Study of Pattern Recognition Techniques in Classifying Gene Expression Profiles

Parneet Kaur
Rutgers University

parneet@eden.rutgers.edu

Abstract

Gene expression profiles are widely studied to classify patients into tumor/non-tumor and/or to the correct tumor category. Since the number of tissue samples examined is usually much smaller than the number of genes examined, efficient data reduction techniques are important. Another interesting problem is to find the most relevant/informative genes in the genome, which can help in successful classification. The aim of this project is to study the effectiveness of linear and non-linear dimensionality reduction methods on performance of classification algorithms by measuring their success rate. In addition, most relevant/discriminative features are found and their success in classification techniques is studied.

1. Introduction

It is known that the performance of a given classifier decreases as the dimension of the features (genes) increases [6]. The classifier performance can be improved by using dimensionality reduction techniques which map the high dimensional feature vector to a low dimensional subspace. Thus, the problem of dimensionality reduction is: *Given n samples (observations) and d genes/peptides (variables), transform the data matrix* (d x n) *from high-dimension to a low dimension subspace by finding m new variables, where m is less than d.* The new m variables can be linear or non-linear combinations of the original d variables. Some classical linear dimensionality reduction techniques are principle component analysis (PCA), linear discriminant analysis (LDA) and classical multidimensionality scaling(CMDS) [13]. Some nonlinear dimensionality reduction techniques are isometric and locally linear embedding which have been shown to perform better than linear methods [3],[11]. Many studies have been done to compare the different dimensionality reduction and pattern classification techniques for high dimensionality datasets[3],[11],[10]. In this project, State Vector Machine [2],C4.5[9] and k-Nearest Neighbor are used for two class classification. For feature selection Genetic Algorithm (GA) [7], [8] is compared with t-statistics.

2. Methods Used

2.1. Dimensionality Reduction

2.1.1 Principal Component Analysis (PCA)

PCA aims to find a reduced dimension,linear subspace specified by orthogonal vectors (also called principle components) which attempt to maintain most of the variability of the data. For given data matrix with n samples and

d variables, eigenvalues e_i and eigenvectors λ_i are computed for the covariance matrix given by:

$$C = AA^T \tag{1}$$

where, $A = [\Gamma_1 \Gamma_2 \cdots\cdots \Gamma_N]$ and Γ_i is a Dx1 vector.
Size of C is (dxd) which is enormous and finding its eigenvalues and eigenvectors is computationally heavy. We can use a theorem in Linear Algebra, which states that eigenvalues e_i and eigenvectors λ_i can be obtained by solving for eigenvalues d_i and eigenvectors μ_i of nxn matrix A^TA as:

$$e_i = Ad_i, \lambda_i = \mu_i \tag{2}$$

These eigenvectors correspond to top n (out of d) eigenvectors of AA^T. The vectors e_i are also called principle components. The higher eigenvalues correspond to eigenvectors that describe more characteristic features and also the highest variance in the data. They produce an orthonormal basis for the subspace within which data can be represented. By selecting the eigenvectors for highest m eigenvalues, each sample Γ_i can be projected onto m-dimensional space ($m \leq n < d$).

2.1.2 Classical Multidimensional Scaling (cMDS)

Multidimensional Scaling is a linear approach that maps the original high dimensional space to a lower dimensional space, with a constraint to preserve euclidean pairwise distances. It finds an embedding with optimal positions for the input points in m-dimensional space through minimization of the least squares error in the input pairwise euclidean distances. The input to this algorithm is a dxd symmetrical matrix E_d. This method is based on the theorem that a Euclidean distance matrix E_d can be approximated into an appropriate Gram matrix B, which is decomposed into $U\Lambda U$. Then, by removing the $d - m$ trailing columns of $X = U\Lambda^{1/2}$, we reduce it to m-dimensional space. This is equivalent of projecting data matrix onto its top m principal component.

