Hopfield Networks in Relevance and Redundancy Feature Selection Applied to Classification of Biomedical High-Resolution Micro-CT Images*

Benjamin Auffarth**, Maite López, and Jesús Cerquides

Volume Visualization and Artificial Intelligence research group,
Departament de Matemàtica Aplicada i Anàlisi (MAIA), Universitat de Barcelona,
C/Gran Via, 585, 08007 Barcelona, Spain
{benjamin,maite,jcerquide}@maia.ub.es

Abstract. We study filter–based feature selection methods for classification of biomedical images. For feature selection, we use two filters — a relevance filter which measures usefulness of individual features for target prediction, and a redundancy filter, which measures similarity between features. As selection method that combines relevance and redundancy we try out a Hopfield network. We experimentally compare selection methods, running unitary redundancy and relevance filters, against a greedy algorithm with redundancy thresholds [9], the min-redundancy max-relevance integration [8,23,36], and our Hopfield network selection. We conclude that on the whole, Hopfield selection was one of the most successful methods, outperforming min-redundancy max-relevance when more features are selected.

Key words: feature selection, image features, pattern classification

1 Introduction

Computerized Tomography (CT) is a technique of producing a 3-dimensional image from a large series of X-ray images taken around a single axis of rotation. The 3-D image is cut into sections (Greek tomos = cutting), so the data is an array of 2-D images which together constitute a volume. The information of density of cell tissue is given in gray intensity levels.

Volume visualization of such CT–slices can help experts in biomedical analysis, such as e. g. inspection of cell tissue and of anatomical structures, or in gaining a better understanding of cell growth (more general: Computer-Assisted Diagnosis). For this purpose, first, a transfer function maps from possible voxel

* This research was supported by the Spanish MEC Project "3D Reconstruction, classification and visualization of temporal sequences of bioimplant Micro-CT images" (MAT-2005-07244-C03-03).

** Corresponding author.

values to RGBA space, defined by colors and opacity (red, green, blue, alpha). Using volume visualization techniques, 2–dimensional projections on different planes can then be displayed.

The opacity of voxels depends on cell tissue that the voxels represent. Therefore, distinguishing between different tissues can enhance the volume visualization. Hence, we break the transfer function between intensity values and optical properties into two parts: i) classification function, and ii) transfer function from tissue to optical properties.

Classification, in the context of this work, refers to the process of distinguishing between different kinds of data, here biomaterial and non-biomaterial, and the result of this process. Our data consist of slices in a 3-D volume taken from CT of bones, in which was artificially introduced a biomaterial for tracing purposes[1]. The introduced biomaterial is the target class and relatively small as compared to the non-target class. The data set originally was of dimensions $423 \times 486 \times 562$, but because 403 slices did not contain any biomaterial, in the current study, dimensionality was reduced to $423 \times 486 \times 159$.

Earlier, working on these data, we introduced a pipeline process for classification and subsequent volume rendering [22]. In classification, instead of a unique single-run classifier, as in most approaches, we applied a learning pipeline consisting of three steps. After initial Gentle Boost [13] classification based on image properties, a conditional random field [20] on an image of reduced scale works on spatial characteristics of uncertain pixels (output of Gentle Boost), and finally we refined the result in a last step. This article aims at extending the framework with a feature selection step.

For organic tissues, as in our case, distributions of intensities overlap considerably [33]. In order to produce a reliable classification model, we extracted characteristics (features) – a process called "feature extraction" (see 4.1) – from images by integrating image intensity within a window around each pixel. With high number of features, classifiers become slow and tend to produce unstable models with low generalization performance, so our problem was then selecting from a number of features, compiling a set of features that would give good performance.

Each of the extracted features can have merits on its own and merits when used in combination with a selection of other features and we did not know beforehand, which of the features to use. "Feature selection" refers to methods dedicated to finding a set of features that together can be more successful than others. Feature selection within the context of pattern classification will be the focus of most of this work.

The outline of this article is as follows: First, in section 2, we explain the concepts of relevance and redundancy filters, briefly survey related research on feature selection, and line out two heuristics for combination of the information from the two filters. In section 3 we present a novel method that uses a Hopfield network with the idea of taking into account more complex redundancy relations

[1] Samples from the data set are available on one of the authors' homepage: `http://www.maia.ub.es/~maite/out-slice-250-299.arff`.

than other methods. In section 4 we describe our experimental benchmarks of several feature selection methods, and interpret conclusions based on the results regarding the best method for feature selection, quality of selection, and finally the best features. Lastly, we draw conclusions in section 5 and outline future work in section 6.

2 Relevance and Redundancy Feature Selection

Feature selection in biomedical research is still often done manually by experts, however due to great quantities of data it is becoming increasingly automatized. A comparison of methods over articles by different authors is difficult, because of incompatible performance indicators, often unknown significance, and different data sets methods are applied to. Saeys et al. [28] review research in feature selection in application to biological data.

Sets of features can be evaluated by either filters, which measure statistical properties or information content, or a performance score of a classifier ("wrapper approach"). There exist many heuristics for choosing subsets of features. Two standard iterative search strategies are forward selection and backward selection. Forward selection, starts from the empty set and adds at each step a feature, which gives the most performance improvement. Backward selection starts from all features, eliminating at each iteration one or several features. Forward-backward algorithms make an initial guess of a useful feature set and then refine the guess by eliminating variables and adding new ones.

