3

Classification of Soft Tissue Tumors by Machine Learning Algorithms

Jaber Juntu[1], Arthur M. De Schepper[2], Pieter Van Dyck[2], Dirk Van Dyck[1], Jan Gielen[2], Paul M. Parizel[2] and Jan Sijbers[1]

[1]*Universiy of Antwerp, Physics Department, Vision Lab.*
[2] *Dept. of Radiology, Antwerp University Hospital, University of Antwerp Belgium*

1. Introduction

MR imaging is currently regarded as the standard diagnostic tool for detection and grading of soft tissue tumors (STT) (De Schepper et al. (2005)). Soft tissue is a term describing all the supporting, connecting or tissues surrounding other structures and organs of the body such as fat, muscle, blood vessels, deep skin tissues, nerves and the tissues around joints (synovial tissues). Soft tissue tumors can grow almost anywhere in the human body. Soft tissue sarcomas, which are the malignant type of STT, are grouped together because they share certain microscopic characteristics, have similar symptoms, and are generally treated in similar ways. Radiologists often look for certain features in the MR image to differentiate benign from malignant STT tumors (Juan et al. (2004); Mutlu et al. (2006)). Although the signal characteristics of both benign and malignant tumors frequently overlap, some MR image features are more highly correlated to the benign or the malignant types of STT, see De Schepper et al. (2000) and De Schepper & Bloem (2007). For example, the most commonly used individual parameters for predicting malignancy are the inhomogeneity (texture) and the intensity (gray level) of the MRI signal with different pulse sequences (De Schepper et al. (2005); Hermann et al. (1992)). Inhomogeneity of the tumor region on T1-weighted MR images is a very good indicator of the malignancy of the tumor because 90% of malignant tumors are inhomogeneous and show a disorganized textured pattern of the MRI signal intensity (Weatherall (1995)). This pattern is formed as a result of the losses of tissue structure and the changes of the extracellular matrix (ECM) by cancer. The study by (Hermann et al. (1992)) reported a sensitivity of 72% and specificity of 87% in predicting malignancy based on visual comparison of texture in the tumor regions in T1-MR images. The reason for the large difference between the sensitivity and the specificity in this study is the difficulty of perceiving texture in some of the malignant tumors. The limited ability for human to perceive and discriminate between textures is well known for quite some time (Julesz (1975); Julesz et al. (1973)). Computer aided diagnostic systems can improve the radiologists performance in identifying the pathological type (i.e. benign or malignant) of a soft tissue tumor from MR images (Meinel et al. (2007)). Eventhough visually comparing the textures of benign tumor and malignant tumor sometimes show no difference, the extracted numerical values by texture analysis are quite different. Figure 1 shows subimages of a benign and a malignant tumors and the values of some of the extracted texture features. Such an example shows that

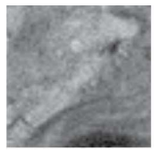
texture analysis can be used for obtaining information that is not visible to the human eye. The reader can refer to (Materka & Strzelectky (1998); Tuceryan & Jain (1998); Wagner (1999)) as excellent references to texture analysis.

In the last few years there has been growing interest in the use of machine learning classifiers for analyzing MRI data. The main aim of this chapter is to train and test several machine learning classifiers with texture analysis features extracted from MR images of soft tissue tumors. The present chapter will also serve as an introductory tutorial by providing a systematic procedure to build and evaluate a machine learning classifier that can be used for practical applications. The typical steps to build machine learning classifier consist of feature extraction, feature selection, classifier training and evaluation of the results. Several studies have tackled the problem of texture analysis for discriminating between benign and malignant tumors for specific type of malignancy, for example, the brain (Mahmoud-Ghoneim et al. (2003)) the liver (Jirák et al. (2002)) and the breast (Huang et al. (2006)). However, most papers did not follow the recommended approach for building machine learning systems (for an example see Salzberg (1997)) and left some unanswered questions. This research aims at answering some questions related to the problem of texture analysis of STT , such as the classifiers complexity, the effect of the training data set on the classifier behaviour and the appropriate size of the training data that can be used to train a machine learning classifier and obtain good generalization performance. In the following sections, we will go through the process of building and testing several machine learning classifiers as shown in Fig. 2.

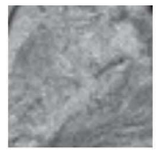
We warn the reader that the training dataset is not meant to train the classifier *per se*, as the name implies, but should be considered as a representative statistical sample from the population of STT . We assume that the training and testing data samples are randomly, identically and independently sampled from the population of STT (i.e, it is an *idd* sample). The process of training and testing the classifier is a sort of statistical parameter estimation problem where in that case the parameter of interest is the error rate of the classifier performance in unseen data. As such, all the experiments in the following sections are in fact to study how the classifier perform in other unseen data from the same STT population. To put a classifier in real practice, the classifier should be trained and tested with several datasets sampled from the same population with the same procedure as outlined in the following sections. Once the classifier evaluation is finished, all the available data can be used to train the final classifier. The classifier should be comprehensively tested based on a prospective study before using the classifier. A shorter preliminary version of this chapter was published in Juntu et al. (2010).

