Breast Cancer Prognosis via Gaussian Mixture Regression

Tiago H. Falk
Electrical and Computer Eng.,
Queen's University
Kingston, ON, Canada
e-mail: tiago.falk@ece.queensu.ca

Hagit Shatkay
School of Computing,
Queen's University
Kingston, ON, Canada
e-mail: shatkay@cs.queensu.ca

Wai-Yip Chan
Electrical and Computer Eng.,
Queen's University
Kingston, ON, Canada
e-mail: chan@post.queensu.ca

Abstract

This paper compares the performance of classification and regression trees (CART), multivariate adaptive regression splines (MARS), and a Gaussian mixture regressor (GMR) method in predicting breast cancer recurrence time in patients that have undergone cancer excision. It is shown that the GMR-based algorithm demonstrates an improved performance compared to CART and MARS. Moreover, GMR performance is comparable to that of a baseline predictor with the advantage of performing automatic feature selection and model optimization.

Keywords *— Prognosis prediction, breast cancer, time-to-recur, automatic feature selection, Gaussian mixture regressor, CART, MARS.*

1 Introduction

Despite much scientific research and public awareness, breast cancer continues to be the most common cancer among Canadian women. According to the Canadian Cancer Society, in 2005, an estimated 21,600 women were diagnosed with breast cancer and approximately 5,300 died of it [1]. Ultimately, one in 9 women is expected to develop breast cancer during her lifetime; one in 27 will die of it. The odds of surviving breast cancer are typically improved by early detection [2]. In turn, the choice of appropriate treatment following surgery is largely influenced by prognosis – the predicted outcome of the disease. Therefore, improving prognostic prediction is an important task.

In traditional medical practice, the most reliable predictive measure is the extent to which cancer is present in the lymph nodes [2]. This measure requires the surgical removal of the nodes, leaving the patient susceptible to infection. Recent studies have shown that alternative measures, obtained directly from the tumor mass, provide useful cues for predicting breast cancer recurrence [3]. For each patient, 30 cellular features are extracted from digital images of cells taken from the tumor (e.g., area, radius, smoothness of cell nuclei); predictors are built based on these features. As a baseline for our study, we use the predictor described in reference [2]. To the best of our knowledge, this predictor exemplifies the state-of-the-art; it is based on linear programming (LP) and takes into account five manually chosen features.

In this paper we examine three alternative predictors. Two are classic algorithms, namely, classification and regression trees (CART) [4] and multivariate adaptive regression splines (MARS) [5]. The third is an extended version of an algorithm based on Gaussian mixture regressors (GMR) [6]. These algorithms are chosen as they have the major advantage of performing automatic feature selection and model optimization. We compare their performance on a publicly available set of records from the Wisconsin Prognosis Breast Cancer Database [7]. Our results demonstrate that GMR significantly reduces the error in the predicted recurrence time compared with the other algorithms. Moreover, GMR achieves performance comparable to the baseline [2], still relying only on information that is easily attainable without invasive removal of lymph nodes, but having the additional advantage of automatic feature selection.

2 Candidate Prognosis Predictors

Classification and regression trees (CART) and multivariate adaptive regression splines (MARS) are widely used in machine learning applications for bioinformatics. For the sake of completeness we briefly introduce them. More details are available in the literature ([4] and [5]). Gaussian mixture regressors, on the other hand, are less popular and are described here in more detail. The description below will focus on the algorithms' ability to perform automatic feature selection and model optimization.

Classic Algorithms

CART is based on a binary-tree structure, in which each parent node is split into two children nodes according to a simple *yes/no* question about the value of a predictor variable [4]. A notion of variable-importance is introduced in CART by means of a purity function. A split is selected such that the data in the child node is "purer" than the data in the parent node. A node is recursively split until a decrease in the impurity function reaches a certain threshold. Feature importance rankings are determined by summing the decrease in impurity produced in the remaining nodes if the split were attained at that specific feature. Scores reflect the contribution each predictor variable has on predicting recurrence times. The feature used to split the root node receives 100% importance.