In the context of this work, we define the feature selection task as follows: given a selection criterion (error function) $\varepsilon(\cdot)$ and an initial feature set X with m features we want to find a subset $X^* \subseteq X$ such that $|X^*| = s$ (s for number of **s**elected features) and $X^* = \arg\min_{\bar{X} \subseteq X, |\bar{X}|=s} \varepsilon(\bar{X})$.

Many approaches to feature selection in bioinformatics are either based on ranks ("univariate filter paradigm") and thereby do not take into account relationships between features, or are wrapper approaches which require high computational costs. We chose a filter-based feature selection approach for being fast and giving good results, which other computation-heavy methods are not guaranteed to achieve (cf. [16]). Filter-based have the additional advantage of providing a clearer picture of why a certain feature subset is chosen through the use of scoring methods in which inherent characteristics of the selected set of variables is optimized. This is contrary to wrapper-based approaches which treat selection as a "black-box" optimizing the prediction ability according to a chosen classifier.

Multivariate filter-based feature selection with the idea to have a set of features of maximal relevance to the target, which are least redundant has enjoyed increased popularity [28]. It has been shown that the best subset of features may not be the set of the best individual features (e. g. [6]). The idea behind combining redundancy and relevance information is simple: you should take the features that together have the highest value for prediction and not the ones which alone have highest prediction value.

Relevance criteria determine how well a variable discriminates between the classes. They are a measure between a feature and the class.

Redundancy criteria should capture similarities of mappings from attributes to classes, i.e. given a predictor function $f \in F : \mathbb{R} \rightarrow C$ then our intuition is that for two non-redundant features X_k and X_l, $f(X_k)$ should be different to $f(X_l)$ (and hopefully provide complementary information). Formally the redundancy between features X_1 and X_2 given class targets $Y \in C^n = \{c_1, \ldots, |C|\}^n$ can be written as

$$\mathrm{Red}(X_1, X_2, Y) = \frac{1}{|C|} \sum_{i=1}^{|C|} \Delta \left(X_1 | Y = c_i, X_2 | Y = c_i \right), \tag{1}$$

where $X_1|Y = c_i$ denotes the distribution of feature 1, given class i (i. e. $\{X_1^l | \forall l, Y^l = c_i\}$, and Δ one of the distributional similarity measures that we applied. Given a relevance measure $Rel()$, features X_1 and X_2, and targets $Y \in C^n$, we can define $\mathrm{Red}(X_1, X_2, Y) = \frac{1}{|C|} \sum_{\forall i \in [1,|C|]} \mathrm{Rel}(X_1|c_i, X_2|c_i)$.

Ding, Peng, et al. [36,23,8] select features in a framework they call "min-redundancy max-relevance" (here short: mRmR) that integrates relevance and redundancy information of each variable into a single scoring mechanism to automatically annotate the fruitfly's embryonic tissue.

Knijnenburg [19] presented a cluster-based approach where variables are first hierarchically (complete linkage) clustered and then from each cluster the most relevant feature is selected. Relevance and redundancy were measured by Pearson Correlation Coefficients. He concluded that cluster-based selection could not improve upon greedy ranking-based selection, but a second approach that integrated relevance and redundancy into a single score (in a way similar to mRmR [8]) did so.

Duch et al. [9] presented an algorithm that proceeds at each step including variables starting from highest relevance and excluding variables that are redundant. Their heuristics is simple, straightforward, and seemed to work.

In the next subsection we will describe the mRmR approach and thereafter describe Duch and Biesiada's [9] greedy heuristics with threshold.

2.1 Minimum Redundancy Maximum Relevance

Ding, Peng et al. [8,23,36] presented minimum redundancy maximum relevance feature selection. The method boils down to a forward scheme[2] maximizing one of two formulas for combination of redundancy and relevance information (mutual information in both cases) by subtraction and division, respectively. These formulas are:

- $\arg\max_i \mathrm{rel}(i, c) - \frac{\sum_j \mathrm{red}(i,j)}{m}$, with i and j being two features, c the matching target, and m the numbers of competing features at each step

[2] Peng et al. [23] also discuss and test a backward scheme but it is given less importance than the forward scheme.

– $\arg\max_i \frac{\mathrm{rel}(i,c)}{\frac{\sum_j \mathrm{red}(i,j)}{m}}$

Peng et al. [23] use mutual information as measure for relevance and redundancy and they refer to the first formula as mutual information difference (MID), to the second as mutual information quotient (MIQ). We will refer to (dropping the reference to mutual information) *mRmRD* and *mRmRQ*,respectively. We implemented the mRmR forward search and integrated it with our redundancy and relevance methods. The algorithm works as lined out in algorithm 1. *best*() is the selection formula, i.e. either quotient or difference. Features are $X_i, i \in [1, \ldots, m]$.

