Learning Spatial Relationships: Some Approaches

R I McKay & R A Pearson
Computer Science Dept, University College,
Australian Defence Force Academy
rim@cs.adfa.oz.au, +61 6 268 8169
rap@cs.adfa.oz.au

P A Whigham
Division of Land and Water Resources
Commonwealth Scientific & Industrial Research Organisation
paw@cs.adfa.oz.au

Presented at the second annual conference of GeoComputation '97 & SIRC '97, University of Otago, New Zealand, 26-29 August 1997

Abstract

We consider three approaches to learning natural resource models involving spatial relationships, based respectively on decision tree learning, genetic programming and inductive logic programming. In each case, the results of spatial learning on a natural resource problem are compared with the results of non-spatial learning from the same data, and improvements in predictivity or simplicity of the models are noted. We argue also that it is highly desirable that spatial learning systems for natural resource problems incorporate mechanisms for the user specification of learning biases.

1. Introduction

1.1. Machine Learning for Natural Resource Problems

With today's increasing emphasis on environmental limits, the need for accurate and timely information on natural resource issues is pressing. In many cases, the information required for decisions may be expensive to obtain, yet data on some of the underlying variables is relatively inexpensive and available in enormous quantity. The problem is to convert this plentiful data into useful information; machine learning and related data mining techniques provide one promising means to do so.

There have been a number of such applications (for example Barbanente et al 1992; Eklund & Salim 1993; Papp, Dowe and Cox 1993; Stockwell et al 1990; Walker & Cocks 1990). Yet the range is perhaps less than one might expect. Part of the reason lies in the form of the readily available, industrial quality learning systems (Breiman et al 1984; Quinlan 1986). These systems are attribute based, rather than relational - thus they cannot directly learn about spatial relationships. Yet spatial relationships are at the core of many, probably most, natural resource problems.

This paper aims to demonstrate the value of spatial learning, by describing a number of experiments using different methods which have been carried out at University College, ADFA.

Of course, we are not alone in such work. Of recent years, spatial regression methods have appeared in statistical packages (Bowman 1997). However it is well known (Stockwell et al 1990) that discrete machine learning methods outperform regression methods on some datasets. Closer to our approach is the work of (Dibble 1994), which uses an evolutionary approach distantly related to the (Whigham 1996) work reported here.

1.2. Why is Spatial Learning Hard

Spatial problems are intrinsically relational rather than attribute based: they are about the relationships between attributes of particular locations and regions, rather than simply about the local values of those attributes. While particular spatial relationships can often be reduced to spatial attributes (see the discussion below), the reduction requires a-priori knowledge, about the significance of particular spatial relationships for the problem at hand, which is often not available.

On the other hand, relational learning is intrinsically difficult. The concept spaces to be searched are orders of magnitude larger than those encountered in attribute-based learning.

Furthermore, there are special difficulties with spatial learning problems. Most attribute-based learning, and much relational learning, makes use of greedy search algorithms, which require each new element of the learned model to contribute significantly toward the accuracy of the model. There is no look-ahead: the new element has to make the contribution on its own, without the assistance of any other element. But spatial relationships typically do not make such isolated contributions: they work together with the attributes of the related locations to contribute toward the reliability of the model.

1.3. The Importance of Bias

The machine learning community has gradually come to appreciate the importance of bias in learning systems, and indeed the impossibility of the once-holy grail of unbiased learning (Wolpert and Macready 1995).

In natural resource problems, it is commonly the case that experts in the field have considerable knowledge about the likely forms of models, even if they do not know the exact model at the time.

Taking all this, together with the inherent computational difficulties of spatial learning, it seems clear that systems which provide the user with opportunities to control the bias of the search, and thus reduce the computational cost of the learning process, will be highly desirable for spatial learning in natural resource problems.

2. Sample Problems

Our work to date has been particularly based on two natural resource learning problems. The first is highly atypical, and is specifically chosen because we already know the answer to the problem, and can thus assess sensibly how different learning systems are behaving in relation to that answer. The second was chosen as a fairly typical example of a natural resource problem, and indeed has previously been intensively studied in a purely attribute-based setting (Stockwell et al 1990)

2.1. The Wetness Index Problem

The wetness index problem derives from a pre-existing expert system, LMAS (Whigham and Davis, 1989). LMAS is used to assist with environmental management at Puckapunyal army base in Victoria, Australia. It predicts, from meteorological records and spatial databases describing the site, the likely ground disturbance effects of a given armoured exercise.

