94-14

*Proceedings of the 1994 Earth Systems Research Institute (ESRI) User Conference*, Pebble Beach, CA, May 1994.

Linda Tischer, Member of Technical Staff, TASC
12100 Sunset Hills Rd.
Reston, VA 22090
Eric Bloedorn, Research Assistant
Machine Learning and Inference Laboratory
George Mason University
Fairfax, VA 23030
Telephone: (703) 834-5000
Fax: (703) 318-7900

# AN APPLICATION OF MACHINE LEARNING TO GIS ANALYSIS

With the large amount of spatially referenced data being produced, there is an increasing demand for improved Geographic Information System (GIS) analysis techniques. Some initial work has already been done in applying Expert System (ES) techniques to assist the GIS analyst, but these techniques have not kept up with advances in Machine Learning (ML). The focus of this paper is the use of modern ML techniques, specifically learning from examples, to induce spatial analysis constraints from positive and negative examples described by multiple thematic and/or spectral sources. These analysis rules can then be used to guide the spatial analysis process when a priori knowledge is limited. This project demonstrates the importance of inter-disciplinary cooperation in the realization of objectives associated with the analysis of large scientific datasets.

## INTRODUCTION

What is a GIS? "In its simplest form, a Geographic Information System (GIS) may be viewed as a database system in which most of the data are spatially indexed, and upon which a set of procedures operate in order to answer queries about spatial entities in the database." (Smith, Menon et al. 1987) Schutzer has defined Artificial Intelligence (AI) as the design and programming of machines to accomplish tasks that people accomplish using their intelligence (Schutzer 1987) . In a GIS, AI techniques can be used for many tasks: to extract features from imagery, automatically scan manuscript maps, manage the editing of geographic data simultaneously with data capture, assess data quality as it enters the system, exploit knowledge about a user query and within the GIS itself in order to speed the search, and provide intelligent user interfaces (Robinson and Frank 1987).

### Artificial intelligence

Efforts to apply artificial intelligence techniques to GIS in subareas such as: map design, terrain/feature extraction, geographic database management, and geographic decision support, date back 20 years or more(Robinson and Frank 1987). Certainly, the ability to learn is one of the main characteristics of intelligence. The most interesting type of learning, for the purposes of this paper, is learning from examples. Learning from examples is a natural and common way for people to learn. In this type of learning, a person formulates a general (possibly incorrect) rule from a small number of examples. In incremental learning, this rule is updated as new examples are encountered. The process of generating rules from examples involves induction. This same process is used by machine learning systems.

Machine learning. Learning is the change in a system that allows the system to improve performance on a task or a task drawn from the same population. This improvement may be an increase in speed, or in the ability to perform new tasks. Machine learning is the study of methods for building machines that can learn. Often the term learning is restricted to learning in which symbolic manipulation tasks are performed such as in the acquisition of rules or the improvement of domain theories. Machine learning differs from adaptive control systems which tune performance based on a feedback loop in its ability to improve performance on not only known cases, but unseen cases as well. In order to achieve this learning, systems must be able to store the results of learning in memory. The results of learning can be stored in many forms including rules, network weights, or nodes and arcs in a decision tree.

For this application of machine learning to GIS, supervised learning from examples, is of most interest. In this approach the improvement is shown in learned rules which can accurately categorize new instances. In supervised learning, a set of positive and negative instances of some concept or set of concepts is provided to the system. These instances are described in terms of attributes and values. The system induces rules which discriminate all of the positive examples from all of the negative examples. The rules are guaranteed to be complete (all positive examples are covered by the positive rules) and consistent (no negative examples are covered by the positive rule) with respect to the training data provided no ambiguous examples are given. A brief description of the AQ algorithm as implemented in AQ17 is given in the following section. This description is taken from the AQ17 user's guide (Bloedorn, Wnek et al. 1993).

The AQ algorithm. The AQ17 program learns decision rules by performing inductive inference over a set of teacher-classified training examples and/or a set of initial rules. Training examples are expressed as conjunctions of attribute values. Initial or induced decision rules are logical expressions in disjunctive normal form. The program performs a heuristic search through a space of logical expressions, until it finds a decision rule which best satisfies the preference criterion and that covers all positive examples, but no negative examples. The program implements the STAR method of inductive learning (Michalski and Larson 1978). It is based on the AQ algorithm for solving the general covering problem (Michalski 1969); (Michalski and McCormick 1971).