Input: rel $\in \mathbb{R}^m$: relevance scores ;
red $\in \mathbb{R}^{m^2}$ redundancy scores ;
s: number of features that need to be selected (assumed $k \geq 1$)
Initialize set $D = \{X_1, \ldots, X_m\}$;
for $i \leftarrow 1; i \leq s;\ \ i++$ **do**
  $S_i \leftarrow best(D)$;
  $D \leftarrow D \setminus S(i)$;
end
Output: S: s features ordered by mRmR

Algorithm 1: mRmR feature selection

2.2 Greedy Algorithm with Redundancy Threshold

Duch et al. [9] in their forward scheme used the Kolmogorov–Smirnov test for measuring redundancy and set the cut–off threshold to a p-value> 0.95.

Algorithm 2 lines out the workings of the implemented algorithm. The Greedy selection scheme could be extended to select a specific number of variables, however this would mean giving up on the strict thresholding and to introduce arbitrariness into feature selection.

Input: m: number of features, ;
rel $\in \mathbb{R}^m$: relevance scores, ;
red $\in \mathbb{R}^{m^2}$: redundancy scores, ϵ: threshold
Initialize sets: set $S \leftarrow \emptyset$, and $C \leftarrow \{c_1, \ldots, c_m\} = \{1, \ldots, m\}$;
while $|C| > 0$ **do**
  $S \leftarrow S \cup \arg_i \max \mathrm{rel}_{c_i}$;
  $C \leftarrow C \setminus \{i\}$;
  $C \leftarrow C \setminus \left\{j | \exists s_i \in S, \mathrm{red}_{s_i,c_j} \geq \epsilon\right\}$;
end
Output: Selected features are in S

Algorithm 2: Greedy feature selection algorithm with Thresholding

3 Hopfield Network for Relevance and Redundancy Feature Selection

The spaces of feature combinations and the corresponding space of their energy or error functions has numerous local optima, which iterative algorithms intrinsically have difficulties dealing with. This inspired us to think of other graph based methods as a manner of partitioning and selecting the best features.

We can form complete graphs from the redundancy matrices, if we think of them as a proximity matrices $D = \mathbb{R}^{m^2}$, where m is the number of variables. We had the idea of the features as nodes, and to add as additional dimension relevance, redundancy constituting inhibitory connections between the features.

The lateral connections represented by the redundancies could create an attractor network that forms basins of attractions, where redundancies are lowest and relevancies are highest. In this manner, the choice of features could come up as an emergent pattern within the configuration space of the network arising from the connections and activations.

A recurrent attractor network with well-studied convergences is a Hopfield network[17,18,31]. A Hopfield network has the advantage of being able of generating arbitrary shapes and providing insight into the number of variables without prior knowledge.

In the simplest form of the Hopfield network, we formalize connections (having an appropriate normalization) as symmetric, real-valued connections w_{ij}, units $S_i \in [0, 1]$ and corresponding bias units I_i. The input to each unit S is

$$n_i \leftarrow \sum_j w_{ij} S_j + I_i \ , \tag{2}$$

where I_i is a bias term of unit i.

In the classical (bipolar) formalization, nodes can be asynchronously (serially) updated at each time step t:

$$S_i(t+1) \leftarrow \begin{cases} 1 & \text{if } n_i(t) > 0 \\ 0 & \text{if } n_i(t) < 0 \\ S_i(t) & \text{otherwise} \end{cases} \tag{3}$$

The energy function of such a network is

$$E = -\frac{1}{2} \sum_{ij} S_i S_j - \sum_i I_i S_i + \sum_i \int_0^{S_i} g_\lambda^{-1}(S) dS, \tag{4}$$

where g_λ is a sigmoidal funcion, often the sigmoid function $\equiv \frac{1}{1+e^{-\lambda x}}$, and λ is its gain, which guarantees the convergence to a continuous local minimum of the energy function over time and the synchronization of clusters of units.

We tried many different parameters, normalizations of activations and connections. Parameters included annealing with different rates, including using e.g. Rprop[25]. Finally, we chose a simple implementation for continuous (graded)

activation (and responses) and asynchronous updating in discrete time-steps[3] with the hyperbolic tangent $\tanh x = \frac{\sinh x}{\cosh x}$ as our sigmoidal function. Weights and activations were normalized in the range $[-1, 0]$ and $[0, 1]$ respectively, with the diagonal of the weight matrix set to 0. We set the noise parameter $u_0 = 0.015$ (cf. [18]) and we fixed the learning rate $\lambda = 0.1$.

The update of the activation S of a neuron i at time step t is then

$$S_i(t+1) \leftarrow (1-\lambda)\, S_i(t) + \lambda \left(1 + \tanh\left(\frac{u_i}{u_0}\right)/2\right), \text{ where} \tag{5}$$

$$u_i = \sum_j w_{ij} \times S_j(t). \tag{6}$$

For application to feature selection, at the end we choose the most highly activated units, thresholding the unit activations:

$$S_i \leftarrow \begin{cases} 1 \text{ if } \sum_j w_{ij} a_j > \theta, \\ 0 \quad \text{otherwise.} \end{cases} \tag{7}$$

In the coming section (4) we will submit the presented feature selection schemes to a testing procedure using information of relevance and redundancies (all combinations). Afterwards we will present results and compare methods.