2. Patients data set and the MR images

A large database of multicenter, multimachine MR images was collected by the *University Hospital Antwerp (UZA)* from different radiology centers for the purpose of conducting scientific research. At the start of this study, there was a real concern that texture features could be more sensitive to image variation due to imaging with different MRI systems or changes in MRI acquisition parameters than variation due to changes in texture as a result of pathological changes. However, a recent study by Mayerhoefer et al. (2005), clearly showed that the difference in texture features extracted from MR images obtained with different machine units seems to have only small impact on the results of tissue discrimination. In the present study, a database of T1-MR images of 86 patients having benign soft tissue tumors and 49 patients having malignant tumors were used in this retrospective study. All malignant and benign masses were histologically confirmed. We discarded all MR images that showed severe



(a) ROI from a benign tumor



(b) ROI from a malignant tumor

Features	Skewness	Perc 1%	Perc 10%	Perc 50%	s(0,2) SumVariance	s(0,4) Contrast	Horzl_fraction	dgr_LngRemph45	GrSkewness	Teta2	Sigma	WavEnHH_s1
Benign ROI	0.26	106	112	119	9.62	2.94	0.49	4.38	0.90	0.71	0.18	0.10
Malignant ROI	0.21	93	45	51	5.78	2.11	0.58	3.41	0.73	0.54	0.25	0.16

Fig. 1. An example of benign and malignant tumors texture

imaging artifacts or that were corrupted by a high level of bias field inhomogeneity signal. From the tumor regions in the MR images, we cut square subimages of size 50×50 pixels for texture features computation. The physical size of that area is not fixed but it depends on the image acquisition parameters. However, the actual size of that area will not effect the values of the extracted features. To increase the size of the training dataset, we selected several tumor regions from the MR images for every patient. Hence, the total size of the dataset available for training consisted of 253 benign and 428 malignant subimages of size 50×50 pixels each. In order to preserve texture information, we avoid preprocessing the subimages. However, histogram equalization was applied to all the tumor subimages since some texture features such as the first order texture features are sensitive to graylevel variation.

3. Texture computation

Texture can be characterized and described in different ways using various sets and combinations of parameters. Most texture features computation was done using the software package MaZda 3.20 which allows the computation of texture features based on statistical, wavelet filtering, and model-based methods of analyzing texture (Castellano et al. (2004)). We also wrote other Matlab programs to calculate some texture features such as the Haralick's texture features to have a better and fine control of adjusting the parameters that effect the extracted features. To ensure the consistency of the calculated texture feature across all the tumor subimages, we wrote a MaZda macro script that reads the tumor subimages and calculates tumor texture with the same texture analysis parameters setting. The extracted texture features were saved in a text file for feature selection and classification. The following is a short description of the texture features that were computed from the tumor subimages, which are also summarized in Table 1 for easy reference:

- *First order statistics:* extract texture statistics based on a function of a single pixel. The simplest approach is to construct a histogram for the image of interest. The histogram is converted into probability function by dividing the values in the histogram by the total

Fig. 2. Block diagram of the chapter

number of pixels in the image. A set of statistical parameters from the probability density function are calculated such as the mean, the variance, the skewness, and the kurtosis.

- *Second order statistics:* the Haralick's texture features and the absolute gradient distribution are used in this study. In this method of texture analysis the correlation between two or more neighborhood pixels is taken into account. Since complex texture patterns are formed by the interaction between more than one pixel, second order statistics might provide extra texture information that can not be extracted based on first order statistics of the texture. The Haralick's texture analysis (Haralick et al. (1973)) is probably the most famous technique of second order texture analysis methods. It is based on the calculation of statistics from a function of two variables that measures the probability of occurrence of a pair of pixels that are separated by d pixels with an angle θ. We calculated 11

different Haralick's features from the co-occurrence matrix. The co-occurence matrix is calculated for every two pixels inclined by an angle θ and separated by a distance d. To take the scaling and rotation of texture into account, we calculated the Haralick's features from the co-occurrence matrices calculated with angles $\{0°, 45°, 90°, 135°\}$ and distances of $\{1, 2, 3, 4, 5\}$ pixels. The absolute gradient texture features are also included to incorporate texture features that are invariant to gray-level scaling caused by bias field inhomogeneity. Every pixel in the image was replaced by the absolute gradient which was calculated from a window of size 3×3 around the pixel by calculating the absolute of the squared summation of the difference between the two pixels above and down the center pixel and the two pixels on the right and left. Doing that for all pixels resulted in a gradient image from which several statistical parameters could be obtained: the mean, the variance, the skewness, and the kurtosis.

- *Higher order statistics:* used to capture texture information which are dependent on the interaction between several neighborhood pixels. We selected two different approaches,
  - the run-length gray-level matrix approach were a consecutive set of pixels with the same gray level value are counted and the result is stored in a 2D matrix indexed by the gray-level value and length of the gray-level run. Several statistics are calculated from the 2D matrix.
  - write a mathematical function or model that describes the texture, for example the autoregressive texture model. The basic idea of autoregressive models for texture is to express a gray level of a pixel as a function of the gray levels of its neighborhood pixels Mao & Jain (1992). The related model parameters for one image are calculated using a least squares technique and are used as texture features. This approach is similar to the Markov random fields.
- *Filtering method:* The image is split into subbands with bandpass filters such as the wavelet transform. The energy of the sub-bands are used as a texture features.

After the texture analysis step, each tumor subimage is encoded by a feature vector as shown in Fig. 3. The texture features are labeled as $\{f_1, f_2, ..., f_{290}\}$ (see Table 1).

Fig. 3. Texture analysis features

4. Feature selection

Feature selection was used to remove redundant features. This step is very important because it improves the performance of the learning models and reduces the effect of the curse of dimensionality. Feature selection also speeds the learning process and improves the model interpretability. Deciding which feature to keep, because it is relevant, and which one to discard, is largely dependent on the context. To perform an unbiased feature selection, we tested several feature selection techniques. We experimented with the following feature selection methods:

Methods	Calculated parameters
First order: $\{f_1, \ldots, f_{10}\}$	
histogram	mean, minimum, variance, skewness, kurtosis 1%, 10%, 50%, 90% and 99% percentiles.
Second Order: $\{f_{11}, \ldots, f_{250}\}$ & $\{f_{271}, \ldots, f_{277}\}$	
coocurrence matrix { angles=$\theta = 0°, 45°, 90°, 135°$ and distances=1,2,3,4,5 }	angular second moment, contrast, sum of squares, inverse difference moment, sum average, correlation, entropy, difference variance, difference entropy.
absolute gradient distribution	mean of absolute gradient, variance of absolute gradient skewness of absolute gradient, kurtosis of absolute gradient.
Higher order: $\{f_{251}, \ldots, f_{270}\}$ & $\{f_{278}, \ldots, f_{282}\}$	
runlength graylevel matrix	short run emphasis moment, long run emphasis moment, run length nonuniformity, fraction of image in run.
autoregressive texture model	$\theta_1, \theta_2, \theta_3, \theta_4, \sigma$.
Filtering technique: $\{f_{283}, \ldots, f_{290}\}$	
wavelet	energies of wavelet coefficients of subbands at successive scales.

