Spurs and Bits: Predicting football results using Bayesian Nets and other Machine Learning Techniques

N.E. Fenton, M Neil, A. Joseph

RADAR group, Queen Mary, university of London

Abstract

Bayesian Networks, BNs, provide a means for representing, displaying and making available in a usable form the knowledge of experts in a given field. In this paper we look at the performance of an expert constructed BN compared with other Machine Learning, ML, techniques in the area of predicting the outcome of football matches involving Tottenham Hotspur. The period under study was 1995-1997 when the expert BN was constructed. The object of the study was to determine retrospectively the comparative accuracy of the expert BN compared to some alternative machine learners and to comment on any insights the learners provide. The additional ML techniques considered were: MC4, a decision tree learner; Naive Bayesian learner; Data Driven Bayesian, that is a learnt network and node probability tables; and a K-Nearest Neighbour learner. The results show that the expert BN is generally superior to the other techniques for this domain.

Key words: BN, learning, comparison

1 Introduction

Bayesian Networks, BNs, provide a means for capturing, displaying and making available in a usable form knowledge, often obtained from experts in a given field. In this paper we look at the performance of an expert constructed BN compared to other machine learning techniques in predicting the outcome,

Email address: norman,martin,adrianj@dcs.qmul.ac.uk (N.E. Fenton, M Neil, A. Joseph).
URL: http://www.dcs.qmul.ac.uk/researchgp/radar/ (N.E. Fenton, M Neil, A. Joseph).

in terms of win, loose, or draw, of all league football matches played by Tottenham Hotspur for the two consecutive season 1995/1996 and 1996/1997. We also outline any knowledge gained with respect to factors which effect the result of the football matches. The outcome of a football match is subject to many complex factors, and more than a little luck. It is in just this type of problem with many complex interacting factors that BNs excel. It is possible for a domain expert, in collaboration with a BN expert, to construct a network detailing the important relationships between the factors involved, and the node probability tables, NPTs, which detail the degree and direction of the effect of each element in the network. In this paper we will compare a BN constructed by an expert[1] in both BNs and the on pitch performance of Tottenham Hotspur during the appropriate seasons, with a naive BN, a BN learnt from statistical relationships in the data, a k-nearest neighbour implementation, and a decision tree. The aim here is to see how the expert constructed BN compares in terms of both predictive accuracy and explanatory clarity for the factors effecting the result of the football matches under investigation.

This paper is divided into six sections. Section 1 is an introduction to the paper. Section 2 discusses how the selection of which data about the football matches would be used to learn from was performed. Section 3 is a brief explanation of the learning techniques used. Section 4 details the results of the learners for each of the data sets used. Section 5 discusses the strengths and weaknesses of each of the learners in the context of the problem being studied. Section 6 looks at some possible directions of future work.

2 Selecting relevant information

There are a large number of factors which could effect the outcome of a football match, in either a positive or negative fashion[2], from the perspective of one of the teams involved. One of the difficulties in any investigation of the relationships involved in a given effect is that to a large extent the assumption of a particular model determines the attributes to study and predetermines the possible relationships that can be found. So, the act of of choosing which model and attributes to study sets a boundary on what can be discovered. The issue here is that it would be reasonable to assume that if the important relationships were already known then there would be no reason to undertake such a study in the first place. Practically the best that can be achieved is to assume what appears to be a reasonable model, determine a set of attributes

[1] N.E. Fenton is an acknowledged expert on BNs and a lesser known, but equally vociferous advocate of and expert on Tottenham Hotspur football club

[2] Interestingly some factors appear, according to many domain *experts*, to have either negative or non existent effect, such as the optical acuity of the referee.

to study, and then if the results achieved do not meet the required standard to amend the chosen model and/or set of attributes using any knowledge gained from the previous analysis. This situation may make it seem that the best initial approach would be to include any and all attributes. However, in general the accuracy of machine learning techniques decreases in the presence of irrelevant information, and the computational cost of the analysis increases with an increasing number of attributes. Thus, our first task is to choose a subset of the available information which appears to be directly relevant to predicting the outcome of the games.

2.1 Constructing an initial model

When approaching a new problem there are two techniques which are commonly used. The first assumes we have some idea how the situation under investigation works, we then attempt to construct a model which represents our best guess at the mechanisms involved. Using this model we can then select the attributes which contribute to the effect under investigation. If the results do not meet our requirements we can adjust the model taking into account anything that was learnt from the previous investigation and repeat until our understanding meets our needs. The second approach assumes little knowledge of the underlying mechanisms involved so rather than construct a formal model instead we look at all the attributes which we think could effect the outcome and try to determine those which have the most significant effect on the situation. This is still in effect the construction of an a priori model, but only a very informal one. However, these assumptions if wildly incorrect will cripple the learning process. In this paper we take the second approach. We do not develop a formal model of the football matches played by Tottenham Hotspur during the period in question, instead we use expert chosen factors and try to determine which are the significant ones.

2.2 Feature Subset Selection

Selecting the relevant features which we believe effect the outcome of a football game is just an instance of the general machine learning issue of Feature Subset Selection, FSS. There are two common automated approaches to FSS, namely those of filters and wrappers. With a filter the raw data is analysed using statistical features, the analysis is thus generally independent of the learning algorithm which is to be used. Those features, individually or in combination, which show the best statistical significance with respect to the problem are retained and others are removed. The advantage of filtering data in this fashion is that the filtering process is relatively simple and not particularly computationally expensive. In contrast a wrapper selects subsets of the available attributes and using the learning technique attempts to find the subset which in combination with the learner produces the best results. Wrappers have the advantage that they will work with effective knowledge of the bias of the learner, but are computationally more expensive than filters. It is also worth noting that choosing which attribute(s) to include or discard while attempting to find the optimum set, is an area of research and no generally good technique exists. A complete test of all possible combinations of attributes would generally be excessively computationally expensive. Our initial analysis is based on an expanded set of expert chosen features and uses no feature subset selection.

2.3 The expert model

The expert BN uses only a few features: the presence of absence of three players, Sherringham, Anderton and Armstrong; the playing position of Wilson represented by him playing in midfield or not; the quality of the opposing team; and whether the game is played in Tottenham Hotspur's home ground or away. We could have used only the same factors for all machine learners, but this seemed to be unnecessarily restrictive as it would not allow the other machine learners to highlight other factors effecting the outcome of the game. The expert constructed a model represented by the BN (see figure 15) which shows how the chosen factors effect the outcome of the football game.

2.4 The general model

We decided to allow the non-expert machine learners to use an expanded set of features compared with those used by the expert constructed BN and also to examine their behaviour using the expert chosen factors. The initial set of factors used for the non-expert learners are all the players who start the match playing for Tottenham Hotspur, the location of the match, being either at the Tottenham home ground or away, and an assessment of the strength of the opposing team as either, poor, average or good. The assessment of team strengths is the same as those used by the expert BN, based on expert judgement, but which also matches closely with the teams final league positions so would appear to be an accurate reflection of the average performance of the teams. The expert BN takes into account the playing position of a single player, Wilson, but we decided to remove this from the general model for the machine learners as the choice of the particular player is not something that would have occurred without the expert, and information on the playing positions of the other players for the games in question was not easily available.

2.5 Known model weaknesses

There are complications here to do with the players present during a game and their position. During a game players can be injured, substituted, have their playing positions changed, or be sent off. The solution chosen to deal with these issues was to use the information about only those players who started the game. So the initial analysis takes no account of how long a player remained on the pitch or their fitness or form at the time the match was played. Similarly Wilson's playing position could change during the course of the match, only his initial playing position was considered. Football teams are often analysed, and organised, in terms of sub-groups covering defence, midfield and attack. Our initial analysis takes no account of these factors, excepting Wilson's playing position, but these are elements that could be introduced if required.

In general terms this problem is not particularly easy from a machine learning perspective. There is not much data to go on. We have the results of two seasons games, a total of 76 matches and a total of 30 attributes, 28 players, location and team ranking. There were changes to the Tottenham Hotspur squad during this period. The simple convention of a player either playing or not was chosen to avoid having missing data entries with regards to squad changes as any player not part of the squad for a given match is accounted for as not playing. The attributes themselves have only a few values: home or away; played or not played; poor, average or good; and the result is win, draw or lose. In addition we know that there are other external factors which effect the outcome of a game. So, even in the best case we expect to have noise, in the form of external unrecorded factors missing in the data. Since players, except Wilson, are only considered from the point of view of playing or not playing, the effect of any player who was always, or nearly always, present will be ignored. This is because the learners can only compare the difference in the outcome of matches with a player present or absent. An example of this is the 1996/1997 season, the contribution of Sol Campbell to the performance of Tottenham Hotspur can't be judged by the type of analysis we use as he was present in all the 38 games.

It is also worth mentioning that both the expert and the general models we are looking at for determining the outcome of a given game are inherently asymmetric. In the case of Tottenham Hotspur we consider the particular players involved in any given match to be significant, however, for all other teams we only have a general rating for their relative strength in the league. In either case it seems obvious that either individual players are important, in which case we would ideally look at the individual players from each team, or they are not, in which case we can simply give a general rating for the teams and ignore the players taking part in any given game. A full model looking at the players for every team in the league would require much more work and since our focus is just on Tottenham Hotspur it is possible the asymmetric model will be acceptable.

3 Machine Learning Techniques

There are a large number of different machine learning techniques each with different strengths and weaknesses. Choosing which is the most appropriate technique often requires an understanding of both the problem domain and the different learning methods. The machine learners used in this analysis were:

MC4 Decision Trees Decision trees provide a visual representation of relationships which appear to effect the situation under investigation. In this instance the apparent effect of the Tottenham Hotspur players, the location in which the game is played, and the strength of the opposing team. Pruning is generally used to reduce the size of the tree and reduce overfitting, in this case the reduction in tree size was our primary aim. The pruning can be controlled in various ways. In this paper the confidence method of pruning was used.

Naive Bayesian Learner The Naive Bayesian Learner makes the simplifying assumption that all the attributes are independent. This leads to a very simple network structure shown in figure 1.

Data Driven Bayesian Learner The complex Bayesian learner as implemented by Hugin attempts to learn the structure of the network by looking at the correlation between the specified attributes. Once the structure has been determined data can then be used to determine the node probability tables. As with all learning techniques reuse of the same data will tend to magnify any bias in the chosen data set, but the small sample of available data, and its use for comparisons rather than absolute accuracy, means that it makes some sense to use it for both structure and NPT values. The strength of a correlation required to trigger the joining of two nodes can be adjusted. Hints can be given about expected relationships, although its not clear quite how these effect the final graph[3].

Expert Constructed Bayesian Network When expert knowledge of a given domain is to be represented as a BN the usual process is for the domain expert(s) and BN expert(s) to jointly construct the BN. Knowledge from the domain expert(s) is used to define the network structure. If a large amount

[3] In practise we were often unable to get what we believed to be likely hints to be accepted

of data is available then the NPTs can be directly learnt and then adjusted if required. However, when there is insufficient data to learn the NPTs these must also be obtained from the expert(s). In this case the domain expert and Bayesian expert happen to be the same person. It is probable that the expert constructed BN is less tuned to the specific data set being used than the other learning techniques. The expert will inevitably have drawn on more general knowledge of the team than the data supplies. This could mean that the accuracy of the expert constructed BN appears weaker for the test data than the other learners, but it might be more generally accurate for data outside of the test set.

K-Nearest Neighbour K-nearest neighbour learners use a *likeness* approach to prediction. That is, they look at the instances most like the test case and usually have some voting method by which the prediction is chosen. The usual measure of *likeness* is Euclidean distance as plotted on an n-dimensional graph where each dimension is one of the supplied attributes. K-nearest neighbour methods can be very accurate providing the test case lies within or close to the range of attributes in the initial data. So, in this case we'd expect it to be a fairly accurate method of prediction if the chosen attributes have a major effect on the outcome of the game. However, this type of learner does little to enhance our understanding of the relationship between the attributes and the outcome of the game.

