Bayesian Classification of Insects

Automatic Detection and Classification of Insects

Based on preliminary experiments and the hardware constraints imposed by the domain (discussed elsewhere), we intended to use extensions of a Bayesian Network classifier for the classification of insect detected by our sensor. Below we briefly review Bayesian classifiers, and explain why we feel it is the perfect classifier for the task at hand.

Bayesian Classification

The Bayesian classifier (Duda et. al. 2002) is a simple classification method, which classifies an instance *j* by determining the probability of it belonging to class C_i. These probabilities are calculated as:

$$P(C_i \mid A_1 = V_{1_j} \,\&\cdots\&\, A_N = V_{N_j}) \qquad (1)$$

where an example is represented as attribute-value pairs of the form A_i=V_i. For concreteness, an attribute in our domain is anything we can measure about the insect or its environment. Examples of attributes are *thorax_length*, *wingbeat_frequency*, *time_of_day* etc. If there are *N* independent attributes, then the probability is proportional to:

$$P(C_i)\prod_k P(A_k = \mathrm{V}_{\mathrm{k_j}} \mid C_i) \qquad (2)$$

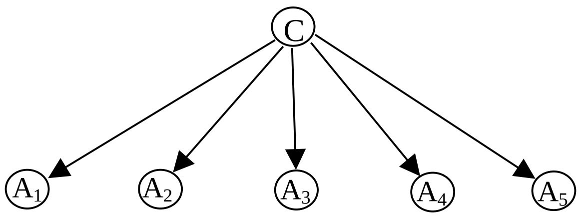

Figure 2 : An example of a Naïve Bayes Network

When this independence assumption is made, the classifier is called naïve Bayes (Ripley 1996). Figure 2 shows the typical illustration of a naïve Bayes classifier. In this case we are trying to predict the class C (i.e male or female, or *Culex pipiens* or *Aedes aegypti*), based on 5 attributes.

The direction of the arrows in the graph encodes the fact that the class depends on these five attributes, and the lack of any arrows between the attributes explicitly encodes the assumption that the attributes are independent (more precisely, the attributes are independent, *given* we know the class). For example, if A_1 is *thorax_length* and A_3 is *wingbeat_frequency*, this classifier explicitly assumes that these two things are unrelated. This is clearly an unrealistic assumption; larger insects tend to have lower wingbeat frequencies.

In spite of these assumptions which are clearly unrealistic for most real world problems, Naïve Bayes has been shown to be competitive with more complex, state-of-the-art classifiers (Dougherty 1995, Kohavi 1994). Nevertheless, if the attributes are known to be related, this can be explicitly encoded by calculating the joint probabilities. For example, before we assumed that attribute 3 (*wingbeat_frequency*) was independent of the rest of the features and thus could be calculated as $P(A_3 = \mathrm{V}_{3_j} \mid C_i)$. If we discover that this attribute depends on attribute 1 (*thorax_length*) we can adjust for this by calculating $P(A_3 = \mathrm{V}_{3_j} \mid C_i \wedge A_1 = \mathrm{V}_1)$, the probability of having some particular frequency *given* that size of the thorax is known. Figure 3 shows a visual representation of this.

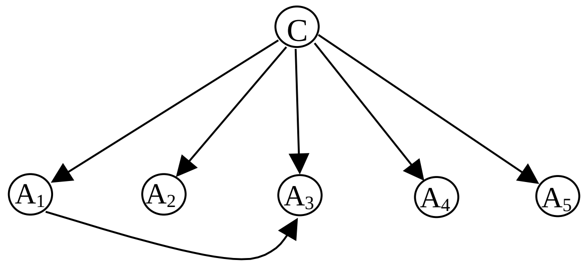

Figure 3: An example of a Bayesian Network

If we allow arbitrary topologies of the graph, the classifier is then known as a Bayesian network. Bayesian networks are known to be the optimal classifier for any problem (Pearl, J. *1988*), given that the correct structure of the graph is specified. In practice, finding the correct structure of the graph is very difficult, although many applicable heuristic algorithms exist (Keogh and Pazzani 1999).