Table 1. Texture analysis methods used in this study and the corresponding texture features

- *Unsupervised feature selection techniques:* these methods do not use the class labels and the selected features are strongly dependent on the sample distribution of the pixels graylevel values. We selected texture features subsets by forward, backward, bidirectional, and greedy stepwise search methods and two feature ranking methods, namely, the chi-squares statistics and the information gain criteria ranking methods.
- *Supervised selection techniques:* these techniques use class labels for guiding the feature selection process, thus, the selected features are the ones that improve the discrimination between benign and malignant tumors. We used the C4.5 decision tree algorithm and the support vector machines as a wrappers.

Table 2 lists all the feature selection techniques that were tested in this study and their selected subset features. It is not surprising that the 8 feature selection methods selected different features subsets because each one has a different measure for feature relevance. However, feature selection methods that belong to the same group generally selected almost similar features. The selected features subsets were used as an input to a simple Bayes classifier to evaluate the efficacy of the texture features subsets. The results of the classification are listed in Table 2. We also listed the classification accuracy ($Acc\%$), the True Positive (TP), the True Negative (TN) and the Area Under the Curve (AUC) of the ROC. The measure that is generally recommended to use is the AUC, since it is a global measure and insensitive to the data distribution. In the last row of Table 2, we included the performance of the Bayes classifier using the full textures features set for comparison. Looking at Table 2, one can notice that the classification results with the feature subsets selected by the feature ranking methods are worse than classification using the full texture feature since their AUC values are 0.72 and 0.75, respectively, while the full texture features classification has an AUC value of 0.78. The best texture features subset was the one that had the highest AUC value. The texture features subset with the highest AUC is the forward selection method which was used for training and testing the classifiers.

5. The trained classifiers

The main purpose of the training data is to infer a mathematical decision function or an algorithm for making prediction. Thereby, a given training data set is used to optimize the parameters of a machine learning classifier, which then results in a simple mathematical function or expression that can be used for making prediction. If the same classifier is trained

Method	The best selected features	ACC%	TP	TN	AUC
Forward selection	$f_4, f_6, f_7, f_8, f_{66}, f_{169}, f_{255}, f_{263}, f_{274}, f_{279}, f_{282}, f_{286}$	76.80	0.80	0.74	**0.87**
Backward selection	$f_4, f_6, f_7, f_8, f_{114}, f_{253}, f_{263}, f_{274}, f_{279}, f_{281}, f_{282}, f_{286}$	77.70	0.80	0.74	0.85
Bidirectional search	$f_4, f_6, f_7, f_8, f_{66}, f_{169}, f_{255}, f_{263}, f_{274}, f_{279}, f_{282}, f_{286}$	77.10	0.79	0.73	0.86
Greedy stepwise search	$f_4, f_6, f_7, f_8, f_{66}, f_{253}, f_{f263}, f_{274}, f_{279}, f_{282}, f_{286}$	78.00	0.83	0.69	0.83
Ranking with chi-squares statistics	$f_7, f_{16}, f_{37}, f_{45}, f_{46}, f_{52}, f_{251}, f_{253}, f_{255}, f_{263}, f_{265}, f_{268}$	67.99	0.65	0.73	0.72
Ranking with information gain	$f_7, f_{16}, f_{37}, f_{45}, f_{46}, f_{52}, f_{251}, f_{253}, f_{254}, f_{255}, f_{268}, f_{282}, f_{286}$	65.34	0.56	0.81	0.75
C4.5 decision tree wrapper	$f_6, f_{21}, f_{38}, f_{49}, f_{56}, f_{64}, f_{118}, f_{164}, f_{253}$	70.77	0.70	0.73	0.78
Best features with SVM wrapper	$f_5, f_6, f_{13}, f_{98}, f_{172}, f_{178}, f_{216}, f_{217}, f_{256}$	78.00	0.86	0.64	0.84
Full texture features set	$f_1, f_2, ..., f_{290}$	73.71	0.74	0.73	0.78

Table 2. Bayes classifier results for the best selected texture features subsets

on a different training data drawn independently and identically from the same problem domain, we expect to obtain a decision function with a similar performance. If the classifier performance stays the same independent of training with a specific training dataset, the classifier then learned how to differentiate benign from malignant tumors from the training data. However, if the classifier performance changes considerably by changing the training dataset, then that classifier can not be used for prediction. However, in principle the decision function (i.e. the classifier) can not be made completely independent from the structure of the training data and the complexity of the learning algorithm. To isolate all contributing factors that might interfere with training the classifier and to minimize the bias in the stated results, we systematically applied several machine learning evaluation strategies. First, we trained several classifiers that belong to different machine learning algorithms on the same texture features data. The selected classifiers are trained with crossvalidation procedure to make better use of the training data. The crossvalidation procedure also tries to minimize the effect of the probability distribution of a specific training dataset on the classifier performance. Second, we study the effect of changing the size of the training data set on the classifiers performance by plotting the learning curves that show the error rate of the trained classifiers as a function of the size of the training data set. Third, we used some statistical tests for comparison between the classifiers performance. We also plotted the ROC (Receiver Operating Curve) and the Cost curves to analyze the classifiers' performance. Finally, we applied the McNemar's statistical test to compare the performance of the best classifier against the radiologists' performance.

From several machine algorithm groups, we selected the following classifiers:

Linear classifier: This classifier assumes that the benign and the malignant classes have the same covariance matrix but different means. It estimates the covariance matrix from the full training data and assigns a new case to the class with the highest probability. Such classifier is able to separate benign and malignant tumors by a simple linear decision surface. The probability distribution of the full training dataset is assumed to be normally distributed.