1. A single positive example, called a seed, is selected and a set of most general conjunctive descriptions of this example is computed (such a set is called a star for the seed). Each of these descriptions must exclude all negative examples.
2. Using a description preference criterion a single description is selected from the star, called the 'best' description. If this description covers all positive examples, then the algorithm stops.
3. Otherwise, a new seed is selected among the unexplained (uncovered) examples, and steps 1 and 2 are repeated until all examples are covered.

The disjunction of the descriptions selected in each step constitutes a complete, consistent and general description of all examples. The preference criterion used in selecting a description from a star is expressed as a list of elementary criteria that are applied lexiographically and with a certain tolerance. The criteria may be simplicity of description (measured by the number of variables used), cost (the sum of the given costs of the individual variables), or other criteria (Michalski and McCormick 1971).

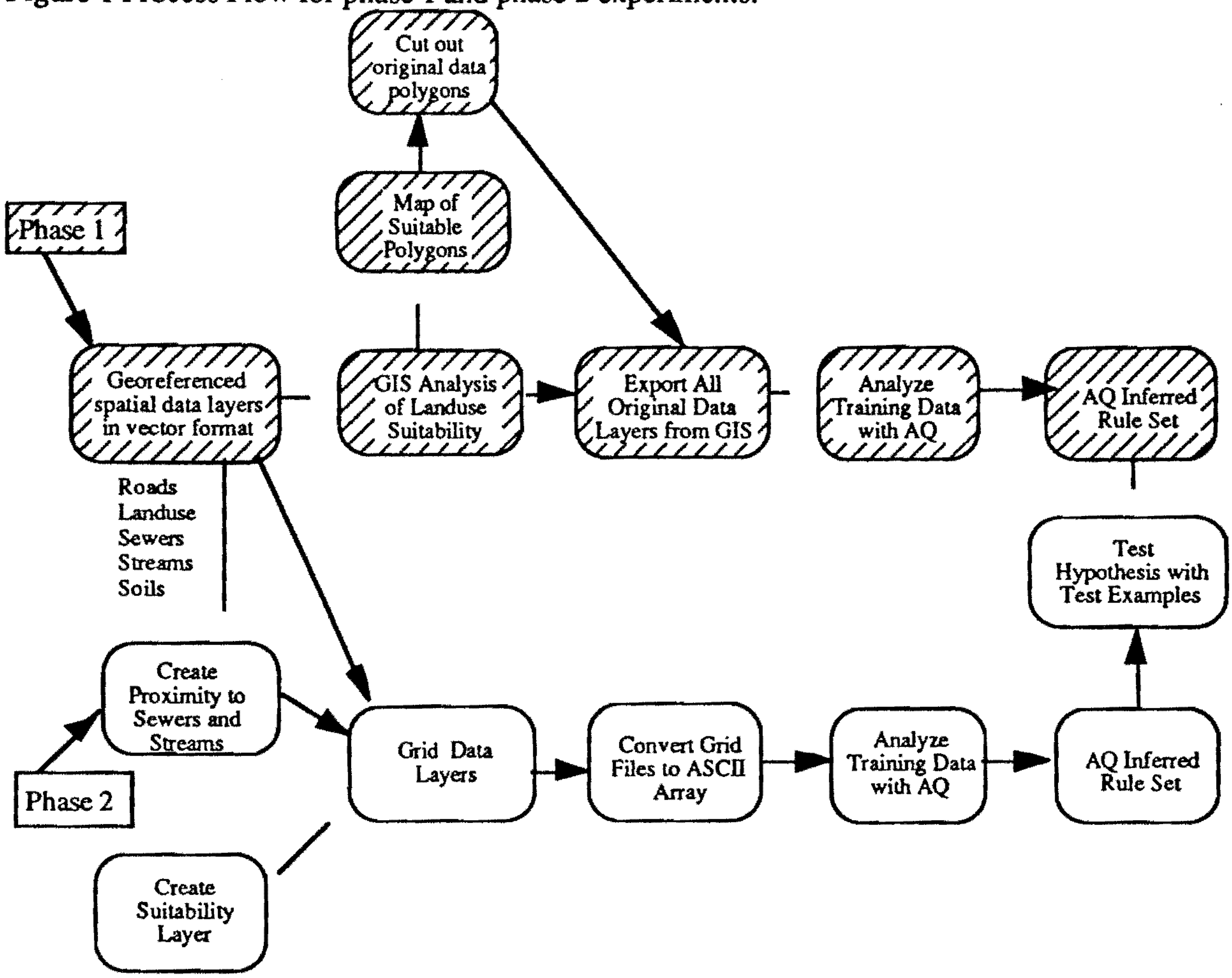
The description of a class is expressed using the variable-valued logic system 1 ($VL_1$), which is a multiple-valued logic propositional calculus with typed variables (Michalski 1974). A class description (called a cover) which is a disjunction of complexes describing all positive examples and none of the negative examples. A complex is a conjunction of selectors, and is the simplest