One module of LMAS uses the landform and slope layers of the GIS describing Puckapunyal to predict the propensity of particular areas to become waterlogged - the wetness index, with 6 possible values: unknown, dry, average, wet, seasonally waterlogged, waterlogged. This module, like the rest of LMAS, was derived through the traditional expert systems process - as an encoding of the pre-existing knowledge of a geographical expert - and was then validated by ground-truthing. A map of the wetness index for Puckapunyal is given in Figure 1.

The wetness index learning problem is this. The system is given a three-layer dataset consisting of the original landform and slope layers, together with a new layer consisting of the wetness indices as derived by the wetness module of LMAS. The dataset consists of 3,272 polygons, together with a table of the adjacencies between polygons. The system is to learn a new set of rules, which are to predict the wetness index as accurately as possible from the landform and slope layers, together with the adjacency relations.

This particular problem is of interest for three reasons. First, we know that there is a perfectly accurate model of this problem - the wetness module of LMAS. Second, we know that the model involves spatial reasoning, so it is likely that spatial learning will be useful for the problem. Finally, we know the form of the LMAS model, so that if a particular learning system fails to learn well, we can investigate why it does not discover the LMAS solution. On the other hand, the problem is artificial, in that the model we are attempting to learn is that which best fits the original expert's model of the situation, rather than some underlying "real World" description.

2.2. The Greater Glider Problem

The greater glider dataset is described in detail in (Stockwell et al 1990); briefly, it consists of a 20*20 grid of cells. For each cell, the values of seven independent variables are recorded: the degree of development (D - 3 categories); whether a stream corridor (ST - 2 categories); stand condition from a forestry perspective (SC - 6 categories); site quality from a forestry perspective (SQ - 4 categories); floristic nutrients (FN - 4 categories); slope (S - 3 categories); and erosion (E - 3 categories) (NB in the study area, all sites were highly eroded, E=3, so the erosion attribute may be effectively ignored). For each cell, we also have a value for the putative dependent variable, the greater glider density (GD - 4 categories, ranging from 0-absent to 3-abundant). A map is given in Figure 2.

3. Simulating Spatial Learning with Attribute-Based Systems

The first series of experiments described here were performed with the aim of demonstrating that the capacity to learn spatial relations could improve the predictivity of machine learning systems applied to natural resource data. The data used was the greater glider dataset described above.

3.1. Experiments

The experiments were conducted using the Rulefinder decision tree induction system (Pearson 1996). Full details of the experiments and results are given in (Pearson and McKay 1996). Briefly, a first experiment was conducted to provide a baseline for comparison by setting up the conditions as similarly as possible to the experiments of Stockwell et al (1996); a second baseline experiment varied the underlying learning conditions to be similar to those of our main experiments as possible, but without incorporating any spatial information. These experiments led into the main work, in which spatial relationships, built from the underlying attributes, were encoded as additional attributes and added to the dataset.

Taking as an example the underlying attribute "site quality", describing the forestry potential of a location, the relationships encoded as attributes for the various experiments were:

experiment 3: distance to nearest location with a particular value of site quality

experiment 5: whether some adjacent location has a particular site quality

experiment 4: whether there was an adjacency chain of a given length (i.e. A adjacent to B adjacent to C ....) to a location having a particular site quality

