Classifying microarray cancer datasets using nearest subspace classification

Michael C. Cohen and Kuldip K. Paliwal

Signal Processing Lab, Griffith School of Engineering, Griffith University, Brisbane, Queensland, Australia
michael.cohen@student.griffith.edu.au, k.paliwal@griffith.edu.au

Abstract. In this paper we implement and test the recently described nearest subspace classifier on a range of microarray cancer datasets. Its classification accuracy is tested against nearest neighbor and nearest centroid algorithms, and is shown to give a significant improvement. This classification system uses class-dependent PCA to construct a subspace for each class. Test vectors are assigned the class label of the nearest subspace, which is defined as the minimum reconstruction error across all subspaces. Furthermore, we demonstrate this distance measure is equivalent to the null-space component of the vector being analyzed.

1 Introduction

DNA microarrays have provided researchers with a new tool for discovering the underlying causes of cancer. Their ability to measure the expression level of thousands of genes simultaneously has allowed researchers to neatly profile the make-up of tumorous cells [1], provide comparisons with unaffected cells [2, 3], other types of cancer [4, 5], and between subclasses of cancer [6–10].

However, the extremely high dimensionality and low sample size of microarray datasets create problems when applying traditional classifiers to the task of cancer classification, referred to as the 'curse of dimensionality' in pattern recognition literature [11]. Ironically, the solution appeared to lie in utilizing Support Vector Machines (SVM) [4, 12], which uses a 'kernel trick' to transform data into a kernel-space with an even greater dimensionality. (Typically, though, linear-SVM is used, which does not increase the number of dimensions, although it does not reduce it either.) Although originally formulated to deal with two-class problems, SVM has since been adapted successfully to perform multi-class classification [13].

There are, however, simpler classifiers that natively handle multi-class problems, such as the k-nearest neighbor algorithm [11], which have remained popular as benchmarks, but not as an active topic for research. The advantages of simple classifiers (such as relatively low memory and time requirements, and the ability to intuitively understand their results) guarantee their popularity and longevity in any pattern classification paradigm.

Arguably the simplest classifier is the nearest centroid classifier [11], where each class is represented by a single centroid, usually the mean of all training

M. Chetty, S. Ahmad, A. Ngom, S. W. Teng (Eds.): PRIB2008,
Supplementary Conf. Proc., pp.170-180, 2008.

vectors within a class. During testing, the distance between a test vector and each class centroid is found (Euclidean distance is typically used) and the vector is assigned the label of the nearest class.

In this paper, we introduce nearest subspace classification, where we utilize the centroid of each class as well as the covariance to create a hyperplane embedded in the original gene space. The hyperplane represents the subspace inhabited by an individual class; hence we refer to the classifier as a nearest subspace classifier.

The formulation and justification of the nearest subspace classifier is outlined in the next section, as well as a description of its antecedents. In Section 3, the nearest subspace classifier is applied to pre-existing microarray cancer datasets, and the results given. The results are discussed in greater depth in Section 4, detailing the strength and weaknesses of nearest subspace classification. Finally, concluding remarks are made in Section 5, with a guide to future developments.

2 Nearest Subspace Classification

Although microarray datasets typically contain thousands of genes, the inherent dimensionality of the data (relevant to cancer classification) is much lower. The fundamental biological processes that give rise to the measured gene expression profiles can be expressed with a combination of tens or hundreds of genes only [14]. Variations in gene expression data not caused by biological processes can be considered noise from the expression profiling process.

Since the data relevant to cancer classification resides on an embedded manifold with a much lower dimensionality than that of the full microarray, it is possible to use dimensionality reduction techniques, such as Principal Component Analysis (PCA) [15] in order to simplify the problem. The difficulty with using PCA is that it can only provide a global linear approximation of the intrinsic manifold. However, by applying class-dependent PCA, we are able to minimize this problem by providing a locally linear approximation of the manifold, tangential to the class means.