Polygon-based data analysis (phase 1)

Data input. For phase 1 of this project, five related georeferenced coverages [ land use (polygons), roads (arcs), sewers (arcs), soils (polygons), and streams (arcs)] taken from the ARC/INFO sample geographic database tutorial, are used as sample datasets (ESRI 1992). Each layer represents thematic data for the same hypothetical geographic location. These example data coverages were chosen because they are representative of GIS analysis input datasets and have already been analyzed using non-AI techniques to produce a land suitability map, and thus make a good controlled example.

Figure 1 shows the Process Flow Diagram for phase 1 and phase 2 of our proposed method. phase 1 involves processing vector-formatted data for use in AQ17, and phase 2 analyzes gridded or raster-formatted data.

Figure 1 Process Flow for phase 1 and phase 2 experiments.




Choosing examples. Positive and negative examples are chosen for use as AQ17 training datasets. Eight such suitable landuse areas or polygons are determined as a result of spatial analysis using ARC/INFO. These "suitable" polygons are used as templates to cut examples from each original coverage. These features and their attributes become the positive training examples. Eight randomly selected "non-suitable" example polygons are digitized and their original input data values are used for the negative training examples. The result will be a training dataset containing vector data values for every positive and negative training example. The query capability within ARC/INFO is used to create positive and negative datasets for desired attribute subsets. The INFO data produced is exported as ASCII flat files. These data are then dropped into an Excel spreadsheet and the columns manipulated to combine attribute data and values for each feature in an appropriate format for use by AQ17.

The initial intent was to retain several attributes in the AQ17 input dataset, but many attributes were deleted as beyond the scope of phase 1. The "area" attribute was deleted from the training set because when the negative examples were digitized in a somewhat random manner, most were of such a size that the 2000 square meter size minimum became a moot question. Also it was discovered that while the "suitable" polygons were homogeneous as a result of the spatial selection process, the negative examples cut through several different soil types and land uses. It would have required additional pre-processing to represent the areas of the negative polygon examples as a single value. The same was true for the "cost" attribute. The subsequent value of the information input from which AQ17 is to induce rules is therefore weakened by this necessity. Soil-code was also dropped as redundant to soil "suit" or suitability. Table 1 shows the data input file used for testing with AQ17 in phase 1.

Table 1. AQ17 data input file used for phase 1.

