Appears in the KDD 1997 Workshop on
Issues in the Integration of Data Mining and Data Visualization

Visualizing the Simple Bayesian Classifier

Barry Becker Ron Kohavi Dan Sommerfield
Data Mining and Visualization
Silicon Graphics, Inc.
2011 N. Shoreline Blvd
Mountain View, CA 94043-1389
{becker,ronnyk,sommda}@engr.sgi.com

Abstract

The simple Bayesian classifier (SBC), sometimes called Naive-Bayes, is built based on a conditional independence model of each attribute given the class. The model was previously shown to be surprisingly robust to obvious violations of this independence assumption, yielding accurate classification models even when there are clear conditional dependencies. The SBC can serve as an excellent tool for initial exploratory data analysis when coupled with a visualizer that makes its structure comprehensible. We describe such a visual representation of the SBC model that has been successfully implemented. We describe the requirements we had for such a visualization and the design decisions we made to satisfy them.

Keywords: Classification, simple/naive-Bayes, visualization.

1 Introduction to the Simple-Bayesian Classifier

In supervised classification learning, a labeled training set is presented to the learning algorithm. The learner uses the training set to build a model that maps unlabeled instances to class labels. The model serves two purposes: it can be used to predict the labels of unlabeled instances, and it can provide valuable insight for people trying to understand the domain. Simple models are especially useful if the model is to be understood by non-experts in machine learning.

The simple Bayesian classifier (SBC), sometimes called Naive-Bayes, is built based on a conditional independence model of each attribute given the class (Good 1965, Duda & Hart 1973, Langley, Iba & Thompson 1992). Formally, the probability of a class label value C_i for an unlabeled instance $X = \langle A_1, \ldots, A_n \rangle$ consisting of n attribute values is given by

$$\begin{array}{ll} \mathrm{P}(C_i \mid X) & \\ = \mathrm{P}(X \mid C_i) \cdot \mathrm{P}(C_i)/\mathrm{P}(X) & \text{by Bayes rule} \\ \propto \mathrm{P}(A_1, \ldots, A_n \mid C_i) \cdot \mathrm{P}(C_i) & P(X) \text{ is same for all label values.} \\ = \prod_{j=1}^{n} \mathrm{P}(A_j \mid C_i) \cdot \mathrm{P}(C_i) & \text{by conditional independence assumption.} \end{array}$$

The above probability is computed for each class and the prediction is made for the class with the largest posterior probability. The probabilities in the above formulas must be estimated from the training set. This model is very robust and continues to perform well even in the face of obvious violations of this independence assumption (Domingos & Pazzani 1996, Kohavi & Sommerfield 1995).

We begin with a discussion of our motivation and requirements for the SBC visualization. We then describe it in detail and then explain why we made certain design decisions.

2 Motivation and Requirements for Visualization

The ability to describe the structure of a classifier in a way that people can easily understand transforms classifiers from incomprehensible black boxes to useful tools that convert the data into knowledge. One advantage of the SBC is that it uses a fairly simple model that users can easily understand.[1] We now describe the visual representation of the SBC model that has been successfully implemented in SGI's MineSet data mining product under the name *Evidence Visualizer.*

Classification of data without any explanation of the underlying model reduces the trust of users in the system and does not help the knowledge discovery process. For example, Spiegelhalter & Knill-Jones (1984) reported that physicians would reject a system that gave insufficient explanation even though it had good accuracy. A visualization accompanying an induced classifier provides a way for users to understand the model used in the classification. A human may choose to reject a classification or the whole model if he or she realizes that

[1] Some SAS users we talked to claimed that CROSSTAB, a procedure for generating cross-tabulated counts, was probably the most frequently used procedure in SAS; the SBC provides a similar function. We believe the visualization may be far more useful than textual tables.

the classifier is basing its decision on factors that are not significant or relevant, or that the classifier is ignoring crucial factors.

In constructing our visualization of the structure of the simple Bayesian classifier, we had a number of design requirements:

1. Users with very little knowledge of statistics should be able to quickly grasp the primary factors (attributes and values) influencing classification.

2. Users should be able to see the whole model and understand how it applies to records, rather than the visualization being specific to every record as was done in the evidence balance sheets described in Spiegelhalter & Knill-Jones (1984, Tukey's discussion) and Madigan, Mosurski & Almond (1997). Showing users the complete model provides a much more powerful knowledge discovery tool.

3. Users should be able to compare the relative evidence contributed by every value of every attribute.

4. Users should be able to see a *characterization* of a given class. We define a characterization for a class as being a list of attribute values or ranges that differentiate that class from others.