2.1.3 Isometric Mapping (Isomap)

Isomap is a non-linear modification of cMDS. This algorithm uses the geodesic distances (distance along the manifold) instead of euclidean distances. To compute the geodesic distances, k-nearest neighbors for each input point are identified using euclidean distances. The geodesic distances for the nearest points can be approximated by the euclidean distances. For the remaining points, geodesic distances are computed using the cost function of shortest path algorithm like Dijkstras or Floyds algorithm. The pairwise geodesic distance matrix G_d is the input to the cMDS algorithm which returns m-dimensional embedding.

2.2. Feature Selection

2.2.1 t-statistics

T-statistics is often used for feature selection. For each gene i, mean μ_1, μ_2 and variance σ_1^2, σ_2^2 are computed for class 1 and class 2. The information content is calculated using the formula:

$$T(i) = \frac{\mu_1 - \mu_2}{\sqrt{\frac{\sigma_1^2}{N_1} + \frac{\sigma_2^2}{N_2}}} \tag{3}$$

The most discriminative features are the ones with highest score. This method uses information from a single feature (gene).

2.2.2 Genetic Algorithm (GA)

Genetic Algorithm (GA) along with k-Nearest Neighbors(kNN) was proposed as a feature selection method by Li et al.[8] Briefly, this method consists of the following steps:

1. Population formation: d' distinct random genes selected from total d genes to form a *chromosome*. Set of chromosomes(100) is constructed to form a *population*. 10 such populations are formed.

2. Fitness: kNN is used to classify the training samples. If a sample and its k-nearest samples have same class, the sample is assigned a score of 1. Mean score of a chromosome is computed as R^2.

3. Selection: Best chromosome with highest R^2 score from each population is used to form next generation, and remaining generations are formed based on the relative fitness.

4. Mutation: First 5 genes of chromosomes forming the new generation are selected for mutation with high probability if d' is greater than 10, else only one gene is selected. Genes selected for mutation are sampled with replacement from all the genes to form new generation.

5. Termination: Generations are allowed to evolve till at least one of the chromosomes reach a threshold fitness. When 10^4 high R^2 chromosomes are selected, the most frequently occurring genes are selected as most relevant features.

This method uses mutual information between features (genes) and hence is expected to perform better than t-statistics which uses information from a single feature (gene). However, it is computationally more expensive.

2.3. Classification

2.3.1 State Vector Machine(SVM)

SVM is a supervised, non-parametric classification method. The inputs to this algorithm are nxm data matrix with m-dimensional feature vector for each of n samples, and nx1 class labels for each of n samples. It separates labeled data with a hyper-plane obtained by maximum margin algorithm. Support vectors are the vectors that lie closest to the separating hyper-plane. Since such high-dimensional data is not linearly separable, a kernel can be used to map the given data to a high-dimensional space where it is linearly separable. The linear hyper-plane in high dimensional space corresponds to a non-linear hyper-plane in the actual m-dimensional space.

2.3.2 C4.5

C4.5 is a decision tree based classifier developed by Ross Quinlan[9]. Using the training data, at each node information gain is computed for selecting each attribute for splitting data. The attribute which gives the highest information gain is selected for data-splitting at the given node.

2.3.3 k-Nearest Neighbors(kNN)

kNN is a supervised, non-parametric classification method. There is no training phase and the input to this algorithm is same as that of SVM. In addition, the user inputs k which is the number of nearest neighbors to be considered for classification. In the classification phase, k is a user-defined constant, test sample is classified by assigning the label which is most frequent among the k training samples nearest to that test sample.