class_neg-events

| poly# | lu-code | strm-code | s-suit | rd-code | sewer |
|---|---|---|---|---|---|
| 2 | 2 | 2 | 2 | 0 | 0 |
| 2 | 4 | 2 | 2 | 0 | 0 |
| 2 | 2 | 2 | 2 | 0 | 0 |
| 3 | 4 | 2 | 2 | 1 | 0 |
| 3 | 2 | 2 | 3 | 1 | 0 |
| 3 | 4 | 2 | 2 | 2 | 0 |
| 3 | 2 | 2 | 2 | * | 0 |
| 3 | 3 | 2 | * | * | 0 |
| 3 | 4 | 2 | * | * | 0 |
| 4 | 4 | 0 | 2 | 1 | 2 |
| 4 | 1 | 0 | 1 | 1 | 2 |
| | | . | | | |
| | | . | | | |
| | | . | | | |
| 6 | 2 | 2 | 2 | 0 | 0 |
| 6 | 2 | 2 | * | 0 | 0 |
| 6 | 2 | 2 | * | 0 | 0 |
| 6 | 4 | 2 | * | 0 | 0 |
| 6 | 4 | 2 | * | 0 | 0 |
| 7 | 4 | 2 | 3 | 2 | 1 |
| 7 | 2 | 2 | 2 | 2 | 1 |
| 7 | 4 | 2 | 2 | 2 | 1 |
| 8 | 4 | 0 | 3 | 2 | 0 |
| 8 | 2 | 0 | 2 | 2 | 0 |
| 9 | 5 | 1 | 3 | 2 | 0 |

class_pos-events

| poly# | lu-code | strm-code | s-suit | rd-code | sewer |
|---|---|---|---|---|---|
| 2 | 3 | 0 | 2 | 0 | 0 |
| 3 | 3 | 0 | 2 | 0 | 0 |
| 4 | 3 | 0 | 2 | 0 | 0 |
| 5 | 3 | 0 | 3 | 2 | 2 |
| 6 | 3 | 0 | 2 | 0 | 0 |

| | | | | | |
|---|---|---|---|---|---|
| 7 | 3 | 0 | 2 | 0 | 0 |
| 8 | 3 | 0 | 2 | 0 | 0 |
| 9 | 3 | 0 | 2 | 0 | 0 |

Learning rules. When the training example datasets are processed with AQ17, the result is an induced rule set or hypothesis. (See Table 2). for rules induced by AQ17 in phase 1.

Table 2. Rules induced by AQ17 (phase 1).

```
Negative_rules
 #  cpx
 1  [lu-code=1,2,4,5]
 2  [strm-code=2]

Positive_rules
 #  cpx
 1  [lu-code=3] [strm-code=0]
```

AQ17 correctly determined that the suitable examples must have a land use of 3 (for brushland), but because only data indicating the presence or absence of a stream was provided instead of proximity to a stream, AQ could only determine that areas where there was no stream (0) would be suitable sites, and was unable to correctly eliminate sites that were too close to a stream to be suitable.

Application of rules. ARC/INFO coverages have a graphic or "arc" portion and a database or "info" portion. Prior to export, the "info" part of each entire data coverage is intersected with the landuse coverage polygons to give every coverage a common ID, the landuse#, with which to quantify and track the features. Attributes for each feature coverage are then queried and all original data from each data layer is exported in ASCII format. The import data is once again edited and features merged into a single input dataset. In the next step, the rules induced from the training dataset are applied to classify the complete datasets, including positive and negative examples and all other data.

Phase 1 results. Without pre-processing that was not within the scope of phase 1 of this project, it was not possible to satisfactorily handle original information concerning area or cost, nor was it possible to take advantage of the coordinate information inherent in GIS data. The ability to designate spatial areas within a certain proximity of a road or stream is easily accomplished within a GIS, but is not easily translatable for use by AQ17, and the negative examples chosen in the study were too large to test for area constraint. There were also problems associated with using vector-formatted data. It was initially easy to designate suitable polygons as features and determine their attributes. It was a little harder to designate and determine the attributes of the negative examples, and it was a real problem to figure out how to quantify data layers that in some cases were lines and in others were polygons. There was no common areal unit crossing all coverages as in the positive training examples. In the rules application part of the project, landuse polygons were somewhat arbitrarily chosen as analysis units and used to provide a polygon numbering scheme to facilitate analysis by AQ17. There was no justification for choosing these units instead of, say soils polygons, or some other arbitrary unit or grid, and this probably limited the granularity of the search. One possible solution might be to try a similar study using raster data layers. The unit of study would be pre-determined as pixels, and would be the same for all layers. This was performed in phase 2.