Each class-dependent PCA transform details the subspace inhabited by an individual class. During classification, the Euclidean distance is found between a test vector and each class subspace. The vector is assigned the label of the nearest class. To understand how this is achieved, let us first review the technique of PCA.

2.1 Principal Component Analysis

PCA is a linear transformation $U \in \mathbb{R}^{\mathrm{D}\times d}$ of a vector x to a d-dimensional subspace (i.e. $U^{\mathrm{T}}x$) such that $d \leqslant D$. U is chosen in such a way that the variance of the translated data is maximised [15]. In other words, it reduces the dimensionality of x in an optimal fashion. In order to perform PCA, we are given a set of labeled training vectors, Z, which consists of a set of N tuples, i.e.

$$Z = \{y_i, \omega_i\}, \qquad \begin{array}{l} y_i \in \mathbb{R}^{\mathrm{D}\times 1} \\ \omega_i \in \{1, 2, \ldots, \mathrm{K}\} \\ i = 1, 2, \ldots, \mathrm{N} \end{array} \tag{1}$$

where y_i is the data of the $i^{\,\mathrm{th}}$ training vector and ω_i is the corresponding label.

For normal PCA, we are only concerned with the dataset $Y = \{y_i\}$, which gives $Y \in \mathbb{R}^{\mathrm{D}\times\mathrm{N}}$. First, the empirical mean is subtracted from the training dataset Y, to create a zero-mean dataset X.

$$\begin{aligned} X &= Y - \mu\,, \\ \mu &= \frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} y_i\,. \end{aligned} \tag{2}$$

A vector is transformed into the d-dimensional subspace using the relation $x_{\mathrm{sub}} = U^{\mathrm{T}} x$, where $U \in \mathbb{R}^{\mathrm{D}\times\mathrm{d}}$ is found by solving the eigenvectors of the generalized eigenvalue problem

$$Cu_i = \lambda u_i\,. \qquad \begin{array}{l} u_i \in \mathbb{R}^{\mathrm{D}\times 1} \\ i = 1, 2, \ldots, \mathrm{D} \end{array} \tag{3}$$

where C is the covariance XX^{T}. Equation 3 results in a set of eigenvectors $U_{full} \in \mathbb{R}^{\mathrm{D}\times\mathrm{D}}$. However, the number of eigenvectors with non-zero eigenvalues, d, is less than or equal to the rank of C , i.e. $\mathrm{N} - 1$. (The presence of noise typically ensures d is not less than $\mathrm{N} - 1$.) Of the full set of D eigenvectors, only the d eigenvectors with non-zero eigenvalues are retained by PCA to form $U \in \mathbb{R}^{\mathrm{D}\times d}$. Since d is typically much less than D, there is a large dimensionality reduction.

An efficient and robust method of obtaining the eigenvectors of the training dataset is by calculating the Singular Value Decomposition (SVD) [16] of X , i.e.

$$X = USV^{\mathrm{T}}\,. \qquad \begin{array}{l} U \in \mathbb{R}^{\mathrm{D}\times\mathrm{N}} \\ S, V \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}} \end{array} \tag{4}$$

Note that SVD calculates N eigenvectors (not $d = N - 1$), so there is at least one eigenvector with a zero eigenvalue. The eigenvalues themselves are determined from the singular matrix S by the relation $\Lambda = \operatorname{diag} S^2$, where Λ is a vector containing the eigenvalues.

2.2 Range-space versus Null-space

The set of eigenvectors for which the corresponding eigenvalue is greater than zero is referred to as range-space. The set of eigenvectors (not explicitly calculated) for which the corresponding eigenvalue is zero is referred to as null-space. The combination of basis vectors in range-space and null-space form the original

space. However, since all of the variance of the training dataset is accounted for in range-space, the null-space eigenvectors are discarded to form a subspace with fewer dimensions. (As the eigenvectors are based on an estimation of the class covariance by the training data only, there will be variance in the null-space when test samples are introduced.) This method of compression, or dimensionality reduction, has been the main use of PCA in pattern recognition.

