Visualizing High-Dimensional Predictive Model Quality

Penny Rheingans
University of Maryland, Baltimore County
Department of Computer Science and Electrical Engineering
rheingan@cs.umbc.edu

Marie desJardins
SRI International
Artificial Intelligence Center
marie@ai.sri.com

Abstract

Using inductive learning techniques to construct classification models from large, high-dimensional data sets is a useful way to make predictions in complex domains. However, these models can be difficult for users to understand. We have developed a set of visualization methods that help users to understand and analyze the behavior of learned models, including techniques for high-dimensional data space projection, display of probabilistic predictions, variable/class correlation, and instance mapping. We show the results of applying these techniques to models constructed from a benchmark data set of census data, and draw conclusions about the utility of these methods for model understanding.

1 Introduction

Discovering interesting information in large, high-dimensional data spaces is a challenging problem. Using inductive machine learning techniques to construct classification models has proven to be one useful approach for solving this problem. A typical machine learning application involves a great deal of manual effort to iteratively construct a representation of the domain (feature engineering), set the parameters of the learning algorithm, induce a set of models, and analyze the resulting models. To support this process, we have developed a set of visualization methods with the goal of improving a user's ability to evaluate the quality of learned models.

Traditional model analysis methods primarily consist of numerical and statistical tools for assessing the quality of a learned model. These tools include classification accuracy, confusion matrices, and receiver operating characteristic (ROC) curves. Our visualization techniques provide a richer representation of the information that the statistical tools summarize by a single number or curve, and are meant to augment, not replace, these statistical tools. To that end, we discuss in this paper how the visualization methods can be used to gain insights into how the behavior of the model varies across the data space. These insights could be used to guide the application development process by pinpointing, for example, regions of the data space (groups of individuals) with high misclassification rates, thus helping the user to determine what additional data to gather, or how to modify the set of features to improve differentiation.

2 Induced Predictive Models

A model is a description of how the world is expected to behave. Typically a model describes the aspects of the world that are relevant to a specific task: e.g., diagnosing a disease, predicting credit risks, or classifying documents by topic area. Here we focus on *classification* tasks, which have the form "Given an object description, classify it into one of *k* classes." Classification methods can be used for both prediction and diagnosis (e.g., "Given an applicant's characteristics, predict whether they will default on a loan," or "Given a patient's symptoms, determine what disease is affecting them"). Probabilistic classification methods give the *probability* of class membership, which is particularly useful in domains containing uncertainty, noisy data, or incomplete object descriptions.

In classification problems, one of the variables is a distinguished *class variable*; we refer to the other variables as *input variables*. (The class variable can be thought of as the dependent variable; the input variables as the independent variables.) The *data space* is the *n*-dimensional space defined by the *n* input variables. In a classification task, the goal is to derive the *class probabilities*, i.e., the marginal probabilities that an instance belongs to each class, given values for some (or all) of the input variables.

The problem of accurately predicting class membership from available information is a key challenge of knowledge discovery. A wide variety of methods have been developed by machine learning and data mining researchers to solve this problem, ranging from decision-tree learning algorithms to nearest-neighbor techniques to Bayesian learning methods. The visualization techniques we have developed are applicable to any learning methods whose output makes predictions that can be interpreted as probabilities, such as probabilistic decision trees or Bayesian networks. In the examples given in this paper, we used the ADULT data set from the UCI Machine Learning Repository [UCI 1999], which is derived from U.S. Census data, to construct classification models. We applied Tree-Augmented Naive Bayes (TAN) [Friedman and Goldszmidt 1996], a Bayesian network learning system that is tailored for classification, to construct the models. Data instances contain fourteen variables (six continuous and eight nominal) and a binary class label indicating income level ($>$ 50K (higher-income) or $<=$ 50K (lower-income)). Input variables include age, sex, race, education, occupation, hours worked per week, native country, type of employer, marital status, and household type. Using a subset of eight of the input variables, we used TAN to construct (from the training data) a model to predict income.