Why Bayesian Classification is Right for Insect Identification in the Field

While there are a host of classification algorithms that could be used for the problem at hand (decision trees, neural networks, nearest neighbor etc), we feel that Bayesian networks outlined above are best. The following considerations lead us to this conclusion:

1) Our autonomous traps will have limited resources. In particular, they will have limited memory, CPU power and battery life. Bayesian networks (once constructed offline in the lab) require only O(|A|) time and space, where |A| is the number of different features. In other words Bayesian networks are extraordinary efficient in terms of both memory requirements and CPU time.
2) Unlike other classification methods that are essentially black box, Bayesian networks allow for the graceful introduction of user knowledge. For example, if we know that *wingbeat_frequenc*y and *thorax_length* are related, we can "tell" the algorithm this. This feature allows us to customize the algorithm to different tasks and locations. For example, if we deploy a trap in Hong Kong, we can encode the fact that the *time_of_year* attribute and the *temperature* attribute are dependent. Whereas if we deploy the trap in Thailand, we can encode the fact that in this location, these particular attributes are (essentially) independent.
3) Bayesian networks are among the most interpretable of classifiers. In other words, we can "ask" the algorithm why it made a particular classification. The algorithm can be made to respond with a structured natural language rule (Przytula and Thompson 2001.,Bouckaert, 2002). For example, when queried as to a particular decision, the algorithm might respond: "Insect #1242 was classified as *Culex pipiens* because *thorax_length > 0.5* ***and*** *500 < wingbeat_frequenc*y *< 750* ***and*** *time_of_day = morning*". This feedback can be particularly useful. By studying the incorrect classifications we may be able to understand why/when the classifier makes mistakes, and fix this by getting new attributes or changing our independent assumptions etc.
4) Bayesian networks simplify flagging anomalies. Most classifiers must make a classification decision, even if the object being classified is vastly different to anything observed in the training phase. In contrast, we can slightly modify the Bayesian Classifier to produce an "*Unknown*" classification. One or two such classifications per week could be ignored, but a spate of them could be investigated in case it is indicative of an infestation of a completely unexpected species.
5) Improving accuracy through sharing data is easily accomplished. It is well understood that the single best thing one can do to improve the accuracy of any classifier is to increase the number of training examples. Classifiers literally learn by experience, and like their human counterparts, the more experience they have the better they become. It has recently been discovered that classifiers can learn from both labeled and unlabeled data (Nigam et al. 2000). As noted elsewhere in this proposal, we will have many traps connected by a network. This fact suggests that the traps could share information to produce better classifications. However, if we use neural networks (or

virtually any other type of classifier), this would require the traps to store and transmit every single data item, an impossible demand on memory and bandwidth. In contrast, with the Bayesian classifier, each trap only records and transmits adjustments to a probability table, a trivial task given current hardware specifications.

References

R. R. Bouckaert. (2002) Low level information extraction: a Bayesian network based approach. TextML 2002

Dougherty, J., Kohavi, R., & Sahami, M. (1995). Supervised and unsupervised discretization of continuous features. *Proc of the 12th Int. Conference on Machine Learning*. (pp. 194-202). Tahoe , Ca: Morgan Kaufmann.

Duda, R, & Hart, P, & Stork, D. (2002).Pattern Classification, Second Edition. SBN: 0-471-05669-3

Keogh, E. & Pazzani, M. (1999). Learning augmented Bayesian classifiers: A comparison of distribution-based and classification-based approaches. In Uncertainty 99, 7th. Int'l Workshop on AI and Statistics, Ft. Lauderdale, FL, pp. 225--230.

Kohavi, R. (1994). Feature subset selection as search with probabilistic estimates. In *AAAI Fall Symposium on Relevance*.

Langley, P. & Sage, S. (1994). Induction of selective Bayesian classifiers. In *Proceedings of the Tenth Conference on Uncertainly in Artificial Intelligence.* p 399-406. San Francisco, CA: Morgan Kaufmann.

Madon MB, Hazelrigg JE, Shaw MW, Kluh S, Mulla MS.Has Aedes albopictus established in California? J Am Mosq Control Assoc. 2003 Dec;19(4):297-300.

Pearl, J. *1988 Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann.

Ripley, B. (1996). Pattern recognition and neural networks. Cambridge University Press.

Wojtek Przytula, Don Thompson (2001). Development of Bayesian Diagnostic Models Using Troubleshooting Flow Diagrams, Proceedings of SPIE 15th Annual Symposium on Aerospace/Defense Sensing, Simulation, and Controls AeroSense 2001; April 16-20, 2001