All the learners used were part of the MLC++ package[4] apart from the complex Bayesian learner which was part of the Hugin tool[5], the Hugin tool was also used to *run* the expert constructed BN. What we're interested in is how accurate each system is in predicting results, and how much does each system add to our knowledge of the cause and effect aspects of the outcome of each game. Its also worth remembering that the machine learning techniques can only draw conclusions from how changes in the included attributes effect results. So, for example, if there is a pivotal player who plays in every match, then the machine learners will not be able to determine the effect of that player and he will effectively be discounted. While this should not effect the accuracy of prediction while the specific player is present, in their absence the accuracy of prediction could be significantly reduced.

[4] Version 2.01 of the MLC++ libraries was used, modified to run under the GNU/Linux operating system. All the MLC++ learners were used with their default settings.

[5] Version 6.1 of this tool was used for this paper

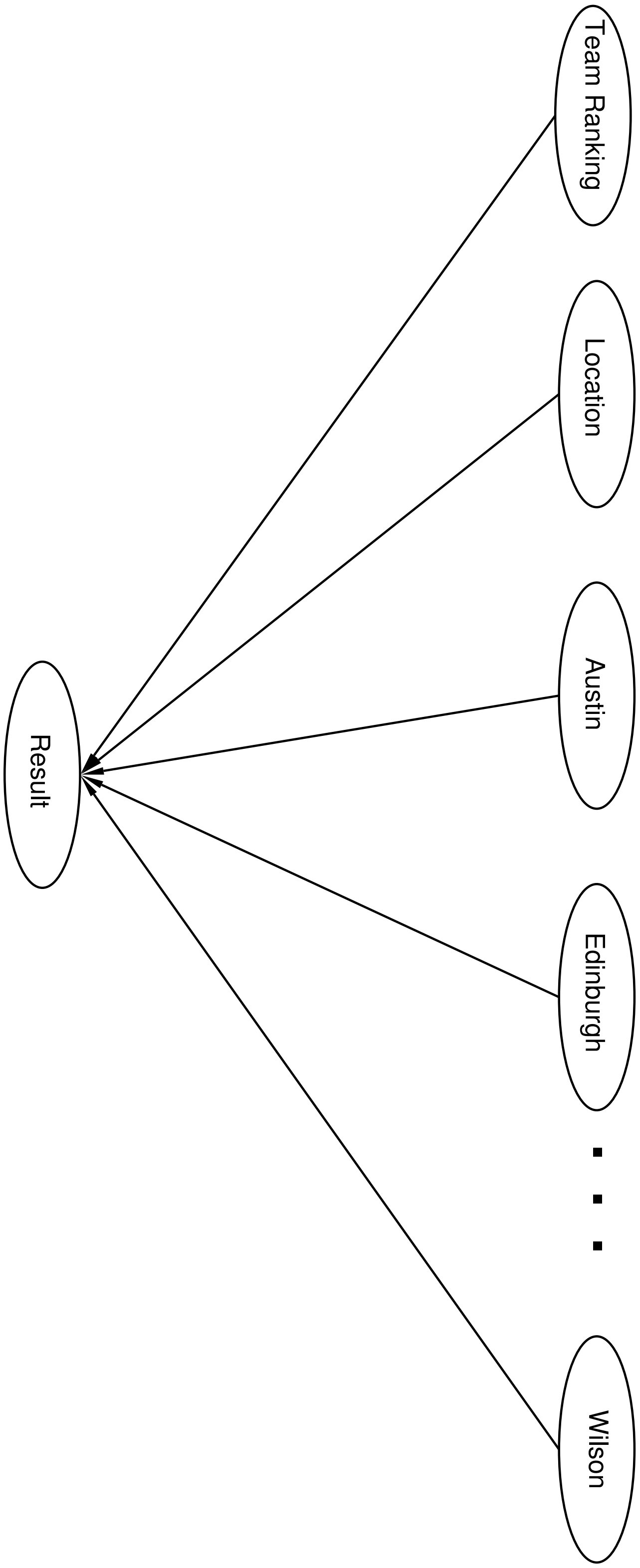

Figure 1. Naive Bayes network (abridged) for the Spurs 1995/1996 season

4 Results

In this section we look at any information provided by each learner about the factors effecting the outcome of the games, and we assess the relative accuracy of the predictions of each learner. We divided the match data into subsets so that some could be used for training and separate data used to check the accuracy of the learners. The data for each season was divided up into three groups of ten matches and one group of eight matches, organised chronologically. We assume that there could be significance to the order of the games and so we maintain the ordering of games and always organise the training so that the training data set is chronologically immediately before the test data set. The training data sets do not apply to the expert constructed BN as its structure and NPTs are already fixed. For comparison we also use each complete season's data for training and test set for the learners. This is really just to get some idea of the model that would be built by the learners for a complete season. In this case the use of the same data for both learning and test cases means that we will tend to overestimate the accuracy of the learners, with the exception of the expert BN. The machine learners were tested with both our general model data and with the data used by the expert BN.

The different learners do not all provide the same sort of prediction. The MC4 and KNN learners usually give a prediction in the form of an unqualified value from the possible range of values. Bayesian Networks do not make predictions in the same format as the MC4 or KNN learners, rather than supply an answer they supply a probability for each of the possible outcomes. This allows for a greater sensitivity of prediction in that the BN not only makes a prediction, but is also able to provide some idea of confidence in the prediction. We chose the outcome with the highest probability as the prediction of the BN. It is possible for a BN to predict that a number of outcomes are equally likely in a given situation. In terms of the correct or incorrect analysis we chose to assume any prediction where more than one outcome had the same joint highest probability as incorrect. This is potentially a harsh analysis as in many cases the information provided by the network would still be useful. An example of this might arise in betting on the outcome of a game, if the BN predicted Win 45% Draw 45% Loss 10% then this would indicate a likely win for an each way bet. However, such an analysis of the potential value of a shared highest probability prediction is beyond the scope of this paper.

In addition to the complete seasons data the learners, with the exception of the expert BN, were given a number of contiguous sets of thirty eight matches divided up into training and test data. The expert BN requires no training data. The sizes of the training and test data sets varied between eight and twenty eight (see table 1). There is one important caveat to the comparisons, the expert BN uses the playing position of Wilson as one of its attributes. This information is neither available to nor used by the other learners in this data set. This begs the question of why would we do this? Why use a different, albeit only slightly different, data sets for the expert BN and the other learners? There are at least two answers to this: a basic knowledge of football could easily lead to the selection of the attributes used with the exception of the playing position of Wilson, this attribute required expert knowledge and understanding to select and so it only relevant in the context of the expert BN; if we are going to include player positions for the learners in general it should be covered in a more comprehensive fashion than the arbitrary selection of one player. In case this mismatch of data sets was a crucial difference we also tested the machine learners with exactly the same data set as that used by the expert BN (see table 2). The results for both the general model data and the expert chosen data are similar and we concentrate on the differences between the results of the learners using general model and the expert BN.

4.1 The MC4 Learner

Decision tree learners like MC4 are good at dealing with relatively static situations, that is, situations in which the relationships between the various attributes are fixed. We were not sure how true this was of the Tottenham Hotspur team, and its performances, over the period being examined. It is certainly true that the starting line-up of the team and the members of the squad vary through the test period and it was thus possible that the performance of this learner could vary wildly. During the tests the performance of the learner while not astounding did at least remain reasonable consistent suggesting that while the changes may have degraded its performance they did not cause any extreme loss of accuracy. This is reasonable in context as you would expect any team to attempt to manage its own changes in a manner that allowed its players to adapt. The overall classification error of the MC4 learner for disjoint training and test data sets in the general model was 69.81% and 61.35% for the expert chosen data.

4.1.1 Complete Seasons

The basic tree produced by MC4 when looking at the general model data for the 1995/1996 season is a fairly simple tree using only 6 of the available 30 attributes, the players Dozzell, Campbell and Nethercott, the location of the game and the opposing team ranking. The tree shows Dozzell as a key player [6],

[6] It is interesting to note that after seeing this analysis the expert stated that while he suspected Dozzell was a key player this was not the general opinion at that time and he thus left Dozzell out of the expert BN

when present Tottenham Hotspur can be expected to beat any bottom ranking team, and to draw with middle or top ranking teams. When Dozzell is absent the location of the match is important, an away game with Nethercott playing would be expected to be a loss, without Nethercott a draw would be expected. A home game without Dozzell and Campbell would be a loss, but with Campbell and without Fox would be a win. Lastly a home game without Dozzell, but with Campbell and Fox would be a loss.

Figure 2 shows an overview of the relationships found by MC4 with indications of the error of each node and the number of cases to which is applies. The error estimates are based on the same data as that used to generate the tree, namely the match results for the 1995/1996 season. Now for the 1995/1996 season the most common result was a win (2) with a total of 16 wins from 38 games. In the simplest case if we always assumed a win regardless of any attributes we'd have an error of 57.89%. So the MC4 analysis give a reduction in the error of 34.57% over the most common outcome using the general model and a reduction of 23.68% using the expert chosen data.

An analysis of the 1996/1997 seasons matches produces the tree shown in figure 3. This is slightly more complex than the tree for the 1995/1996 season, using 8 rather than 6 attributes. In this tree the presence of the unlucky Mr Nethercott or Mr Mcveigh would signal a loss. Against a high rated team we'd expect a loss, against a low ranking team with Mr Anderton or Mr Fox playing we'd expect a win. Against a medium ranking team at home with Mr Wilson playing we'd expect a draw or otherwise a win, and away with Mr Anderton playing we'd expect to loose or otherwise to draw. The most common result for Tottenham Hotspur in the 1996/1997 season was a loss with 18 games being lost. Assuming the most common result of a loss we'd have an error of 52.63%. We can see that the MC4 analysis gives a reduction in the error of 31.58% using the general model and a reduction of 21.05% using the expert chosen data. Thus it appears that MC4 is consistent in providing a reduction in the classification error of about 31.5% compared with the most common classification.

4.1.2 Separate Training and Test Data - Single Season

The performance of the MC4 learner was, as expected, less impressive when it was only given part of a season's data and used to predict the remainder. The trees for the first ten, twenty and thirty games in the 1995/1996 season are shown in figures 4, 5, and 6 respectively. Similarly the trees produced by the MC4 learner for the first ten, twenty and thirty games in the 1996/1997 season are shown in figures 7, 8, and 9 respectively. The performance of the MC4 learner in this situation was generally quite poor. The classification error for the tests using general model data from 1995/1996 season was 9.26% worse