We have identified a set of *model characteristics* that may be visualized in order to understand and analyze a model's behavior. Some of the characteristics we have explored are *class probability* at each point in the data space; the *decision boundary* that delineates the region of instances that are predicted to be members of a class; *misclassifications*, which correspond to instances whose actual class label does not match the predicted class label; *misclassification types*, e.g., false positives and false negatives for binary classes; and *meta-attributes* of the model, such as the distribution and density of the training data used to build the model, and the confidence assigned to each estimate.

3 Traditional Model Analysis

A number of measurements have been developed by machine learning and statistics researchers to assess model performance. The most commonly used are classification accuracy, confusion matrices, and receiver operating characteristic (ROC) curves. In most machine learning research, model evaluation focuses on a single metric, such as classification accuracy. Classification accuracy is a single number that indicates the percentage of correctly classified instances in a test set. For the example model shown here, predictive accuracy on the test set is 81 percent.

Test data		Predicted class label	
		lt50K	gt50K
Actual class label	lt50K	10124	1236
	gt50K	1526	2174

Table 1: Confusion matrix for the census domain

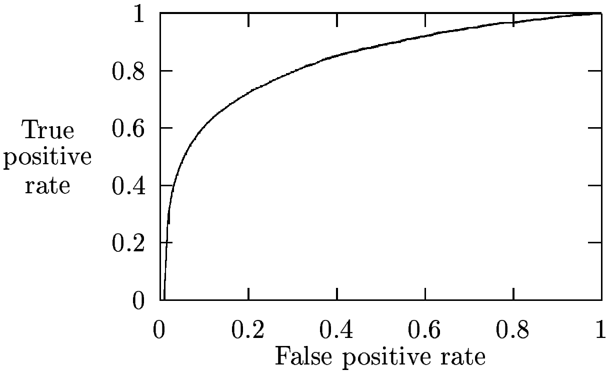

Figure 1: ROC curve for the census domain

A *confusion matrix* is often used to show the types of misclassifications made by a model. The confusion matrix (Table 1) is a two-dimensional table that indicates actual class label along one dimension and predicted class label along the other dimension. Each matrix entry indicates how many instances with the corresponding actual class label were predicted by the model to have the corresponding predicted class label. Entries along the diagonal correspond to correctly classified instances. For a binary class, there are two off-diagonal entries, corresponding to false positives (negative instances with a predicted positive label) and false negatives (positive instances with a predicted negative label).

ROC curves are used to assess the performance of the model as misclassification costs are varied. By changing the prediction threshold, a given model can be biased towards making more false positive predictions (lowering the threshold) or more false negative predictions (raising the threshold). The ROC curve (Figure 1) plots the false positive rate against the false negative rate.

None of these methods really address the question of *when* the model performs poorly, specifically what sort of instances tend to be misclassified. Visualization of model characteristics in the context of the data space complements the statistical tools by providing better insights into the nature of the model's performance.

4 Related Work

Although many researchers have studied techniques for visualizing data sets, and others have developed techniques to view model structure directly, there has been relatively little effort focused on visualizing learned models in the data space. A notable exception is the MineSet data mining package [SGI 1999], which includes several techniques for visualizing models, such as scatterplotting of misclassified instances. The display space is generated by manual feature selection, so the behavior of the complete model can be difficult to perceive. Visualization of classifiers in the MineSet paradigm was described by [Becker 1998].

A wide variety of techniques have been developed to perform dimension reduction of high-dimensional data. These include parallel coordinates [Inselberg and Dimsdale 1990], multiparameter icons [Pickett et al. 1990], and a host of interactive techniques developed by dynamic statistics researchers [Cleveland and McGill 1988]. Many of these approaches only work for discrete-valued variables.

There are also other techniques that produce clusters in 2D space based on the similarity of data instances have been used to perform the projection of high-dimensional data to 2D. These techniques include multi-dimensional scaling [Cox and Cox 1994] and relevance maps [Assa *et al.* 1997]. Other applications of SOM techniques to information visualization include the visualization of customer characteristics [Rushmeier *et al.* 1997].