On the other hand, MARS is constructed as a weighted sum of basis functions, or more specifically, truncated spline functions [5]. Variable importance scores are found by measuring the effects the variable has in fitting the data by dropping it from the model. The most important variable is the one that, when omitted, degrades the model fitness the most. In both CART and MARS, feature variables receive an importance score ranging from 0% to 100%. Features that receive a 0% importance rating are discarded.

Gaussian Mixture Regressors

Gaussian mixture regressors (GMR), as proposed by Ghahramani and Jordan [8], do not perform automatic feature selection. Earlier attempts of using GMR made use of CART or MARS as feature selection tools [9]. Here, the GMR-based algorithm [6] is extended to perform sequential feature selection. In the sequel, we define Gaussian mixture models (GMM) and GMRs, and then provide details of our feature selection extension.

Let $\mathbf{u}$ be a K-dimensional vector. A Gaussian mixture density is a weighted sum of M component densities, $p(\mathbf{u}|\boldsymbol{\lambda}) = \sum_{i=1}^{M} \alpha_i b_i(\mathbf{u})$, where α_i are the mixture weights ($\alpha_i \geq 0,\ \ i = 1, \ldots, M,$ and $\sum_{i=1}^{M} \alpha_i = 1$), and $b_i(\mathbf{u})$ are K-variate Gaussian densities with mean vector $\boldsymbol{\mu}_i$ and covariance matrix $\mathbf{\Sigma}_i$. The parameter list, $\boldsymbol{\lambda}$=$\{\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_M\}$, defines a particular Gaussian mixture density, where $\boldsymbol{\lambda}_i = \{\boldsymbol{\mu}_i, \mathbf{\Sigma}_i, \alpha_i\}$. GMM parameters are commonly estimated via the EM algorithm [10]. We use the *k-means* algorithm to initialize the model parameters.

GMM-based regressors (GMR) rely on modelling the joint density of the predictor variable vector ($\mathbf{x}$) with the target variable (y), i.e., $\mathbf{u} = [y, \mathbf{x}]$. The mean vector and the covariance matrix of the i^{th} GMM component become:

$$\boldsymbol{\mu}_i = (\mu_i^y,\ \boldsymbol{\mu}_i^x) \text{ and } \mathbf{\Sigma}_i = \begin{pmatrix} \Sigma_i^{yy} & \mathbf{\Sigma}_i^{yx} \\ \mathbf{\Sigma}_i^{xy} & \mathbf{\Sigma}_i^{xx} \end{pmatrix}, \text{ respectively.}$$

Given the GMM parameters, the minimum mean-square-error regression function $\hat{f}$ is the conditional expectation of the target variable, given the predictor variables [8]:

$$\hat{f}(\mathbf{x}) = E[y|\mathbf{x}] = \sum_{i=1}^{M} h_i(\mathbf{x})[\mu_i^y + \mathbf{\Sigma}_i^{yx}(\mathbf{\Sigma}_i^{xx})^{-1}(\mathbf{x} - \boldsymbol{\mu}_i^x)]. \quad (1)$$

The function $\hat{f}$ above is a weighted sum of linear models, with weights $h_i(\mathbf{x})$ representing the probability that the i^{th} GMM component generated the vector $\mathbf{x}$ and given by

$$h_i(\mathbf{x}) = \frac{\dfrac{\alpha_i}{|\mathbf{\Sigma}_i^{xx}|^{1/2}}\ e^{\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i^x)^T(\mathbf{\Sigma}_i^{xx})^{-1}(\mathbf{x}-\boldsymbol{\mu}_i^x)\right)}}{\displaystyle\sum_{k=1}^{M} \dfrac{\alpha_k}{|\mathbf{\Sigma}_k^{xx}|^{1/2}}\ e^{\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k^x)^T(\mathbf{\Sigma}_k^{xx})^{-1}(\mathbf{x}-\boldsymbol{\mu}_k^x)\right)}}. \quad (2)$$

If GMM covariance matrices are restricted to be diagonal, Eq. (1) simplifies to

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{M} h_i(\mathbf{x})\mu_i^y. \quad (3)$$

The primary objective in feature selection and model optimization is to find, among n candidate feature variables, a subset of variables $\mathbf{x} = \{x_1, \ldots, x_m\}$, $m < n$, and a mapping $\hat{f}(\mathbf{x})$, such that $\hat{f}$ approximates the target variable y. The GMR-based algorithm performs feature selection while progressively constructing $\hat{f}$ using formulas (1) or (3).