Grid-based data analysis (phase 2)

Data input. In phase 2, five data coverages (soils, landuse, sewer and stream buffer, cost/ha, and suitability) were gridded into 70 row x 54 column data layers (40-meter square cells). Each data layer had 3780 cells resulting in an 18,900 cell AQ17 input file. The fairly coarse grid cell size was chosen so that the resulting AQ17 input file would not be excessively large, but for this example, the induction process took less than 1 second so it is felt that larger input files could be easily handled. This opens up the potential for increasing both the size and number of the data layers, without greatly impacting the time spent in the learning process.

Learning rules. When the training example datasets are processed with AQ, the result is an induced set of rules. The training percentages used were varied to see how the amount of information provided in the training set affected the resulting rules and predictive accuracy. This is known as a learning curve. Generally the predictive accuracy of the rules will be initially very poor with small training set sizes, but will increase as the training set size nears 100%. AQ is guaranteed to completely and consistently cover the training set so it is guaranteed to reach the 100% mark when all of the data is used for training. As can be seen in figure 1, the predictive accuracy achieved the 100% accuracy mark from only 10% of the total data. This accuracy was maintained as the training percentage increased. The changes in the learned rules are also interesting to notice as the percentage of training data is varied. The learned rules for the 1%, 5% and 10-90% iterations are given below.

Results of phase 2.

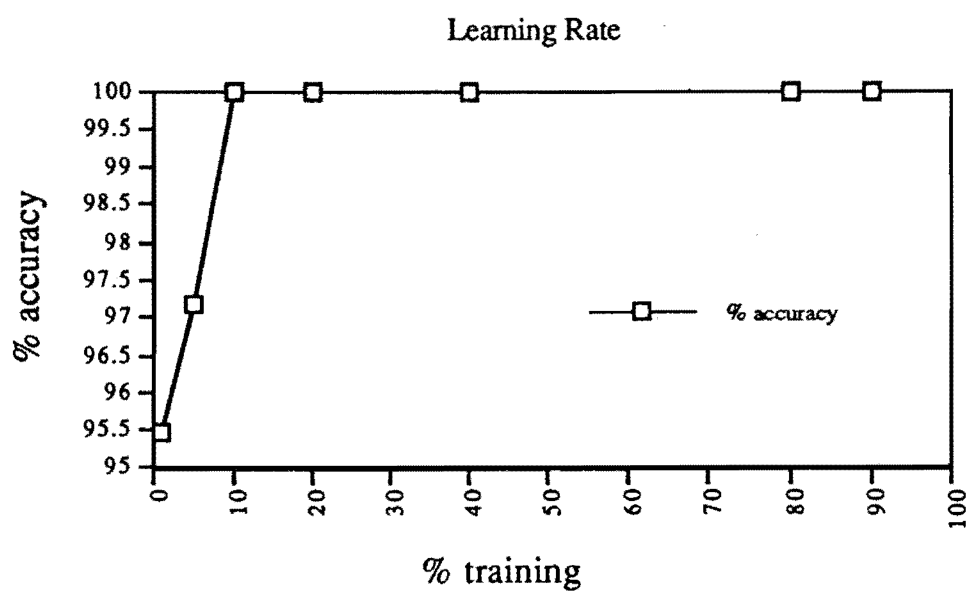


Figure 2. Learning rate for phase 2 experiment.

Negative-rules
# cpx
1 [sewer_and_stream=0]
2 [soils=0..2]
3 [landuse=2]

Positive-rules
# cpx
1 [sewer_and_stream=1] [landuse=3]

Table 3. Rules learned from 1% of the example data.

statement in $VL_1$. A selector relates a variable to a value or a disjunction of values, for example [temperature = cold], or [x < 5]. The general form of a selector (condition) is:

[L # R]

where L, called the referee, is an attribute, and R, called the referent is a set of values in the domain of the attribute in L, # is a relational symbol which can be one of the following: =, <,>,>=,<=,<>.

## Geographic information systems

A GIS is typically defined as a system having the capability to store, manipulate, and display spatial data. A GIS's strengths lie in its ability to link spatial and descriptive data and maintain the spatial relationships between the map features, integrating graphic and tabular data (ESRI 1992). It is extremely important to integrate approaches and procedures from other disciplines. This is because many other disciplines such as statistics, computer science, and database study focus on the same basic problems related to reasoning about spatially-indexed data, but have been around longer and may already have answers to some of the problems still perplexing GIS engineers. Cross-discipline communication and cooperation are not always easy, as it seems that there are barriers between disciplines such as terminology and familiarity with discipline-specific concepts.