4 Experiments and Results

We conducted experiments in order to find out which selection schemes and which relevance and redundancy measures perform best. For the experiments we need to extract a set of features from the images and compute measures of redundancy and relevance. After describing methods corresponding to these, we come to our experimental design, and methods of statistical validation. After this we look at results.

4.1 Feature Extraction

A standard method for compact image encoding is a method called Laplacian pyramids [3]. For their computation, an image is iteratively smoothed by computing averages in constant windows as low-pass filters. The bottom level of this representation (g_0) is the original image. The Laplacian pyramid is then the sequence of difference maps between two levels at the pyramid $L_n = g_n - g_{n+1}$, for $0 \leq n < N$, with N denoting the number of levels in the smoothed pyramid.

Gabor filters (e.g. [12] have received considerable attention because the characteristics of simple cells in the primary visual cortex of some mammals can be approximated by these filters. They are used a lot in pattern recognition and

[3] Our attempts at converging at a good implementation were streamlined considerably by Hervé Abdi [1].

texture segmentation. Gabor filters, in contrast to the other features presented here, incorporate orientation information.

In eye-tracking studies, Reinagel and Zador [24] gave evidence for increased luminance contrast in fixated regions as compared to control points (fixated points on different images). They defined luminance contrast (LC) as the variance of luminance within a patch (a rectangular patch for practical purposes) divided by the mean intensity of the image. Given a patch P of pixel intensity from image I and around a pixel (x, y): $\mathrm{LC}_P = \frac{\delta_P}{\mu_P}$. Another texture function from neuropsychological research is texture contrast [10]. The texture contrast (TC) of a patch is the standard deviation of the luminance contrast values in the patch standardized by the luminance contrast mean of the image. More formally, given a patch $\bar{P}$ from LC_I: $\mathrm{TC}_{\bar{P}} = \frac{\delta_{\bar{P}}}{\mu_{\bar{P}}} = LC_{\bar{P}}$.

We extracted 10 features from the Laplacian Pyramid, 100 Gabor features (10 orientations at 10 scales), 9 features from luminance contrast, 7 features from texture contrast, and intensity. We added 50 probes which have a function in performance assessment; a good feature selection method should eliminate most of these probes. 25 probes were standard normal distributed, 24 uniformly distributed in the interval $(0, 1)$. The last probe was a variable of zeros.

4.2 Relevance and Redundancy Criteria

Due to the uncertainty of the true distribution underlying data, we prefer non-parametric and model-free metrics. Non-parametric tests have less power (i. e. the probability that they reject the null hypothesis is smaller) but are more robust to outliers than parametric tests.

The four relevance criteria that we used in experiments are: Symmetric Uncertainty (SU), Spearman Rank Correlation Coefficient (CC), Value Difference Metric (VDM), and Fit Criterion (FC). In [2], we showed how a measure of probability difference, presented before as the "value difference metric" [29], can be adapted as a relevance criterion. We also use a measure, which we call "fit criterion", presented in [2].

As for redundancy criteria, we used seven measures: Kolmogorov-Smirnov test on class-conditional distributions (KSC), Kolmogorov-Smirnov test ignoring classes (KSD), Value Difference Metric adapted to redundancy (RVDM) [2], Redundancy Fit Criterion (RFC) [2], Spearman Rank Correlation Coefficients (CC), Jensen-Shannon Divergence (JS), and the Sign-test (ST).

As for discretization, we use histograms. Conforming to Cromwell's rule of avoiding probabilities of 1 and 0 (except for logical true and false), we apply the Laplacian rule of succession by calculating the probabilities of bin i with frequency count n_i as $\tilde{p}(i) = \frac{n_i+1}{k+\sum_{j=1}^{k} n_j}$. In order to avoid any problems with optimization of a bandwidth or bin number and because of impracticality of mixture modeling, we chose a rigid bin number of 100.

4.3 Experimental Design

We benchmarked first each relevance and redundancy criterion on its own ("unitary filters"), then all 28 combinations of mentioned relevance and redundancy measures with the selection methods mRmRQ, mRmRD, Greedy, and Hopfield. For the threshold in Greedy we used all thresholds possible in combination with the redundancy measure. As for unitary filters, for relevance measures, the s highest relevant features were used and for redundancy measures, at each step the most redundant feature with all the remaining features is removed until the desired numbers of features s are left. We also introduced a baseline of random selection.

We selected feature sets of sizes $[4, 8, 12, 16, 20, 30, 45, 60, 80, 100]$. We emphasized feature sets of sizes ≤ 30 because that was were they were the greatest differences between the different methods. The reported experiments and comparisons are based on the set of 177 features and their respective relevance measures and mutual redundancies. We used three classifiers for benchmarking: Naïve Bayes, GentleBoost, and a linear Support Vector Machine.