The algorithm starts with an empty feature set, S. Features from a candidate feature set are added to S progressively. To determine which candidate feature to add, the algorithm tentatively adds to the current feature set one feature that is not already in it, forming an augmented feature set. The joint density of the target variable and the augmented feature set is modelled with a GMM, with model parameters $\lambda = (\alpha,\ \mu,\ \Sigma)$ estimated using the EM algorithm. The accuracy of the GMR using λ is then calculated. The above steps are repeated for every candidate feature and corresponding GMM. The candidate feature that produces the least regression error is admitted and added into the feature set S. The algorithm stops selecting features either when it reaches a pre-determined threshold for the training ratio (ratio between the number of parameters that have to be estimated during the training phase and the total number of records in the training set) or when the desired (pre-specified) number of features has been selected.

Using the notation "EM" to stand here for the GMM parameter estimation via the EM algorithm, $\hat{f}_k$ for the mapping function with k variables, I for the feature indices, and assuming D is the desired number of features, the algorithm is summarized as follows:

```
Initialization: Let $I = \{1, \ldots, n\},\ S = \emptyset,\ k = 1$;
Step 1: $\lambda_i \leftarrow \mathrm{EM}(y,\ S \cup \{x_i\}),\ \forall i \in I$;
Step 2: $i_k \leftarrow \arg\min_{i \in I} \sum_j (y_j - \hat{f}_k(S \cup \{x_i\}|\lambda_i))^2$;
Step 3: $I \leftarrow I - \{i_k\},\ S \leftarrow S \cup \{x_{i_k}\},\ k \leftarrow k + 1$.
Go to step 1 if $k < D$, else stop.
```

This algorithm description has focused on hard-decision search, where only a single feature variable, which minimized the estimation error, was kept. With a linear increase in computational complexity, the algorithm is extended to perform *N-candidate soft-decision* sequential search, where at each iteration N features are kept as candidates. In this extended version, at each iteration, the N features that assume the top-N ranks in minimizing the estimation error are kept as candidates. A tradeoff between complexity and performance can be adjusted by tuning the parameter N. If the ultimate goal is to find D features out of n candidate features, then N candidates are kept in iterations $i = 1, 2, \ldots, D-1$. At iteration $i = 1$, the algorithm selects the N best features out of the n available candidates. At iterations $1 < i < D$ the N best ranked feature combinations, out of the $N(n-i+1)$ possible feature combinations, are kept. Finally, at iteration $i = D$, the single best feature is kept. This last best feature and its ancestor features constitute the set of features selected by the search process.

3 Experiments and Results

The experiments described here use the Wisconsin Prognosis Breast Cancer (WPBC) Database [7]. The database is comprised of 253 records from malignant patients for which follow-up data was available. Features are extracted by a program called Xcyt [3] and depend on the analysis of cellular images. Ten features are computed for each cell nucleus: *area*, *radius*, *perimeter*, *symmetry*, *number* and *size* of concavities, *fractal dimension* of the boundary, *compactness*, *smoothness*, and *texture*. A complete description of the features can be found in reference [3]. The mean value, extreme value (e.g., biggest size, most irregular shape), and standard deviation of each of these cellular features are computed for each image, resulting in a total of 30 real-valued features. Moreover, two traditional features, tumor size and number of involved lymph nodes are added to the feature set, resulting in a total of 32 features per record.

For a subset of the patients, cancer recurs and the time-to-recur (TTR) is recorded. For some patients, only the time of their last checkup, or disease free survival time (DFS) is available. Since DFS is assumed to be a lower bound of TTR, all available cases are used [2]. TTR prediction is regarded as a function estimation problem, where a mapping between cellular features and expected recurrence time is found. It is now clear why feature selection is such an important step in TTR prediction. The number of features is relatively large (32) compared to the available number of training records (253). In this situation, overfitting is most likely, and using a subset of the features is expected to lead to better generalization capability.