5 Model Visualization

Information visualization applications are typically characterized by non-spatial, high-dimensional, non-continuous data spaces. Visualizing such data requires transforming the data to a spatial display space, then applying representation techniques to the transformed data points. The model characteristics we are visualizing are non-spatial and high-dimensional, but the dimensions are a mixture of continuous and nominal variables. Thus, while this application has a number of typical information visualization characteristics, the continuous nature of the data necessitates some additional design choices, particularly in the projection process.

In all of the visualizations shown here, we use a background colormap in saturations of green to show the class probabilities at each point in the display space. Locations with high class probabilities are bright green; locations with low class probabilities are black.

5.1 Data Space Projection

A computer monitor or printed page can only display two dimensions. These two dimensions can be intuitively extended to three with the addition of computer graphics 3D shape cues such as obscuration, perspective, and view point control. While we can represent additional high-dimensional coordinate information using other display parameters, such as time, color, size, or opacity, these more abstract display parameters are not as easily or directly interpreted as spatial position. In practice, the display space is only three-dimensional; therefore, high-dimensional data must be projected down to three dimensions to be visually represented.

One obvious, and common, method for the projection of high-dimensional data into a lower-dimensional display space is the selection of a subset of available dimensions. The plane of the projected space corresponds to an axis-parallel plane through the data space, with all points orthogonally projected onto the plane.

Figure 2 shows a 2D display space created using feature selection. The vertical axis is education; the horizontal axis, hours worked. Each location in this display space represents a high-dimensional subspace where education and hours worked are fixed, but other values can vary over their entire range. Feature selection has the advantages of being simple to perform and intuitive to understand. Unfortunately, the straightforward feature selection display often does not adequately capture the complex structure of the model in the high-dimensional data space, since instances with very different characteristics along other dimensions are aggregated.

In order to achieve a display space that better represents the high-dimensional structure of the data space, we have also used a set of projection techniques based on self-organizing maps (SOM) [Kohonen 1997]. Figure 3 shows a representation of the model where the data space has been projected to two dimensions using a SOM. In a SOM, neighboring locations in the display space correspond to neighboring locations in the data space, unlike feature selection where points far apart in the data space can map to identical locations in the display space. We are currently performing data space projection using a public-domain package that implements self organizing maps [Kohonen *et al.* 1996].

A SOM is constructed from a set of discrete instances, commonly the data set to be explored. In our application, the continuous data space must be sampled to yield a discrete set of instances.

We have experimented with constructing SOMs from the input instance set used to construct the model, the test set of instances used to perform traditional model analyses, and a set of samples generated from the model, which reflects the population characteristics of the input set. The latter was used to create the SOM in Figure 3. Each of these choices produces a map that is specific to a particular model. Alternatively, one can build a “model-neutral” SOM using a set of constructed instances that regularly sample the entire data space, as seen in Figure 4. This includes instances that, while not impossible, may be extremely unlikely. Note that the high-probability region is much less contiguous in this visualization than in Figure 3. The model-neutral map allocates display space relatively equally to all regions of the data space, rather than predominantly to regions of the data space corresponding to instances that are more likely to appear in the data.

5.2 Display of Model Characteristics

The *decision boundary* (in this case, the boundary between positive and negative predictions when a prediction threshold of 50% is used) is shown by a white line. Figure 2 shows the probability distribution for 2D feature selection. Since each point in the display space is equivalent to an attribute vector with missing values, the class probability can be computed by marginalizing over the missing values. Figure 3 shows the probability distribution for the SOM projection. When building the SOM from the model, probability is treated as just another variable for similarity clustering, resulting in maps that typically have high coherence of predicted probabilities.

5.3 Display Space Interpretation

The most difficult aspect of projecting high-dimensional spaces into a 2D display space is understanding the correspondence of the projected space to the original data space. In Figures 5 and 6, attribute contours are overlaid on the probability distribution to show how the input variables correlate with the predicted class and with each other. These contours show how the Cartesian grid of the data space has been warped by the projection process. The 2D feature selection projection (Figure 6) is easy to understand, but the attribute contours are useful as an indicator of the scale of the variables. In this projection, each attribute contour represents a data space hyperplane that is orthogonal to the 2D display plane. The correlations of the selected input variables with the class are apparent: individuals with more education (towards the top) and who work more hours (towards the right) tend to make more money, with the education correlation being somewhat stronger. (Notice that working *too* hard doesn’t really pay off: a good lesson for us all.)