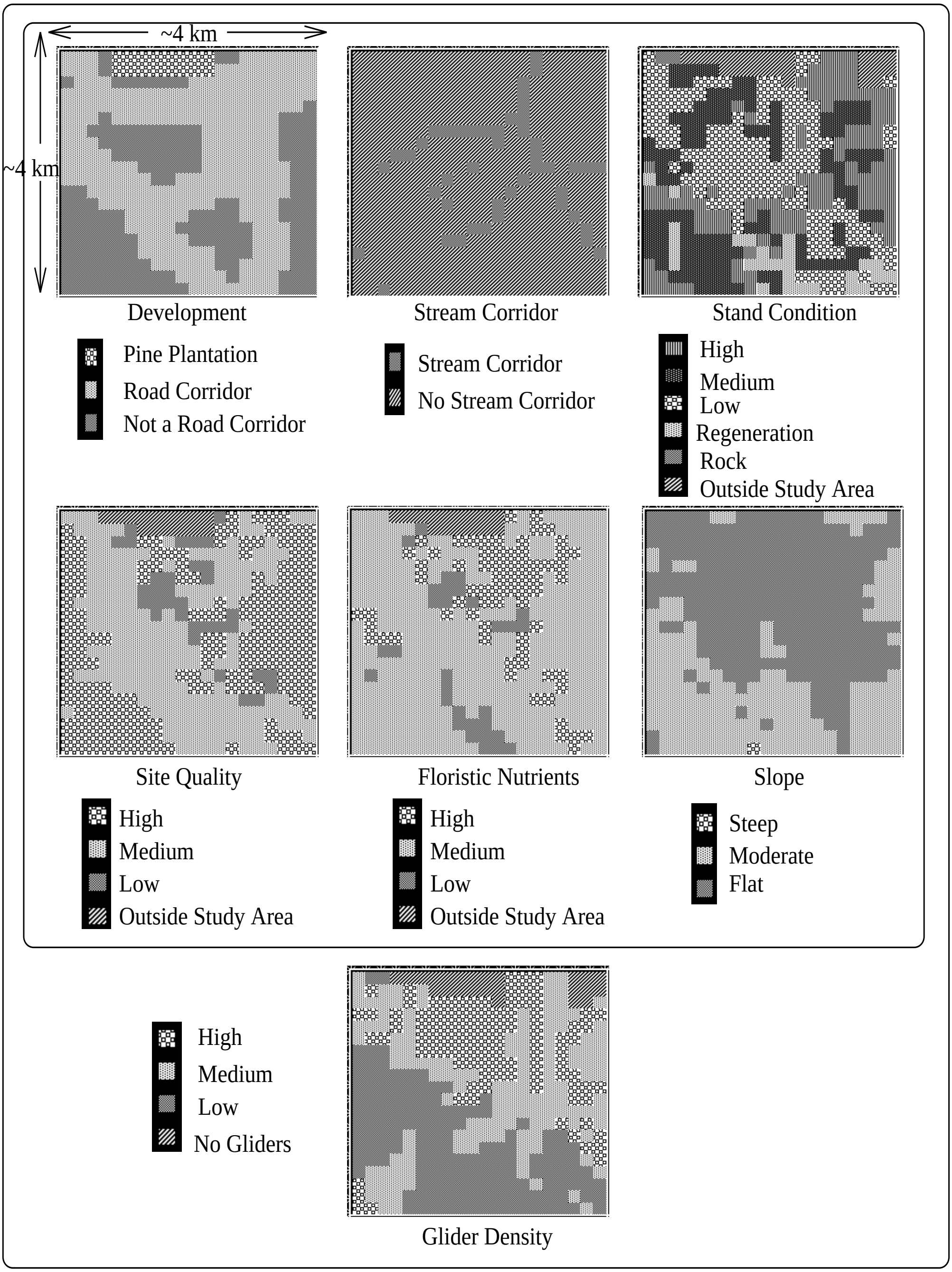
Finally, each of the above experiments was split into two experiments, according to whether values of the learning attribute - the glider density (at sites other than the particular location in question) - were incorporated amongst the spatial relationships encoded (e.g. in experiment 3a, "distance to the nearest site having a glider density of 3" was not encoded as an attribute in the dataset; in experiment 3b, it was so encoded).

3.2. Results

Size and accuracy of decision trees induced from the greater glider dataset.