5. Users should be able to infer record counts and confidence in the shown probabilities so that the reliability of the classifier's prediction for specific values can be assessed quickly from the graphics. The precise numbers can always be made accessible through interaction with the visualization, but most of the time a visual cue should be what prompts a user to desire such a number.

6. The system should handle many attributes—on the order of hundreds—including attributes with hundreds of values without creating an incomprehensible visualization or a scene that is impractical to manipulate.

7. Users should be able to interact with the visualization to perform classifications. Specifically, users should be able to classify data directly in the visualizer and watch the predictions change as they select values for attributes.

Given the above desiderata, we constructed a visualization called *Evidence Visualizer.*

3 Our Proposed Visualization

The Evidence Visualizer displays the structure of the SBC and allows users to interact with it, examine specific values, show probabilities of picked objects, and ask what-if questions. Figure 1 and 2 show the two possible displays that users see.

There are two panes in the Evidence Visualizer. The right pane shows a large pie with prior probabilities for the possible label values. As the user interacts with the visualization by choosing values for attributes, the slices update to show the posterior probabilities. The left pane consists of rows of pie charts, one for each attribute. The attributes are sorted in

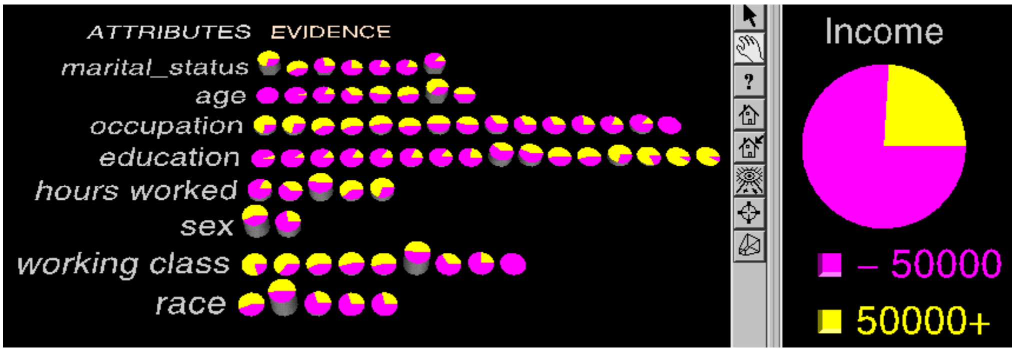


Figure 1: The evidence visualizer's pie chart display of the SBC model. The height of each pie represents the number of instances for that value or range.

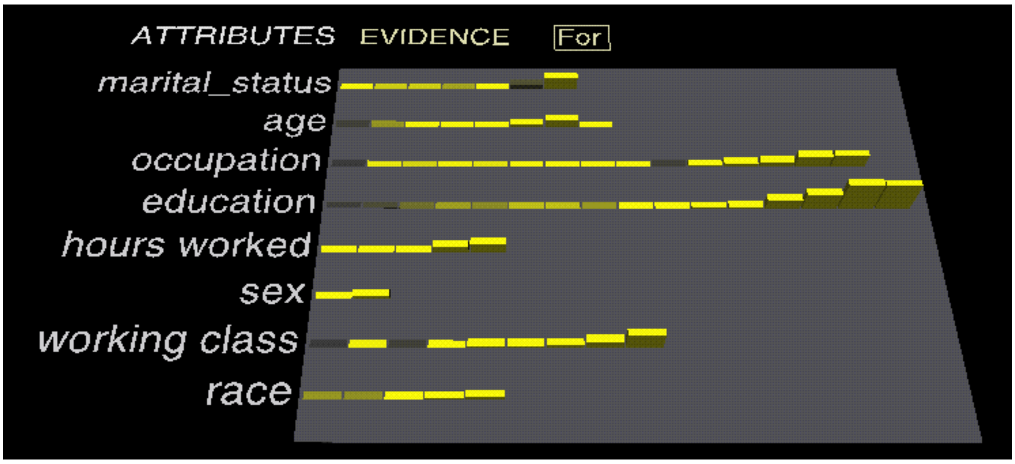
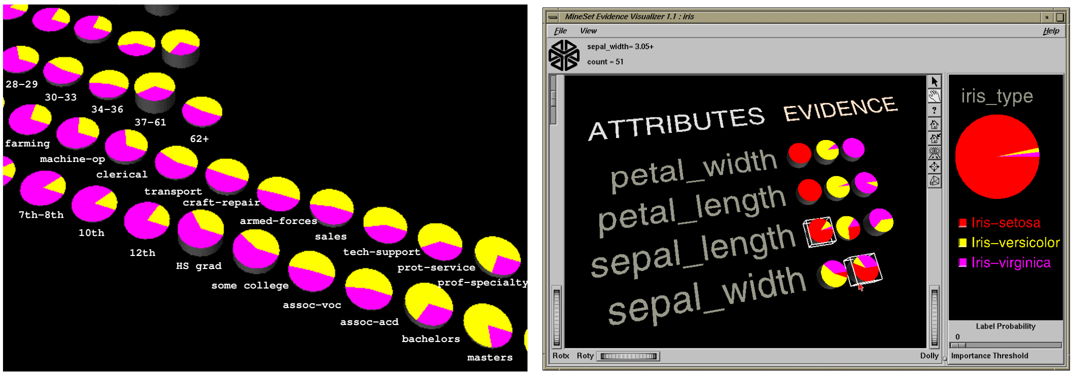