The attribute contours are even more important in the SOM projection, since there is no intuitive interpretation of the projected space. Figure 5 shows the education and hours worked contours. These contours show how severely the hyperplanes of constant attribute value in the data space are distorted through the projection process. In this projection, the hyperplanes represented by the attribute contours are not generally orthogonal to the display space, and may not even be represented by contiguous contour lines. As before, notice that predicted high-income earners tend to have greater education levels (higher saturation of blue in the contours) and work more hours (red saturation). Figure 7 shows contours of another pair of attributes, those of education and sex.

6 Visual Model Analysis

By providing a visual representation of the model characteristics, our visualization methods add depth to a user’s understanding of traditional statistical model analysis measurements. Seeing the number of instances and their class labels against the background of the predicted probability distribution gives the user a visual understanding of the number and types of misclassifications. The graded color representing the probability distribution and the instance classifications shown against this background gives the user a visual interpretation of the information conveyed by the ROC curve. The visual display, however, also provides information not present in the ROC curve. Specifically, when the decision threshold is increased (decreased), where are false negatives (positives) introduced?

6.1 Instance Mapping

Test instances can be plotted in the display space in order to compare model predictions to actual classifications. Each test instance is displayed in the projected space as a small sphere-shaped glyph. We use glyph size to indicate the number of instances at a given point. (A linear scale is used, but is clamped at 10 instances, since otherwise large glyphs dominate the picture.) A continuous color map is used to show the proportion of class labels in the set of collocated instances. Yellow shows predominantly positive instances, red shows predominantly negative instances. Orange glyphs indicate points where there are roughly equal numbers of positive and negative instances. This representation allows a user to easily identify false positives (red glyphs inside the decision boundary) and false negatives (yellow glyphs outside the decision boundary).

Figure 8 shows the test instances on the 2D feature selection projection. Notice that because the projection groups very different locations together at each projected point, a high proportion of the instance glyphs are orange (indicating a mix of positive and negative instances at a location). Also, a relatively small percentage of the space is used for the high-class-probability part of the model, so the visual impact is that most of the space is taken up by negative instances (in fact, about 25% of the training instances are positive).

Figure 9 shows the test instances on the SOM projection. Notice that many of the misclassified instances are located in the vicinity of decision boundary, a region of the data space where their presence is expected and is not a major reason for concern. Comparing this pictures to Figure 4, one can see that the model-neutral SOM has much tighter instance clustering, because much of the display space represents regions of the data space that do not appear in the data (for example, very young and very old people). The model-neutral SOM can be useful in understanding how the data and model are biased towards particular regions of the data space; the model-dependent SOM in Figure 9 is more useful in understanding the model itself, since differences between high-probability and low-probability regions are emphasized in the SOM construction.

6.2 Display Space Queries

The visualizations can be queried interactively (by clicking at a point on the map) to generate a summary of the instances that correspond to any given region. For example, near the upper right of Figure 9, there is a region with two large orange glyphs (mixed positive and negative instances). This region corresponds to a group of males in private industry who work long hours (60 hours a week), have moderate education (typically some college), and work as professionals or managers. Querying further, we can get a summary of the positive instances (true positives) and negative instances (false positives) in the region. Upon inspection, there are few differences between these two groups along the dimensions included in this data set. A knowledge engineer could use this analysis process to identify groups of individuals who are not easily differentiated with respect to the class of interest. Such a conclusion might lead to further data gathering (to identify features that would differentiate the high- and low-income earners), or might simply indicate that the model was not reliable for that particular group.

7 Lessons Learned

Existing data mining software packages primarily use feature subset projection techniques for visualizing high-dimensional datasets and model predictions. Our investigation has shown that this approach is inadequate for complex, high-dimensional domains, because the 2D (or even 3D) feature selection display does not adequately capture the structure of the domain.