Results in the two baseline experiments were very comparable with Stockwell et al (1990), with error rates of 47.5% and 47.75% respectively, and trees of very similar structure. Experiments 3 to 5 gave dramatically improved

Experiment	1	2	3a	3b	4a	4b	5a	5b
Tree Size	21	5	27	23	68	13	39	15
Error Rate (%)	47.5	47.75	29.5	29.0	28.75	31.75	34.5	31.75
Std Dev (%)	NA	6.74	5.68	3.71	4.64	7.86	6.78	6.46


~4 km
~4 km
Development
Pine Plantation
Road Corridor
Not a Road Corridor
Stream Corridor
Stream Corridor
No Stream Corridor
Stand Condition
High
Medium
Low
Regeneration
Rock
Outside Study Area
Site Quality
High
Medium
Low
Outside Study Area
Floristic Nutrients
High
Medium
Low
Outside Study Area
Slope
Steep
Moderate
Flat
High
Medium
Low
No Gliders
Glider Density

error rates, ranging from 28.75% to 34.5%.

The tenfold cross-validation method, which Rulefinder uses to estimate error rates, also permits the estimation of standard deviation of the error rates. It is thus possible to say that the results in experiments 3 through 5 are significantly different from the results in experiments 1 and 2 (and thus from the Stockwell et al (1996) results) at the 1% confidence level; but they are not significantly different from each other.

One other point to note: the trees learnt here may be approaching the limit of what can be learnt from this data, due to inherent noise and/or missing variables. As shown in Stockwell et al, simply looking at cases in which pairs of cells with the same values for all the independent attributes nevertheless have differing values of the learning attribute, gives an error rate of 24.2%, with a standard deviation of 1.2%. While one should be careful in extrapolating this to spatial learning - since spatial learning in effect provides additional independent attributes by which cells may be distinguished - the similarity of these error rates may not be entirely coincidental.

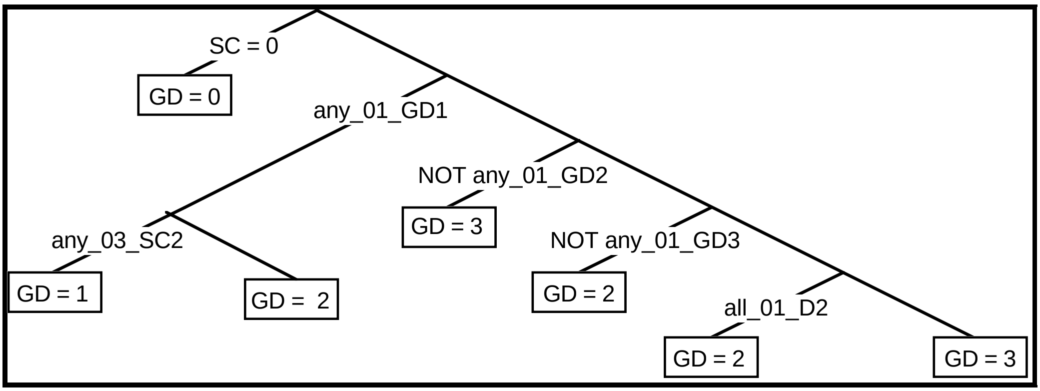
3.3. Discussion

There is always the possibility that the decision trees in experiments 3 to 5 are overfitted to the data. The pruning process in decision tree learning normally provides some protection against this. However the incorporation of spatially derived attributes in the dataset implies that it is not possible any longer to guarantee the independence of the training and test sets, and thus overfitting cannot be ruled out.

However, consideration of the meanings of the decision trees gives some degree of protection against overfitting: on the assumption that the search space of decision trees is sparsely populated with sensible explanatory trees, it is highly likely that any overfitting will be accompanied by meaningless expressions at the tips of the decision trees. Analysis of experiments 3 to 5 suggests that the largest decision trees generated - a 68-node tree in experiment 4a, and possibly a 39-node tree in experiment 5a - may be somewhat overfitted, but that the other treees, which are roughly comparable in size with those of Stockwell et al (1996), are unlikely to be overfitted. A detailed discussion may be found in Pearson & McKay (1996). The smallest tree, that from experiment 4b, is shown in Figure 3.

Thus our final conclusion is that the incorporation of spatial information into a learning process can lead to significant improvements in the predictivity of the models generated. However, the process used is relatively clumsy. It requires the experimenter to know ahead of time which spatial attributes are important, so that they can be incorporated into attributes for use in the learning process. Further, it requires the experimenter to write special-purpose programs to translate the selected spatial relationships into tabular attribute form.