Figure 2: The evidence visualizer's bar chart display of the SBC model. The height shows evidence for the selected class. The bars become less saturated as the number of instances decreases, signifying a wider confidence interval.



sepal_width= 3.05+

count = 51

Figure 3: Closeup on some attribute values (left) and selection of specific value ranges to see the posterior probability (right). Users can highlight a pie chart by moving the cursor over it. A message then appears at the top showing the attribute value (or range) and the number of instances having that value (or range). The pie's height is proportional to this number. Pointing to the items in the legend on the right pane, shows the numerical probabilities corresponding to the slice size.

order of importance computed as the conditional entropy (Cover & Thomas 1991) of each attribute and the label. The left pane will switch to a bars representation of evidence for a label once a specific label is selected. There is one pie chart or bar for each discrete value or range of the attribute.

3.1 The Pie Chart Representation

In the pies representation on the left, the height of a pie is proportional to the number of instances having that attribute value. The sum of the pie heights for every row is constant. The slices of the pies represent the **evidence**, defined as the normalized conditional probabilities of an attribute value (or range) A, given a class label C_i of n possible classes. Specifically, the size of each slice for attribute representing class label value C_i is

$$P(A \mid C_i) \,/\, \sum_{i=1}^{n} P(A \mid C_i) \; .$$

The size of a pie slice indicates the amount of evidence to "add" to the class label matching the slice if we know that an instance has the given attribute value that matches the specific pie. If the slices are of equal size for an attribute value, knowing that an instance has this specific attribute value adds equal evidence to all classes, indicating that the posterior probability will not change, and thus this attribute value is irrelevant according to the SBC model.

Users may interact with the visualization by selecting values for the attributes and observing how the posterior probability (pie chart on the right) changes. For example, selecting the pie for *sepal-length* < 5.45 inches and the pie for *sepal-width* > 3.05 inches shows that an iris with these characteristics is probably an iris-setosa (Figure 3).

3.2 The Bar Chart Representation

The bars representation gives evidence that is *additive* as opposed to multiplicative. Formally, starting from Equation 1 repeated below:

$$\mathrm{P}(C_i \mid X) \propto \prod_{j=1}^{n} \mathrm{P}(A_j \mid C_i) \cdot \mathrm{P}(C_i) \; ,$$

we can take negative logs of each side (a small epsilon is added to avoid logs of zero if a Laplace correction is not applied). Class i with the smallest value of $-\log\left(\mathrm{P}(C_i \mid X)\right)$ is predicted, which is equivalent to the class i with the smallest value of the following expression

$$-\sum_{j=1}^{n} \log\left(\mathrm{P}(A_j \mid C_i)\right) - \log\left(\mathrm{P}(C_i)\right) \; .$$

In this mode of the *Evidence Visualizer*, which displays evidence against the selected class, each bar's height is proportional to $-\log\left(\mathrm{P}(A_j \mid C_i)\right)$ and the base height (representing the prior evidence) on which all bars stand as proportional to $\log\left(\mathrm{P}(C_i)\right)$. This shows evidence *against* each class because the class with the smallest sum is predicted. In a complementary mode, showing evidence for the selected class, the bar height shows the sum of the log-evidence against *all other* classes as shown in Figure 2. A bar for class i is then proportional to

$$-\log\left(1 - \mathrm{P}(A_j \mid C_i)\right)$$

and the base is proportional to $\log\left(1 - \mathrm{P}(C_i)\right)$. This mode was found to be more intuitive to users.