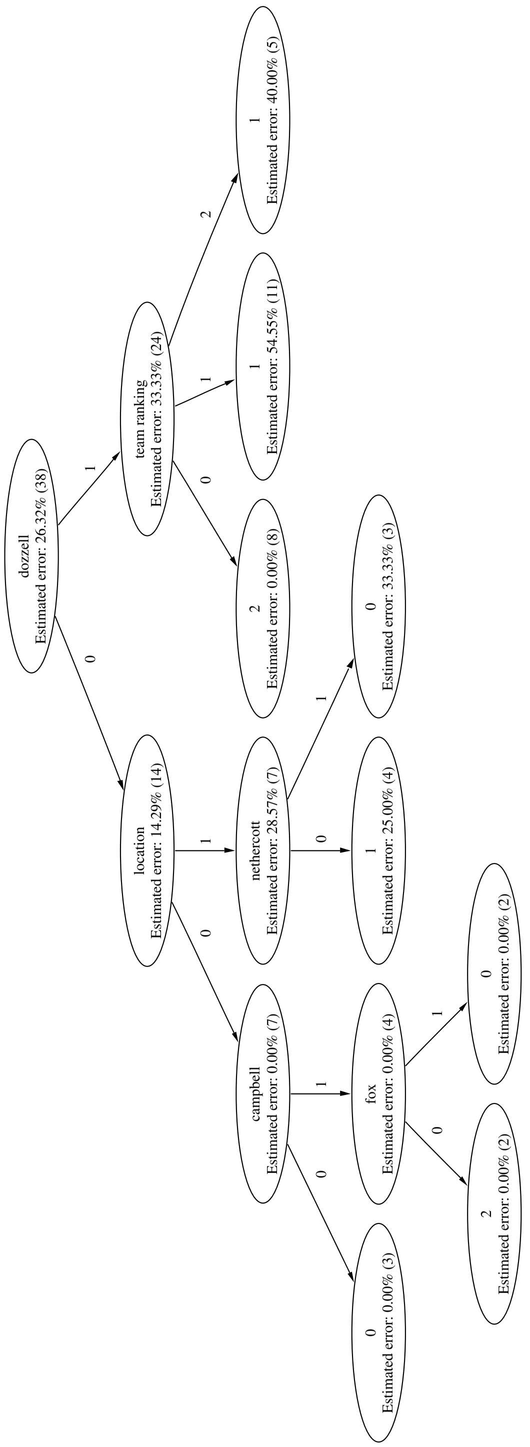
dozzell
Estimated error: 26.32% (38)
0
1
location
Estimated error: 14.29% (14)
team ranking
Estimated error: 33.33% (24)
0
1
0
1
2
campbell
Estimated error: 0.00% (7)
nethercott
Estimated error: 28.57% (7)
2
Estimated error: 0.00% (8)
1
Estimated error: 54.55% (11)
1
Estimated error: 40.00% (5)
0
1
0
1
0
Estimated error: 0.00% (3)
fox
Estimated error: 0.00% (4)
1
Estimated error: 25.00% (4)
0
Estimated error: 33.33% (3)
0
1
2
Estimated error: 0.00% (2)
0
Estimated error: 0.00% (2)

Figure 2. Decision Tree for the general model 95/96 season with error estimates

than choosing the most common result, and the same tests for the 1996/1997 season showed an increase in the error of 9.25% compared with choosing the most common result. The learner faired slightly better with the expert chosen data giving an increase in error of 7.41% and 5.55% for the 1995/1996 and 1996/1997 seasons respectively. The performance of the learner did not seem

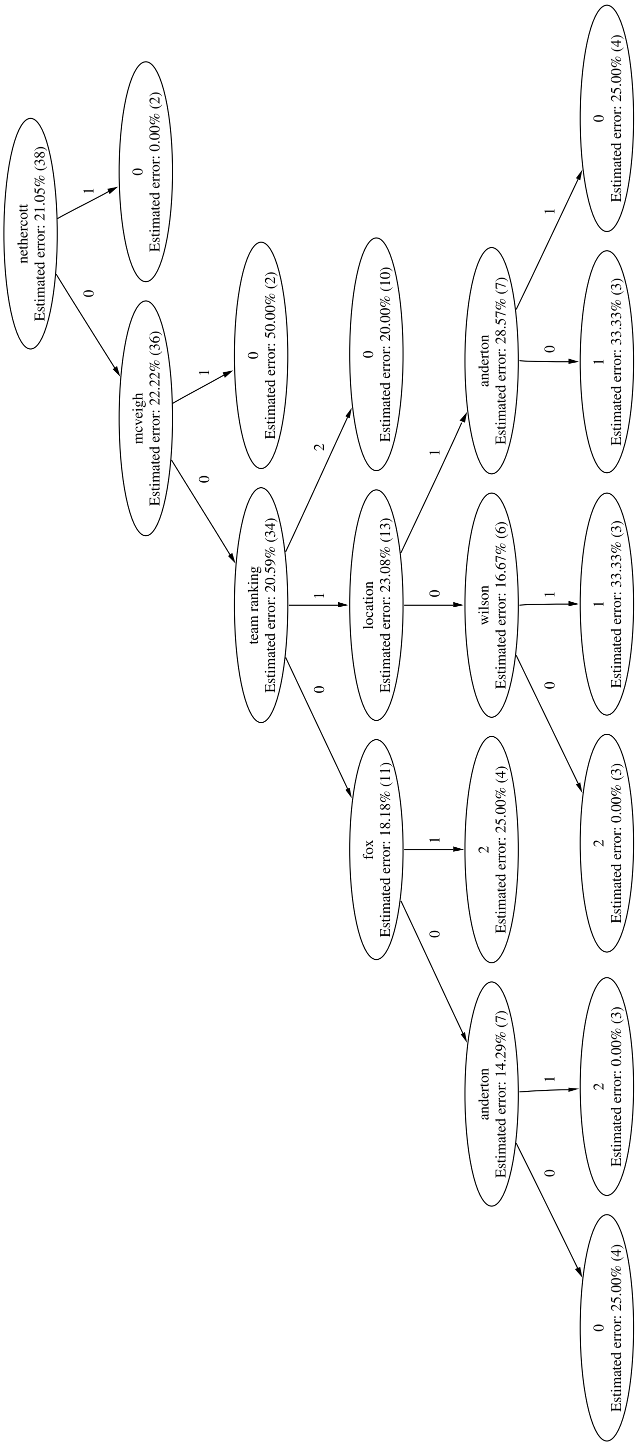
nethercott
Estimated error: 21.05% (38)
0
1
mcveigh
Estimated error: 22.22% (36)
0
Estimated error: 0.00% (2)
0
1
team ranking
Estimated error: 20.59% (34)
0
Estimated error: 50.00% (2)
0
1
2
fox
Estimated error: 18.18% (11)
location
Estimated error: 23.08% (13)
0
Estimated error: 20.00% (10)
0
1
0
1
anderton
Estimated error: 14.29% (7)
2
Estimated error: 25.00% (4)
wilson
Estimated error: 16.67% (6)
anderton
Estimated error: 28.57% (7)
0
1
0
1
0
1
0
Estimated error: 25.00% (4)
2
Estimated error: 0.00% (3)
2
Estimated error: 0.00% (3)
1
Estimated error: 33.33% (3)
1
Estimated error: 33.33% (3)
0
Estimated error: 25.00% (4)

Figure 3. Decision Tree for the general model 96/97 season with error estimates

to improve with increasing amounts of training data. The trees built by MC4 with increasing data can be seen to develop towards that built with the full season's data. Anderton appears in all except the last tree which demonstrates a weakness of the tree system. Anderton played in only six games in the 1995/1996 league season, three in the first period and three in the last period.

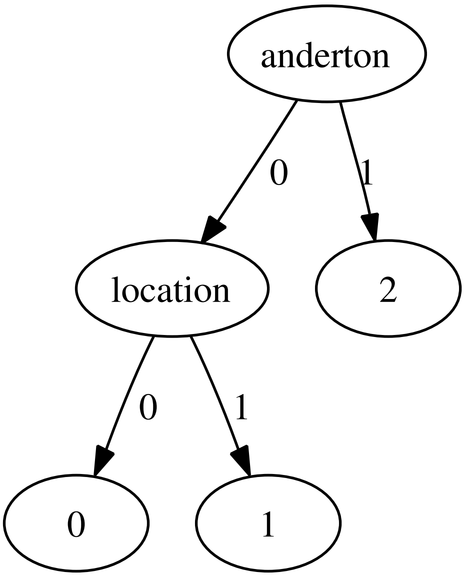

Figure 4. Tree from MC4 for period 1 of the general model 95/96 season

So in effect his three games early in the year are given disproportionate weight.

4.1.3 Separate Training and Test Data - Cross Seasons

Figures 10 and 18 show the trees produced by the MC4 learner for two of the five cross season periods, the tree for the full season comparison remains that shown in figure 2. Once again the performance of the learner over all five cross season periods, for the general model, was quite poor on this task often predicting the outcomes of the test games less well than assuming the most common result from the training games. The classification error for the general model averaged over all the cross season tests was 6.37% higher using the MC4 learner than assuming the most common value from the test data. The tree covering the end of the 1995/1996 season, period 4, and the beginning of the 1996/1997 season, period 1, is the largest of the trees for any two period group. This may indicate that significant changes take place between seasons, which would not be contradicted by the slight drop in performance of cross season tests compared to similar intra-season tests. There is also a drop in the predictive ability of the most common test result which means that overall for the cross seasons tests the classification error from the MC4 learner was 6.37% worse than that from choosing the most common test result. Over the same period the expert chosen data gave a better result with an average reduction in the error of 13.48%.

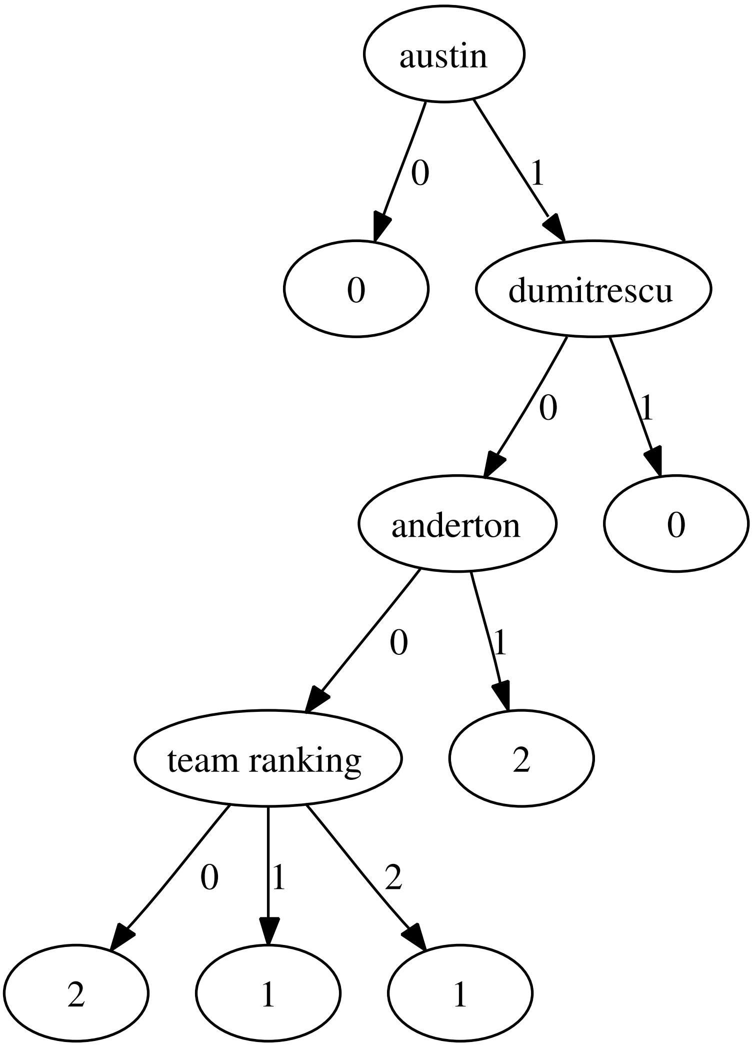

Figure 5. Tree from MC4 for the general model periods 1 and 2 of the 95/96 season

4.2 Naive Bayesian Learner

While the attributes of the problem do not adhere to the strict independence assumption of the naive Bayesian learner we would expect there to be a reasonable match and thus for this learner to perform relatively well. This is reflected in that for non-overlapping training and test data sets on the general model this learner came second overall with a classification error of 61.19%. Interestingly on the expert chosen data the naive Bayesian learner only came in fifth best with a classification error of 64.26% suggesting that perhaps the expert was making use of dependencies in the data to aid choosing attributes.