We would naturally prefer that the learning system be able to discover the important spatial relationships for itself,



while permitting the user to narrow the focus of the learning to particular classes of spatial - or other - relationships if such knowledge is available. Thus a prime focus of our work has been on learning systems which can work directly with spatial relationships, but permit the user to vary the bias of the learning space search.

4. Genetic Programming and Geospatial Relations

The work on context free grammars for genetic programming (CFG-GP) discussed here is reported in detail in the doctoral thesis of P A Whigham (1996). It builds upon the genetic programming paradigm of Koza (1992). However, in the genetic programming paradigm, the description language is a by-product of the GP system and is not amenable to user variation except through re-building the underlying system.

In line with our conviction that useful geospatial learning systems will require simple mechanisms by which the user may specify the search space the learning system is to use, CFG-GP provides a context-free grammar in which the user defines a grammar for the language the learning system is to use for the specific problem (this work follows on from the Grendel system (Cohen 1994), which used context free grammars similarly, but within the inductive logic programming paradigm).

The greater glider dataset contains a number of hard constraints. For example, a small proportion of the cells are rated as "outside the study area". These cells have their glider density set arbitrarily to zero. This causes little problem to deterministic learning systems such as decision tree systems: these rapidly learn that "outside the study area" implies "glider density zero", and are thus free to ignore those cells from that point on (indeed, this is the top-level decision in virtually all the decision trees we have generated from this data).

A stochastic learning paradigm has problems with such hard constraints, since the system will always be prepared, even though with low probability, to re-visit these constraints and to try alternatives. Whatever mechanism is used to evaluate the success of the system will thus incorporate some penalty for this willingness to try alternatives.

Fortunately, CFG-GP incorporates a mechanism for investigating this effect. The user may explicitly incorporate the hard constraint into the search language used by the system, so that the option of revisiting the constraint is no longer available.

4.1. Experiments

CFG-GP was first applied to the greater glider dataset in non-spatial mode. A number of experiments were conducted, starting off with a simple attribute language describing the dataset, then extending this with two hard constraints: the "outside search area" constraint described above, and a second explicitly requiring the system to learn descriptions for each of the four glider density classes (otherwise the system may simply ignore density classes which are sparsely represented in the data).

The language was then extended with additional spatial expressions. For each possible value V of each of the underlying attributes A, and for each distance D, the system is permitted to derive the boolean expression determining whether there is a cell within distance D of the current cell, in which the attribute A has the value V.

For computational reasons (genetic programming is computationally very expensive), the values of D were limited to be either 1 or 2, though the decision tree work above suggests that distance values up to 5 may be meaningful in this dataset.

4.2. Results

In the simplest attribute learning example above, the system achieved an error rate of 47.5 ± 3.4% (based on 6 trials). Incorporating the hard constraints mentioned above improved the learning somewhat, to an error rate of 42.9 ± 3.2% (6 trials). Finally, addition of spatial expressions gave error rates of 32.8 ± 1.7% (6 trials). The best ruleset was:

if ((stand_condition = rock)
  or ((slope > flat within distance 2)
    and (stand_condition = regeneration within distance 4)
    and (floristic_nutrients > medium within distance 5)))
then glider_density = low
else **if** ((slope > flat within distance 2)
    or (stand_condition = regeneration within distance 4))
  then glider_density = medium
  else glider_density = high

4.3. Discussion

In non-spatial learning, CFG-GP achieved similar results to Stockwell et al (1990), and to the Rulefinder results reported above (the incorporation of hard constraints improved the learning, but the improvements are only marginally significant). Significant improvements were obtained by the incorporation of spatial information into the learning; the improvements are very comparable with those achieved by Rulefinder, providing further confirmation that the improvements in error rate are real, and not just the result of overfitting the data.

5. Inductive Logic Programming and Geospatial Relations

We have previously (McKay 1994) reported negative results in the application of ILP systems to geospatial learning problems. Our analysis there pointed out that the lack of results were not due to inherent limitations of the ILP paradigm, but were particularly related to specific assumptions made in the greedy algorithms used.

Specifically, the systems assumed that useful relationships either directly reduce dataset noise (without the assistance of subsidiary attributes), or are determinate. Unfortunately, spatial relationships such as distance, relative orientation etc. do not have either of these properties, so that spatial relationships would never be tested by these algorithms.