The SOM-based displays are promising in that the display space is utilized more efficiently, and the actual model behavior is reflected more completely than in feature selection. However, the correspondence of the SOM displays to the data space is also more difficult to understand. The attribute contours and querying techniques we have developed provide some initial tools for interpreting the SOM display, but further work is needed in this area.

The nature of the SOM display space depends heavily on the particular process used to build it (e.g., which instances are used to seed the SOM construction process and which similarity metric is used). Although this could be seen as a disadvantage, it provides more flexibility in constructing the SOM to meet particular projection criteria. We are currently exploring different projection metrics (such as region preservation and decision boundary smoothness) and how the SOM construction process affects these metrics.

We are exploring other dimension reduction techniques, such as principal components analysis (PCA), as alternatives to the SOM approach. However, while these techniques may provide useful for certain types of domains, the key issues of representing complexity and of data space/display space correspondence will remain the same. Similarly, we are investigating 3D visualization methods to add richness to the visualizations, but adding another display dimension does not solve the fundamental challenges of high-dimensional model visualization. We believe that the key to making these methods widely useful lies in developing additional annotation techniques to clarify the correspondence between the data space and the projected display space.

Many of the techniques that we have presented here are also completely applicable to the direct visualization of data instances without the initial creation of a model. Some of the techniques used here, in particular the display of instance glyphs using feature selection or SOMs, are already in common use for that purpose. The annotation of high-dimensional projections using attribute contours can be done with direct visualization of discrete data instances, since the projection is continuous even if the data is not, but contours have not previously been used this way, to our knowledge.

8 Conclusions

The construction of induced predictive models can aid in the process of knowledge discovery, but can be limited by the complexity of such models. We have presented visualization techniques that assist a user in understanding and analyzing a predictive model. These techniques augment standard statistical tools by allowing the user to see multiple characteristics of the model's behavior across the data space. Using a benchmark data set, we have demonstrated how visualization of model characteristics facilitates both model understanding and analysis, indicating aspects of model performance not available from standard statistical tools.

Acknowledgments

This work was partially supported by NSF CAREER Grant 9996043 (Dr. Rheingans) and DARPA's HPKB program (Dr. desJardins).

References

[Assa *et al.* 1997] J. Assa, D. Cohen-Or, and T. Milo, "Displaying data in multidimensional relevance space with 2D visualization maps," IEEE Visualization '97, IEEE Computer Society Press, 1997, pp. 127-134.

[Becker 1998] B. Becker, "Research report: Visualizing decision table classifiers," Information Visualization '98, IEEE Computer Society Press, Los Alamitos CA, 1998.

[Cleveland and McGill 1988] W. Cleveland and M. McGill, *Dynamic Graphics for Statistics*. Wadsworth and Brooks/Cole, 1988.

[Cox and Cox 1994] T. Cox and M. Cox, *Multidimensional Scaling*, Monographs on Statistics and Applied Probability, Chapman & Hall, London, 1994.

[Friedman and Goldszmidt 1996] N. Friedman and M. Goldszmidt, "Building classifiers using Bayesian networks," Proceedings of the Thirteenth National Conference on Artificial Intelligence, Portland, OR, pp. 1277-1284. AAAI Press, 1996.

[Inselberg and Dimsdale 1990] A. Inselberg and B. Dimsdale, "Parallel coordinates: A tool for visualizing multi-dimensional geometry," In *IEEE Visualization '90*, pp. 361–375. IEEE Computer Society Press.

[Kohonen 1997] T. Kohonen, *Self-Organizing Maps*, Second Edition, Springer, Berlin, 1997.

[Kohonen *et al.* 1996] T. Kohonen, J. Hynninen, J. Kangas, and J. Laaksonen. "SOM_PAK: The Self-Organizing Map program package," Technical Report A31, Helsinki University of Technology, Laboratory of Computer and Information Science, FIN-02150 Espoo, Finland, 1996.