Quadratic classifier: This classifier is more complex than the linear classifier since it estimates different matrices for the means and covariance of the benign and the malignant classes. Such classifier is able to separate the benign and the malignant tumors by a quadratic nonlinear decision surface. The probability distributions of the benign and the malignant classes are assumed to be normally distributed but not necessary with the same covariance matrices.

Nonparametric density estimation classifiers: Parzen classifier and k-NN nearest neighborhood classifier. Both classifiers estimate the empirical probability density function of the benign

and the malignant classes from the training data instead of assuming certain probability distribution function such as the linear and quadratic classifiers.

Decision trees classifier: Such classifier uses logical rules to separate the benign form the malignant tumors regardless of the probability distribution of the training data.

Back-propagation neural network: The *NN*-classifier separates the tumors by high nonlinear decision surface. The neural network uses an iterative optimization algorithm to find the weights of the neural network from the training data.

Support vector machine classifier: The SVM classifier simplifies the classification problem by transforming the input space into high dimensional space such that the classification problem become a linear one and easier to solve. The SVM classifier does not depend on the probabilistic distribution of the training dataset and has the ability to generalize quite well for classification problems of varied degrees of complexities. During the training process, a quadratic optimization algorithm is used to iteratively adjust the complexity of the decision function to adopt to the problem domain.

In the following sections, we describe several tests that were performed to study the effect of the size of the training data set on the classifier performance. Additionally, we tested the complexity of the decision function, analyzed the classifier performance and statistically compared the performance of two classifiers. Finally, we tested the classifier performance against the radiologists' performance.

6. The size of the training data and the classifiers performance

The classifier learns the classification function from the training data. The training data represents a small sample from the population of soft tissue tumors and hence the size of the training data has an impact on the trained classifier. We run the learning curve test to study the effect of the size of the training data set on the classifier performance. Using a small subset of the training data, we tuned the parameters for each classifier as follows. The back-propagation neural network has two hidden layers, an input layer of 12 nodes (i.e, number of selected texture features by the forward selection method) and an output layer with two nodes corresponding to the benign and the malignant classes. The SVM classifier is trained with an RBF kernel which is tuned with a grid search algorithm that resulted in a ($\sigma = 10000$) and a cost coefficient ($C = 1.0$). We used the PRTOOLS 4.0 matlab toolbox to run this experiment. We left the parameters of the decision trees and the Parzen classifier to their default values, which forces the PRTOOLS toolbox to tune them automatically to their best values. We trained the 7 classifiers with different sizes of the training data set. At each specific size of the training data set, we measured the error rate of all the classifiers. For each specific size of the training data, we repeated the experiment 10 times and the average error rate was calculated. Figure 4 shows the learning curves of the 7 trained classifiers. The learning curves show some interesting facts about the problem domain. First, the learning curves are smooth which is a good indicator of the classifiers stability against changes in the training data distribution . The smoothness of the learning curves is also a necessary condition for carrying some statistical tests that we used to compare the classifiers performance(Dietterich (1998)). Second, the 7 classifiers learned very well with few training samples. Most classifiers achieved an error rates between 0.251 and 0.198 after training with as few as 50 training samples. As we increase the size of the training data set, the error rate decreases very slowly after training by 50 samples. This observation indicates that a small training data set is sufficient to get good generalization performance. Increasing the size of the training set after certain

limit seems to have little impact on improving the classifiers performance any further. The third observation is related to the complexity of the classifiers. Simple classifiers such as the k-NN nearest neighborhood classifier and the SVM with an RBF kernel with large bandwidth achieved lower error rates compared to the neural network classifier. This observation is an indication that the decision surface that separates the benign from the malignant tumors based on texture features is a very simple mathematical function which we investigate further in the following section. Classification problems that procedure linear or simple decision function are less likely to overfit the training data and often generalize and predict very well in unseen data.

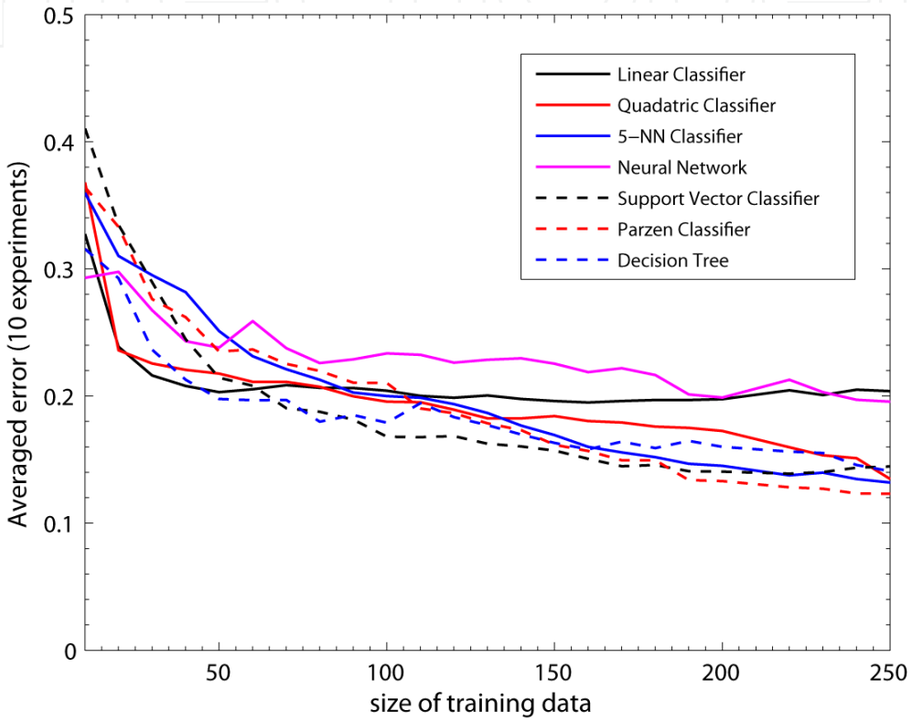


Fig. 4. The learning curves of the 7 trained classifiers

7. The complexity of the decision function

The learning curves from the last section showed that classifiers which produce simple decision functions generalize better since they have the smallest error rate on the testing samples. To check that conclusion we ran a test using an SVM classifier with a polynomial kernel that produces a polynomial decision function with a varied degree of complexity. We varied the degree of the polynomial kernel gradually from 1 to 20 and at each degree of the polynomial, we run the experiment 10 times using a crossvalidation procedure. Each point in the learning curves is the average of the error rates of ten different experiments. Figure 5 shows the error rate of the polynomial classifier versus the degree of the polynomial kernel function. The plot clearly shows that the error rate is minimum at a polynomial decision function of the 4^{th} degree. The error rates for the linear classifier (a 1^{st} degree polynomial) and the quadratic classifier (a 2^{nd} degree polynomial) are large since they under-fit the training data. A polynomial classifier higher than the 4^{th} degree also have high error rate since it