Since that time, we have carried out further experiments with the more recent Progol system (Muggleton 1995), which does not make determinacy assumptions. Progol learns logical rules, in the form of prolog programs. Progol does not handle noise well, so we have not gained any useful results in learning from the greater glider dataset. However experiments with the wetness index dataset have yielded some interesting results

5.1. Experiments

In the first experiment, progol was run on the wetness index as described above. The second experiment was identical, except that the table of adjacencies was deleted from the dataset, so that progol could only learn attribute descriptions of the dataset.

5.2. Results

Progol learns a complete description of the dataset on which it is run. If necessary, it will generate rules for the dataset cell by cell, in order to do so. Unlike Rulefinder and CFG-GP, it does not provide for a separation of learning and test datasets. Thus results from Progol do not give meaningful error estimates. The only meaningful comparison we can make is between the sizes of the rulesets learnt in each run. Note also, that these rules have been learned from positive data only: since progol was unable to deduce that "dry" and "average" are incompatable, it was prepared to learn identical rules for both. Further work, to ameliorate this problem, is in progress.

The first run, incorporating adjacencies, described the dataset with 6 rules, using 20 conditions (note that the land unit types are ordered):

wi(A,wet) **if** land_unit(A,B) **and**
  B > floodplain_seasonally_inundated
wi(A,dry) **if** land_unit(A,B) **and**
  B < dam **and** B > floodplain_seasonally_inundated
wi(A,average) **if** land_unit(A,B) **and**
  B < dam **and** B > floodplain_seasonally_inundated
wi(A,wet) **if** A adjacent_to B **and**
  slope(B,C **and** C > -3
wi(A,seasonally_waterlogged) **if** slope(A,B) **and**
  A adjacent_to C **and** land_unit(C,D) **and**
  D < sand_dunes **and** D > floodplain_seasonally_inundated
wi(A,waterlogged) **if** A adjacent_to B **and** B adjacent_to C **and**
  slope(C,D) **and** D > -2.

The second run, omitting adjacencies, required 10 rules and 40 conditions.

The original expert ruleset, when expressed in the above language, has 13 rules and 42 literals.

5.3. Analysis

The most important result is that experiment 1, using spatial learning, learnt a very much simpler model of the dataset than experiment 2, using purely attribute learning. The big difference lies in only one of the wetness index values: in experiment 1, "wet" cells are described in one spatial and one non-spatial rule, using 8 literals. In experiment 2, 5 non-spatial rules are required, using 25 literals.

Secondly, it is interesting that progol has learnt a model which is simpler, in this language, than the original expert ruleset. The comparison is not entirely fair, however: the expert ruleset was originally expressed in a completely different language, and its present size is partly a result of the translation process. Moreover, the expert ruleset did know about such issues as mutual exclusiveness of wetness values. Nevertheless, it is fair to say that the spatial learning process has produced a ruleset which is smaller and simpler than the non-spatial process, and of expert quality in these respects.

6. Conclusions

Learning systems which can take spatial relationships into account may learn more accurate models than non-spatial learning systems, in real-World natural resource problems. The genetic programming and inductive logic programming paradigms both provide mechanisms with which to attack such problems. So far, greater success has been achieved with GP approaches than with ILP, but this does not seem to be due to any inherent limitations of ILP. Assuming that ILP systems able to handle both noise and indeterminacy become available, the choice between the two may come down to ease of use vs computational complexity: correctly setting up an ILP system may require greater understanding than an equivalent GP system, but the GP system is likely to use more computational resources. As an indication, the CFG-GP work reported above required cpu-days on a SUN SPARC 1000. ILP is also computationally expensive, but more on a scale of cpu-hours than cpu-days.

All existing relational learning systems are computationally expensive; this is unlikely to change, as relational learning is an inherently difficult task. But experts working with geospatial datasets typically have considerable knowledge about constraints on the likely structure of models of those datasets - often arising from knowledge about the physical and other processes involved. Thus it is highly desirable that learning systems for use in geospatial problems permit the user to incorporate this knowledge in the search strategy of the learning system involved. The Grendel and CFG-GP systems mentioned above (along with many other learning systems) give indications of how this may be achieved. A useful by-product of the use of such biases is the possibility of assembling a body of knowledge about useful biases for geospatial learning, and thus of the overall structure of spatial knowledge.