Cross-discipline communications. Statisticians concentrating on statistical theories and concerns may not be aware of how techniques commonly used by them for years would help in GIS analyses. AI scholars concentrating on perfecting their algorithms and applying them to challenging benchmark examples may never see a GIS or think of how their hard won knowledge could be applied there. Even in the GIS field, techniques commonly used with tabular or vector-based data are not thought of when the data are derived from imagery; nor do imagery scientists or environmentalists always find it easy to communicate with geographers or statisticians.

Vector GIS. Even within the GIS field, there are communications breakdowns. Proponents of the two main formats of GIS, vector and raster, are as loyal to their choices as are Mac and PC owners. Some tasks are best handled in the raster format and some in the vector format, so what is needed is either a GIS engineer with a working knowledge of both, or, from the user's point of view, a system so user-friendly that it obviates the need to be aware of either format. The former is available for a price and the latter is at least now accepted by most in the GIS field as a goal. In a vector GIS, the Earth's features are mapped to a flat, two-dimensional surface as an x,y (Cartesian) coordinate or combination of ordered coordinates representing locations. A point, or single x, y coordinate might be a city, or school, or building; an arc is a series of ordered x,y coordinates representing linear features, such as drainage patterns or roads; and an area or polygon is a series of x,y coordinates defining arcs that enclose the area. An areal feature might be a soil class or land use classification. All the attribute information about a particular feature is stored in a database record.

The attributes can be seen as column headings in a feature table. If the feature table for soils were viewed, one might see attributes such as soil type, suitability, or sand content. These attributes store all the information about that particular feature in the database. These data files are known as feature attribute tables. Any two tables can be connected if they share a common attribute. A relate uses a common item to establish temporary connections between corresponding records in two tables, while a relational join relates and merges two attribute tables using their common item. Relate and join concepts are fundamental to vector GIS. When a spatial overlay is performed, it is really a spatial join that uses the location of the associated geographic features, rather than just using a common item in two tables.

Raster GIS. In a raster-based GIS, each data layer is represented by a regular grid of cells whose size depends on the resolution of the imagery or data being used. Each cell has a value. For an image, this value usually represents the electromagnetic intensity recorded by the remote sensor for the land area represented by that image pixel. If the data layer is a classified layer or thematic map, the pixel value is the class to which that pixel is assigned during the classification process. If a vector coverage is gridded, it is necessary to create a grid layer to represent each attribute you want to capture, since each pixel can have only one value. This type of representation is used most with scanned data, thematic map layers, and imagery (ESRI 1992).

AI/GIS software. In addition to AQ17, there exist many AI software packages, or GIS packages that use AI. Each has its own methodology and intended use. The MAPS spatial database is a large-scale image/map database system for Washington, D.C. that contains approximately 200 high resolution aerial images, a digital terrain database, and a variety of map databases from the Defense Mapping Agency. MAPS has been used as a component for an automated road finder/follower, a stereo verification module, and a knowledge-based system for interpreting scenes from aerial imagery (McKeown 1987). INDUCE 4 is a "general-purpose incremental inductive learning program that transforms symbolic descriptions of real-world events into more general and more useful descriptions of these events."(Bentrup, Mehler et al. 1987). INDUCE 4 learns from examples and is capable of incremental learning. INLEN was designed to analyze databases and discover patterns in them. It is made up of several integrated parts: a relational database, a knowledge base, and machine learning and inference capabilities. INLEN learns rules from examples, creates classifications, and discovers equations (Kaufman, Michalski et al. 1991). It was influenced by AQ15, a predecessor of the AQ17 software (Bloedorn, Wnek et al. 1993) which was described earlier in this paper, and is used for the project described in the remainder of this paper.