Recently, though, the null-space vectors have been used to obtain important information in classification problems. Principal component null-space analysis (or PCNSA [17, 18]) finds an approximate null-space to transform the training dataset into a subspace where all training vectors are located at approximately the same point. During testing, a vector that does not belong to a particular subspace is likely to be a greater distance from the class mean than a vector of the true class, thereby allowing a distance-based classifier to be constructed.

2.3 Class-dependent PCA

The classifier discussed in this paper, adapted from Sharma et al. [19], also uses this reasoning, but to a different purpose. In this method, the subspace of each class is calculated using class-dependent PCA. The distance of a test vector to each subspace is defined as its reconstruction error. To understand this quantity, let us review the reconstruction process. A test vector is projected into range-space using the simple formula,

$$x_{\text{range}} = U^{\mathrm{T}} x\,. \qquad x_{\text{range}} \in \mathbb{R}^{d\times 1} \tag{5}$$

To reconstruct the data into the original space, the projected data is then multiplied by U, i.e.

$$\begin{aligned} \hat{x} &= U x_{\text{range}} \\ &= U\left(U^{\mathrm{T}} x\right)\,. \end{aligned} \tag{6}$$

Since only the basis vectors in range-space are taken into account, there is some reconstruction error involved, corresponding to the null-space component of x,

$$\begin{aligned} e &= x - \hat{x} \\ &= x - U\left(U^{\mathrm{T}} x\right)\,, \end{aligned} \tag{7}$$

as shown in Fig. 1 for the trivial case $x, U \in \mathbb{R}^{2\times 1}$.

For this simple demonstration, where only a single class is illustrated, it is true to state the distance from x to the subspace is the null-space component of x . However, the distinction loses clarity when multiple classes are taken into account. This is due to the fact each class will have its own class size, N_k (i.e. the number of training vectors in the k^{th} class) and therefore a subspace with its own number of non-zero eigenvalues. To avoid irregularities in distance

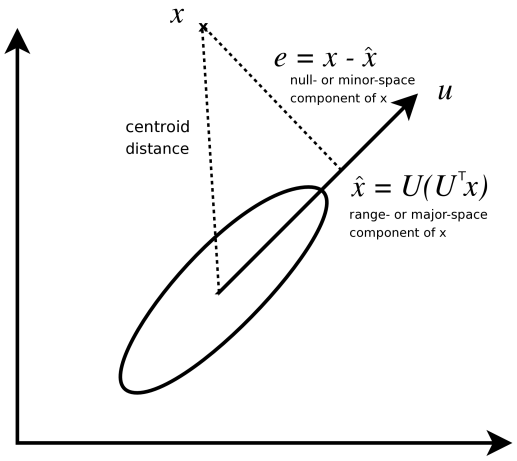

Fig. 1. When a test vector x is projected into range-/major- space, there is a reconstruction error $e = x - \hat{x}$, which is due to the null-/minor- space component of x. The vector e also quantifies the displacement of x from the subspace U . It is this distance measure which forms the heart of the nearest subspace classifier. The oval represents the shape of the class covariance.

measurements, the number of eigenvectors to quantify each subspace is set to be the minimum rank of all class covariance matrices, i.e.

$$d = \min_k N_k - 1 \,. \tag{8}$$

Note that classes with more than the minimum number of training vectors neglect eigenvectors with non-zero eigenvalues. Therefore, the reconstruction error is no longer just the null-space component of x. For this reason, a different nomenclature is used. We refer to $\hat{x}$ as the major-space component of x, and $x - \hat{x}$ as the minor-space component of x. This is analogous to an ellipse, which has major and minor axes. The subspace of the k^{th} class, U_k , is a set of basis vectors that quantifies major-space for that particular class.

2.4 Nearest-subspace Classifier

The nearest subspace classifier consists of a training phase and a test (or classification) phase. During training, we are given a set of labeled training vectors Z defined by eq. 1. First, the data is separated into classes

$$A_k = \{y_i : \omega_i = k\} \quad . \qquad \begin{array}{l} i = 1, 2, \ldots, \mathrm{N} \\ A_k \in \mathbb{R}^{\mathrm{D} \times \mathrm{N_k}} \\ k = 1, 2, \ldots, \mathrm{K} \end{array} \tag{9}$$

Then the class-dependent mean is found for each class, and the mean subtracted from the class dataset

$$\begin{aligned} \mu_k &= \frac{1}{N_k}\sum_{i=1}^{N_k} a_{ki} \qquad a_{ki} \in \mathbb{R}^{\mathrm{D}\times 1} \\ X_k &= A_k - \mu_k \qquad k = 1, 2, \ldots, \mathrm{K} \end{aligned} \tag{10}$$

SVD is performed to give the subspace U_k

$$X_k = U_k S V^{\mathrm{T}} \qquad k = 1, 2, \ldots, \mathrm{K} \tag{11}$$

Finally, the subspace is trimmed to retain the d eigenvectors (see eq. 8) for which the corresponding eigenvalues are highest.

During the classification phase, we are given an independent test vector $y \in \mathbb{R}^{\mathrm{D}\times 1}$ (i.e. test data is strictly prohibited from the training phase). For each class, the mean is subtracted to give

$$x_k = y - \mu_k \qquad k = 1, 2, \ldots, \mathrm{K} \tag{12}$$

The sample is assigned the label of the nearest subspace, i.e. the class with the least reconstruction error

$$\omega = \arg\min_k \left\| x_k - U_k \left(U_k^{\mathrm{T}} x_k \right) \right\| \tag{13}$$

Notice that this distance metric is very similar to the nearest centroid classifier, with only the modifying factor $U_k \left(U_k^{\mathrm{T}} x_k \right)$ providing any difference.

3 Experimentation and Results

Nine microarray cancer datasets available on the internet were gathered to test the performance of the nearest subspace classifier on a variety of microarray sources. To assess classification ability, the results were compared against two other simple classifiers k-nearest neighbor and nearest centroid. For all experiments, one nearest neighbor was used. Results for SVM classification on these datasets have been widely reported, and were not considered for this publication.

3.1 Datasets

Two types of microarray datasets were used; oligonucleotide microarrays (produced by Affymetrix) and cDNA microarrays. For Affymetrix data, preprocessing steps from [20] were used, including gene filtering1 and logarithmic transformation. For cDNA datasets, missing data was imputed by replacing missing values with the average expression level of the gene across all samples. Although this is not an optimal procedure, the focus of the paper is to determine the relative performance of the nearest subspace classifier, for which the row-average

method is adequate. Standardization of data (zero empirical mean, unit variance), which is recommended in SVM studies, proved to be of little use in this case, and was not implemented.

A quick summary of the properties of each dataset is provided in Table 1.

Table 1. A quick summary of each dataset (K refers to the number of classes, D the number of genes, and N is the total number of samples (test and training)). For more details on each database, consult the references.

Dataset	Type	K	N	D	D_{orig} [a]
ALL/AML [6]	Affy.	3	72	3571	5327
BRN [b] [5]	cDNA	13	254	7452	7452
COLON [2]	Affy.	2	62	1224	2000
GCM [4]	Affy.	14	198	10820	15009
LUNG [9]	Affy.	5	203	1741	12600
MLL [8]	Affy.	3	72	8681	11225
PDL [7]	Affy.	6	248	12011	12600
PROST [3]	Affy.	2	102	404	10509
SRBCT [10]	Affy.	4	83	2308	2308

[a] the number of genes prior to gene filtering

[b] The skin cancer class was removed due to small sample size

3.2 Experiment Methodology

Each database was tested three times; first with the full set of genes, then with the top 1000 ranking genes (selected using BW-ratio, as per [20]), then the top 100 ranking genes.

For each of the three experiments, 3-fold stratified cross-validation was utilized, with 50 iterations, giving a total of 150 trials per experiment. Gene selection was performed independently during each trial.

3.3 Results

The overall results of the experiments are given in Tables 2 to 4. No results are given for the PROST dataset for the 1000 gene experiment because the dataset contained an insufficient numbers of genes after gene filtering.

The results show the nearest subspace classifier generally outperforms nearest neighbor and nearest centroid algorithms, with gene selection and without. When it is not the best performing classifier, it is always within 1.1% of the greatest performance, and is often less than half of one percent off. On the other hand, it can provide significant improvements, as exemplified by the GCM, PROST and SRBCT results. The classification performance on the BRN, MLL and PDL datasets also show promise.

Table 2. Classification performance (% accuracy) of 1-nearest neighbor (NN), nearest centroid (NC) and nearest subspace (NS) classifiers over 150 trials, using all genes after gene filtering. Highest accuracy for each dataset is in bold type. The standard deviation of classification performance appears in brackets.

Dataset	NN	NC	NS
ALL/AML	83.5 (6.8)	85.8 (6.5)	**89.0 (4.6)**
BRN	91.9 (2.2)	94.6 (1.9)	**95.0 (1.6)**
COLON	**76.9 (8.0)**	75.8 (12.1)	76.2 (7.1)
GCM	63.6 (4.0)	54.3 (4.9)	**71.6 (4.2)**
LUNG	93.0 (2.5)	92.8 (2.8)	**94.4 (2.1)**
MLL	92.7 (4.8)	**94.7 (4.4)**	94.6 (4.5)
PDL	87.7 (3.3)	**93.4 (2.4)**	93.1 (2.2)
PROST	78.1 (5.8)	66.7 (8.1)	**87.2 (4.9)**
SRBCT	91.6 (4.2)	89.6 (7.0)	**95.2 (3.7)**

Table 3. Classification performance (% accuracy) of 1-nearest neighbor (NN), nearest centroid (NC) and nearest subspace (NS) classifiers over 150 trials, using the top 1000 ranked genes. Highest accuracy for each dataset is in bold type. The standard deviation of classification performance appears in brackets.

Dataset	NN	NC	NS
ALL/AML	82.3 (6.2)	86.2 (6.2)	**87.6 (5.9)**
BRN	94.4 (1.8)	94.1 (2.0)	**94.9 (1.8)**
COLON	**77.3 (7.4)**	74.6 (14.0)	76.7 (6.3)
GCM	64.4 (4.3)	55.3 (5.0)	**69.7 (4.6)**
LUNG	93.2 (2.4)	92.6 (2.6)	**94.5 (2.3)**
MLL	96.0 (3.9)	93.0 (5.0)	**97.0 (2.9)**
PDL	97.1 (1.6)	97.5 (1.6)	**98.8 (1.1)**
PROST	-	-	-
SRBCT	98.2 (2.7)	94.5 (5.3)	**99.4 (1.9)**

Table 4. Classification performance (% accuracy) of 1-nearest neighbor (NN), nearest centroid (NC) and nearest subspace (NS) classifiers over 150 trials, using the top 100 ranked genes. Highest accuracy for each dataset is in bold type. The standard deviation of classification performance appears in brackets.

Dataset	NN	NC	NS
ALL/AML	81.1 (6.0)	**86.0 (4.7)**	85.9 (5.7)
BRN	90.4 (2.7)	86.9 (2.9)	**90.9 (2.7)**
COLON	81.0 (6.0)	**81.1 (8.8)**	80.5 (6.8)
GCM	61.0 (4.8)	59.7 (5.0)	**66.5 (5.0)**
LUNG	93.0 (2.6)	93.2 (2.8)	**93.7 (2.4)**
MLL	94.9 (3.4)	94.5 (3.5)	**96.4 (3.2)**
PDL	96.2 (1.8)	**96.7 (1.7)**	95.6 (2.0)
PROST	80.5 (5.9)	74.3 (9.2)	**84.9 (5.4)**
SRBCT	98.9 (2.1)	96.4 (3.5)	**99.2 (2.1)**

4 Discussion

The nearest subspace classifier, whilst similar in structure to the nearest centroid classifier, represents a consistent improvement in classification accuracy.

Although the accuracy is not always comparable to the most popular classifier, SVM, the nearest subspace classifier represents a fundamentally different approach to classification of microarray data. SVM is a maximum margin classifier, where outliers are used to construct classification boundaries. Nearest subspace classification (as mentioned in the introduction) attempts to create a series of subspaces, which form a locally linear approximation of the intrinsic manifold. This is potentially significant because quantifying the manifold of gene expression data could prove useful towards understanding the biological processes that define cancer.

As microarray technology improves, three things are likely to occur: the use of more probes will result in a more complete picture of gene profiles (as well as increase the importance of gene selection); a decrease in cost will make databases more abundant and allow more samples to be included; and finally, an increase in precision will result in less process noise on the data, and allow the manifold to be constructed more clearly. These last two developments in particular will assist the performance of manifold-learning techniques.

Manifold learning algorithms have recently been shown to provide performance capable of matching, or even exceeding, that of SVM [21].

4.1 Paremetric versus Non-parametric Classification

Due to the large dimensionality and small training dataset size of microarray datasets, non-parametric classifiers, such as SVM and k-nearest neighbor, have remained the most popular method of classification. While they have produced meaningful results, it is in some way disadvantageous not to have information about the underlying distribution of the data. To undertake such tasks as manifold learning, which could potentially have benefits in the way we understand cancer diagnosis, we must therefore turn to parametric methods.

The nearest centroid classifier is a parametric classifier, which assumes all statistical sources are Gaussian, with identical spheroid covariance matrices. (In spite of its simplicity, or maybe because of it, the nearest centroid classifier performed exceedingly well on certain datasets. When it proved to be significantly worse than the nearest neighbor classifier, the nearest subspace classifier proved to be significantly better). The nearest subspace classifier represents an added layer of sophistication, where the class covariances are assumed to be ellipsoid, allowing a greater level of distinction between classes.

Although the added complexity has improved results, the difficulty of estimating the covariance of a class from as few as four training vectors (as occurred during three-fold stratified cross-validation) can render the problem almost nonsensical. It would be interesting to see what effect estimating the class covariance matrix by other than the normal covariance formula would have on classification performance.

5 Conclusion

The nearest subspace algorithm is a fast, efficient and accurate method of classification for microarray cancer datasets. It represents an improvement in classification performance on the nearest centroid and nearest neighbor algorithms. This suggests manifold learning in the bioinformatics domain, specifically for microarray analysis, is a viable topic for research.

While PCA was utilized to obtain the class-dependent subspaces, there are other methods, such as Independent Component Analysis (ICA) and kernel PCA that might be tested to see how they compare. PCA remains attractive, however, with its ease of reconstruction and simple formulation.

Also worth investigating is the estimation of the class covariance matrices, used to obtain the eigenvectors and eigenvalues of the subspace, by means other than the traditional covariance formula. Methods such as Parzen windowing and regularization may prove useful in overcoming the small number of training vectors inherited by certain datasets.

References

1. Alizadeh, A.A., Eisen, M.B., Davis, E.E., Ma, C., Lossos, I.S., Rosenwald, A., Boldrick, J.C., Sabet, H., Tran, T., Yu, X., Powell, J.I., Yang, L., Marti, G.E., Moore, T., Hudson, J., Lu, L., Lewis, D.B., Tibshirani, R., Sherlock, G., Chan, W.C., Greiner, T.C., Weisenburger, D.D., Armitage, J.O., Warnke, R., Levy, R., Wilson, W., Grever, M.R., Byrd, J.C., Botstein, D., Brown, P.O., Staudt, L.M.: Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling. Nature **403**(6769) (February 2000) 503–511
2. Alon, U., Barkai, N., Notterman, D.A., Gish, K., Ybarra, S., Mack, D., Levine, A.J.: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proc. Nat. Acad. Sci. U.S.A. **96**(12) (June 1999) 6745–6750
3. Singh, D., Febbo, P.G., Ross, K., Jackson, D.G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A.A., D'Amico, A.V., Richie, J.P., Lander, E.S., Loda, M., Kantoff, P.W., Golub, T.R., Sellers, W.R.: Gene expression correlates of clinical prostate cancer behavior. Cancer Cell **1**(2) (March 2002) 203–209
4. Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Yeang, C.H., Angelo, M., Ladd, C., Reich, M., Latulippe, E., Mesirov, J.P., Poggio, T., Gerald, W., Loda, M., Lander, E.S., Golub, T.R.: Multiclass cancer diagnosis using tumor gene expression signatures. Proc. Nat. Acad. Sci. U.S.A. **98**(26) (December 2001) 15149–15154
5. Munagala, K., Tibshirani, R., Brown, P.O.: Cancer characterization and feature set extraction by discriminative margin clustering. BMC Bioinformatics **5** (2004) 21
6. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science **286**(5439) (October 1999) 531–537
7. Yeoh, E.J., Ross, M.E., Shurtleff, S.A., Williams, K.W., Patel, D., Mahfouz, R., Behm, F.G., Raimondi, S.C., Relling, M.V., Patel, A., Cheng: Classification, subtype discovery, and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling. Cancer Cell **1**(2) (2002) 133–143

8. Armstrong, S.A., Staunton, J.E., Silverman, L.B., Pieters, R., den Boer, M.L., Minden, M.D., Sallan, S.E., Lander, E.S., Golub, T.R., Korsmeyer, S.J.: Mll translocations specify a distinct gene expression profile that distinguishes a unique leukemia. Nat. Genet. **30** (Jan 2002) 41–47
9. Bhattacharjee, A., Richards, W.G., Staunton, J., Li, C., Monti, S., Vasa, P., Ladd, C., Beheshti, J., Bueno, R., Gillette, M., Loda, M., Weber, G., Mark, E.J., Lander, E.S., Wong, W., Johnson, B.E., Golub, T.R., Sugarbaker, D.J., Meyerson, M.: Classification of human lung carcinomas by mrna expression profiling reveals distinct adenocarcinoma subclasses. Proc. Nat. Acad. Sci. U.S.A. **98** (Nov 2001) 13790–13795
10. Khan, J., Wei, J.S., Ringnér, M., Saal, L.H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C.R., Peterson, C., Meltzer, P.S.: Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nat. Med. **7**(6) (June 2001) 673–679
11. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification (2nd Edition). Wiley-Interscience (November 2000)
12. Vapnik, V.N.: The Nature of Statistical Learning Theory (Information Science and Statistics). Springer (November 1999)
13. Statnikov, A., Aliferis, C.F., Tsamardinos, I., Hardin, D., Levy, S.: A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. Bioinformatics **21**(5) (March 2005) 631–643
14. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. Machine Learning **46**(1-3) (2002) 389–422
15. Bishop, C.M.: Pattern Recognition and Machine Learning (Information Science and Statistics). Springer (August 2006)
16. Strang, G.: Linear Algebra and Its Applications. Brooks Cole (July 2005)
17. Vaswani, N., Chellappa, R.: Principal components null space analysis for image and video classification. **15**(7) (July 2006) 1816–1830
18. French, L., Ngom, A., Rueda, L.: Fast protein superfamily classification using principal component null space analysis
19. Sharma, A., Paliwal, K.K., Onwubolu, G.C.: Class-dependent pca, mdc and lda: A combined classifier for pattern classification. Pattern Recognition **39**(7) (2006) 1215–1229
20. Dudoit, S., Fridlyand, J., Speed, T.P.: Comparison of discrimination methods for the classification of tumors using gene expression data. J. Am. Stat. Assoc. **97**(457) (2002) 77–87
21. Lee, J., Zhang, C.: Classification of gene-expression data: The manifold-based metric learning way. Pattern Recogn. **39**(12) (2006) 2450–2463