overfit the training data. This explains why in Fig. 4 that the simple linear classifier and the neural network classifier both have high error rates compared to other classifiers, because the linear classifier is too simple and the neural network classifier is too complex for the problem domain. That also explains why the SVM classifier has a good classification performance because it is very flexible and can adept to classification problems of varied complexity.

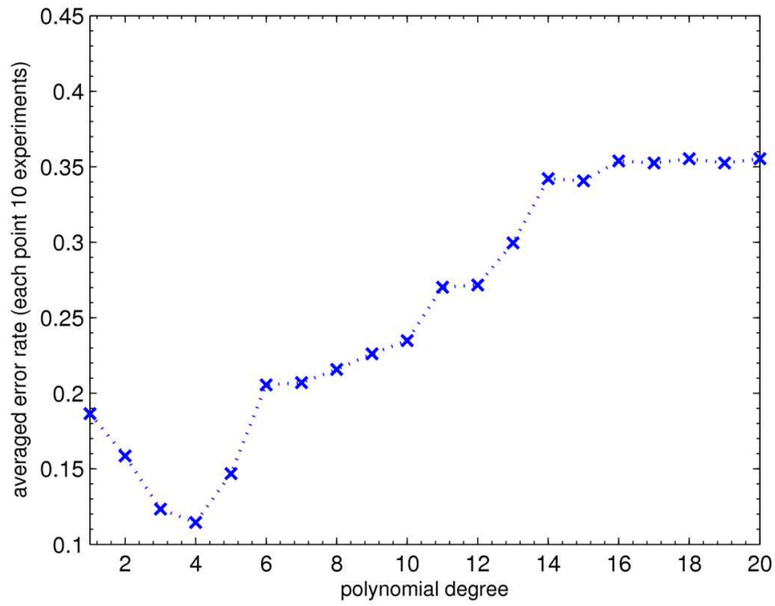


Fig. 5. The error rate versus the complexity of a polynomial classifier

8. Analyzing the classifiers performance

To gain more insight into the classifiers' performance, we trained the 7 classifiers using the full data set with a 10-folds crossvalidation procedure. In Fig. 6 and Fig. 7, we plotted the ROC curves and the Cost curves of the 7 classifiers. In the ROC curves plot, the best curves are at the top of the plot. In the ROC curves, we see that the classifiers are ranked, according to an increase in performance, as follow: the decision trees, the neural networks, the linear classifier, the quadratic classifier and the k-NN classifier. However, there is an ambiguity about the ranking of the Parzen and SVM classifiers because their ROC curves intersect. In the Cost-curve plot, the classifiers are ranked in the same order as the ROC curves. However, this time the curves of the best classifiers are at the bottom of the plot. The Cost-curves of the Parzen classifier and the SVM classifier have the same normalized expected cost value for a probability cost function (PCF) between 0.45-0.75 where both curves intersect. For a value of PCF < 0.45, the SVM classifier performance is better than the Parzen classifier while for the value of PCF > 0.75 the Parzen classifier performance is better. In other words, both classifiers perform equally well if the cost of classifying benign and malignant tumors is kept the same. However, if we would like to change the cost of classifying benign and malignant tumors, for example, we decided to give more cost for missing malignant tumors than missing benign tumors then both classifiers perform differently (see Holte & Drummond (2011)). The later observation explains why the SVM and Parzen classifier have an overlapping performance which is easy to explain from the ROC curves.

