

Comparative evaluation of support vector machines and probabilistic neural networks in superficial bladder cancer classification

P. Spyridonos^{a,*}, P. Petalas^a, D. Glotsos^a, D. Cavouras^c, P. Ravazoula^b and G. Nikiforidis^a

^a*Computer Laboratory, School of Medicine, University of Patras, Rio, Patras 265 00, Greece*

^b*Department of Pathology, University Hospital, Rio, Patras, 265 00, Greece*

^c*Department of Medical Instrumentation Technology, Technological Education Institution of Athens, Ag. Spyridonos Street, Aigaleo, 122 10 Athens, Greece*

Abstract. *Purpose:* In this paper we address the demanding diagnostic problem of classifying tumors according to the degree of their malignancy by investigating the efficiency of Support Vector Machines (SVMs) and Probabilistic neural networks (PNN).

Material and methods: 129 cases of urinary bladder carcinomas were diagnosed as high or low-risk according to the WHO grading system. Each case was represented by 36 automatically extracted nuclear features. Two different classification designs based on SVMs and PNNs were tested according to their ability in differentiating superficial urinary bladder carcinomas according to the degree of malignancy. Best feature combination for each classification scheme was obtained performing an exhaustive search in feature space and employing the leave-one-out method.

Results: Both classification models (SVM and PNN) resulted in a relatively high overall accuracy of 85.3% and 83.7% respectively. Descriptors of nuclear size and chromatin cluster patterns were participated in both best feature vectors that optimized classification performance of the two classifiers.

Conclusion: The good performance and consistency of the SVM and PNN models render these techniques viable alternatives in the diagnostic process of assigning urinary bladder tumors grade.

Keywords: Support vector machines, probabilistic neural networks, image analysis, urinary bladder tumors

1. Introduction

Bladder cancer is the fifth most common cancer in the western male population [1]. Microscopic visual analysis of histopathological material provides an index of disease severity and tumor grading according to the degree of malignancy determines the choice and form of treatment [2]. However, the recognition of a variety of histopathological findings in tissue biopsies by human requires high level of skill and knowledge [3]. In addition, inter and intra-observer low reproducibility has been shown to influence the quality of diagnosis [4]. Recent developments in digital image analysis techniques and classification systems offer solutions to objectively grading tumors and evaluating new prognostic features. Previous studies aimed at bladder cancer classification, have described methods based on quantitative structural and textural tissue features [5,6]. More recent approaches have proposed the application of Bayesian

*Corresponding author: Panagiota Spyridonos, Ph.D., Computer Laboratory, School of Medicine, University of Patras, 265 00 Rio-Patras, Greece. Tel.: +30 61 997745; Fax: +30 61 997882; E-mail: Spyridonos@med.upatras.gr.

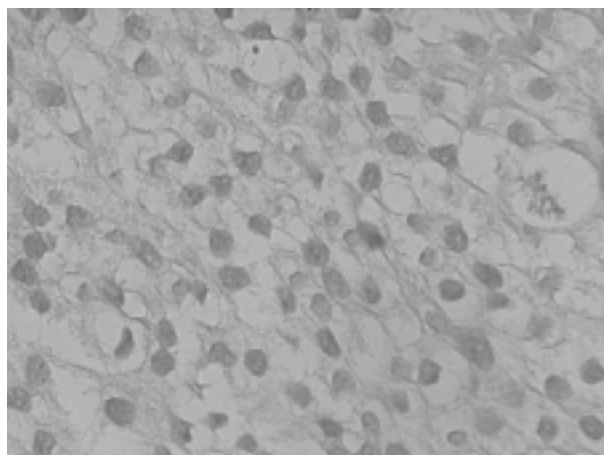


Fig. 1. Tissue sample of bladder carcinoma.

Belief Networks in grade diagnosis of bladder cancer utilizing histological features estimated subjectively by pathologists [7]. So far, little effort has been made to investigate the potential of morphological and textural nuclear features in computer-based grading systems [8]. The first thing that must be taken under consideration for designing a recognition system is to achieve the best possible classification performance for a particular task. Experimental results from different classification designs would be the basis for choosing the optimal classification model for the particular application [10].

In this study a comparative evaluation of Support Vector Machines (SVMs) was performed exploring their ability to classify superficial bladder carcinomas as low or high risk according the WHO grading system [11]. SVMs were selected due to their ability to generalize and maintain good performance in unseen data [12]. On the other hand, PNNs learn to build nonlinear decision boundaries, which approach to optimal Bayes classifier decision surfaces [13].

The contribution of the present study lies in the use of robust, and fast, algorithms for addressing the demanding problem of diagnosis of tumor malignancy which is crucial for patient management. Additionally, there is no previous work where PNN and/or SVMs have been explored as pattern recognition schemes incorporating information from histological images for the automatic classification of tumors.

2. Material and methods

129 cases with bladder cancer were collected from the University Hospital of Patras in Greece. Of the 129 patients, 92 were diagnosed as low grade and 37 as high grade from two independent pathologists. Images (fields) from tissue specimens were captured using a light microscopy imaging system. Each digitized image ($768 \times 576 \times 24$ -bit resolution) was converted into an 8-bit gray scale image for further processing and analysis (Fig. 1).

From each case a representative sample of nuclei (about 70) was isolated using an automatic segmentation technique (Fig. 2) [8].

Two kinds of quantitative parameters were estimated:

- Morphological features related to nuclear size and shape distribution
- Textural features related to nuclear chromatin organization [14].

Table 1
Construction of a Truth Table

Classification performance			
Patient Outcome	Low-risk	High-risk	Accuracy
Low-risk	n11	n12	$\frac{n_{11}}{n_{11}+n_{12}}\%$
High-risk	n21	n22	$\frac{n_{22}}{n_{21}+n_{22}}\%$

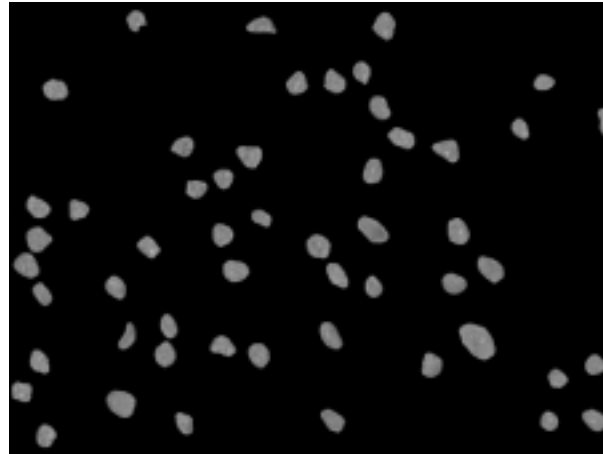


Fig. 2. Processed image indicating the segmented nuclei.

In the training process of SVM and PNN no iterative procedures are used and no feedback paths are required. The latter enabled us to perform an exhaustive search method for the optimal selection of the best feature vector combination. The problem of optimum feature selection is defined as follows: Given the set of 36 features select a subset of size m that leads to the smallest classification error. Features were combined in all possible ways (combinations of 2, 3, 4, 5, 6 features). Each time the feature subset was inserted into the classifier and its performance was obtained using the leave-one-out method [15]. According to this method, all cases except one are used for designing the classifier. The one left out is then used for testing. The procedure is repeated as many times as the number of cases. Results are presented in truth tables revealing the classification accuracy of each classifier. More specifically, accuracy is determined as

$$\% \text{ Overall Accuracy} = \frac{n_{11}+n_{12}}{n_{11}+n_{12}+n_{21}+n_{22}}$$

Where n_{11} and n_{22} presents the number of the correct classified sample data and n_{21} and n_{21} presents the number of the misclassified sample data. The results are presented in the form of the so-called truth table (Table 1).

2.1. Support vector machines

SVMs represent novel classification techniques based on statistical learning theory developed by Vapnik and Chervonenkis (VC) [12]. The basic idea behind SVMs is twofold: Map the input feature space into a higher dimension using a transformation function (kernel) and then define the maximal margin hyperplane that separates optimally the training data.

Let S be a set of l vectors $x_i \in R^n$, ($i = 1, 2, \dots, l$) in a n -dimensional space. Each vector x_i belongs to either of two classes identified by the label $y_i \in \{-1, 1\}$. If the two classes are linearly separable, then there exists a hyperplane, defined by:

$$w \cdot x + b = 0, \tag{1}$$

which divides S leaving all the vectors of the same class on the same side.

Then the Maximal Margin Hyperplane (MMH), which maximizes the distance of the closest vectors of the different classes, is given by the solution to the problem minimize

$$\frac{1}{2} \|w\|^2 \tag{2}$$

$$\text{with } y_i (w \cdot x_i + b) \geq 1 \tag{3}$$

$$\text{where } \frac{b}{\|w\|} \text{ is the distance between origin and hyperplane.} \tag{4}$$

Karush-Kuhn-Tucker (KKT) theorem gives the solution to this quadratic programming problem [9]. By the denotation of the l non-negative Lagrange multipliers $\alpha = (a_1, a_2, \dots, a_l)$, the problem is equivalent to Wolfe dual problem:

$$\text{maximize } \sum_{i=1}^l a_i - \frac{1}{2} \sum_{i,j=1}^l y_i y_j a_i a_j x_i^T x_j \tag{5}$$

$$\text{subject to } \sum_{i=1}^l a_i y_i = 0 \text{ and } a_i \geq 0 \tag{6}$$

$$\text{The solution of } w \text{ is: } w = \sum_{i=1}^l a_i y_i x_i \tag{7}$$

Since most of the a_i are usually null, the vector w is a linear combination of an often relatively small percentage of the vectors x_i . These vectors are termed support vectors and they are the only vectors of S needed to determine the MMH. In case of non-linear separating hyper surfaces, each vector x in input space is mapped into a vector $z = \Phi(x)$ in a higher dimensional feature space. Instead of the dot product $\langle \Phi(x), \Phi(y) \rangle$, a non-linear function $K(x, y)$, named kernel, is used. The KKT transform to:

$$\text{Maximize } \sum_{i=1}^l a_i - \frac{1}{2} \sum_{i,j=1}^l y_i y_j a_i a_j K(x_i x_j) \tag{8}$$

$$\text{Subject to } \sum_{i=1}^l a_i y_i = 0 \text{ and } a_i \geq 0 \tag{9}$$

$$\text{Where } K(x_i, x_j) = \Phi^T(x_i) \cdot \Phi^T(x_j) \text{ the kernel function.} \tag{10}$$

Mercer conditions define possible kernel functions.

Table 2
Truth table demonstrating classification results using the SVM classifier

Classification performance			
Patient Outcome	Low-risk	High-risk	Accuracy
Patient Outcome	Low-risk	High-risk	Accuracy
Low-risk	84	8	91.3%
High-risk	11	26	70.3%
Accuracy			85.3%

In this work SVM classifier was constructed using the Gaussian radial basis function (RBF) kernel indicated in Eq. (11).

$$K_{\text{RBF}}(x, x_i) = \exp\left(-\frac{1}{2}(x - x_i)^T \sum^{-1} (x - x_i)\right) \tag{11}$$

The optimization problem of finding the Lagrange multipliers (Eqs (8)–(9)) was solved by using the routine quadprod [16] provided with the MATLAB optimization toolbox. The realization of this problem in Matlab code is provided in Appendix I.

2.2. Probabilistic neural networks

PNN architecture is an implementation of the Bayes classifier, which minimizes the expected risk of classifying patterns in the wrong category. Without any real knowledge of the underlying probability distribution form of the training data, PNN are using non-parametric estimation methods based on the use of Parzen windows [13].

PNN consists of an input layer followed by two computational layers and one output unit. The first computational layer has *k* units, one for each training pattern and computes distances from the input vector to the training input vectors. Thus a vector whose elements indicate how close the input is to training input is produced. This vector is exponentiated by a radial basis activation function and is passed as an input to the next computational layer. In the second computation layer, the pattern-layer outputs are selectively connected to two summation units depending on the class of patterns they represent. The output unit produces a binary output signal indicating the most probable class membership for a particular input vector. (Matlab code is available in Appendix II).

3. Results

Both classification models (SVM and PNN) resulted in a relatively high overall accuracy of 85.3% and 83.7% respectively, in discriminating low from high-risk tumors. Considering the SVM classifier, the best feature vector combination comprised two morphological features describing nuclear shape distribution (range of area and standard deviation of area) and two textural features derived from co-occurrence matrices encoding chromatin clusters pattern. The textural features were the cluster shade for inter-pixel distance $d = 1$ and $d = 3$. As shown in Table 2, 91.3% (84/92) low-risk cases were correctly classified whereas 11 cases were misclassified as high-risk. On the other hand, high-risk classification success was 70.3% (26/37) with 8 cases incorrectly assigned as belonging to the low-risk group. For the PNN classifier the best classification result was obtained utilizing a 4 dimensional feature vector consisting of

Table 3
Truth table demonstrating classification results using the PNN classifier

Patient Outcome	Classification performance		Accuracy
	Low-risk	High-risk	
Low-risk	81	11	88.0%
High-risk	10	27	73.0%
Accuracy			83.7%

three morphological features (the nuclear concavity, range of roundness and standard deviation of area) and one textural feature from co-occurrence matrices the cluster shade for inter-pixel distance $d = 1$. Low-risk cases were classified with 88% accuracy whereas high-risk with 73% (Table 3).

4. Discussion

Computer-based methods for the automatic description of the degree of tumors malignancy have been the major goal for many research groups in the field of image analysis and pattern recognition applications [5–7]. Although many research groups have indicated that nuclear features carry useful diagnostic and prognostic information, less effort has been made to design automated classification systems based exclusively on these features. Utilizing the diagnostic potentiality of nuclear features, two different classification designs based on SVMs and PNNs were tested according to their ability in differentiating superficial urinary bladder tumors. Although the theory of SVMs was developed in the late seventies [12] and PNNs were introduced about ten year ago [13], it is only now receiving increasing attention in the medical domain. The most attractive characteristic of SVMs is that provide bounds on the generalization error of the model in the framework of structural risk minimization. On the other hand, decision boundary implemented by the PNNs asymptotically approaches the Bayes optimal decision surface. Additionally, the selected classifiers' setting is simple and computationally efficient. Both classification models exhibited relatively high overall accuracy. SVMs gave an overall accuracy of 85.3%. In the low-risk group the correct classification was 91.3%. High-risk cases were classified with an accuracy of 70.3%. PNNs exhibited an overall accuracy of 83.7%. 88% and 73% were the correct classification rates for the low and high-risk groups respectively. The rather smaller classification accuracy in the high risk group compared with the respective of the low-risk group might be due to the different size of data sets (37 high-risk cases versus 92 low-risk). Worth mentioning is that both classification models indicated relevant features during the optimization process of their performance. Descriptors of nuclear size and chromatin cluster patterns, such as the standard deviation of area and the cluster shade, were participated in both best feature vectors that optimized classification performance of the two classifiers. The latter enforces the belief that certain nuclear features carry significant diagnostic information. Concluding, the good performance and consistency of the SVM and PNN models render these techniques viable alternatives in the diagnostic process of classifying tumors according to their degree of malignancy.

Appendix I

**%MATLAB code for SVM classifier design employing an exhaustive
%search procedure and testing with the Leave One Out Method**

```

m=input('m'); %m is the feature vector dimension
N=36;% N is the total number of features
vector=nchoosek([1:N],m);
T=[-1*ones(1,size(setA,1)) ones(1,size(setB,1))]; %Label for each pattern
data=[setA;setB];
%setA: low-risk cases, setB:high-risk cases
%each row from setA/setB is treated as a pattern
[v1 v2]=size(vector);
to=clock;
for m=1:v1 % Exhaustive search procedure
    %data normalization
    v=vector(m,:);
    sif=data(:,v);
    co=cov(sif);
    sif=sif';
    [U,L]=eig(co);
    mo=mean(sif')';
    for i=1:size(data,1)
        po(:,i)=(L^(1/2))*U'*(sif(:,i)-mo);
    end
    set1=po(:,1:size(setA,1));
    set2=po(:,size(setA,1)+1:size(data,1));
    P=[set1 set2]';
    lowrisk_falses=0;
    highrisk_falses=0;
    for k=1:size(data,1) %Leave one out method
        Ttest=T(k);
        Ptest=P(k,:);
        Ttrain=T;
        Ttrain(k)=[];
        Ptrain=P;
        Ptrain(k,:)=[];
        rows = size(Ptrain,1);
        for i=1:rows
            for j=1:rows
                H(i,j)=T(i)*T(j)*exp(-0.5*((Ptrain(i,:)-Ptrain(j,:))...
                    *inv(cov(Ptrain,1))*(Ptrain(i,:)-Ptrain(j,:))'));
            end
        end
        f=-ones(rows,1);
        [xo,fval,exitflag,output,lambda] = quadprog(H,f,[],[],...
            Ttrain',0,zeros(rows,1), ones(rows,1));
        alphay=xo.*Ttrain;
        w=alphay.*Ptrain;
        idx=find(xo>0.1);
    end
end

```

```

sv=Ptrain(idx,:);
for i=1:rows
    ko(i)= alphas(i)*exp(-0.5*((Ptest-Ptrain(i,:))...
        *inv(cov(Ptrain,1))*(Ptest-Ptrain(i,:))'));
end
score=sum(ko);
Label=Ttest;
if (Label==-1&score>0)
    lowrisk_falses=nonrec_falses+1;
end
if (Label==1&score<0)
    highrisk_falses=rec_falses+1;
end
end
f1=nonrec_falses;
f2=rec_falses;
f(m,:)=[f1 f2];
end
s=sum(f');
mo=min(s);
idx=find(s==mo)
vector(idx,:) %find the best vector
fo=f(idx,:) %find the minimum misclassification rate for each diagnostic category
((size(setA,1)-fo(1,1))/size(setA,1))*100% specificity
((size(setB,1)-fo(1,2))/size(setB,1))*100% sensitivity

```

Appendix II

%MATLAB code for Probabilistic Neural Network design employing an exhaustive search procedure and testing with the Leave One Out Method

```

T=[2*ones(1,size(setA,1)) ones(1,size(setB,1))];
%setA: low-risk cases, setB:high-risk cases
%each row from setA/setB is treated as a pattern
data=[setA;setB];
m=input('m');% m is the feature vector dimension
vector=nchoosek([1:36],m); %36 is the total number of features
[v1 v2]=size(vector);
for m=1:v1 % Exhaustive search procedure
    %data normalization
    v=vector(m,:);
    sif=data(:,v);
    co=cov(sif);
    sif=sif';
    [U,L]=eig(co);
    mo=mean(sif')';

```

```

for i=1:size(data,1)
    po(:,i)=(L^(1/2))*U*(sif(:,i)-mo);
end
set1=po(:,1:size(setA,1));
set2=po(:,size(setA,1)+1:size(data,1));
P=[set1 set2];
lowrisk_falses=0;
highrisk_falses=0;
for j=1:size(data,1)
    Ttest=T(j);
    Ptest=P(:,j);
    Ttrain=[T(1:j-1) T(j+1:size(data,1))];
   Ptrain=[P(:,1:j-1) P(:,j+1:size(data,1))];
    Tctrain=ind2vec(Ttrain);
    at=radbas(netprod(dist(Ptrain',Ptest),(0.8326/0.06)));
    a=vec2ind(compet(Tctrain*a));
    if (Ttest==2&a==1)
        lowrisk_falses=nonrec_falses+1;
    end
    if (Ttest==1&a==2)
        highrisk_falses=rec_falses+1;
    end
end
f1=lowrisk_falses;
f2=highrisk_falses;
f(m,:)=[f1 f2];
end
s=sum(f');
m=min(s);
idx=find(s==m)
vector(idx,)%find the best vector
fo=f(idx,:) %find the minimum misclassification rate for each diagnostic category
((size(setA,1)-fo(1,1))/size(setA,1))*100%spesificity
((size(setB,1)-fo(1,2))/size(setB,1))*100%sensitivity

```

References

- [1] S. Parker, T. Tony, S. Bolgen and P. Wingo, Cancer Statistics 1997, *CA Cancer J Clin* **47**(5) (1997), 2–27.
- [2] D. Bostwick, D. Rammani and L. Cheng, Diagnosis and grading of bladder cancer and associated lesions, *Urol Clin North Am* **26** (1999), 493–507.
- [3] G. Wied, G. Bahr and P. Bartels, Laboratory organization in the detection and diagnosis of early cervical neoplasia, *Obstetrics Gynec Survey* **24** (1969), 935–966.
- [4] E. Ooms, W. Aderson, C. Alons, M. Boon and R. Veldhuizen, Analysis of the performance of pathologists in grading of bladder tumors, *Human Pathology* **14** (1983), 140–143.
- [5] H.-K. Choi, J. Vasko, E. Bengtsson, T. Jarkans, U. Malmstrom, K. Wester and C. Bush, Grading of transitional cell bladder carcinoma by texture analysis of histological sections, *Analytical Cellular Pathology* **6** (1994), 327–343.

- [6] T. Jarkans, J. Vasko, E. Bengtsson, H.-K. Choi, U. Malmstrom, K. Wester and C. Bush, Grading of transitional cell bladder carcinoma by image analysis of histological sections, *Analytical Cellular Pathology* **18** (1995), 135–158.
- [7] R. Montironi, Improving Inter-Observer Agreement and Certainty Level in Diagnosing and Grading Papillary Urothelial Neoplasms: Usefulness of a Bayesian Belief Network, *European Urology* **41**(4) (2002), 449–457.
- [8] P. Spyridonos, V. Ravazoula, D. Cavouras, K. Berberidis and G. Nikiforidis, Computer-based grading of Hematoxylin-Eosin stained tissue sections of urinary bladder carcinomas, *Medical Informatics and the Internet in Medicine* **26**(3) (2001), 179–190.
- [9] P. Spyridonos, D. Cavouras, V. Ravazoula and G. Nikiforidis, A computer-based diagnostic and prognostic system for assessing urinary bladder tumor grade and predicting cancer recurrence, *Medical Informatics and the Internet in Medicine* **7**(2) (2002), 111–122.
- [10] J. Kittler, M. Hatef, P. Duin and J. Matas, On combining classifiers, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **20**(3) (1998), 226–239.
- [11] F. Mostofi, L. Davis and I. Sesterhenn, *WHO Histologic Typing of Urinary Bladder Tumors*, Springer, Berlin, 1999.
- [12] V. Kechman, *Learning and Soft Computing*, MIT, 2001.
- [13] D. Specht, Probabilistic neural networks, *Neural Networks* **3**(1) (1990), 109–118.
- [14] R. Harralick, K. Shanmugum and I. Dinstein, Textural features for image classification, *IEEE Transactions on System, Man and Cybernetics* **3**(6) (1994), 610–621.
- [15] S. Theodoridis and K. Koutroubas, *Pattern Recognition*, Academic Press, 1999.
- [16] N. Christianini and J.S. Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press, 2000.
- [17] S. Fukuzawa, T. Hashimura, M. Sasaki, H. Yamabe and O. Yoshida, Nuclear morphometry for improved prediction of the prognosis of human bladder carcinoma, *Cancer* **10**(76) (1995), 1790–1796.