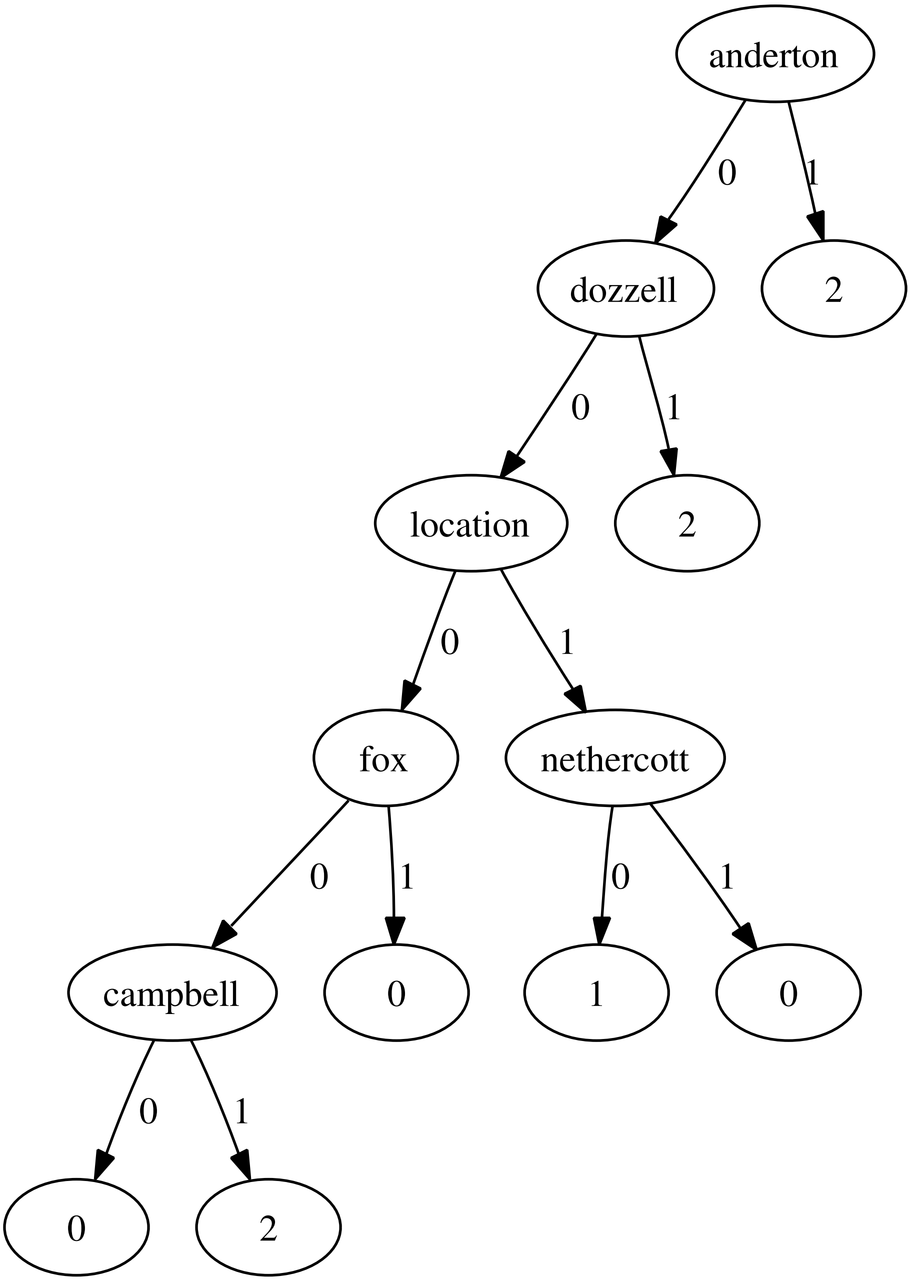

Figure 6. Tree from MC4 for periods 1, 2, and 3 of the general model 95/96 season

4.2.1 Complete Seasons

For the 1995/1996 season the Naive Bayesian learner correctly predicted the result of 26 and 22 of the 38 games in the general and expert models respectively. This is a reduction in the classification error of about 26.31% and 15.78% and compared with assuming the most common result. While this is not quite as good as the performance of the MC4 classifier it is still quite impressive. The naive Bayesian classifier gives no direct indication of the importance of any given attribute. However, looking at the NPT for the classi-

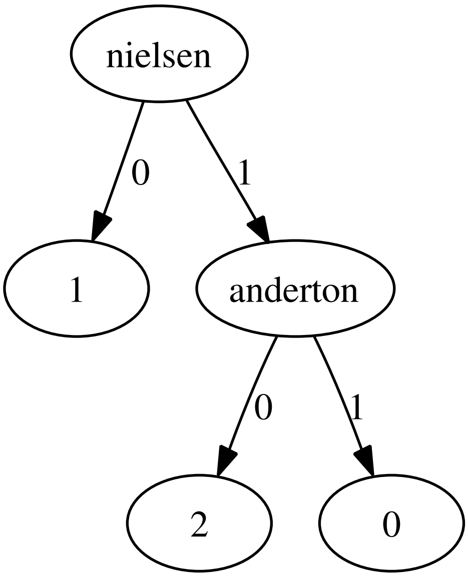

Figure 7. Tree from MC4 for period 1 of the general model 96/97 season

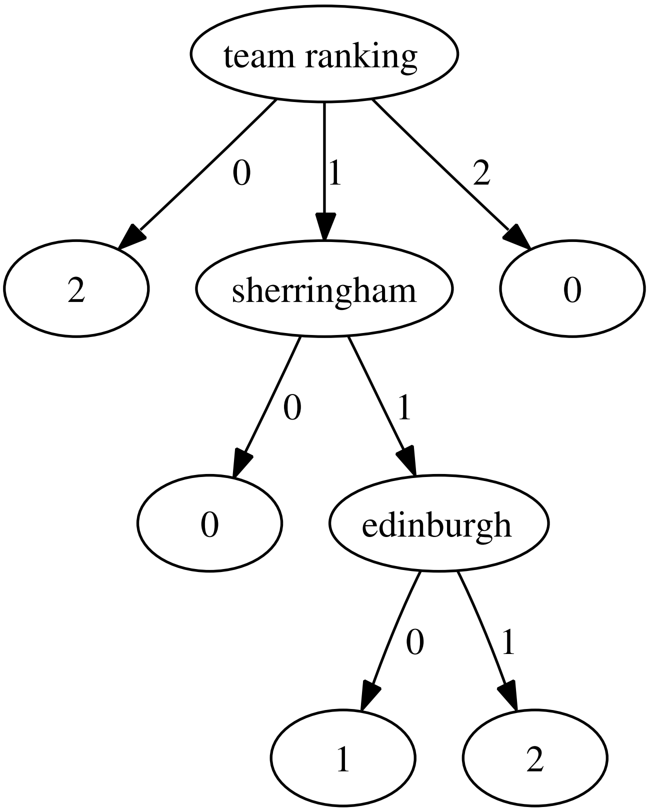

Figure 8. Tree from MC4 for periods 1 and 2 of the general model 96/97 season

fier in the general model we can see that the six most significant attributes in descending order are: Team Ranking, Dozzell, Edinburgh, Anderton, Dumitrescu and Calderwood. There is some, limited, agreement between MC4 and the naive Bayesian learner on the significant attributes, they agree on the

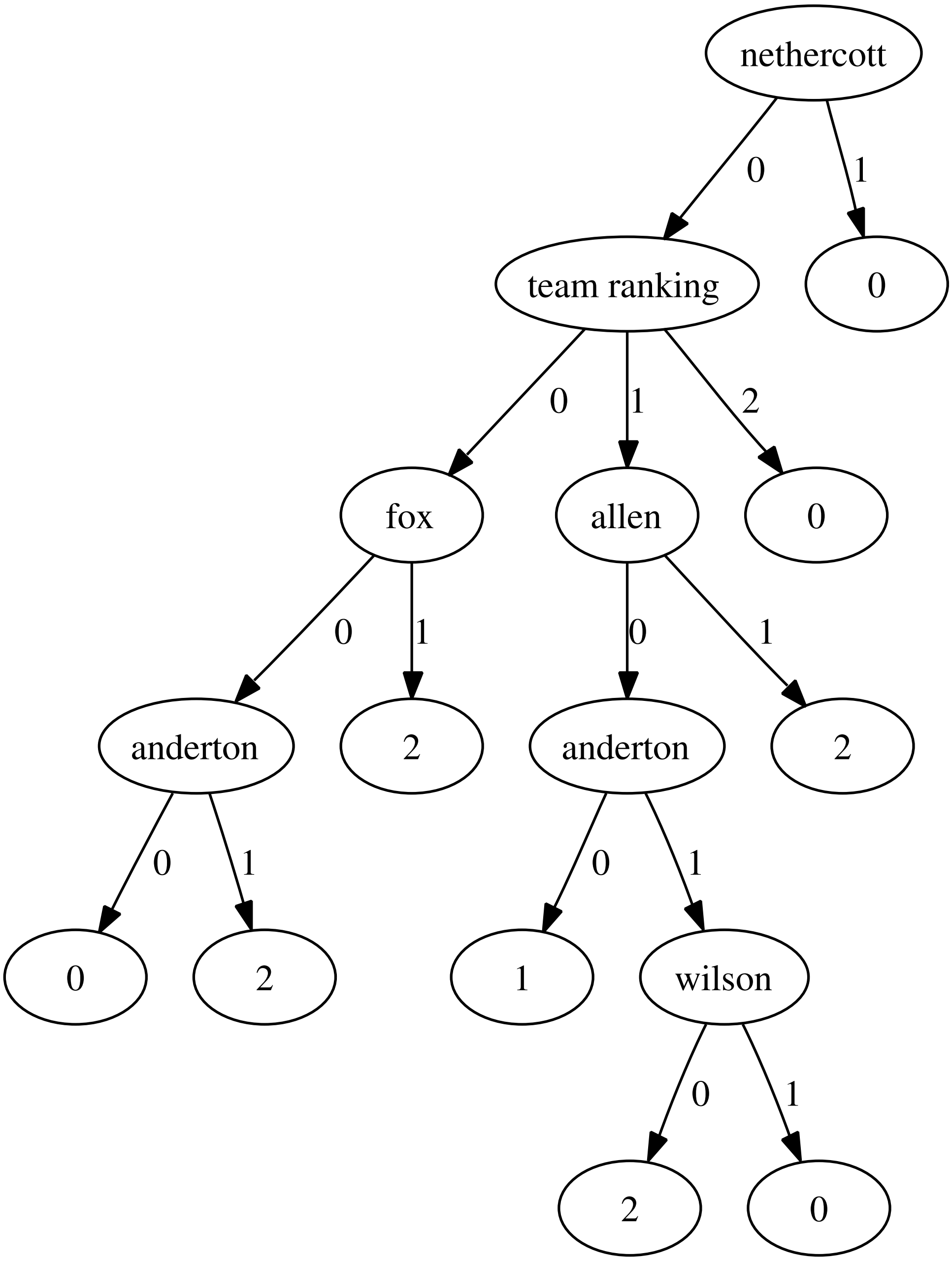

Figure 9. Tree from MC4 for periods 1, 2, and 3 of the general model 96/97 season

two most important of the thirty attributes for the 1995/1996 season. For the 1996/1997 season the Naive Bayesian learner correctly predicted the result of 31 and 25 of the 38 games for the general and expert models respectively. This is a reduction in the classification error of about 34.21% and 18.42% compared with the assuming most common result. So the performance of the naive Bayesian learner is comparable to that of MC4 for the 1996/1997 season. However, for the general model the Naive Bayesian learner ranks Allen as the second highest weighted attribute and this attribute does not appear at all in the MC4 decision tree.

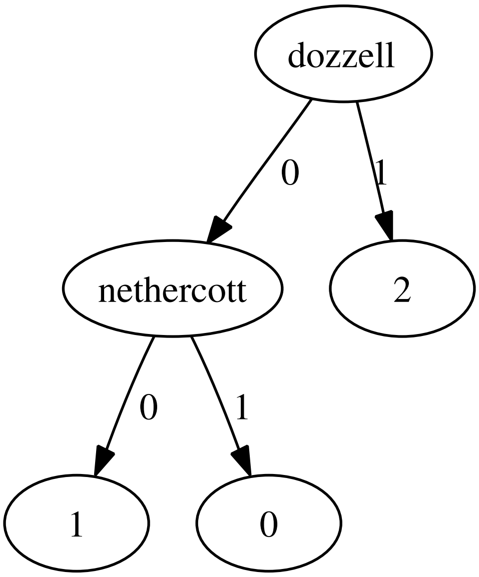

Figure 10. Tree from MC4 for periods 2 and 3 of the general model 95/96 season

4.2.2 Separate Training and Test Data - Single Season

When given disjoint training and data sets the performance of the naive Bayesian learner also decreased as expected. The results for the 1995/1996 season showed the average classification error to be 7.41% and 3.70% higher for the general and expert data sets respectively than that obtained by using the most common test result. However, for the 1996/1997 season the general model classification error was 7.41% lower than that achieved by the most common value classifier while that for expert data set model was 3.70% worse than the most common result. Most classifiers achieved better results for the 1996/1997 season than the 1995/1996 season which may indicate greater stability in the team in the later season.

4.2.3 Separate Training and Test Data - Cross Seasons

The cross season results for the naive Bayesian learner were roughly comparable to its in-season results. Overall it achieved a classification accuracy of 33.09% and 35.29% for the general and expert model which only bettered the most common classifier by 0.98% and 3.18% respectively. Ignoring the case using the same training and test data for the complete seasons, the naive Bayesian learner came out second best overall on the general model and fifth overall on the expert model.

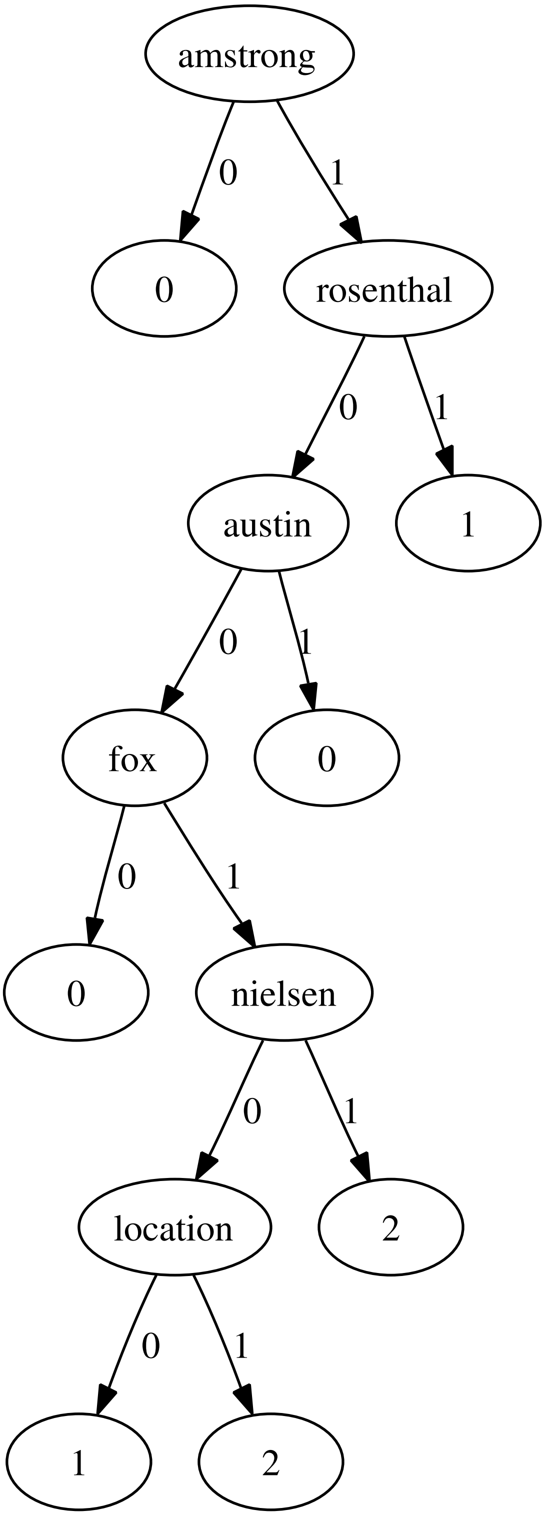

Figure 11. Tree from MC4 for the general model periods 4 and 1 of the 95/96 and 96/97 seasons

4.3 Data Driven Bayesian Learner

The BNs for the data driven Bayesian learner were generated using the structural learning wizard from the Hugin Developer version 6.1 program. The process used was to run the program using an initial `Level of Significance` of 0.1. If no link directed to the **result** node was formed the process was rerun doubling the `Level of Significance` until a network with at least one link directed to the **result** node was achieved. Since in this problem all of the nodes except the **result** node have their values specified any nodes in the network with no links directed to the **result** node were removed. The remaining network was used for the testing. The overall classification error of the various learnt networks for disjoint training and test data sets was 67.69% and 67.38% for the general and expert models respectively.

4.3.1 Complete Seasons

Using the structural learning wizard in Hugin Developer 6.1 a network was learned using the data for the 1995/1996 season. With a `Level of Significance` of 0.1 and removing all the nodes not directly connected to the **result** node the network, for the general model, shown in figure 12 was obtained. This network includes rather odd dependencies between the players. it is possibly significant that the two nodes with the greatest number of dependencies are **dozzell** and **wilson**. We know from our other analysis that these are two important players, but with the network as show we are unable to usefully include them. A crucial feature of this network is the **result** node has no children and its only parent is the **team_ranking** node. Since in this problem the data for all the nodes except **result** are specified, we can infer the outcome of the game simply by knowing the quality of the opposition, the other attributes become irrelevant if the **team_ranking** is specified. What we are seeing here is a general issue when constructing BNs from data. If the nodes represent items whose value is usually known, then there is limited scope for induction in the network. See section 5 for further discussion on this issue. Using the quality of the opposing team it is possible to correctly predict the outcome of 21 of the 38 games for the 1995/1996 season. This amounts to a reduction in the classification error of 13.15% over choosing the most common result. Using the expert chosen data the network obtained was that shown in figure 13. The network based on the expert selected data correctly predicted 23 of the 38 games for the season a reduction in error over choosing the most common result of 18.42%. The Hugin BN learnt network for the general model 1996/1997 season is shown in figure 14. This is exactly the same as the learnt network for the expert selected data. This tiny network, is, after trimming away any redundant nodes and links, all that could reasonably be extracted from the Hugin software using the data for the 1996/1997 season, to provide a prediction of the result. This particular network was extracted using a `Level of Significance` of 0.1 for both models. It is interesting to note that this is effectively the same network as that constructed using the general data for the 1995/1996 season, that is, the **result** node has no children and its only parent is the **team_ranking** node.

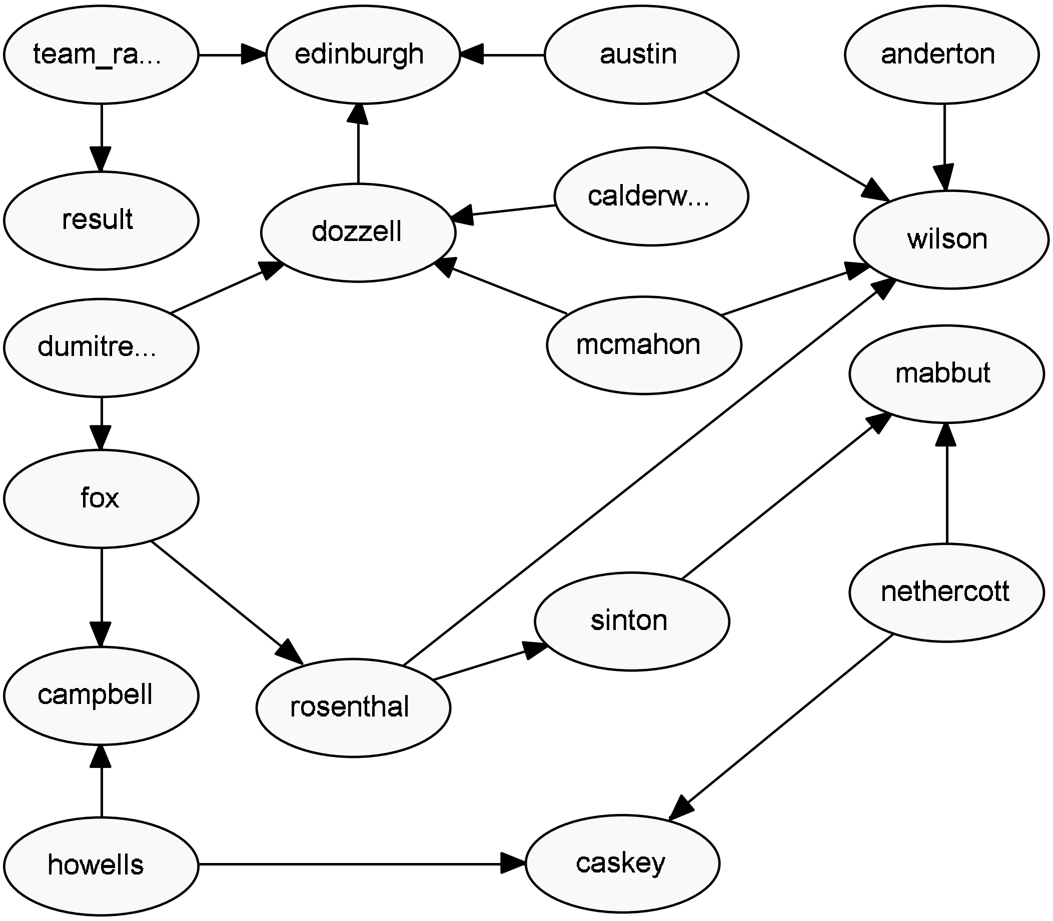

Figure 12. Learnt BN for the general model 95/96 season with `Level of Significance` 0.1

4.3.2 Separate Training and Test Data - Single Season

The networks learnt for the season 1995/1996 and 1996/1997 training sets are shown in Appendix A. It is interesting to note that for the general model the attributes chosen by the Hugin learner for the periods in 1995/1996 season are a subset of those chosen by the MC4 learner for the same periods. There is a less strong relationship for the general model between the chosen attributes of the Hugin and MC4 learners for the 1996/1997, but still a lot of shared attributes. This is reasonable given that both learners are presumably choosing attributes with a strong correlation with the result.

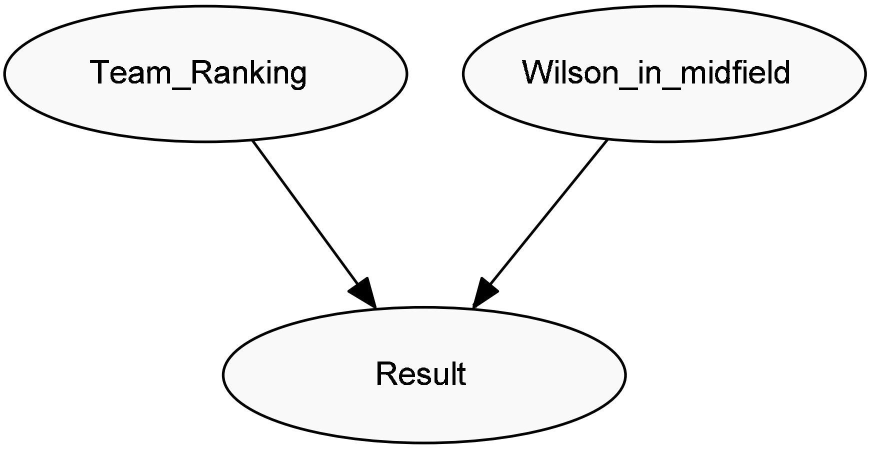

Figure 13. Learnt BN for the expert model 95/96 season with `Level of Significance` 0.1

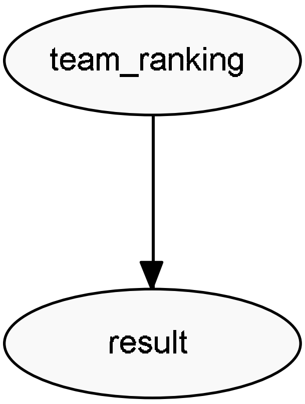

Figure 14. Learnt BN for the general model 96/97 season with a `Level of Significance` 0.1

4.3.3 Separate Training and Test Data - Cross Seasons

The networks learnt for the cross season training and testing are shown in Appendix A. Similar to the intra-season networks there is a striking similarity between the attributes chosen by the Hugin learner and the MC4 algorithm for the general model. All of the learnt BNs reduce to very simple networks. This suggests that either the learning algorithm selects simple networks or that a few strong correlations exist. A learnt network for the general model non test periods 3 and 4 of the 1996/1997 season, figure 30, show a slightly larger network suggesting that it was the strength of correlations rather than the learner which determined the small networks. We encountered a problem with the network produced by the Hugin learner for the period 2 and 3 general model data in the 1995/1996 season. This network crashed when we tried to run it so no results could be obtained for this training period.

4.4 Expert Constructed Bayesian Network

The expert constructed BN is shown in figure 15 below. Unlike the other learners this one does include reference to a single player's position of play. An important point about the expert constructed BN is that it has nodes, **Attack**, **Spurs_Quality** and **Performance**, which do not directly represent any of the supplied attributes or the result. These nodes are a result of the model the expert has built to describe the performance of Tottenham Hotspur and define more detailed relationships between the attributes and the result than those provided by the other learners. Another difference with the expert constructed BN is that is does not use the supplied training data for any of the tests. The structure of the network and the value of the NPTs has all been fixed by the expert. This means it is unable to take into account any change which may occur outside of the expert chosen attributes. Despite these limitations the expert BN was the most accurate predictor of the outcome of the Tottenham Hotspur games with a classification error over the disjoint training and test data sets of 40.79%.

4.4.1 Complete Seasons

The expert BN is the only learner we would not expect to appear overly accurate when looking at a complete season's data for both training and testing as it does not use training data, and is therefore not subject to the overfitting that we would expect to effect the others learners. The expert BN did better than the most common value predictions for both the 1995/1996 and 1996/1997 seasons with a classification error of 40.79%. However, on this task it was outperformed by all the other learners. The expert BN was the best classifier overall for the disjoint training and test data sets with a classification error of 40.79% and the best classifier on all bar one of the individual disjoint subsets.

4.4.2 Separate Training and Test Data - Single Season

The expert BN had its poorest performance on the data for the 1995/1996 season. This is not difficult to understand given that: Sherringham played in every match for Tottenham Hotspur during that season; Anderton played only 6 matches in the season; Armstrong played in all bar one game of the season; Wilson only played in midfield in 3 games in the season. Thus given its chosen set of attributes there was little variation the expert BN could produce over the 1995/1996 season. However, it is worth noting that with classification errors of 50.00% and 40.74% for the 1995/1996 and 1996/1997 seasons respectively, it was still the best classifier for the intra-season data.

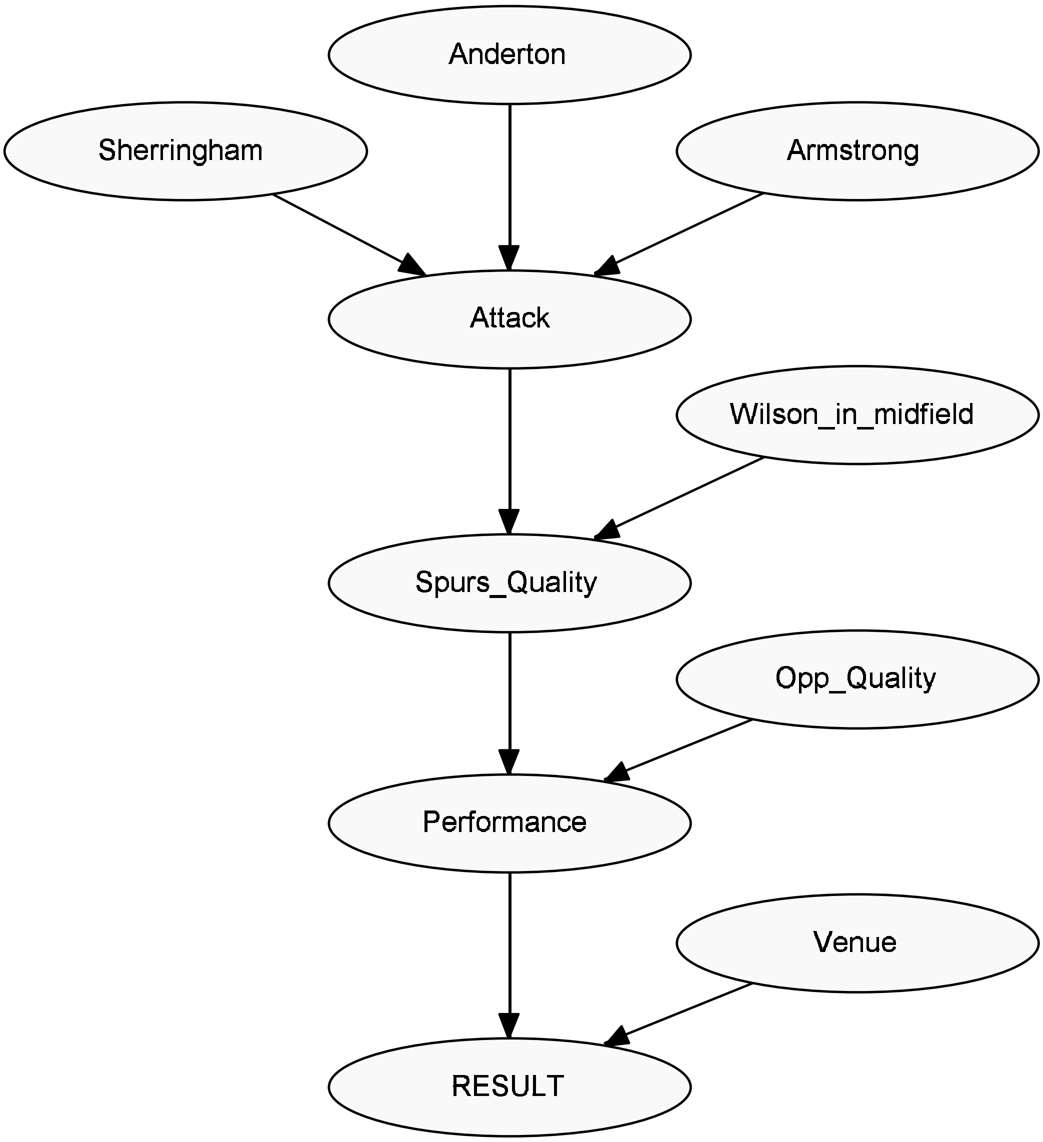

Figure 15. Expert constructed BN for Tottenham Hotspur's performance

4.4.3 Separate Training and Test Data - Cross Seasons

The expert BN produced the best results of any of the classifiers for every one of the cross season test periods. Since it does not use the training data, any changes that occur between season not involving its key attributes are ignored. This is really a case of the expert being able to select the key features, FSS, and thus remove any other features which could adversely effect its predictions. However, in the case of something like a football team where over the course of a few seasons all the players may change it does potentially limit the useful lifetime of any given expert constructed BN.

4.5 K-Nearest Neighbour

The IB classifier from the MLC++ library is a version of the k-nearest neighbour algorithm. In effect the KNN algorithm constructs a graph with as many dimensions as we have attributes. We are not aware of an easy to interpret representation for graphs of high dimension so we provide no visual representation of the model constructed by this learner. Overall for the disjoint training and test data sets KNN proved to be an average performer with a classification error of 65.02% and 62.94% for the general and expert models respectively. However, as expected with the same training and test data provided KNN performs exceptionally.

4.5.1 Complete Seasons

Using the KNN algorithm as a 3-nearest neighbour classifier on the data for the 1995/1996 season it correctly predicts the result of 37 of the 38 games for the general model and an average of 29.5 games for the expert model data. This amounts to an error reductions of 55.26% and 28.94% compared with assuming the most common result and a classification errors of 2.63% and 28.95% for the general and expert models respectively. For the 1996/1997 season the KNN algorithm again correctly predicts the result of 37 of the 38 games for the general model and 32 of 38 games for the expert model. This gives classification errors of 2.63% and 15.79% and error reductions of 50.00% and 36.84% compared with assuming the most common result.

4.5.2 Separate Training and Data - Single Season

With separate training and test data sets the performance of the KNN learner dropped dramatically, and interestingly providing more training data did not seem to improve its performance. The overall classification error for the 1995/1996 season for both general and expert models was 61.11% and for the 1996/1997 season it was 68.52% and 66.67% for the general and expert models respectively.

4.5.3 Separate Training and Data - Cross Seasons

Cross season performance was generally a bit weak for the KNN learner. This might be because of an inability to filter out unimportant attributes involved in cross season changes. However, with an overall classification error for the cross season test periods of 65.44% for the general and 61.03% for the expert model the KNN classifier still outperformed the most common classifier if only by 2.45% and 6.86%.

4.6 Validation and Overfitting

In this problem we know that we have both noisy data and unknown, possibly random, effects. So, we would not expect to get a completely accurate classification for the outcome of a given game. We have only a small sample of data a situation that will tend to cause a strong bias towards the specific data set. However, what we are interested here is in the relative performance of each learner and, since each learner would be expected to generate the same dataset bias, the comparisons should be valid. We also have a situation in which the underlying mechanisms that determine the performance of the football team, the members of the team, their playing positions, fitness and tactics can all change. We would not expect our chosen attributes to account for all of the likely variations so its difficult to determine what a reasonable level of predictive accuracy to expect.

4.7 Predictive Accuracy

Tables 1 and 2 show the relative accuracy of the different learners in predicting the outcome of the games for the 1995/1996 and 1996/1997 seasons using the general and expert model data respectively. When using the same training and test data for the complete seasons all of the learners perform significantly better than the most common assumption with KNN as the best performer. When disjoint training and test data sets were used the performance of the KNN learner dropped significantly and the expert BN outperformed all the other learners. The learners generally performed similarly with both the general and expert chosen data sets. This would generally indicate that the superior performance of the expert BN was not due to the specific data set chosen for it.

5 Conclusions

The process of machine learning, and learning in general, provides us with two tangible benefits, understanding and prediction. While it is true that the better our understanding the better we should be able to make predictions, it is possible to make accurate predictions with limited understanding, as with a Chinese room. We can treat these as qualitative and quantitative results from the learning process. The understanding we gain from the learning process allows us to construct models which reflect what we have learned about

Table 1
Comparison of learner accuracy with general model data.