As for Naïve Bayes we relied on our own implementation for multi-valued attributes using 100 bins for discretization. Given m features $X_1, \ldots, X_m$ and corresponding targets $Y \in C^n$, classifying a pattern $x = \{x_1, \ldots, x_m\}$ by Naïve Bayes means $\text{argmax}_{c \in C}\ p(c) \prod_{i=1}^{m} p(x_i | c = Y)$. As for GentleBoost we used Antonio Torralba's matlab toolbox [27] and fixed the iterations to 50, which seemed to be a good trade–off between speed and performance. As for SVM [5] we used libsvm 2.84 [4], accessed from within MATLAB using an interface by Michael Vogt [30] from Technical University Darmstadt. We made the cost function to compensate for unequal class priors, by setting the weight of the less frequent class to $\max\left(\frac{\sharp(Y=c_2)}{\sharp(Y=c_1)}, \frac{\sharp(Y=c_1)}{\sharp(Y=c_2)}\right)$. Further, we set the SVM complexity parameter C to 1 which seemed to be a good choice and in the right order of magnitude. We tried out several normalization methods. Comparisons showed that classification performance being approximately equal, there was a notable loss of speed with classification after normalization according to Graf et al. [15], with z-normalization performing faster than normalization between $[0, 1]$ or $k[-1, 1]$. Consequently, features were z-normalized.

The whole set of experimental conditions can be obtained by combining selection schemes with corresponding relevance and redundancy measures, classifiers, and numbers of features. Greedy, Hopfield, mRmRQ, and mRmRD, were tried out with the 28 redundancy and relevance combinations, all classifier, at each number of features. Unitary filters with their redundancy or relevance measures, were combined with a classifier and a number of features. Random selection ran with each classifier at each number of features. In total we had 3700 experimental conditions.

In order to have many validations at acceptable speed – we made 10 random samplings of size $n/10$ and for each sampling we did 5-fold cross-validation. As for random feature selection, we did 10 random samplings of the data of size $n/10$ and tested 10 random selections of features in 5-fold cross-validation.

4.4 Statistical Evaluation

As performance measure, we used the area under the curve (AUC) throughout the analysis and — following the recommendations of Janez Demŝar [7], who surveyed the state of the art of comparing classifiers — we did not base our statistics on performances of single folds but took averages (medians[4]) over folds.

4.5 Results

Statistics were extracted from performance vectors and are given over all three classifiers (Naïve Bayes, GentleBoost, and SVM). For feature selection, what is the "best" method depends on how many features there are, which is the application, and what computational resources are available.

We will focus on three questions:

1. Which is the best feature selection scheme?
   (a) In particular, are there differences with respect to numbers of features?
2. Are the best methods the ones with fewest probes?
3. What is the best feature set?

Question 1 includes feature selection schemes, measures of redundancy and relevance (short: RR measures), and combinations of relevance and redundancy. Apart from an overall winner according to our experimental setup, we will look at which selection scheme gives the best results. We will have to look whether there are differences between the methods with respect to RR measures.

As for question 1.a, we will analyze, if relative performance of the different methods is the same when the number of attributes selected increases. We will have to decide which is a good feature size for our classification task and with respect to this decision decide on the best selection scheme.

As for question 2, we look at probe frequency and see whether a selection scheme with good performance is automatically one with few probes.

Question 3 deals with the final result of our feature selection: which are the best features for our classification task?

In table 1, the first column gives the name of the method (the selection scheme followed by redundancy and relevance measures), ordering (column two) follows the mean rank of performance (third column), win–loss statistics (W/L) from statistical tests based on ranks at all feature numbers respectively show

[4] According to the central limit theorem, any sum (such as e. g. a performance benchmark), if of finite variance, of many independent identically distributed random features will converge to a Gaussian distribution. This is however not necessarily to expect for only 5 values, i. e. from 5-folds of cross-validations. After finding partly huge differences between means (which are usually taken) and medians over cross-validations, in pre-trial runs, we decided to take the more robust median (which in case of normal distributions is equal to the arithmetic mean anyway). As for the error-bar, we plot the interquartile range (short: IQR), which is the difference between values at the first (25%) and the third quartile (75%).

	index	mean rank	F/N W/L	SR W/L
mRmRD	1	2.10	3/0	4/0
Hopfield	2	2.85	2/0	2/0
Red	3	3.50	1/0	2/0
mRmRQ	4	3.70	1/0	1/1
Rel	5	3.95	1/1	1/1
Greedy	6	5.00	1/2	1/3
rand	7	6.90	0/6	0/6

Table 1. Ranking of all Selection Schemes

in column four and five: Friedman test with Nemenyi post-hoc test (F/N) and Wilcoxon Signed Rank Test (SR, also called the *Mann-Whitney U test*).[5]

According to table 1, mRmRD is overall winner followed by Hopfield. mRmRQ is by Wilcoxon Signrank worse than mRmRD. Hopfield and unitary redundancy filters are not statistically worse than mRmRD. Random feature selection is clearly (and statistically significantly) worse than all other selection schemes. Greedy is the worst non-random scheme. Unitary redundancy filters come high up in third place.