The approach taken by Mangasarian *et al.* [2], which we use as a baseline, was to remove features, one-by-one, retrain the model and test on a tuning set. The five features which showed the best performance on the tuning set were used to train the final model. In the sequel, CART, MARS and GMR are tested as possible predictors of TTR. Such algorithms may be preferred over the baseline as feature selection and model optimization is performed iteratively and automatically. This is invaluable when additional features need to be tested and user supervision is limited.

Experimental Setting

We apply CART, MARS and GMR to the task of predicting breast cancer recurrence time. Performance is assessed by the *average absolute error*, measured in months, between predicted recurrence times and actual TTR. Following the baseline, we set the number of features to five. To maintain an adequate training ratio we restrict the number of Gaussian components to $M \leq 4$ and use the diagonal covariance GMR. Ten-fold cross validation is used in the performance evaluation. The WPBC database is randomly divided into 10 data sets of almost equal size. Training and testing is performed in 10 trials, where, in each trial, one of the data sets serves as a test set and the remaining 9 are combined to serve as a training set. Each data set serves as a test set only once. The ten resulting errors are averaged to obtain the overall expected error. Note that Mangasarian *et al* [2] did not use this procedure, but rather used a leave-one-out cross validation, which is less robust [11].

Results

Table I shows the absolute error, given in months, produced by CART and MARS, and the *reduction* in error attained with three versions of GMR: GMR-1 uses hard-decision search, GMR-2 uses 2-candidate soft-decision search, and GMR-2P is similar to GMR-2, but built on preprocessed data, as described in the sequel. The reduction in prediction error, also given in months, is reported relative to CART and MARS in columns “Reduction GMR-*i*” (for GMR version “*i*”). As can be seen, CART and MARS have similar performance figures for most cross validation trials, with MARS achieving somewhat better accuracy. Moreover, all three GMR schemes outperform both CART and MARS. An average reduction in prediction error of approximately 7 months and 5.5 months (15-20% reduction) can be attained relative to CART and MARS, respectively. The differences shown in Table I are highly statistically significant ($p < 0.05$) according to the 2-sample t-test.

Note that an average error reduction of 5.8% is achieved by GMR-2 relative to GMR-1. This exemplifies the tradeoff between complexity and accuracy inherent in the N-candidate soft-decision search capability of the proposed algorithm. The data pre-processing of GMR-2P is similar to that of the baseline – data is scaled to have mean 0 and a variance of 1, and principal components analysis is performed to decorrelate the features. Since training data is limited, GMR is restricted to a small number of diagonal covariance Gaussian components; this restriction can degrade performance if selected features are highly correlated.

The top five features selected by the algorithms, including the baseline, can be seen in Table II. As mentioned previously, the most predictive feature in traditional medical practice is the extent to which cancer is present in the lymph nodes. This requires a microscopic examination of the lymph nodes, that in turn must be surgically removed from the patient. This invasive procedure leaves the patient more susceptible to infection and lymphedema, a severe swelling of the arm [2]. It is, therefore, desirable that the algorithms depend only on cellular features and not on the number of lymph nodes. To this end, GMR-1 and GMR-2P are preferred over GMR-2.

Comparing GMR-2 with GMR-2P, it can be seen from Tables I and II that the advantage of data pre-processing is not in the performance gain, but in the fact that the updated model depends only on easily attainable cellular features, an extremely desirable outcome. We note that most of the features selected by GMR-2P are gleaned from those selected by MARS, GMR-1 and GMR-2. Finally, to compare against the baseline, a leave-one-out test is performed and an average error of approximately 26 months is attained. This performance is comparable to that achieved by the baseline (24 months) [2] with the proposed algorithm having the advantage of performing automatic feature selection and model optimization.

4 Conclusions

We have investigated the use of three data mining algorithms to predict breast cancer recurrence times. It is noted that the GMR-based algorithm, extended to perform N-candidate soft-decision feature selection, outperforms classic algorithms such as CART and MARS. Diagonal covariance matrix GMRs, built on five cellular features, achieve an average 10-fold cross validation error of 27.91 months

TABLE I

Absolute prediction errors, given in months, for the candidate algorithms. Column "Reduction GMR-i" shows the reduction in prediction error, also given in months, by using GMR version "i," relative to CART and MARS.