[Mitchell 1980] T. Mitchell, "The need for biases in learning generalizations," Rutgers University Technical Report CBM-TR-117, May 1980.

[Pickett et al. 1990] R. Pickett, H. Levkowitz, and S. Seltzer, "Iconographic displays of multiparameter and multimodality images," In *Proc. of Visualization in Biomedical Computing*, pp. 58–65. IEEE Computer Society Press.

[Provost and Fawcett 1997] F. Provost and T. Fawcett, "Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions," Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, Huntington Beach, CA, 1997.

[Rushmeier *et al.* 1997] H. Rushmeier, R. Lawrence, and G. Almasi, "Case Study: Visualizing Customer Segmentations Produced by Self Organizing Maps," IEEE Visualization '97, IEEE Computer Society Press, Los Alamitos CA, 1997, pp. 463-466.

[SGI 1999] Silicon Graphics, Inc., *MineSet User's Guide*, SGI Document 007-3214-002, 1999.

[UCI 1999] University of California, Irvine, *UCI Machine Learning Repository*, http://www.ics.uci.edu/˜mlearn/MLRepository.html, 1999.

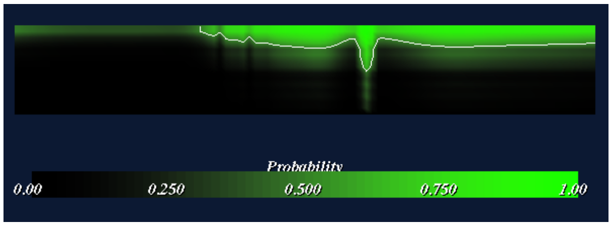

Figure 2. *Probability distribution for 2D feature selection. Vertical axis = education; horizonal = hours worked.*

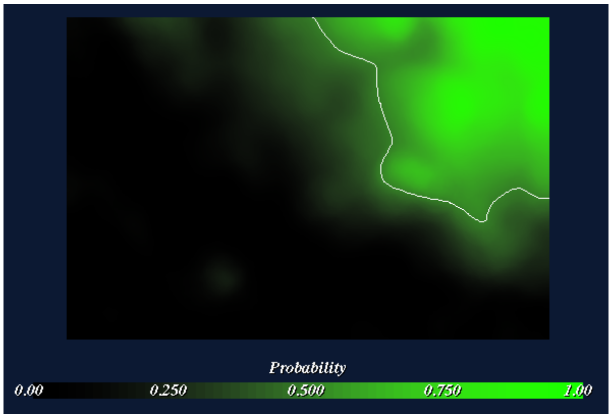

Figure 3. *SOM probability map constructed from model.*

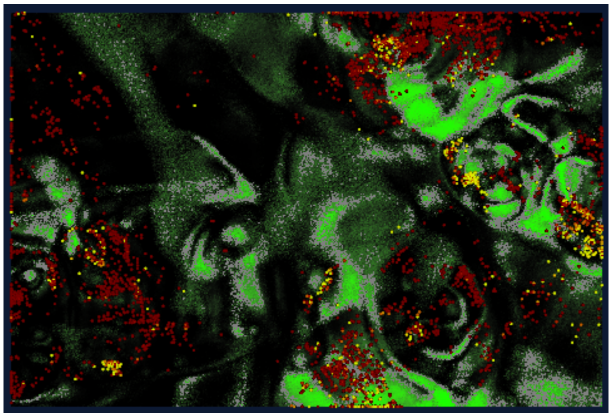

Figure 4. *SOM "model-neutral" probability map constructed from instances spanning the data space.*