THE PROJECT

The objective of this project is to compare the results of conventional suitability analysis done in the Commercial-Off-The-Shelf (COTS) GIS software, ARC/INFO, with the results of analysis of the same data using the artificial intelligence software, AQ17. For the ARC/INFO analysis, constraints are provided by experts to guide the analysis process. This process results in the designation of suitable building sites as described below. For the AQ17 analysis, only original data for positive and negative examples of suitable building sites are provided. This process starts with examples, and induces a set of positive and negative rules that characterize the examples given. These rules should be very similar in content to the constraints provided for the GIS analysis. ARC/INFO is run on a SUN SPARCstation 10 with the UNIX operating system, and AQ17 is run on a SUN SPARCstation 2, also using UNIX. The suitability constraints provided with the ARC/INFO tutorial are as follows:

> A local university plans to construct a small lab and office building for performing research and extension projects in aquaculture. They've narrowed the area down to a coastal farming area near several small towns but need to select a specific site that meets the following requirements: The site must be at least 2,000 square meters in size, must overlie soils suitable for construction of buildings, not forested because of clearing expenses and because the regional agricultural plan prohibits converting farmland in this area, and the current land use must be brushland. A local ordinance designed to prevent rampant development allows new construction only within 300 meters of existing sewer lines and a recent national water quality act requires that no construction occur within 20 meters of streams (ESRI 1992).

Michalski, R.S. On the Quasi-Minimal Solution of the Covering Problem. International Symposium on Information Processing. Bled, Yugoslavia: 1969. A3 (Switching Circuits): 125-128.

Michalski, R.S. (1974). Variable-Valued Logic: System VL1. International Symposium on Multiple-Valued Logic. 323-346.

Michalski, R.S.; and Larson, J.B. Selection of Most Representative Training Examples and Incremental Generation of VL1 Hypotheses: the Underlying Methodology and the Description of Programs ESEL and AQ11. Dept. of Computer Science, University of Illinois, 1978.

Michalski, R.S.; and McCormick, B.H. Interval Generalization of Switching Theory. University of Illinois, 1971.

Robinson, Vincent B; and Frank, Andrew U. "Expert Systems for Geographic Information Systems." Photogrammetric Engineering and Remote Sensing 53 (1987): 1435-1441.

Schutzer, Daniel. "Theories and Concepts." In Artificial intelligence: An Applications-Oriented Approach, pp. 1-2, 86-93. New York: Van Nostrand Reinhold Company, 1987.

Smith, Terence R.; Menon, Sudhakar; Star, Jeffrey L.; and Estes, John E. "Requirements and Principles for the Implementation and Construction of Large-Scale Geographic Information Systems." International Journal of Geographical Information Systems 1 (1987): 13-31.

The positive rules in Table 3. state that the data of most importance in characterizing the suitable sites are the proximity to sewers and streams, and the brushland land use category. A value of 1 for the sewer_and_stream rule means that the positive example cell was far enough away from streams and close enough to a sewer line to meet suitability requirements; a value of zero means that this constraint was not satisfied. A landuse value of 3 stands for a cell having the desired brushland landuse type. Other values for landuse stand for other land use types. Because of the small percentage of examples used for this run, the soils feature seems to have had an influence on the makeup of the negative rules. Also AQ is only able to induce a negative rule for landuse land use category 2. As can be seen in Table 4. below, when the 5% of the examples is used, AQ is better able to categorize the landuse examples, both positive and negative.

| Negative-rules | Positive-rules |
|---|---|
| # cpx | # cpx |
| 1 [landuse=4..7] | 1 [sewer_and_stream=1] [landuse=3] |
| 2 [landuse=0..2] | |

Table 4. Rules learned from 5% of the example data.

The positive rules are the same for this run as in Table 3., but because of the larger number of examples provided to AQ17, the soils data are no longer necessary to discriminate between suitable and unsuitable sites, and are therefore dropped from the rules.

| Negative-rules | Positive-rules |
|---|---|
| # cpx | # cpx |
| 1 [sewer_and_stream=0] | 1 [sewer_and_stream=1] [landuse=3] |
| 2 [landuse=4..7] | |
| 3 [landuse=1..2] | |

Table 5. Rules learned from 10%, 20%, 40%, 80% and 90% of the example data.

The rules shown in Table 5. show the further refinement of positive and negative rules as a result of increasing the percentage of data examples used in the AQ17 analysis. These rules correctly categorize all of the example cells.