7. References

Barbanente, A, D Borri, F Esposito, P Leo, G Maciocco and F Selicato (1992) Automatically Acquiring Knowledge by Digital Maps in Artificial Intelligence Planning Techniques. Theories and Methods of Spatio-Temporal Reasoning in Geographic Space. A U Frank, I Campari and U Formentini, editors. *Springer Lecture Notes in Computer Science* 639. Springer Verlag, Berlin, Germany, 1992, pp 379 - 401.

Bowman, a (1997) S+ SpatialStats: A Spatial Statistics module for S-Plus. *Maths&Stats Newsletter.* CTI Statistics, University of Glasgow, Glasgow, Scotland, 1997.

Breiman, L, J H Friedman, R A Olshen and C J Stone (1984) Classification and Regression Trees. Wadsworth Inc, Belmont, USA, 1984.

Cohen, W W (1994) Grammatically Biased Learning: Learning Logic Programs Using an Explicit Antecedent Description Language. *Artificial Intelligence* 68(2), 1994, pp 303-366.

Dibble, C (1994) Beyond Data: Handling Spatial and Analytical Contexts with Genetics Based Machine Learning. *Sixth International Symposium on Spatial Data Handling* Edinburgh, Scotland, September 1994

Eklund, P W and A Salim (1993) An Experiment with Automated Acquisition of Classification Heuristics *Conference on Advanced Remote Sensing* University of NSW, Sydney, Australia, 1993, volume 2, pp 83 - 94.

Koza, J R (1992) Genetic Programming: On the Programming of Computers by Means of Natural Selection. Bradford, MIT Press, Cambridge, USA, 1992.

McKay, R I (1994) Relational Learning for Geospatial Problems. *ICARCV Workshop on Spatial and Temporal Interaction: Representation and Reasoning*, Singapore, 1994.

Muggleton, S (1995) Inverse Entailment and Progol. *New Generation Computing Journal* 13, 1995, pp 245 - 286.

Papp, E, D L Dowe and S J D Cox (1993) Spectral Classification of Radiometric Data using an Information Theory Approach *Conference on Advanced Remote Sensing* University of NSW, Sydney, Australia, 1993, volume 2, pp 223 - 232.

Pearson, R A (1996) Single Pass Constructive Induction with Continuous Variables. *ISIS: Information, Statistics and Induction in Science*. D L Dowe, K B Korb and J A Oliver, editors. World Scientific, Singapore, 1996, pp 31 - 42.

Pearson, R A and R I McKay (1996) Spatial Induction for Natural Resource Problems: A Case Study in Wildlife Density Prediction. Technical Report CS11/96, School of Computer Science, University College, University of New South Wales, Canberra, Australia, 1996. Submitted to *AI Applications in Natural Resource Management.*

Quinlan, J R (1986) Induction of Decision Trees. *Machine Learning* 1, 1986, pp 81 - 106.

Stockwell, D R B, S M Davey, J R Davis and I R Noble (1990) Using Induction of Decision Trees to Predict Greater Glider Density. *A I Applications in Natural Resource Management* 4(4), 1990, pp 33 - 43.

Walker, P A and K D Cocks (1990) Habitat: a Procedure for Modelling a Disjoint Environmental Envelope for a Plant or Animal Species. *Global Ecology and Biogeography Letters* 1, 1990, pp 448 - 461.

Whigham, P A (1996) Grammatical Bias for Evolutionary Learning. PhD Thesis, University College, University of New South Wales, Canberra, Australia, 1996.

Whigham, P A and J R Davis (1989) Modelling with an Integrated GIS/Expert System *Ninth Annual ESRI Users Conference*, Palm Spring, ESRI, Redlands, USA, 1989.

Wolpert, D H and W G Macready (1995) No Free Lunch Theorems for Search. Santa Fe Working Paper, Santa Fe Institute, Santa Fe, USA, 1995.

8. Acknowledgements

We wish to thank Dr P Laut, Dr R Davis and Ms S Cuddy for use of the wetness index dataset, and Dr S Davey and Dr D Stockwell for use of the greater glider dataset.