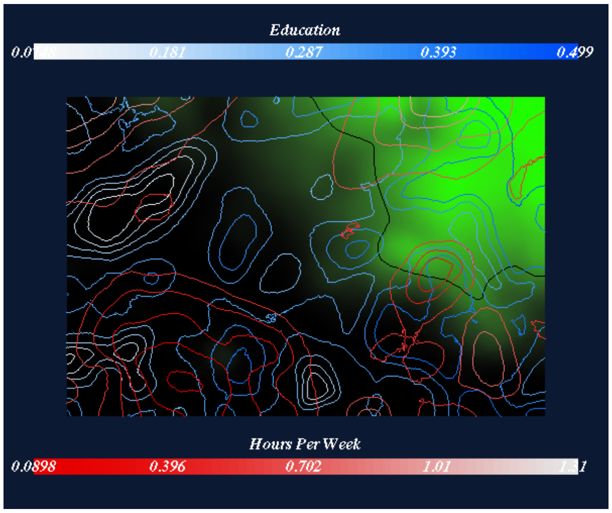

Figure 5. *SOM probability map with education and hours worked contours.*

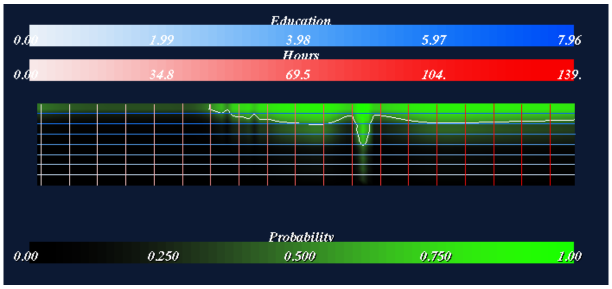

Figure 6. *Probability distribution for 2D feature selection with attribute contours.*

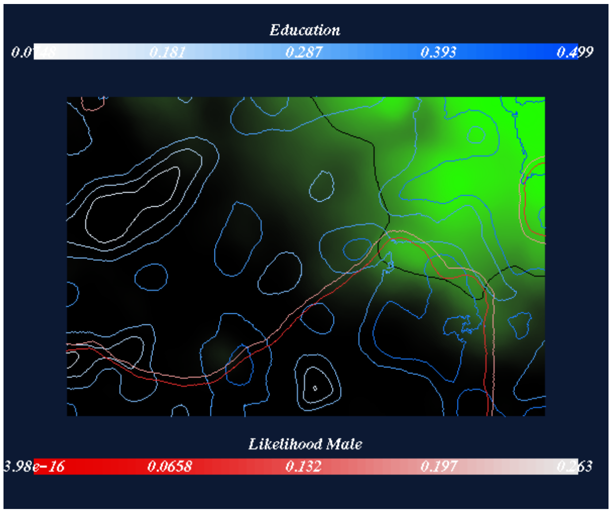

Figure 7. *SOM probability map with education and sex contours.*

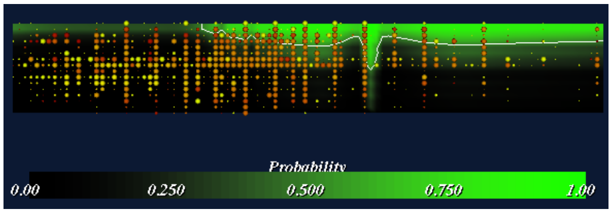

Figure 8. *2D feature selection projection with test instances.*

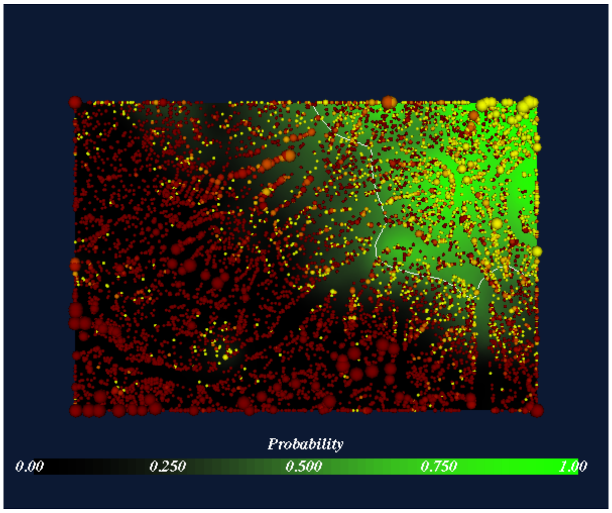

Figure 9. *SOM probability map with test instances.*