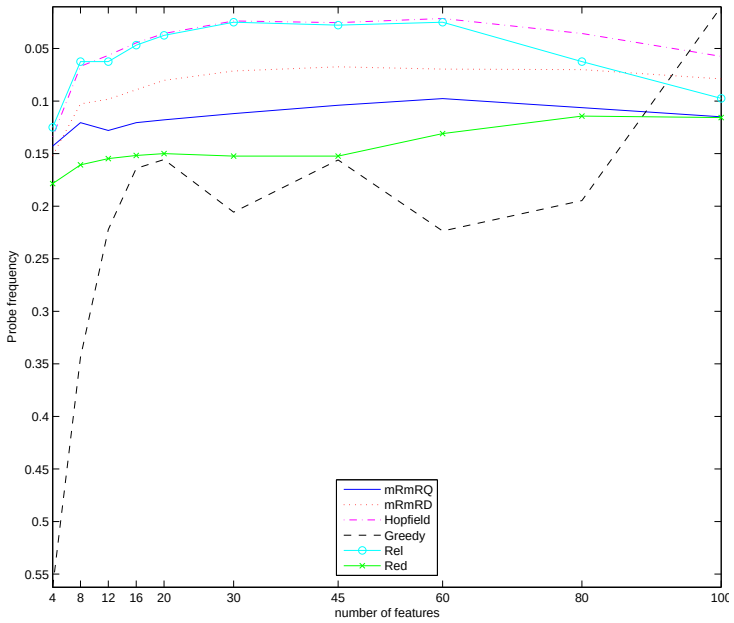

Fig. 1. Normalized Probe Frequencies of all Selection Schemes

[5] We included the Greedy schemes using a threshold of $\frac{|s_{\text{design}} - s_{\text{Greedy}}|}{s_{\text{design}}} \leq 0.1$.

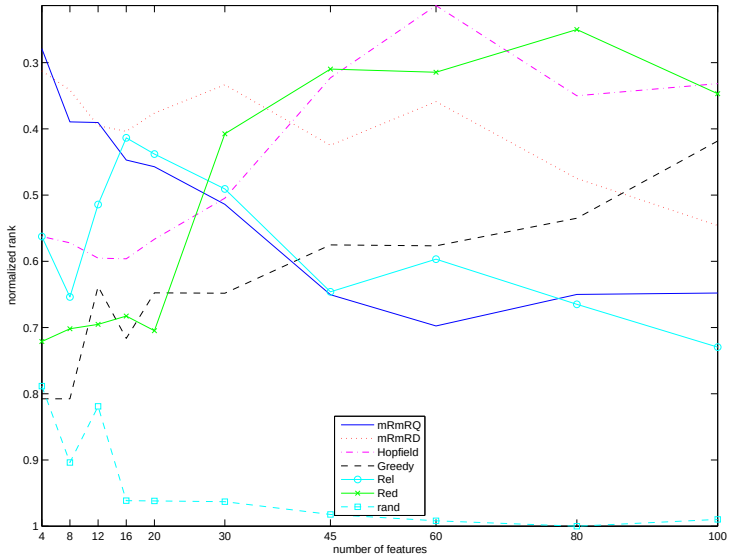

Fig. 2. Normalized Median Rankings of all Selection Schemes

Fig. 2 shows changes with different numbers of features. Because of the complications with the Greedy scheme, rankings of selection methods at all numbers of features were normalized by the total number of competing methods with their different combinations of methods. The medians for each selection scheme are depicted in fig. 2. We see that with more features all selection schemes become better than random choice because of the inclusion of less probes. MRmRQ starting as best at 4 variables, adding more features improves relatively less than most other selection schemes. Hopfield, unitary redundancy filters, and Greedy see best improvements as compared to other methods as compared to mRmRQ/D and unitary relevance filters. We observe that Hopfield copes better with higher feature spaces than other methods and constitutes one of the best methods from 45 features on.

Fig. 1 shows the expected frequency of probes in the selection of schemes. Frequencies are normalized by numbers of features in the selected set. We see that Greedy, the worst selection scheme, was very resistant to probes, however chosen features could obviously not have been the most useful ones. The same is true for (unitary) redundancy measures. Redundancy and Greedy curves show an increasing probe tolerance (as was to expect), the Greedy curve exhibiting a steep rise from 80 to 100 features. Unitary relevance filters and Hopfield let in most probes as compared to the other measures. mRmRD/Q were in the mid-field.

Over all classifiers, Spearman correlations of normalized ranks and inverse (subtracting from 1) normalized probe frequencies over all RR combinations at

each number of features ranged between −0.08 and 0.6. This suggests to us that low probe frequencies are not sufficient for good classifier performance. The correlations follow a curious pattern: they start at medium range with 4 features (0.5), go down (until −0.08) at 45 features, and climb up again. This suggests low probe frequencies being relatively important at low numbers of variables and when there are few to choose from (but not in-between). This explains for higher numbers of features the success of redundancy filters and the revival of Greedy.

As we will see below in table 2, the probe of zeros (feature index 177), enjoyed some popularity. This seems to be a problem that comes from the skewed class-distributions in our data, which makes that 0 can be to 90% associated with one class.