This kind of representation is excellent for characterizing a class of interest. If one selects a different class, the heights and colors of the bars change to represent the new class. The colors become less saturated (*i.e.*, grayer) if the confidence interval for the estimated evidence is large, signifying that the estimate is based on a small number of instances. The tool is interactive and users can highlight a bar by moving the cursor over it. Once a bar representing A_i is highlighted, statistical information is shown, including $P(C_j \mid A_i)$, $P(A_i \mid C_j)$, the additive evidence, 95% confidence intervals for the probabilities, and the instance count.

The additive evidence can be interpreted as the information content in bits (Cover & Thomas 1991). High evidence values will increase the class posterior probability more. This evidence can be summed in order to determine which class is being predicted by the model (unlike probabilities, which must be multiplied). This is analogous to a race between runners, each representing a class. Each time an attribute value is selected, each runner for each class is advanced by the corresponding amount of evidence. The predicted class is represented by the runner that advanced the furthest.

Because only the relative distances are important, we found it useful to subtract the evidence of the class with the smallest evidence from the rest. This is analogous to measuring relative distances from the slowest runner. When there are values that add similar evidence to all classes, the bar heights will be low. In two class problems, every attribute value will have at most one bar with a positive height. This method accentuates the importance of differences in evidence as opposed to the absolute values.

The visualizer orders the values (or ranges) depending on the attribute type. If the attribute is ordered (continuous ranges are commonly discretized) there is a natural order defined. If the attribute is nominal, there are a variety of ways that the visualizer provides to order the values:

1. The values could be sorted alphabetically. Sorting values alphabetically can aid in locating specific values when many are present. For example, when an attribute describes the country of birth.

2. The values can be sorted by decreasing number of instances with the leftmost values having the greatest height and decreasing to the right. The values further to the right would have less statistical significance and can be ignored by the user or not drawn below some threshold count.

3. The values can be sorted by decreasing size of conditional probability value for a specified label value. This makes the values that give the most evidence for the given class toward the left. As can be seen for the education attribute shown in Figure 1 and 3 (left), this often has the effect of ordering an attribute in a natural way. The user selects the class value to use for sorting, with a default to the value with the largest prior probability.

4. A method which we have not yet implemented involves automatic grouping of values. If certain values result in similar conditional probabilities, we can group them together in a single pie.

We have found the visualizer to be very useful in aiding knowledge discovery and understanding patterns. Customers of MineSet reported that end users find it useful and easy to understand.

4 Design Decisions

Our proposed visualization satisfies the requirements outlined in Section 2. We now describe in detail some of the choices made and the reasons for making them.

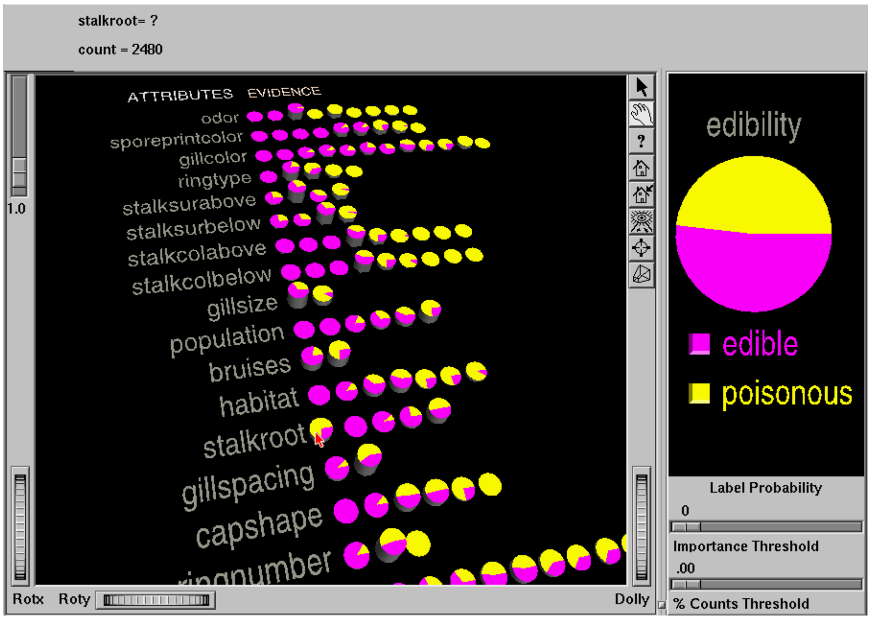
We found that a three dimensional representation is the best way to accommodate large numbers of attributes and values. Three dimensional navigation combined with perspective allows the user to closely examine an area of interest while maintaining context in the whole visualization. The predicted distribution of classes was included in a separate two dimensional pane to make it readily accessible.

We display probabilities as both pies and bars because each representation has distinct advantages in helping a user understand the operation of the simple Bayesian classifier. We found that a pie chart was optimal for showing probabilities because the angles subtended by the slices and representing relative conditional probabilities always sum to one. This allows us to easily display a full probability distribution in a small space. We recognize some of the drawbacks of pie charts noted by Tufte (1983, p. 178), but in our case they have several advantages that outweigh their disadvantages. Pie charts are used pervasively to represent distributions in magazines and newspapers making them understood and recognized by everyone. The use of pie charts allows many probability distributions to be displayed simultaneously, something we have not been able to achieve with other graphical representations.

Because the pie charts are of the same size and laid out on a line, they are similar to charts used by Consumer Reports to represent product quality through a set of circles filled with varying amounts of green and black to signify good and bad aspects respectively. Tufte (Tufte 1983, p. 174) lauded this approach as “a particularly ingenious mix of table and graphic.”

The use of the third dimension allows us to show the number of records underlying a particular distribution as the height (z coordinate) of the pie chart. This display gives users a quick way to gauge the reliability of any given distribution by rotating the scene.

The pies representation is strongest when the user is interested in distributions over all classes. The alternate representation, using bars, is more useful when the user is interested in properties of a specific class, and its use during interactive classification closely parallels the use of an evidence balance sheet to explain a result (Spiegelhalter & Knill-Jones 1984). The use of log-probabilities for the bars coupled with subtracting the minimum bar height from all bars makes this representation ideal for understanding the effect each attribute value has on the final prediction. Prediction is accomplished simply by adding up bar heights (evidence) corresponding to the users choices for each class, and picking the class with the highest total evidence.

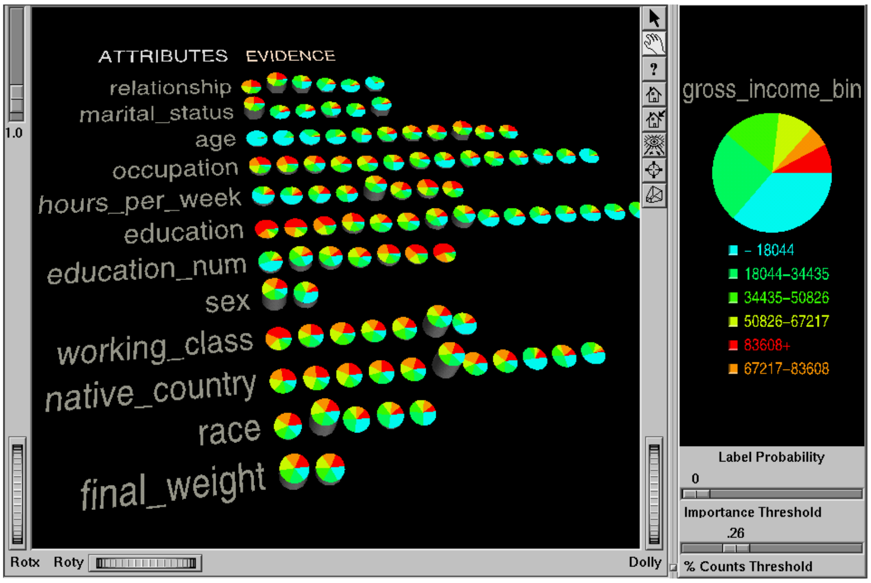

stalkroot= ?

count = 2480

Label Probability 0

Importance Threshold .00

Figure 4: An example of the evidence visualizer for the Mushroom dataset, where the goal is to determine mushroom edibility. One can see that Odor on the top was ranked as an excellent discriminator; the values are perfect discriminators except for the value "none," which is represented by the third pie from the left. The pointer points to the first value of Stalkroot, which is slightly offset to help the user understand that the first value is a missing (null) value.


Label Probability 0

Importance Threshold .26

Figure 5: An example of a binned attribute used as label. The goal to understand factors affecting income. Because the label—gross_income—was binned and has a natural ordering, the colors assigned form a continuous spectrum from green to red, where red is the highest range. Also note that the class labels on the right are arranged according to their proportion of the expected probability distribution; The largest slice appearing at the top. One can quickly see that salary rises with age.

Nominal attributes with many values pose problems because they flood the visualization with a large number of pies which may be difficult to interpret or compare. We provide two mechanisms for solving this problem; first, we order the values based on user preference (alphabetical, counts, probabilities). Second, we provide an option to remove values based on less that a user-specified small percentage of the data, typically 1% or less, thus eliminating the least significant portions of the scene. These two mechanisms combined make it easier for users to focus on the key values.

It is very important to display reliability of the probability estimates through the visualization. The pie charts display accomplishes this by showing the number of records behind each distribution as the height; higher pies are more reliable. The bars mode uses height to show evidence, so we had to pick an alternate method. We lower the saturation of a bar as its confidence worsens, effectively graying out bars with little support. This display of confidence is essential in the bars mode because probability distributions estimated from little data tend to be heavily skewed, resulting in large bars which pop out during visualization. Making these bars gray prevents them from heavily influencing the user.

If there are many classes it can be difficult to locate the name of the class that is predicted. For this reason, the class labels listed on the right pane are ordered by decreasing prior probabilities (Figure 5). The exact probabilities can be determined by moving the cursor over a particular class label.

Our visualization is also designed to handle unknown (null) values. Because nulls are handled differently in the classification process, probability distribution pies representing null attribute values appear in a special leftmost column, offset slightly from the other values. They also may not be selected by the user; to leave an attribute's value undefined the user simply leaves all attribute values in a specific row unselected. For an example of unknown handling, see the first value of attribute "stalkroot" in figure 4.

In some cases the classes are an ordered list. This usually occurs as a result of binning a continuous variable in order to use it as a class label. In this case the class colors are not assigned randomly: a continuum is used to indicate the low to high ordering of the classes. Users can hence easily identify values which lend strong evidence for predicting a class at one end of the range or the other as shown in Figure 5.

To allow the user to see datasets with hundreds of attributes, we compute the *importance* of each attribute and display them in the scene ordered by this measure. Technically, the importance value is conditional entropy (Cover & Thomas 1991) of each attribute and the label. Attribute with low conditional entropy have little effect on the posterior probability. A slider button on the bottom right (see Figure 1) allows users to remove lower-ranked attributes from the visualization.

To aid understanding with respect to the actual data used to build the classifier, it is possible to *drill through* to the actual instances which produced certain graphics. For example, a user may select the pies corresponding to education=masters and occupation=clerical and see the instances in the dataset that have these two values.

5 Summary

We described a visual representation of the SBC model that has been successfully implemented and can help one to understand the underlying model and the importance of specific attributes and attributes values in the classification process. The visualization gives insight into how classification is done, as well as allowing users to answer what-if questions through interactions with the visualization, something we found very useful.

We described the desiderata for such a visualization and the specific design decisions that we made to meet the requirements.

Acknowledgments We would like to thank everyone on the SGI MineSet team, especially Joel Tesler and Gerald Sangudi who helped with the conception of the *Evidence Visualizer*.

References

Cover, T. M. & Thomas, J. A. (1991), *Elements of Information Theory*, John Wiley & Sons.

Domingos, P. & Pazzani, M. (1996), Beyond independence: conditions for the optimality of the simple Bayesian classifier, *in* L. Saitta, ed., 'Machine Learning: Proceedings of the Thirteenth International Conference', Morgan Kaufmann, pp. 105–112.

Duda, R. & Hart, P. (1973), *Pattern Classification and Scene Analysis*, Wiley.

Good, I. J. (1965), *The Estimation of Probabilities: An Essay on Modern Bayesian Methods*, M.I.T. Press.

Kohavi, R. & Sommerfield, D. (1995), Feature subset selection using the wrapper model: Overfitting and dynamic search space topology, *in* 'The First International Conference on Knowledge Discovery and Data Mining', pp. 192–197.

Langley, P., Iba, W. & Thompson, K. (1992), An analysis of Bayesian classifiers, *in* 'Proceedings of the tenth national conference on artificial intelligence', AAAI Press and MIT Press, pp. 223–228.

Madigan, D., Mosurski, K. & Almond, R. G. (1997), 'Graphical explanation in belief networks', *J. Comp. and Graphical Statistics* p. to appear.

Spiegelhalter, D. J. & Knill-Jones, R. P. (1984), 'Statistical and knowledge-based approaches to clinical decision-support systems, with an application in gastroenterology', *Journal of the Royal Statistical Society A* **147**, 35–77.

Tufte, E. R. (1983), *The Visual Display of Quantitative Information*, Graphics Press, Cheschire, CT.