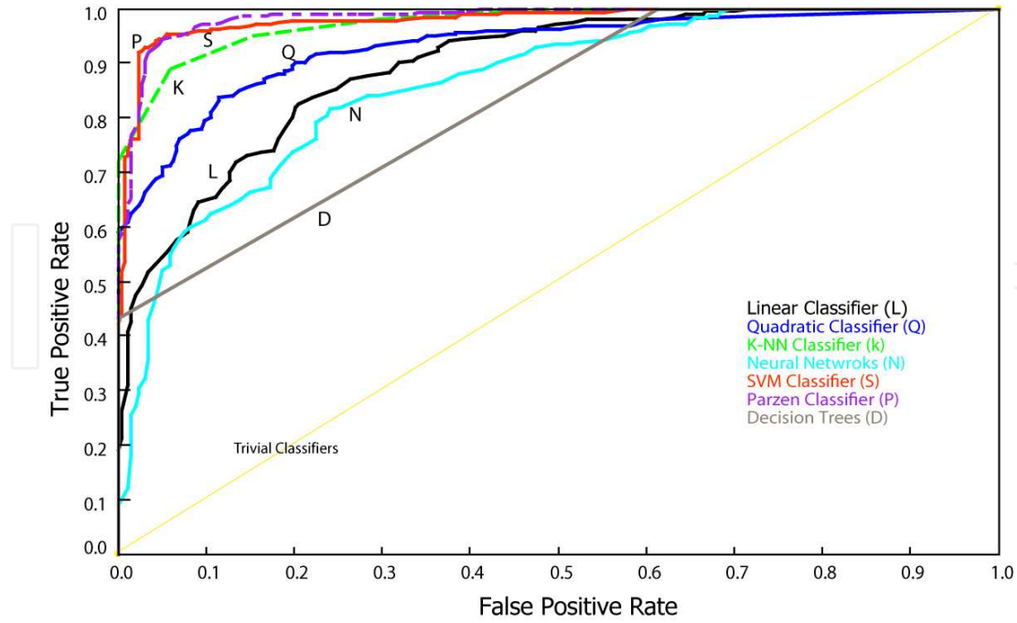


Fig. 6. ROC curves of the trained classifiers

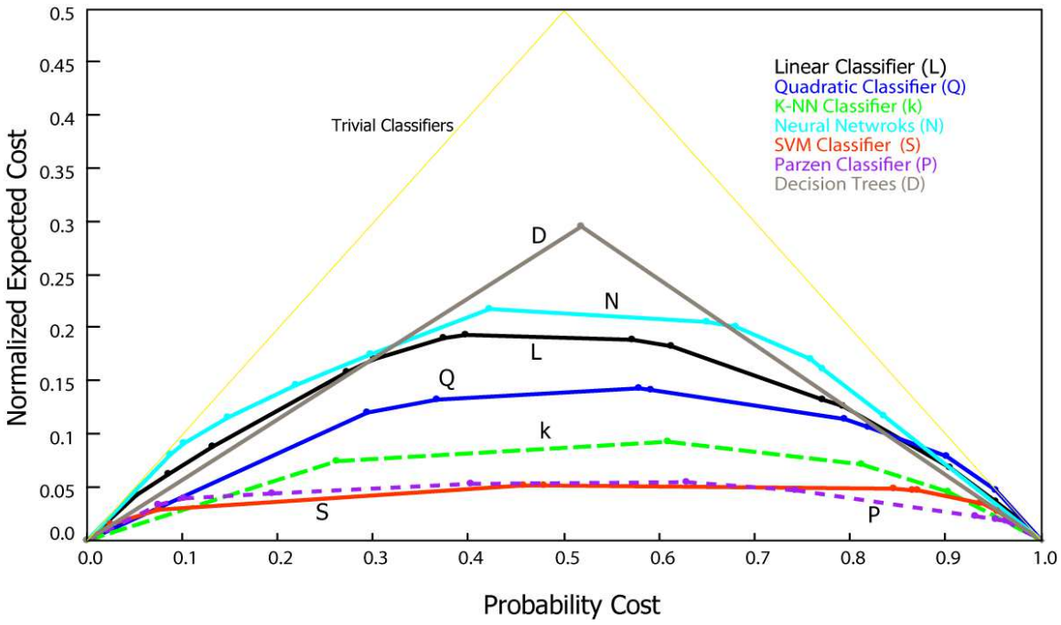


Fig. 7. Cost curves of the trained classifiers

9. Statistical comparison between two classifiers

Classifier performance is a function of several factors including the statistical distribution of the training and testing data, the internal structure of the classifier and the inherent randomness in the training process. Even if we train two different classifiers with the same dataset their classification error rates will not be necessary the same. That is because classifiers are trained with different algorithms and with different optimizations criteria and different parameter settings. The most effective way to compare classifiers is to empirically train

and test the classifiers using multiple training and testing data. This procedure is repeated several times and then some statistical tests should be applied to assess their performance. Dietterich (1998) described an 5×2 *cv* algorithm that can be used to statistically compare the performance of two machine learning classifiers in the same classification problem. The name of the test is an abbreviation for "*5 iterations 2-fold crossvalidation paired t-Test*". The same test can be used to check if one classifier outperforms another classifier on a specific classification task. Let D be a dataset which is divided into five folds $F_1, F_2, .., F_5$ and let A and B be two classifiers that their performance will be compared. Let $p_j^{\{i\}}$ stands for the difference in errors between the two classifiers in iteration j fold replication i. Then, the steps of the algorithm are as follows:

- divide the first fold F_1 into two equal-sized parts t_1 and t_2. Train both classifiers A and B using t_1 and test them using t_2 to obtain two error estimations e_A^1 and e_B^1. Calculate the difference in errors $p^{(1)} = e_A^1 - e_B^1$
- swap t_1 and t_2 such that the classifiers are trained with t_2 and tested with t_1 . Re-train both classifiers and calculate new errors and new difference in errors $p^{(2)} = e_A^2 - e_B^2$
- for this crossvalidation run, calculate the mean $\bar{p} = \frac{p^{(1)}+p^{(2)}}{2}$ and the variance $s^2 = (p^{(1)} - \bar{p})^2 + (p^{(2)} - \bar{p})^2$
- repeat the same procedure for the remaining folds $\{F_2, ..., F_5\}$

Let $p_1^{(1)}$ denotes the difference $p^{(1)}$ from the first run, and s_i^2 denote the estimated variance for run $i, i = 1, ..., 5$. Calculate the $\tilde{t}$-statistics using:

$$\tilde{t} = \frac{p_1^{(1)}}{\sqrt{(1/5)\sum_{i=1}^{5} s_i^2}} \tag{1}$$

Note that only one of the ten differences is used in the above expression. Dietterich (1998) has shown that under the null hypothesis, $\tilde{t}$ is approximately a t-distributed with 5 degrees of freedom. The test can be used to check if two constructed classifiers have a similar error rate on new example. The null hypothesis indicates that the two classifiers have the same error rate and the alternative hypothesis indicates different error rates. We reject the null hypothesis with 95 percent confidence if $\tilde{t}$ is larger than the tabulated t-statistics.
Note that, there are 10 different values that can be placed in the numerator of Eq.(1) leading to 10 possible statistics. Selecting different values in the numerator of Eq.(1) should not effect the results of the test. Practically, this is not always the case as shown in Alpaydin (1999), which proposed a modified test called *the combined* 5×2 *cv* . The modified Dietterich test combines the results of the 10 possible statistics and uses more degrees of freedom which promises to be more robust and has better statistical power than the original Dietterich test. The new test calculates:

$$\tilde{f} = \frac{\sum_{i=1}^{5} \sum_{j=1}^{2} \left(p_i^{(j)}\right)^2}{2\sum_{i=1}^{5} s_i^2} \sim F_{n,m} \tag{2}$$

and tests the estimated $\tilde{f}$ against an F-statistics with 10 and 5 degrees of freedom. Reject the null hypothesis if $\tilde{f}$ is larger than the tabulated F-statistics value (i.e., $F = 4.74$), otherwise, accept the null hypothesis.