♯features	selection method	LP	Gabors	LC	TC	Int.	RN	RU	Zeros
4	mRmRQ, CC+CC	4.42	0.89	0.00	0.00	0.00	0.00	0.00	44.25
8	mRmRD. CC+VDM	6.64	1.11	0.00	0.00	0.00	0.00	0.00	0.00
12	mRmRD, CC+VDM	7.38	0.89	0.00	2.11	0.00	0.00	0.00	0.00
16	Redundancy Filter VDM	0.00	1.33	4.92	0.00	0.00	0.00	0.00	0.00
20	Redundancy Filter VDM	0.00	1.33	4.92	0.00	0.00	0.00	0.00	0.00
30	mRmRQ RVDM FC	5.31	1.00	1.97	0.84	0.00	0.00	0.00	0.00
45	mRmRQ, RVDM+FC	3.54	1.14	2.62	0.56	0.00	0.00	0.00	0.00
60	Hopfield. KSC+CC	0.00	1.24	2.95	2.95	2.95	0.00	0.00	2.95
80	Hopfield, KSD+FC	0.89	1.28	2.21	2.21	2.21	0.00	0.00	2.21
100	Hopfield, KSD+FC	1.77	1.27	1.77	1.77	1.77	0.00	0.00	1.77

Table 2. Best Selection Methods for each Number of Features

Table 2 lists from 4 to 100 numbers of features (first column) the selection method (second column) that provided the best performing feature set. At the end of the second column we put the redundancy and relevance measures (cf. 4.2). From column 3-10 you see normalized frequencies of features from Laplacian Pyramid (LP), Gabor filters, luminance contrast (LC), texture contrast (TC), intensity (Int.), random normal probes (RN), random uniform probes (RU), and the zero probe. The frequency of each feature type in the selected set was divided by the frequency expected from prior probabilities. E.g. as for Gabor filters, the a-priori probability is $100/177 \approx 0.56$. For 100 features, the expected number of Gabor features is $0.56 \times 100 = 56$. The figures corresponding to each number of features and feature type tell how much the frequencies found for the type exceed or fall behind expectations. E.g. for 100 features from Gabor filters there were 1.27 times more than expected.

Laplacian Pyramids, intensity, texture contrast, and luminance contrast appear prominently (relative to their proportion in the feature set). There are many Gabor filters present. It is remarkable that only few probes are selected (however the zeros each time).

5 Conclusions

In this paper, we presented a new method for feature selection based on redundancy and relevance of features. Approaches that use neural networks for

feature selection (e.g. [35]) or that use feature selection before feeding data into neural networks exist in manifold (e.g. [32]. However, to our knowledge, a non-supervised neural network has not been used before for feature selection in the minimal redundancy and maximal relevance framework. As a recurrent artificial neural network attractor model, the Hopfield network [17,18], shares phenomenology with the associative memory function of the cortex. Similarly in both, if several patterns are presented simultaneously, a rivalry process leads to competition, from which stable states result in perceptual groupings. In the brain, this process possibly functions by synchronization of neural cell oscillation [14].

On the whole, for all the tested features we saw that mRmRD was the best combination scheme. Curiously, the selections with least numbers of probes are not necessarily the best ones. We observed a log-shaped performance curve over number of features, unsaturated until 100 for some methods. Therefore we decided that the best selection came from the best method at 100 features, a Hopfield network using the Kolmogorov-Smirnov test as redundancy measure and FC relevance. Selection with Hopfield networks showed improvements for higher-dimensioned feature sets.

Not explained yet, but what deserves mention is that the SVM classifier was the best classifier, followed by GentleBoost and Naïve Bayes. Performance estimations obtained could be optimistically biased, because we used only one data set for estimation and the methods were partly chosen for the expected aptness in the domain.

6 Future Work

We identify four lines of future work. These concern relevance and redundancy measures, discretization methods, extension to multi-class classification, and more numbers of features.

Though not reported in this article, due to space limitations[6], we obtained some interesting results comparing different relevance and redundancy measures. As for redundancy, Jensen–Shannon Divergence, was best, followed by VDM, and the sign–test. As for relevance measures, VDM and FC were good. Some measures had difficulties with the zero–probe and this should be taken care of. Future research could try out other relevance and redundancy measures and try them out on several data sets.

Liu et al. [21] systematize and test several methods of discretization and again found the minimum description length best performing. Ding, Peng et al. [23] chose to discretize data using $mean \pm \alpha\sigma$, with $\alpha \in [0, 2, 0.5]$. We have not tried out this method, neither did we try out minimum description length.

Although this study was limited to binary classification, its methods are not and it remains to be seen how our feature selection scales up from a two-class problem to multi-class domains with thousands of features.

[6] For relevance and redundancy measures check [2] for details

In our experimental design we emphasized few numbers of features (70% below 50), which turned out favorably for mRmRD. The Hopfield network selection seems to perform well for high-dimensional feature spaces and could be used in analysis of complex data. It stands out to test the methods in higher dimensional feature spaces. Studies in this direction for the NIPS feature selection challenge are in preparation.

References

1. Hervé Abdi. *Les reseaux de neurones*. Presses Universitaires de Grenoble, 1994.
2. Benjamin Auffarth. Classification of biomedical high-resolution micro-ct images for direct volume rendering. Master's thesis, University of Barcelona, Barcelona, Spain, 2007.
3. P. J. Burt and E. H. Adelson. The laplacian pyramid as a compact image code. *IEEE Trans. Communications*, 31:532–540, 1983.
4. Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.
5. C. Cortes and V. Vapnik. Support-vector network. *Machine Learning*, 20:273–297, 1995.
6. Thomas M. Cover. The best two independent measurements are not the two best. *IEEE Transactions on Systems, Man, and Cybernetics*, 4:116–117, 1974.
7. J. Demsar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.
8. C. Ding and H. Peng. Minimum redundancy feature selection from microarray gene expression data. In *Second IEEE Computational Systems Bioinformatics Conference*, pages 523–529, 2003.
9. W. Duch and J. Biesiada. Feature selection for high-dimensional data: A kolmogorov-smirnov correlation-based filter solution. In M. Kurzynski, E. Puchala, M. Wozniak, and A. Zolnierek, editors, *Advances in Soft Computing*, pages 95–104. Springer, 2005.
10. W. Einhäuser, W. Kruse, K.-P. Hoffman, and P. König. Differences of monkey and human overt attention under natural conditions. *Vision Research*, 46(8-9):1194–1209, 2006.
11. T. Fawcett. Roc graphs: Notes and practical considerations for researchers. technical report, HP Laboratories, Palo Alto, 2004.
12. I. Fogel and D. Sagi. Gabor filters as texture discriminator. *BioCyber*, 61:102–113, 1989.
13. J. Friedman, T. Hastie, and R. Tibshirani, *Additive logistic regression: a statistical view of boosting*, Tech. report, Department of Statistics, Stanford University, 1998.
14. P. Fries, J. H. Reynolds, A. E. Rorie, and R. Desimone. Modulation of oscillatory neuronal synchronization by selective visual attention. *Science*, 291:1560–1563, 2001.
15. A.B.A. Graf and S. Borer. Normalization in support vector machines. In B. Radig and S. Florczyk, editors, *DAGM 2001: Pattern Recognition.*, LNCS 2191, pages 277–282. Springer, 2001.
16. I. Guyon, S. Gunn, A. Ben-Hur, and G. Dror. *Result analysis of the NIPS feature selection challenge*, volume 17, pages 545–552. MIT Press, 2004.
17. J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. In *Proceedings of the National Academy of Science*, volume 79, pages 2554–2558, 1982.

18. J. J. Hopfield. Neurons with graded responses have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences*, 81:3088–3092, 1984.
19. Theo A. Knijnenburg. Selecting relevant and non-relevant features in microarray classification applications. Master's thesis, Delft Technical University, Faculty of Electrical Engineering, 2628 CD Delft, 2004.
20. John Lafferty, Andrew McCallum, and Fernando Pereira, *Conditional random fields: Probabilistic models for segmenting and labeling sequence data*, Proc. 18th International Conf. on Machine Learning, Morgan Kaufmann, San Francisco, CA, 2001, pp. 282–289.
21. Huan Liu, Farhad Hussain, Chew Lim Tan, and Manoranjan Dash. Discretization: An enabling technique. *Data Min. Knowl. Discov.*, 6(4):393–423, 2002.
22. Maite López-Sánchez, Jesús Cerquides, David Masip, and Anna Puig, *Classification of biomedical high-resolution micro-ct images for direct volume rendering*, Proceedings of IASTED International Conference on Artificial Intelligence and Applications (AIA 2007) (Austria), IASTED, 2007, pp. 341–346.
23. Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.
24. Pamela Reinagel and Anthony Zador. Natural scene statistics at center of gaze. *Network: Comp. Neural Syst.*, 10:341–350, 1999.
25. M. Riedmiller and H. Braun. Rprop – description and implementation details. Technical report, Universitat Karlsruhe, 1994.
26. F. Rossi, A. Lendasse, D. François, V. Wertz, and M. Verleysen. Mutual information for the selection of relevant variables in spectrometric nonlinear modelling. *Chemometrics and Intelligent Laboratory Systems*, 2(80):215–226, 2006.
27. B.C. Russell, A. Torralba, K.P. Murphy, and W.T. Freeman. Labelme: a database and web-based tool for image annotation. MIT AI Lab Memo AIM-2005-025, MIT CSAIL, September 2005.
28. Yvan Saeys, Iñaki Inza, and Pedro Larrañaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, August 24, 2007.
29. C. Stanfill and D. Waltz. Toward memory-based reasoning. *Communications of the ACM*, 29(12):1213–1228, 1986.
30. Michael Vogt. `http://pc228.rt.e-technik.tu-darmstadt.de/~vogt/de/software.html`. [Online; accessed 9-October-2007].
31. D.W. Tank and J.J. Hopfield. Simple "neural" optimization networks: An a/d converter, signal decision circuit, and a linear programming circuit. *ieeetcas*, 33:533–541, 1986.
32. O. Valenzuela, I. Rojas, L.J. Herrera, A. Guillén, F. Rojas, L. Marquez, and M. Pasadas. Feature selection using mutual information and neural networks. *Monografias del Seminario Matematico Garcia de Galdeano*, 33:331–340, 2006.
33. D. Xu, J. Lee, D.S. Raicu, J.D. Furst, and D. Channin. Texture classification of normal tissues in computed tomography. In *The 2005 Annual Meeting of the Society for Computer Research*, 2005.
34. Lei Yu and Huan Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *ICML*, pages 856–863, 2003.
35. L. Yu and H. Liu. Redundancy based feature selection for microarray data. *ACM SIGKDD 2004*, pages 737–742, 2004.
36. Jie Zhou and Hanchuan Peng. Automatic recognition and annotation of gene expression patterns of fly embryos. *Bioinformatics*, 23(5):589–596, 2007.