	Number of correct predictions by learner					
Train Period - Test Period	Most Common	MC4	naive BN	Hugin BN	expert BN	KNN
95/96 - 95/96 season	16(42.11%)	28(73.68%)	26(68.42%)	21(55.26%)	20(52.63%)	37(97.37%)
96/97 - 96/97 season	18(47.37%)	30(78.95%)	31(81.58%)	26(68.42%)	25(65.79%)	37(97.37%)
Average for full Seasons	17(44.74%)	29(76.32%)	28.5(75.00%)	23.5(61.84%)	22.5(59.21%)	37(97.37%)
period 1 - period 234 95/96	12(42.86%)	8(28.57%)	9(32.14%)	8(28.57%)	14(50.00%)	12(42.86%)
period 12 - period 34 95/96	7(38.89%)	6(33.33%)	6(33.33%)	3(16.67%)	10(55.56%)	7(38.89%)
period 123 - period 4 95/96	2(25.00%)	2(25.00%)	2(25.00%)	2(25.00%)	3(37.50%)	2(25.00%)
Sum for 1995/1996 periods	21(38.89%)	16(29.63%)	17(31.48%)	13(24.07%)	27(50.00%)	21(38.89%)
period 1 - period 234 96/97	11.5(41.07%)	10(35.71%)	13(46.43%)	11(39.29%)	19(67.86%)	11(39.29%)
period 12 - period 34 96/97	7.5(41.67%)	7(38.89%)	10(55.56%)	3(16.67%)	10(55.56%)	5(27.78%)
period 123 - period 4 96/97	5(62.50%)	2(25.00%)	5(62.50%)	2(25.00%)	3(37.50%)	1(12.50%)
Sum for 96/97 periods	24(44.44%)	19(35.19%)	28(51.85%)	16(29.63%)	32(59.26%)	17(31.48%)
period 23 95/96 - period 4/1 95/97	6(33.33%)	4(22.22%)	6(33.33%)	unavailable	9(50.00%)	7(38.89%)
period 234 95/96 - period 1 96/97	4(40.00%)	2(20.00%)	4(40.00%)	3(30.00%)	6(60.00%)	3(30.00%)
period 34 95/96 - period 12 96/97	8(40.00%)	6(30.00%)	8(40.00%)	11(55.00%)	15(75.00%)	7(35.00%)
period 4 95/96 - period 123 96/97	6(20.00%)	8(26.67%)	6(20.00%)	10(33.33%)	22(73.33%)	8(26.67%)
period 4/1 95/97 - period 23 96/7	6.67(33.33%)	7(35.00%)	8(40.00%)	7(35.00%)	16(80.00%)	7(35.00%)
season 95/96 - season 96/97	13(34.21%)	8(21.05%)	13(34.21%)	20(52.63%)	25(65.79%)	15(39.47%)
Sum for cross Season periods	43.67(32.11%)	35(25.74%)	45(33.09%)	51(43.22%)	93(68.38%)	47(34.56%)
Overall Average Percentage	40.05%	41.72%	47.86%	39.69%	59.21%	50.58%
Overall disjoint training/data	38.48%	30.19%	38.81%	32.31%	59.21%	34.98%

Table 2
Comparison of learner accuracy with expert model data

	Number of correct predictions by learner					
Train Period - Test Period	Most Common	MC4	naive BN	Hugin BN	expert BN	KNN
95/96 - 95/96 season	16(42.11%)	25(65.79%)	22(57.89%)	23(60.53%)	20(52.63%)	27(71.05%)
96/97 - 96/97 season	18(47.37%)	26(68.42%)	25(65.79%)	26(68.42%)	25(65.79%)	32(84.21%)
Average for full Seasons	17(44.74%)	25.5(67.11%)	23.5(61.83%)	24.5(64.47%)	22.5(59.21%)	29.5(77.63%)
period 1 - period 234 95/96	12(42.86%)	8(28.57%)	7(25.00%)	8(28.57%)	14(50.00%)	9(32.14%)
period 12 - period 34 95/96	7(38.89%)	5(27.78%)	9(50.00%)	0(0.00%)	10(55.56%)	8(44.44%)
period 123 - period 4 95/96	2(25.00%)	4(50.00%)	3(37.50%)	2(25.00%)	3(37.50%)	4(50.00%)
Sum for 1995/1996 periods	21(38.89%)	17(31.48%)	19(35.19%)	10(18.52%)	27(50.00%)	21(38.89%)
period 1 - period 234 96/97	11.5(41.07%)	11(39.26%)	12(42.86%)	13(46.43%)	19(67.86%)	7(25.00%)
period 12 - period 34 96/97	7.5(41.67%)	6(33.33%)	8(44.44%)	6(33.33%)	10(55.56%)	8(44.44%)
period 123 - period 4 96/97	5(62.50%)	4(50.00%)	2(25.00%)	2(25.00%)	3(37.50%)	3(37.50%)
Sum for 1996/1997 periods	24(44.44%)	21(38.89%)	22(40.74%)	21(38.89%)	32(59.26%)	18(33.33%)
period 23 95/96 - period 4/1 95/97	6(33.33%)	7(38.89%)	7(30.89%)	7(30.89%)	9(50.00%)	8(44.44%)
period 234 95/96 - period 1 96/97	4(40.00%)	7(70.00%)	3(30.00%)	6(60.00%)	6(60.00%)	5(50.00%)
period 34 95/96 - period 12 96/97	8(40.00%)	14(70.00%)	9(45.00%)	11(55.00%)	15(75.00%)	11(55.00%)
period 4 95/96 - period 123 96/97	6(20.00%)	6(20.00%)	8(26.67%)	4(13.33%)	22(73.33%)	7(23.33%)
period 4/1 95/97 - period 23 96/97	6.67(33.33%)	6(30.00%)	8(40.00%)	6(30.00%)	16(80.00%)	8(40.00%)
season 95/96 - season 96/97	13(34.21%)	22(57.89%)	13(34.21%)	21(55.26%)	25(65.79%)	14(36.84%)
Sum for cross Season periods	43.67(32.11%)	62(45.59%)	48(35.29%)	55(40.44%)	93(68.38%)	53(38.97%)
Overall Average Percentage	40.05%	45.77%	42.26%	40.58%	59.21%	47.21%
Overall disjoint training/data sets	38.48%	38.65%	35.74%	32.62%	59.21%	37.06%

the relationships between the attributes and the relative importance of each attribute. In terms of the football matches it lets us see which of the selected attributes are the crucial factors effecting the outcome of a game, and gives some clues as to the relationships between some of those factors. The predictive ability of the learners allows us to foretell the likely outcome of a novel situation. In this instance to predict the outcome of a game which has yet to be played. However, since the data we're using is historical in this case it does not apply.

5.1 Knowledge Gained

The different learning techniques vary in what they provide in terms of understanding of the interrelationships between the attributes and the outcome of a game. The MC4 learner identifies those attributes which have the largest effect on the outcome of the game. It shows their relationships to each other in terms of their effect on the outcome of the game. This is a very simplified model of the game itself. The naive Bayesian learner does not construct a model as such, its model is predefined, that is, it assumes that all of the attributes are independent and that every attribute will have some effect on the result. The learning process for the naive Bayesian learner is then simply one of discovering the relative strength, and polarity, of the effect of each attribute with respect to the result. The Hugin learner looks for correlations between the values of the attributes including the result. Once a network is constructed using the correlations that lie within the required sensitivity, then the NPTs can be learnt from the available data. The expert constructed BN represents the knowledge of the expert, that is, it is a model is the expert's belief of the interrelationships between the attributes and their relative importance. KNN does not construct a model as such, it simply uses the existing data and provides a *likeness* comparison with any test data. Thus KNN does not significantly enhance our understanding. One of the limitations of all the non expert methods used here is that they only use the supplied attributes. This is particularly limiting in its effect on the learnt BNs. In a problem where most of the supplied attributes have defined values the possible network structures for a learnt BN are very restricted and, in effect, become just reduced versions of the naive Bayesian model. To overcome this limitation the learners need to add meta-attributes which represent some combined effect of observed, and possibly unobserved, attributes. In modelling we often try to group related attributes together in a way that creates meta-attributes which are not directly observed, but which can aid in both understanding and analysis. These constructed meta-attributes help to clarify and structure our models. An example of this is the **Attack** node in the expert constructed network. This is not an observed attribute, but its value is inferred from the directly observed attributes **Sherringham**, **Anderton** and **Armstrong**. While they are not observed the nodes **Attack**, **Spurs_Quality** and **Performance** help build a model of the games Tottenham Hotspur played. This model gives us some additional insight into how the observed attributes effect the outcome of the game[7].

5.2 Predictive Value

From the table of results it is clear that the KNN learner provides the best prediction when the same data is used for both training and testing. The general *likeness* matching of KNN is a powerful predictive tool when working within a known data set. The expert constructed BN is the best predictor of the outcome when disjoint training and test data sets are used. The way we judged results in terms of their being correct or incorrect can make BNs appear poorer performers than might be the case. Consider that a BN provides not just a win, draw, or loose result, but a prediction of the likelihood of each outcome. A BNs prediction can be used in ways that the simple win, draw or loose results of the other learners cannot. In the experiments the BNs sometimes gave two possible outcomes equal probabilities, and sometimes the difference in probabilities between two outcomes was small. In practise this probability information could be used to enhance decisions made on the basis of the probable results. This is a strength of BNs that is lost in simple right or wrong comparisons.

If we were to assume the outcome of the matches was random and choose a random value for each outcome then we would expect to achieve a classification accuracy of around 33.33%. A number of the learners' classification accuracy is around this figure for disjoint training and test data sets. This might mean that neither of the chosen set of attributes is suitable for the task of prediction, but the performance of the expert BN would tend to contradict this assumption.

6 Future Work

There are a number of obvious directions in which future work could be done. As pointed out this method of prediction is inherently asymmetric. It should be possible to construct a more symmetrical model using similar data for all the teams in the league. However, while this might provide a more accurate prediction of the outcome of the games involving Tottenham Hotspur, it would also involve at least multiplying the amount of computational work by the number of additional teams in the league. Since each of the other teams only

[7] Assuming the model is correct that is

contributes to 1/19th of the results this additional work may not be justified. The data could be expanded by, for example, adding the position occupied by each player, but then the issue of position changes would need to be considered. Alternatively it might be possible to provide the learners with more structure to learn by building a model of the team performance and then using the learners on the parts of the model. An example of this would be to define the team in terms of attack and defence. Each player could be characterised as belonging to attack, defence or both and an analysis could be performed to evaluate the strength of the possible groupings. The strengths of the attacking and defending player combinations could then be used in predicting the result.