Exp#	$e_A^{(1)}$	$e_B^{(1)}$	$p^{(1)}$	$e_A^{(2)}$	$e_B^{(2)}$	$p^{(2)}$	s^2
1	0.3853	0.1618	0.2235	0.3588	0.2029	0.1559	0.0023
2	0.3382	0.1735	0.1647	0.1353	0.1706	-0.0353	0.0200
3	0.4265	0.1794	0.2471	0.3176	0.2000	0.1176	0.0084
4	0.3824	0.1735	0.2088	0.3618	0.1529	0.2088	0.0
5	0.3912	0.1794	0.2118	0.3529	0.1647	0.1882	0.0003

Table 3. Error rates, differences and variances s^2 of the SVM classifer (A) and the Parzen (B) using 5×2-fold crossvalidation on tumors' texture.

We selected two classifiers from Fig. 7, namely, the SVM and the neural networks classifiers. We run the test to check whether both classifiers have similar performance or have different performance. The results of running the 5-iterations 2-fold crossvalidation algorithm are summarized in Table 3. Using Eq.(2), we calculated $\tilde{f} = 5.58$ which is larger than the the theoretical F-statistics value. Hence, the null hypothesis that both classifiers have similar error rates was rejected. Therefore, according to *the combined* 5×2 *cv* test, the SVM classifier had better performance than the neural network classifier with 95% statistical confidence. In conclusion, the test shows that some classifiers can have better performance than other classifier when trained with the same training dataset.

10. Machine learning versus radiologists performance

An important question is how machine learning classifiers perform compared to radiologists. In the previous section, we used the modified 5×2 *cv* Dietterich test to compare two classifiers. However, we can not use the same test to compare a classifier performance against the radiologists diagnosis since the radiologist results can not be repeated. Instead, we applied the McNemar's test (Alpaydin (2001)). To apply McNemar's test, we first have to express the results of the radiologists and the SVM classifier as depicted in Table 4: Second, we

N_{00}: Number of examples misclassified by both	N_{01} : Number of examples misclassified by the classifier but not the radiologists
N_{10}: Number of examples misclassified by radiologists but not the classifier	N_{11}: Number of examples correctly classified by both

Table 4. A table used to perform McNemar's test.

construct two hypothesis: the null hypothesis H_0 is that there is no difference between the error rates or accuracies of the radiologists and the classifier and the alternative hypothesis H_1 is that the radiologists and the classifier have different performance. If the null hypothesis is correct, then the expected counts for both off-diagonal entries in Table(4) are $\frac{1}{2}(N_{01} + N_{10})$. The discrepancy between the expected and the observed counts is measured by the following statistics:

$$\frac{(|N_{01} - N_{10}| - 1)^2}{N_{01} + N_{10}} = \tilde{\chi}^2, \tag{3}$$

which is, approximately, distributed as χ^2 with 1 degree of freedom. First, we run several experiments to find an optimal classifier. The best classifier so far was the SVM classifier.
The results of the SVM classifier against the radiologists are summarized in Table 5. Using Eq.3, we obtained $\tilde{\chi}^2 = 12.85$ which is larger than the tabulated $\chi^2 = 3.48$. Hence, we rejected

	laboratory results		
SVM results		*malignant*	*benign*
	malignant	405	23
	benigin	25	228

	laboratory results		
physician results		*malignant*	*benign*
	malignant	134	45
	benigin	32	552

Fig. 8. The SVM and the radiologists confusion matrices

$N_{00} = 39$	$N_{01} = 16$	$N_{00} + N_{01} = 55$
$N_{10} = 45$	$N_{11} = 581$	$N_{10} + N_{11} = 625$
$N_{00} + N_{10} = 84$	$N_{01} + N_{11} = 597$	$N = 681$

Table 5. A table constructed for the McNemar's test

the null hypothesis that both the radiologists and the SVM classifier have similar error rates. Therefore, the SVM seems to perform slightly better than the radiologist. This last conclusion should, however, be taken with a grain of salt because it is based on statistical analysis of the SVM classifier with a limited training data set that does not represent the full distribution of the soft tissue tumors.

The McNemar's test does not tell us about the strength between the agreement or the disagreement between the radiologists and the SVM classifier to validate the previous test so we evaluated the kappa statistics ($\kappa = 0.5$) which is larger than 0 which shows that the results of the McNemar's test is correct. Finally, the confusion matrix of the SVM classifier is shown in Fig. 8. The radiologist performance is also shown in Fig. 8.

11. Conclusions

We demonstrated that texture analysis of soft tissue tumors and machine learning algorithms can be used as a tool for objective evaluation of MR images and the results correlate well with the laboratory results. We ran several tests and come up with some interesting observation related to the problem of texture analysis of soft issue tumors. First, texture features combined with machine learning algorithms seems to perform as well as radiologists since computer can extract more information related to signal homogeneity in T1-MRI than what human can do based only on visual perception. Second, we do not need a large training data set to train a machine learning classifier and obtain a good classification performance since texture features correlate very well with the pathology of the tumor. Moreover, simple classifiers such as a Parzen classifier or an SVM classifier can effectively separate benign from malignant tumors.

12. Acknowledgments

Thanks to the *University Hospital Antwerp (UZA), Dept. of Radiology* for providing the MR images. The authors would like to thank Prof. Robert Holte for providing the Cost Curve software.

13. References

Alpaydin, E. (1999). Combined 5 x 2 cv F test for comparing supervised classification learning algorithms, *Neural Computation* 11(8): 1885–1892.

Alpaydin, E. (2001). Assessing and comparing classification algorithms.

Castellano, G., Bonilha, L., Li, L. & Cendes, F. (2004). Texture analysis of medical images, *Clinical Radiology* 59: 1061–1069.