Table 1. Datasets

Dataset	Samples	Dimensionality (Genes)
Colon Cancer [1]	40 (Tumor) 22 (Normal)	2000
DLBCL-Harvard [12]	58 (DLBCL) 19 (FL)	7129
Leukemia [4]	20 (ALL) 14 (AML)	7129

2.4. Performance Evaluation

Performance of the dimensionality reduction methods and feature selection methods is evaluated based on the success rate of classifiers to correctly classify the test samples in reduced dimensional space. Computational time is also computed as a performance measure of the classifiers. Even though computational performance is not a major criteria for this project, it is interesting to study because the techniques discussed in this project can also be applied to real time applications where computational performance is very importance.

3. Experimental Setup

3.1. Datasets

The datasets used for this project are publicly available and shown in Table 1.

3.2. Experiment Design

The training and test sets are formed as described below:

1. Leave One Out Cross Validation (LOOCV): For n samples in a database, training is done on $n-1$ samples and the testing is done on the remaining sample. This process is repeated n times making sure that each sample is tested only once.

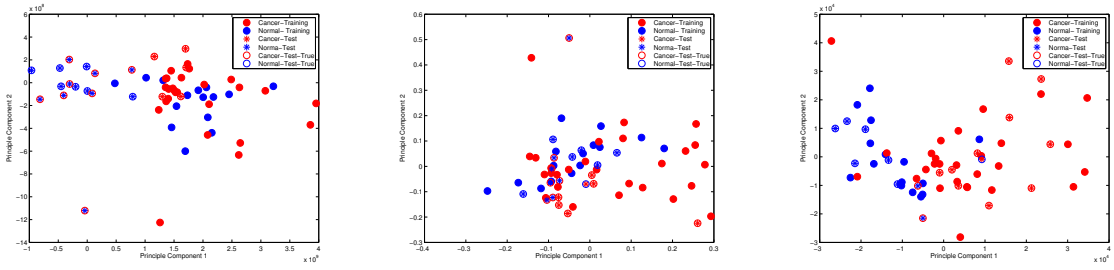
2. Training is done on one-third of the randomly selected samples from total n samples in dataset and testing is done on the remaining two-third samples. This process is repeated 100 times to compute the average classifier performance.

For the given datasets, we have a binary class problem, with two classes - normal/tumor or two types of tumor. For each of the datasets, the dimensionality of the dataset is reduced using the methods in section 2.1 and then performance of classification methods in section 2.3 is evaluated on the test set using methods in section 2.4. Similarly, feature selection methods in section 2.2 are evaluated based on classifier performance..

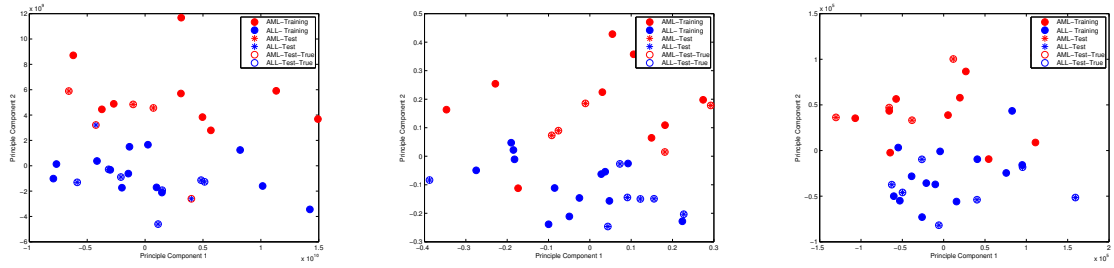
4. Results and Analysis

Initially, I used LOOCV explained in section 3.2 for computing test set error. Later, I used the second method because using this method we can measure the mean and variance over iterations of experiments which can be analyzed to observe the mean and variance of error or success rate of classifiers .

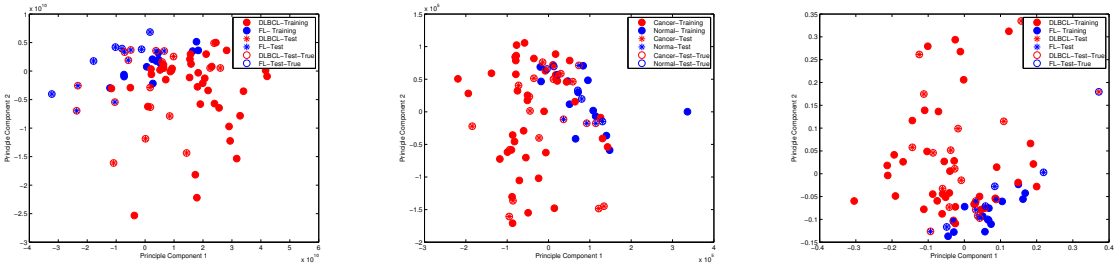
Using the dimensionality reduction techniques we can reduce the dimensionality to two, and then represent the samples in a 2-dimensional space. Figure 1 shows the 2-dimensional representation of the given datasets



(a) DR using PCA in Colon Cancer Dataset (b) DR using cMDS in Colon Cancer Dataset (c) DR using ISOMAP in Colon Cancer Dataset



(d) DR using PCA in Leukemia-AMLALL Dataset (e) DR using cMDS in Leukemia-AMLALL Dataset (f) DR using ISOMAP in Leukemia-AMLALL Dataset

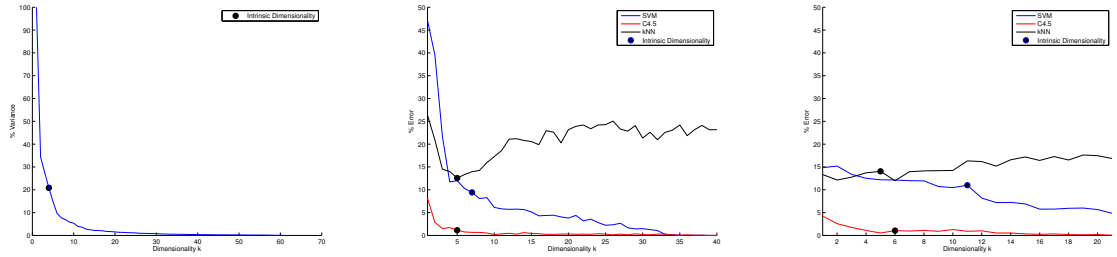


(g) DR using PCA in DLBCL-Harvard Dataset (h) DR using cMDS in DLBCL-Harvard Dataset (i) DR using ISOMAP in DLBCL-Harvard Dataset

Figure 1. Data representation in 2 dimensional space obtained by different dimensionality reduction (DR) techniques

using all the three DR methods. Training samples, true class of test samples and obtained test-sample class from SVM classification are shown. The overlap between the two classes reduces when Isomap is used. PCA and cMDS have more overlap between two classes when compared to Isomap. Two dimensional representation is only for visualization and in practice, a relatively higher dimension is considered to avoid over fitting when using a classifier.

Intrinsic dimensionality is defined as the dimensionality to which a dataset should be reduced based on the training set. Figure 2 illustrates how intrinsic dimensionality is selected. For PCA, as shown in figure 2(a), dimensionality is chosen such that it corresponds up to 98% of total variance. For cMDS and Isomap, training set error is plotted and the dimension for which the classifier error-rate is minimum is selected for dimensionality reduction, hoping that it will also give the least test set error. However, reduction in training-set error necessarily doesn't imply reduction in test set error. As shown in Figure 3, using SVM on colon cancer dataset, as the



(a) PCA: Intrinsic dimensionality from 98% variance criteria.

(b) MDS: Intrinsic dimensionality from training data classification error rate.

(c) ISOMAP: Intrinsic dimensionality from training data classification error rate.

Figure 2. Intrinsic dimensionality selection for colon cancer dataset

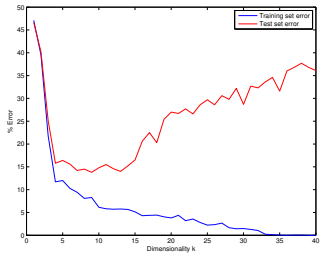


Figure 3. Colon dataset training and test set error using cMDS. Note that training set error reduces with increasing dimensionality whereas test set error increases with increasing dimensionality.

Table 2. Success Rate of classifiers for different dimensionality reduction techniques

Classification Method	Dataset	% Success Rate			
		No DR	PCA	MDS	ISOMAP
SVM	Colon Cancer	83.3	65.3	85.8	85
	Leukemia-ALLAML	86	71.3	91.6	92.5
	DLBCL-Harvard	96.4	86.1	94	96.2
C4.5	Colon Cancer	72.7	63.8	71	79.2
	Leukemia-ALLAML	81.8	70.2	86.7	83.33
	DLBCL-Harvard	80.5	72.3	82.8	93.6
kNN	Colon Cancer	80.8	57.5	81.5	86.2
	Leukemia-ALLAML	75.5	65.2	86	88.33
	DLBCL-Harvard	86.8	73.5	89.2	93.9

dimensionality is increased, training set error reduces but the test set error increases. So, when choosing the dimensionality based on training set error, it is taken care that the dimensionality for which the error is close to zero is not selected. On all the three datasets, dimensionality around which the training set error stabilizes and is less than 10% is selected. Figure 2(b) and (c) show intrinsic dimensionality selection for cMDS and Isomap, respectively.

Table 2 shows the success rate with and without dimensionality reduction techniques. It is observed that for

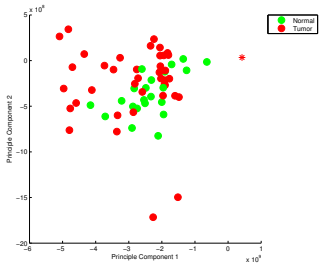
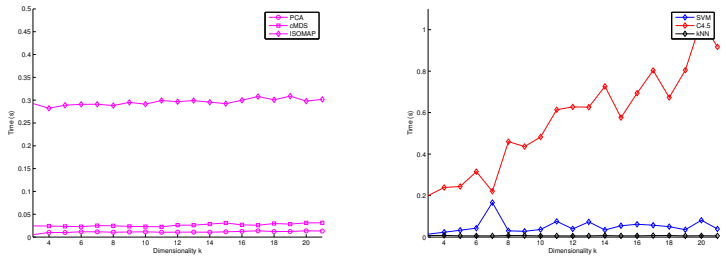


Figure 4. Analyzing kNN performance in 2-dimensional subspace obtained by PCA. $*$ represents test data and its color indicates its class (tumor). All other samples are training samples.



(a) Dimensionality reduction techniques (b) Classification Techniques

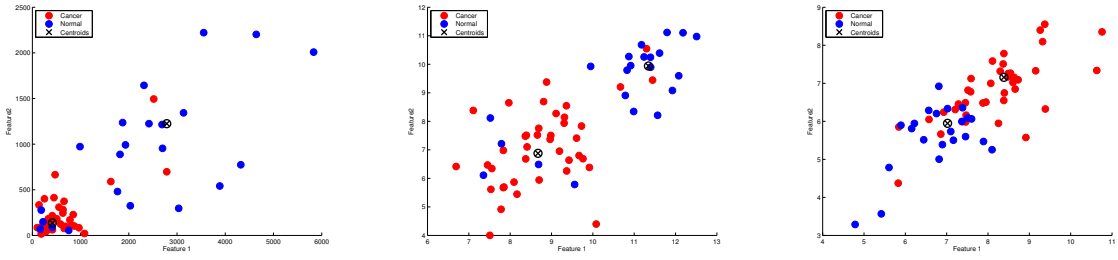
Figure 5. Computational time the with varying dimensionality

all the given datasets, on an average, using Isomap improves the performance of all the three classifiers. On the other hand, using PCA results in reduced success rate. Isomap is a non-linear DR technique and is expected to perform better than linear DR methods because of the underlying non-linear nature of gene expression data. An example of PCA deteriorating the kNN classifier performance is as shown in figure 4. The test sample (true class: tumor) has at least three of its nearest neighbors belonging to wrong class(normal). Clearly, for at least $K = 5$, kNN will have wrong result. In d dimensional space, the class was correctly identified. In general, among the three classifiers, SVM performs the best and C4.5 performs the worst. It is worth mentioning here that C4.5 is an unstable classifier.

For Isomap, I used Dijkstras algorithm implementation in MATLAB. The performance was very poor and it would not give results for a single run even after 6 hours. Later I used an open source fast implementation of Dijkstra's algorithm [5], which is faster than the MATLAB implementation (around an hour for each run). The best approach for this method is to use C++ implementation and I finally used the C++ code by Tenenbaum et al which is significantly fast. Figure 5 shows the computational time for DR and classification methods. Among DR method, Isomap is computationally expensive but the time doesn't vary much with increasing dimensionality. I use, number of nearest neighbors, $\kappa = 7$ for all the datasets. This value was selected based on observation on training set error for colon cancer dataset.

Among the classifiers, C4.5 is more computationally intensive than SVM and kNN and moreover, as the dimensionality increases, time requirement significantly increases. Note that even though computational time is not an issue for this project, it is interesting to study it for real-time classifiers.

For feature selection, t-statistics and GA/kNN are used for Colon Cancer dataset. Figure 6 shows the data



(a) GA/kNN (b) log_2+ GA/kNN (c) log_2+ t-statistics

Figure 6. Feature Selection: 2-D representation using top two features

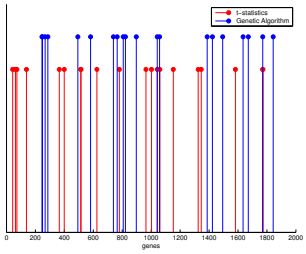


Figure 7. Comparing t-statistics and GA top 20 features

Table 3. Success Rate of classifiers for feature selection techniques

Feature Selection Method	Test Set Success Rate		
	SVM	C4.5	kNN
All Features	82.25	70.96	83.87
t-test	77.41	72.58	72.58
GA	85.16	74.51	81.29
GA + Bagging	88.17	84.30	84.51

representation in 2-dimensional using the top two features. Data is transformed by applying log_2 only for better representation. Note that GA has less overlap for two classes as compared to t-statistics.

GA/kNN is sensitive to the choice of populations formed. So, the method is repeated five times and it was found that among the top 100 genes, 65 genes are always among the top 100. These 65% genes are used to classify the samples. Number of nearest neighbors used is 5. Figure 7 shows the top 20 features from both the methods. Note that some features overlap, however it can be concluded that the most discriminative features found by t-statistics need not be most important for feature classification. Table 3 shows the success rate of classifiers for both the methods. t-statistics reduced the classification success rate for both SVM and kNN. However, GA improves the success rate.

Another experiment conducted on features selected using GA is by bagging using feature selection. 5 out of top 65 features are sampled randomly and the new feature space is used for training the classifier and testing. This is repeated 20 times. The results are shown in figure 8. It is observed that mean error and variance for all the classifiers is reduced using this method.

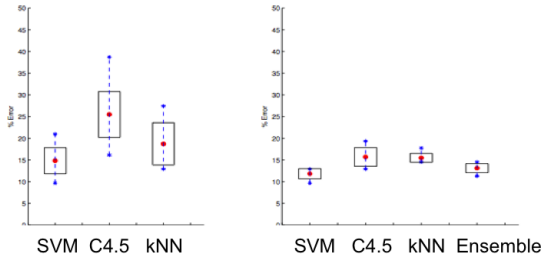


Figure 8. Error rate of classifiers with feature selection. (a) Error rate with feature selection using GA. (b) Error rate with feature selection using GA and Bagging using feature sampling.

An ensemble learner is then formed using the output of SVM,kNN and C4.5 classifiers (using GA/kNN and bagging with feature sampling) where the class is assigned to test samples based on majority voting. The ensemble classifier is found to perform better than C4.5 and kNN classifiers.

In this study, I do not compare feature selection and dimensionality reduction techniques because I implemented feature selection for only one dataset and moreover, I believe that feature selection by GA/kNN can be further improved by tuning the parameters of the algorithm.

5. Conclusion and Future Work

My conclusions from this project are as follows:

1. Isomap(non-linear method) performed better than PCA and cMDS (linear methods) for the given datasets.
2. SVM performed the best among the three classifiers.
3. C4.5 is computationally intensive and computational time increases with increase in dimensionality.
4. GA/kNN is found to be a better feature selection method than t-statistics for colon cancer dataset.
5. For feature selection using GA/kNN, when random feature selection is used, mean error rate and variance of classification methods on test data set reduced.

To study the methods in more detail, some more experiments can be conducted. For Isomap, the effect of varying the number of neighbors ca be studied. It is expected that as the number of neighbors is increased, non-linearity condition is no longer true and the result approaches that of cMDS. Another interesting experiment is to study result of varying the chromosome size for GA/kNN method.

References

[1] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences*, 96(12):6745–6750, 1999. 4

[2] T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914, 2000. 1

[3] L. G., R. C, and M. A. Investigating the efficacy of nonlinear dimensionality reduction schemes in classifying gene- and protein-expression studies. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 5:368–84, 2008 Jul-Sep 2008. 1

[4] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537, 1999. 4

[5] J. Kirk. Dijkstra's shortest path algorithm, Mar. 2012. 7

[6] J. L. McLachlan. Mclachlan, geoffrey j.: Discriminant analysis and statistical pattern recognition. john wiley and sons, inc., new york, chichester, brisbane, toronto, singapore 1992, 526 pp. 92.00, isbn 0-471-61531-5. *Biometrical Journal*, 35(7):784–784, 1993. 1

[7] L. Li, T. Darden, C. Weinberg, and L. Pedersen. Gene assessment and sample classification for gene expression data using a genetic algorithm/k-nearest neighbor method. *Comb. Chem. High Throughput Screen*, 4:727–739, 2001. 1

[8] L. Li, C. R. Weinberg, T. A. Darden, and L. G. Pedersen. Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the ga/knn method. *Bioinformatics*, 17(12):1131–1142, 2001. 1, 3

[9] R. J. Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993. 1, 4

[10] C. Romualdi, S. Campanaro, D. Campagna, B. Celegato, N. Cannata, S. Toppo, G. Valle, and G. Lanfranchi. Pattern recognition in gene expression profiling using dna array: a comparative study of different statistical methods applied to cancer classification. *Human Molecular Genetics*, 12(8):823–836, 2003. 1

[11] J. Shi and Z. Luo. Nonlinear dimensionality reduction of gene expression data for visualization and clustering analysis of cancer tissue samples. *Comput. Biol. Med.*, 40(8):723–732, Aug. 2010. 1

[12] M. A. Shipp, K. N. Ross, P. Tamayo, A. P. Weng, J. L. Kutok, R. C. Aguiar, M. Gaasenbeek, M. Angelo, M. Reich, G. S. Pinkus, T. S. Ray, M. A. Koval, K. W. Last, A. Norton, T. A. Lister, J. Mesirov, D. S. Neuberg, E. S. Lander, J. C. Aster, and T. R. Golub. Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. *Nature Medicine*, 8(1):68–74, Jan. 2002. 4

[13] K. Y. Yeung and W. L. Ruzzo. Principal component analysis for clustering gene expression data. *Bioinformatics*, 17(9):763–774, 2001. 1