## ACKNOWLEDGMENTS

Both authors would like to thank Dr. Larry Kerschberg for his encouragement and support. We would also like to thank Michael Hieb for his comments and suggestions.

Eric is a research assistant in the Machine Learning and Inference laboratory at George Mason University. The Laboratory's research is supported in part by the National Science Foundation under grant No. IRI-9020266, in part by the Advanced Research Projects Agency under the grant No. N00014-91-J-1854, administered by the Office of Naval Research, and the grand No. F49620-92-J-0549, administered by the Air Force Office of Scientific Research, and in part by the Office of Naval Research under grant No. N00014-91-J-1351.


## CONCLUSIONS

The benefits of this simple project were several-fold. It was possible to select positive and negative training examples in ARC/INFO and import them into AQ17. Using this data, AQ17 was able to induce rules that successfully predicted some of the constraints that guided the original ARC/INFO suitability analysis: landuse type and relationship to streams and sewers. This concept, if refined could be used to "train" on sites about which there are no ready characterizations. The only information available to aid in analysis might be the existence and identification of these examples, such as toxic waste sites in inaccessible countries.

These rules were simple, but the rules generated by the AI software could be as complex as need be, because in this approach, only the computer needs to use them. However, experts may still analyze them and justifications for predicting can be made (as opposed to using a neural net). Another advantage of this approach is extraction of useful information without the time and expertise needed for some approaches. The examples are the experts. Data from imagery, cultural, economic, political, environmental, social , etc. could be represented as layers of a GIS. Further, there are already tremendous numbers of existing databases and GISs that, if integrated and made accessible, could provide data for such a project. If this were the case, the examples trained on might include data of many types and might help locate high risk areas for many types of problems. The learning process can be iterative and can be improved by entering some known rules at the start. The input can be weighted and the search for the best rules made more or less intense. Even if the rules are imperfect, they could still be used as a "cue" to highly probable areas.

This method could be automated using Arc Macro Language (AML) and/or C so that the user could graphically select positive and negative examples sites (well sites, weather collection stations, or anyplace difficult, expensive, or hard to access), collect data for only those sites through multiple data layers, and use AQ to induce rules and determine which data layers contributed most to the rules. Only the feature layers that contributed to the rules would be targeted for comprehensive data collection, making it unnecessary to collect data layers that are not important to the analysis. For example, in the phase 2 run, it was very time-consuming to prepare and transfer the soils information, but it turned out that it wasn't needed to solve the problem at hand. This was determined with data samples from only 10% of the total training examples, both positive and negative. Therefore it is unnecessary to collect soils data for the other 3000+ sites, assuming that the examples used were truly representative. This would save time and money.

This project was a good exercise in communication as the authors learned to see GIS as a practical application of machine learning and a supplementary GIS analysis tool. We feel that the integration and analysis problems remaining can be solved and that further coordination between our two disciplines is not only possible, but highly desirable.

## REFERENCES

Bentrup, John A.; Mehler, Gary J.; and Riedsel, Joel D. INDUCE 4: A Program for Incrementally Learning Structural Descriptions From Examples. University of Illinois at Urbana-Champaign, 1987.

Bloedorn, Eric; Wnek, Jamusz; Michalski, Ryszard S.; Kaufman, Ken; Imam, Ibrahim F.; and Bala, Jerzy. AQ17: A Multistrategy Learning System: the Method and User's Guide. 17 ed. Fairfax, VA: Center for Artificial Intelligence, George Mason University, 1993.

ESRI. Understanding GIS: The Arc/Info Method. 6 ed. Redlands, CA: Environmental Systems Research Institute, Inc., 1992.

Kaufman, Kenneth A.; Michalski, Ryszard S.; and Kerschberg, Larry. "Mining for Knowledge in Databases: Goals and General Description of the INLEN System." In Knowledge Discovery in Databases, pp. 525. Ed. Gregory Piatetsky-Shapiro and William J. Frawley. Menlo Park: The MIT Press, 1991.

McKeown, David M. "The Role of Artificial Intelligence in the Integration of Remotely Sensed Data with Geographic Information Systems." IEEE Transactions on Geoscience and Remote Sensing GE-25 (1987): 330-348.