De Schepper, A. M. & Bloem, J. L. (2007). Soft tissue tumors : grading, staging, and tissue-specific diagnosis, *Topics in Magnetic Resonance Imaging* 18(6): 431–444.

De Schepper, A. M., De Beuckeleer, L., Vandevenne, J. & Somville, J. (2000). Magnetic resonance imaging of soft tissue tumors, *European Radiology* 10(2): 213–223.

De Schepper, A., Vanhoenacker, F., Parizel, P. & Gielen, J. (eds) (2005). *Imaging of Soft Tissue Tumors*, 3rd edn, Springer.

Dietterich (1998). Approximate statistical tests for comparing supervised classification learning algorithms., *Neural Computation* 10(7): 1895–1923.

Haralick, R.M., Shanmugan, K. & Dinstein, I. (1973). Textural features for image classification, *IEEE Transactions on Systems, Man and Cybernetics* 3(6): 610–621.

Hermann, G., Abdelwahab, I., Miller, T., Kelin, M. & Lewis, M. (1992). Tumor and tumor-like conditions of the soft tissue: Magnetic resonance imaging features differentiating benign from malignant masses, *Br J Radiol* 65: 14–20.

Holte, R. C. & Drummond, C. (2011). Cost-sensitive classifier evaluation using cost curves, *Proceedings of The 24th Florida Artificial Intelligence Research Society Conference (FLAIRS-24)*.

Huang, Y., Wang, K. & Chen, D. (2006). Diagnosis of breast tumors with ultrasonic texture analysis using support vector machines, *Neural Computing & Applications* 15(2): 164–169.

Jirák, D., Dezortová, M., Taimr, P. & Hájek, M. (2002). Texture analysis of human liver, *Journal of Magnetic Resonance Imaging* 15(1): 68–74.

Juan, M., García-Gómez, Vidal, C., Luis Martí-Bonmatï£¡, Joaquín, G. & et al. (2004). Benign/malignant classifier of soft tissue tumors using MR imaging, *Magnetic Resonance Materials in Physics, Biology and Medicine* 16: 194–201.

Julesz, B. (1975). Experiments in visual perception of texture, *Sci Am* 232: 34–43.

Julesz, B., Gilbert, E., Shepp, L. & Frisch, H. (1973). Inability of humans to discriminate between visual textures that agree in second-order statistics, *Perception* 2: 391–405.

Juntu, J., Sijbers, J., De Backer, S., Rajan, J. & Van Dyck, D. (2010). Machine learning study of several classifiers trained with texture analysis features to differentiate benign from malignant soft-tissue tumors in T1-MRI images, *J. Magn. Reson. Imaging* 31(3): 680–689.

Mahmoud-Ghoneim, D., Toussaint, G. & Jean-Marc, C. (2003). Three dimensional texture analysis in MRI: a preliminary evaluation in gliomas, *Magnetic Resonance Imaging* 21(9): 983–987.

Mao, J. & Jain, A. K. (1992). Texture classification and segmentation using multiresolution simultaneous autoregressive models, *Pattern Recognition* 25(2): 173 – 188.

Materka, A. & Strzelectky, M. (1998). Texture analysis methods- a review, *Technical University of Lodz 1998, COST B11-techincal report* 11: 873–887.

Mayerhoefer, M. E., Breitenseher, M. J., Kramer, J., Aigner, N., Hofmann, S. & Materka, A. (2005). Texture analysis for tissue discrimination on T1-weighted MR images of knee joint in a multicenter study: Transferability of texture features and comaprison of feature selection methods and classifiers, *J Mag Reson Imaging* 22: 674–680.

Meinel, L. A., Stolpen, A. H., Berbaum, K. S., Fajardo, L. L. & Reinhardt, J. M. (2007). Breast MRI lesion classification: Improved performance of human readers with a backpropagation neural network computer-aided diagnosis (CAD) system, *Journal of Magnetic Resonance Imaging* 25(1): 89 –95.

Mutlu, H., Silit, E., Pekkafali, Z., Basekim, C., Ozturk, E., Sildiroglu, O., Kizilkaya, E. & Karsli, A. (2006). Soft-tissue masses: Use of a scoring system in differentiation of benign and malignant lesions, *Clinical Imaging* 30(1): 37–42.

Salzberg, S. L. (1997). On comparing classifiers: Pitfalls to avoid and a recommended approach, *Data Mining and Knowledge Discovery* 1: 317–327.

Tuceryan, M. & Jain, A. K. (1998). Texture analysis, *in* C. H. Chen and L. F. Pau and P. S. P. Wang (ed.), *The Handbook of Pattern Recognition and Computer Vision (2nd Edition)*, World Scientific Publishing Co., pp. 207–248.

Wagner, T. (1999). Texture analysis, *in* B. Jane, H. Haubecker & P. Geibler (eds), *Handbook of Computer Vision and Applications, Vol.2, Signal Processing and Pattern Recognition*, Academic Press, chapter 12, pp. 275–308.

Weatherall, P. (1995). Benign and malignant masses, MR imaging differentiation, *Mag Reson Clin N Am* 3: 669–694.

Soft Tissue Tumors
Edited by Prof. Fethi Derbel

ISBN 978-953-307-862-5
Hard cover, 270 pages
Publisher InTech
Published online 16, November, 2011
Published in print edition November, 2011

Soft tissue tumors include a heterogeneous group of diagnostic entities, most of them benign in nature and behavior. Malignant entities, soft tissue sarcomas, are rare tumors that account for1% of all malignancies. These are predominantly tumors of adults, but 15% arise in children and adolescents. The wide biological diversity of soft tissue tumors, combined with their high incidence and potential morbidity and mortality represent challenges to contemporary researches, both at the level of basic and clinical science. Determining whether a soft tissue mass is benign or malignant is vital for appropriate management. This book is the result of collaboration between several authors, experts in their fields; they succeeded in translating the complexity of soft tissue tumors and the diversity in the diagnosis and management of these tumors.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen