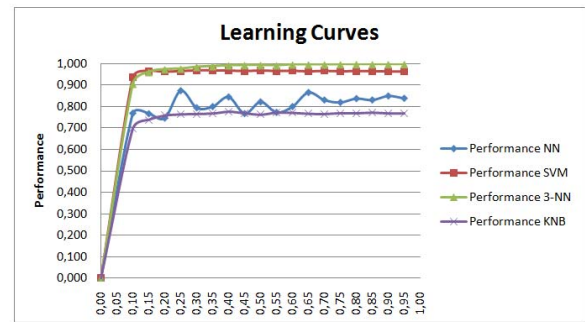


Figure 7. Relation between training set dimension and accuracy (DataBase B)

Observing the learning curves (Fig. 7) it is possible to obtain some conclusions. In fact with approximately 10% of the training set the accuracy is almost at their pick using SVM and 3-NN. The NN has a more oscillatory curve and the KNB produces the worst accuracy. Further experiments using distinct number of hidden nodes may be needed to further validate the approach of NN. In order to observe the accuracy given by a database composed only with the components an analogous experience were also made and the results are in table V.

TABLE V. ACCURACY AND TIME OF EXPERIENCE WITH 4 COMPONENTS

	Classifier	SVM	NN	3-NN	KNB
Data Base C	Accuracy (%)	10,25	46,69	95,91	50,21
	Time	1h52'23	51'49''	3'54''	11''

The time to complete the process is very slow comparing with the dataset with 26 attributes making the process faster except for the SVM. However, the accuracy is low meaning a loss of model prediction. The 3-NN guarantees approximately 96% of accuracy. This fact provides the information that using 3-NN with four components is analogous of using the 26 attributes to construct the model. The learning curves in this case are represented in Fig. 8.

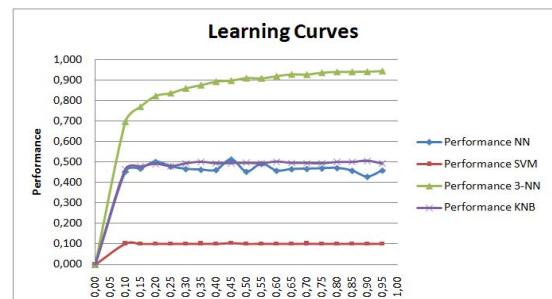


Figure 8. Relation between training set dimension and accuracy (DataBase C)

In terms of accuracy the learning process using four components has lower accuracy than using the 26 attributes. The 3-NN needs almost 65% of the training set in order to give a result of 93% of accuracy. The initial slope of the learning curve using 3-NN is more marked than the others three learning curves which indicates a more quickly phase of learning with a smaller training set. For testing the four classifiers with a different database and with different games were performed experimental work with a Database D with 2 games of the most

recent FC Portugal. The objective of this experience is to study how well the previous classifiers (using database B as the training set) can predict the formations of FC Portugal team but in different situations of games and strategies.

TABLE VI. ACCURACY AND TIME

Classifier	SVM	NN	3-NN	KNB
Accuracy (%)	51,65	45,77	46,67	26,47
Time (")	4'58"	21'11"	3'55"	29"

In these conditions the best performance were produced by SVM with an accuracy of 51,65%. The 3-NN and the Neural Networks had similar performances in terms of accuracy, however, NN used the in terms of consumed seven more units of time. These results reveal that using the components the SVM produces worst results, but in terms of adaptation to a new kind of data is more flexible with the 26 attributes. Finally the last experiment was performed to detect the opponent team playing against FC Portugal. The first results achieved using the players positions can easily predict the opponent teams. Almost every algorithms here tested could achieve almost 100% of examples correctly classified in the first 20 cycles of the testing game. The justification for this result relies on the positions of the opponent players which are typically the same, in each game, before the start of the game due to the fact that the players are directly positioned on their starting position. So instead of using all the attributes another experience using PCA was made. Using 4 components, that explains 89,4% of the total variance, it also was achieved an accuracy of 100% except using the KNB. The accuracy using KNB was 99,98% since one cycle was predicted as an example of Brasil but really was from Hellios and one case was from Brasil and was predicted as Hellios cycle.

## VI. CONCLUSIONS AND FUTURE WORK

The comparative study of SVM, Neural Networks, k-Nearest Neighbor and Kernel Naïve Bayes was conducted by analyzing theoretical and experimental concepts behind the machine learning algorithms. Experiences were performed in order to obtain results about predictions of formations of FC Portugal team and their opponent teams. The conclusions obtained revealed that if a model is trained with certain games the 3-NN shows better results in predicting formations with those games and even using the Principal Component Analysis. However in reality the training games are not the same that we wish to predict in a competition. Thus, when another test set is applied with a different data set of games the results produced by SVM are, in terms of accuracy, the best. Another important point to be referred is the time consumed to implement and finalize the projects in RapidMiner. Applying cross-validation for all schemes the NN is the most expensive in terms of computational time consumed. For future work it is important to test and apply other algorithms and test several others measures in order to confirm the results here obtained. Also, a more thoughtful cleaning process of the database should be performed. This process should be conducted taking in consideration the opinion of experts that may correctly classify the example near the transitions between formations, stating the exact point where the formation has completely changed.