Cross Validation	CART	MARS	Reduction GMR-1		Reduction GMR-2		Reduction GMR-2P	
			CART	MARS	CART	MARS	CART	MARS
Trial 1	33.34	33.02	3.27	2.95	5.2	4.88	7.85	7.53
Trial 2	33.83	32.88	2.16	1.21	4.9	3.95	4.73	3.78
Trial 3	35.96	35.65	7.03	6.72	7.72	7.41	6.68	6.37
Trial 4	37.87	36.81	4.61	3.55	7.45	6.39	3.91	2.85
Trial 5	39.34	37.94	3.17	1.77	4.38	2.98	7.98	6.58
Trial 6	37.37	38.25	5.47	6.35	5.47	6.35	4.81	5.69
Trial 7	34.48	32.49	6.53	4.54	6.95	4.96	9.42	7.43
Trial 8	33.96	34.81	6.65	7.50	8.87	9.72	8.10	8.95
Trial 9	30.70	27.29	5.01	1.60	5.64	2.23	6.74	3.33
Trial 10	31.08	24.35	5.98	-0.75	10.39	3.66	8.62	1.89
Average	**34.79**	**33.35**	**4.99**	**3.54**	**6.70**	**5.25**	**6.88**	**5.44**

TABLE II

Top five features selected by each prognosis prediction algorithm.

Top Features	CART	MARS	GMR-1	GMR-2	GMR-2P	Baseline
Average Compactness	✓					
Average Fractal Dimension	✓	✓			✓	✓
Average Perimeter		✓				✓
Average Radius		✓	✓		✓	✓
Average Texture			✓		✓	
Average Area				✓	✓	
Average Concavity				✓		
Extreme Smoothness	✓					
Extreme Compactness	✓					
Extreme Fractal Dimension	✓	✓				
Extreme Symmetry		✓	✓	✓		
Extreme Texture			✓	✓		
Extreme Perimeter						✓
Extreme Area						✓
Standard Deviation of Perimeter			✓			
Standard Deviation of Symmetry					✓	
Number of Lymph Nodes				✓		

and an average leave-one-out error of approximately 26 months. It is thus shown that GMR prediction capability is comparable to that of the baseline, with the advantage of performing automatic feature selection and model optimization. We emphasize that, similar to the baseline, our algorithm depends only on cellular features. Thus, the routine and potentially hazardous removal of lymph nodes can be avoided.

References

[1] http://www.cancer.ca.

[2] O. L. Mangasarian, W. N. Street, and W. H. Wolberg, "Breast cancer diagnosis and prognosis via linear programming," *Operations Research*, vol. 43, no. 4, pp. 570–577, July 1995.

[3] W. N. Street, "Cancer diagnosis and prognosis via linear-programming based machine learning," Ph.D. dissertation, University of Winsconsin, 1994.

[4] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees.* Wadsworth & Brooks, 1984.

[5] J. H. Friedman, "Multivariate adaptive regression splines," *The Annals of Statistics*, vol. 19, no. 1, pp. 1–141, March 1991.

[6] T. H. Falk and W.-Y. Chan, "A sequential feature selection algorithm for GMM-based speech quality estimation," in *Proc. of the 13th European Signal Processing Conference*, Sept. 2005.

[7] ftp://ftp.cs.wisc.edu/math-prog/cpo-dataset/machine-learn/WPBC/.

[8] Z. Ghahramani and M. I. Jordan, "Supervised learning from incomplete data via an EM approach," in *Advances in Neural Information Processing Systems*, vol. 6, 1994, pp. 120–127.

[9] T. H. Falk and W.-Y. Chan, "Feature mining for GMM-based speech quality measurement," in *Proc. of the Asilomar Conf. on Signals, Systems and Computers*, Nov. 2004, pp. 2290–2294.

[10] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Statistical Society*, vol. 39, pp. 1–38, 1977.

[11] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning.* Springer, 2001.