APPENDIX A - Additional Diagrams

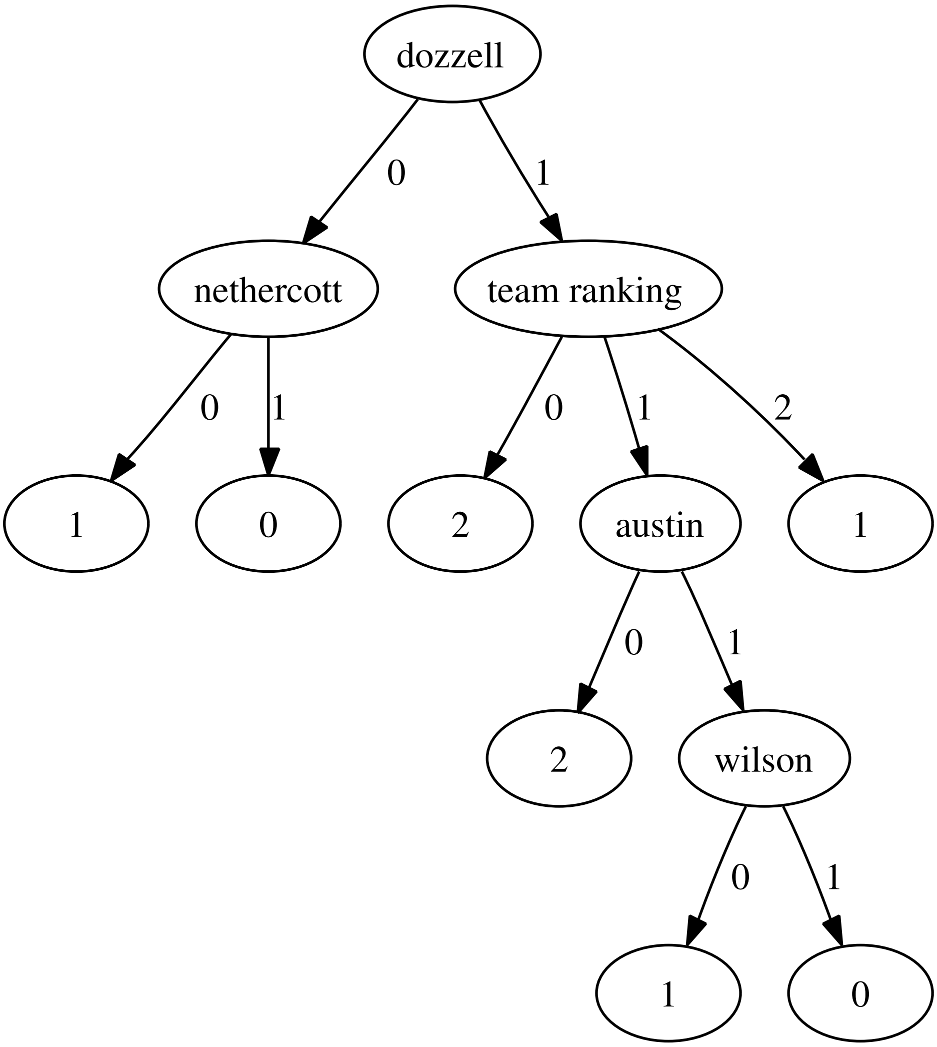

Figure 16. Tree from MC4 for the periods 2, 3, and 4 of the 1995/1996 season

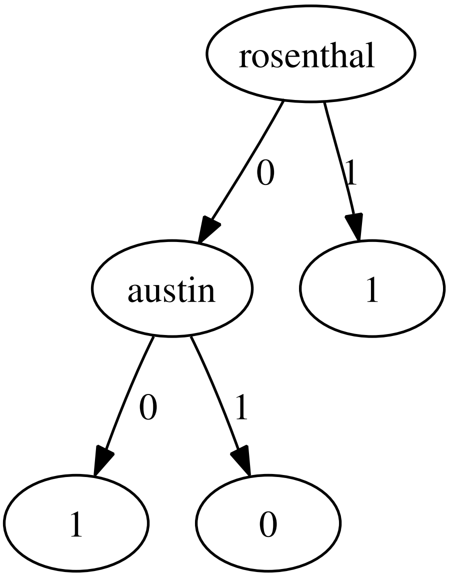

Figure 17. Tree from MC4 for period 4 of the 1995/1996 season

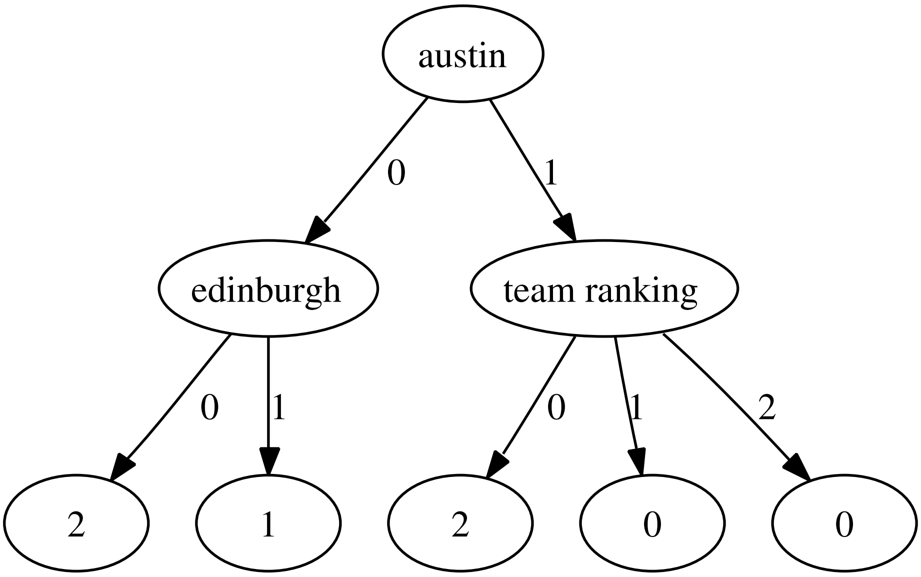

Figure 18. Tree from MC4 for the periods 3 and 4 of the 1995/1996 season

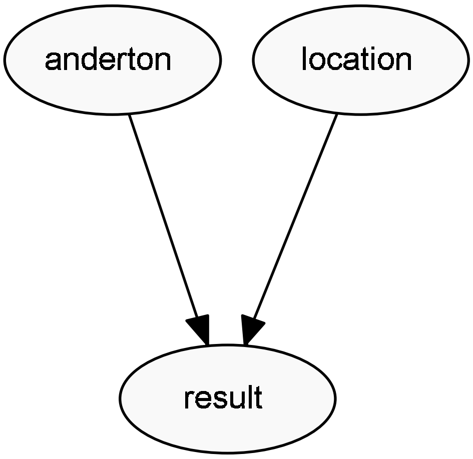

Figure 19. Data Driven BN for period 1 of the 1995/1996 season

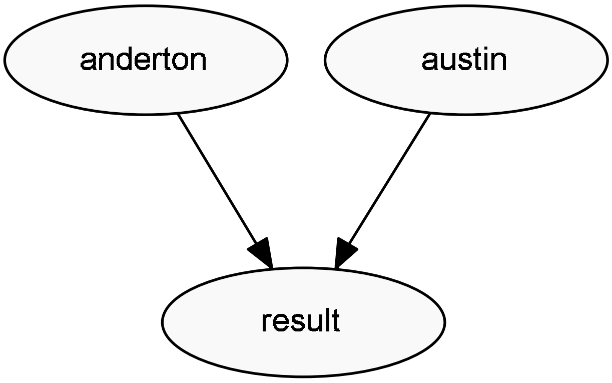

Figure 20. Data Driven BN for the periods 1 & 2 of the 1995/1996 season

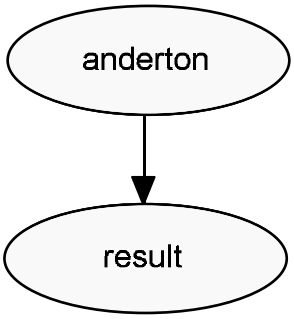

Figure 21. Data Driven BN for periods 1,2 & 3 of the 1995/1996 season

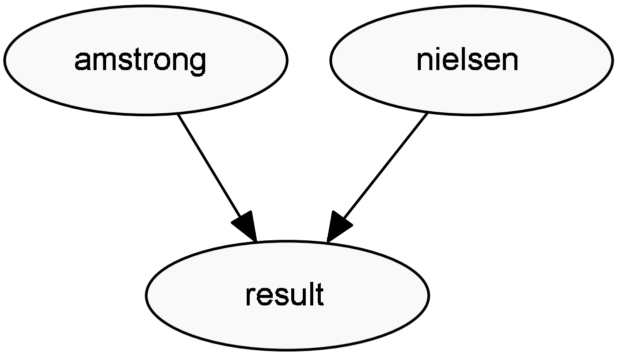

Figure 22. Data Driven BN for period 1 of the 1996/1997 season

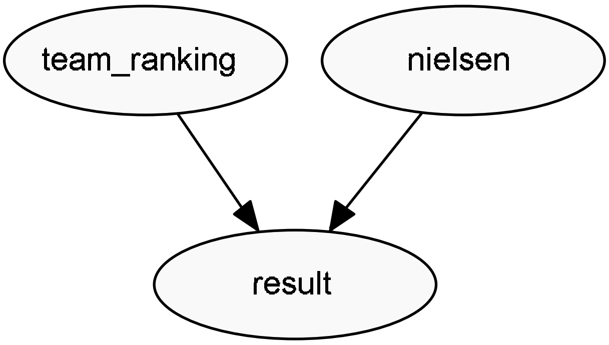

Figure 23. Data Driven BN for periods 1 & 2 of the 1996/1997 season

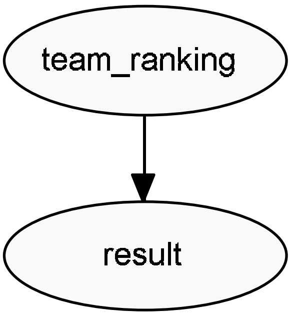

Figure 24. Data Driven BN for the periods 1, 2 & 3 of the 1996/1997 season

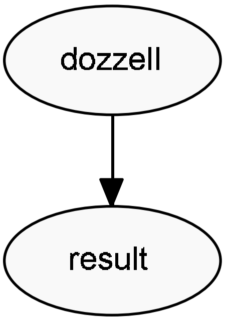

Figure 25. Data Driven BN for periods 2 & 3 of the 1995/1996 season

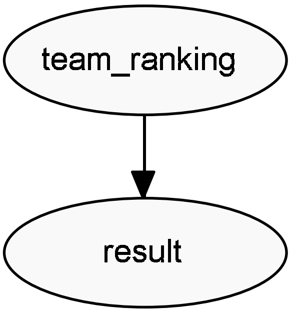

Figure 26. Data Driven BN for the periods 2, 3 & 4 of the 1995/1996 season

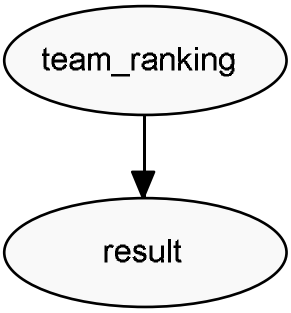

Figure 27. Data Driven BN for periods 3 & 4 of the 1995/1996 season

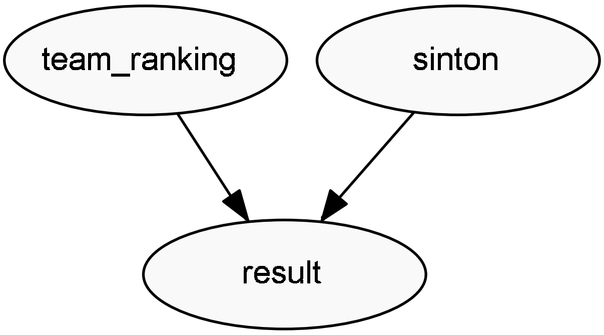

Figure 28. Data Driven BN for period 4 of the 1995/1996 season

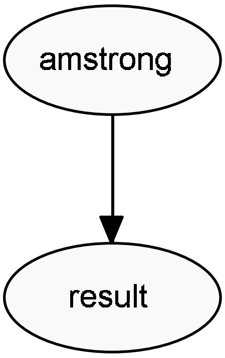

Figure 29. Data Driven BN for periods 4 & 1 of the 1995/1996 and 1996/1997 seasons

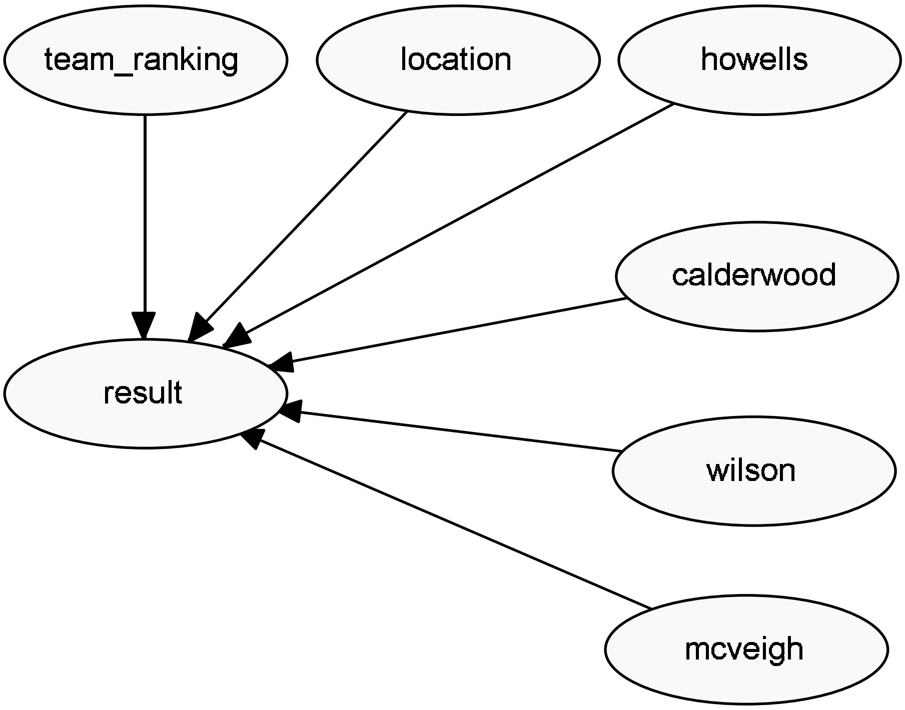

Figure 30. Data Driven BN for periods 3 & 4 of the 1996/1997 season