Finally further tests are needed in the opponent classification task, using a higher number of different opponents that use similar strategies and similar formations concept.

## ACKNOWLEDGMENTS

The authors would like to acknowledge to FCT – Portuguese Science and Technology Foundation (PhD Scholarship FCT/ SFRH / BD / 44541 /2008), DETI/UA – Dep. Electrónica, Telecomunicações e Informática and ESTSP/IPP – Esc. Sup. Tecnologia Saúde Porto-IPP. This work was partially supported by project FCT-PTDC/EIA/70695/2006 – “ACORD: Adaptive Coordination of Robot Teams”.

## REFERENCES

- [1] Hiroaki, K.; Minoru, A.; Yasuo, K.; Itsuki, N.; Eiichi, O. and Hitoshi, M.; “RoboCup: A challenge problem for AI and robotics”, Springer LNCS Berlin, pp. 1-19, April 2006. ISBN: 978-3-540-64473-6
- [2] Lau, N. and Reis, L. P., “FC Portugal-High-level Coordination Methodologies in Soccer Robotics”, Robotic Soccer, Book edited by Pedro Lima, Itech Education and Publishing, Vienna, Austria, pp. 167-192, December 2007, ISBN 978-3-902613-21-9.
- [3] Reis, L. P. and Lau, N., “FC Portugal Team Description: RoboCup 2000 Simulation League Champion”, in P. Stone, T. Balch and G. Kraetzschmar, editors, RoboCup-2000: Robot Soccer World Cup IV, Springer LNAI, Vol. 2019, pp.29-40, 2001, ISBN 3-540-42185-8.
- [4] Reis, L. P.; Lau, N. and Oliveira, E., “Situation Based Strategic Positioning for Coordinating a Team of Homogeneous Agents” in M. Hannebauer et al. Editors, Balancing Reactivity and Social Deliberation in MAS – From RoboCup to Real-World Applications, Springer LNAI, Vol. 2103, pp. 175-197, Berlin, 2001, ISBN 3-540-42327-3.
- [5] Almeida, R.; Reis, L. P. and Jorge, A. M., “Analysis and Forecast of Team Formation in the Simulated Robotic Soccer Domain”. in L.S.Lopes, et. al. editors, 14th Port. Conf. on Artificial Intelligence, EPIA'2009, Aveiro, LNAI 5816, Springer, pp. 239-250, Oct 12-15, 2009
- [6] Vapnick, Vladimir, “The Nature of Statistical Learning Theory”. Springer-Verlag, 1995. ISBN 0-387-98780-0.
- [7] RapidMiner, “RapidMiner Report the Future,” Available at [http://rapid-i.com/content/blogcategory/38/69/]. Consulted April 2009.
- [8] Robocup, “Robocup 2009”, Available in [http://www.robocup2009.org/]. Consulted May 2009.
- [9] Eibe, Frank; Mark, Hall; Len, Trigg, “Weka 3: Data Mining Software in Java,” Available at [http://www.cs.waikato.ac.nz/~ml/weka/index.html]. Consulted May 2009.
- [10] Data Mining Software-A Brief History of Data Mining. Online available at: http://www.data-mining-software.com, Consulted April 2010
- [11] Kunzle, Philippe, “Vehicle control with neural networks”, Available at [http://www.gamedev.net/reference/programming/features/vehiclenn/]. Consulted June 2009.
- [12] Tan, Pang-Nin; Steinback, Michael; Kumar, Vipin, “Introduction to Data Mining”, Pearson Education, Inc, 2006. ISBN: 0321321367.
- [13] Zhang H., “The Optimality of Naive Bayes”, Faculty of Computer Science, University of New Brunswick, Fredericton, New Brunswick, Canada, American Association for Artificial Intelligence, 2004.
- [14] The MathWorks, Inc., “Accelerating the pace of engineering and Science”, Available at [http://www.mathworks.com/ access/helpdesk/help/toolbox/stats/br039qw-1.html]. Consulted April 2010.
- [15] Bennet, K., Campbell, C., “Support Vector Machines: Hype or Hallelujah?”, SIGKDD Explorations, Vol 2, Issue 2, ACM SIGKDD, Dec 2000.
- [16] Anguita, D.; Boni, A.; Ridella, S.; Riviaccio, F.; Sterpi, D., “Theoretic and Pratical Model Selection Methods for Support Vector Classifiers”, Dep. Biophysical and Electronic Eng. Univ. Genova, Italy.
- [17] Duntelman, G. H., “Principal Component Analysis”, SAGE Publications, Inc., USA 1989, ISBN: 0-8039-3104-2