POLITECNICO DI TORINO

PhD in Information and System Engineering
XXII cycle

III Facoltà di Ingegneria
Settore scientifico ING-INF/05

PhD Thesis

Extraction of biological knowledge by means of data mining techniques

Author:
Alessandro Fiori
Matr. 143494

Supervisor:
Prof. Elena Baralis

April 2010
A.A. 2009/2010

In memory of Sweet

Acknowledgments

I would like thank Prof.ssa Elena Baralis for her qualified support and her availability. Her advices were precious and fundamental for the development of my PhD work. I thank her because she believed in me and offer me the possibility of working to very interesting projects. I will be grateful to her.

I want to thank Prof. Gerhard Weikum from Max-Planck-Institut für Informatik because he gave me the opportunity to spend six months abroad to work with his research group. I appreciated his help and his continuous patient support.

My PhD and my university career probably would not have been possible without the people that accompanied me in these years. I want to thank all my research group colleagues, Daniele, Giulia, Paolo, Tania and Vincenzo, for their constant support and suggestions. Another thank goes to Elisa Ficarra for her help in understanding bioinformatics problems.

A special thank goes to my family for their constant belief and support. I will be always grateful to them. I want also thanks my dogs, Sweet and Sophie, because they gave me a lot of love. This thesis is in memory of Sweet, my "brother".

Finally, a special acknowledgement to Maria. Her love and support through these years was very important. Words alone can not adequately express my gratitude.

Contents

1

Introduction

Bioinformatics and computational biology involve the use of techniques including applied mathematics, informatics, statistics, computer science, artificial intelligence, chemistry, and biochemistry to solve biological problems usually on the molecular level. The core principle of these techniques is using computing resources in order to solve problems on scales of magnitude far too great for human discernment. The research in computational biology often overlaps with systems biology. Major research efforts in this field include sequence alignment, gene finding, genome assembly, protein structure alignment, protein structure prediction, prediction of gene expression and protein-protein interactions, and the modeling of evolution. The huge amount of data involved in these research fields makes the usage of data mining techniques very promising. These techniques, starting from many sources, such as the results of high throughput experiments or clinical records, aims at disclosing previously unknown knowledge and relationships.

Different data sources became available in recent years. For example, DNA microarray experiments generate thousands of gene expression measurements and provide a simple way for collecting huge amounts of data in a short time. They are used to collect information from tissue and cell samples regarding gene expression differences. Compared with traditional tumor diagnostic methods, based mainly on the morphological appearance of the tumor, methods relying on gene expression profiles are more objective, accurate, and reliable [61].

Moreover, document collections of published papers are an interesting data source to retrieve background knowledge on specific topics. Analyzing the most relevant parts of research papers and performing on demand data integration for inferring new knowledge and for validation purposes is a fundamental problem in many biological studies. The growing availability of large document collections has stressed the need of effective and efficient

techniques to operate on them (e.g., navigate, analyze, infer knowledge). Given the huge amount of information, it has become increasingly important to provide improved mechanisms to detect and represent the most relevant parts of textual documents effectively.

Analyzing different data sources is necessary to model gene and protein interactions and build a knowledge background of biological processes. Microarray data analysis allows identifying the most relevant genes for a target disease and group of genes with similar patterns under different experimental conditions. Feature selection and clustering algorithms are the most widely used approaches to face these issues.

Feature selection indeed allows the identification of the genes which are relevant or mostly associated with a tissue category, disease state or clinical outcome. An effective feature selection reduces computation cost and increases classification accuracy. Furthermore, when a small number of genes is selected, their biological relationship with the target disease is more easily identified, thus providing additional scientific understanding of the problem.

Clustering is a useful exploratory technique for gene expression data as it groups similar objects together and allows the biologist to identify potentially meaningful relationships between the genes. The genes belonging to the same cluster are typically involved in related functions and are frequently co-regulated [38]. Thus, grouping similar genes can provide a way to understand functions of genes for which information has not been previously available. Another employment of clustering algorithms is to identify group of redundant genes and then select only a representative for each group to perform a dimensionality reduction [72].

Although these techniques retrieve good models of biological processes, they are strong dependent by the data employed in the experimental analysis. Moreover, in many biomedical application a background knowledge is not available. Thus, text mining methods applied on published research papers may provide powerful tools to validate the experimental results obtained by other data analysis studies. Initially, analyzing and extracting relevant and useful information from research papers was manually performed by molecular biologists [64]. In last years summarization approaches allow facing with this problem in an automatic way.

The aim of this thesis is to exploit data mining techniques to solve specific biological problems, i.e., i) identifying the most relevant genes related to a target disease, ii) grouping together genes which show a similar behavior under different conditions, and iii) automatically extracting biological information from research papers. Particularly, a new feature selection method has

been developed to identify the most relevant genes and thus improve the accuracy of prediction models for sample classification. Furthermore, to study the correlations among genes under different experimental conditions, a new similarity measure has been integrated in a hierarchical clustering algorithm. Finally, a new summarization approach in order to provide an automatic tool to extract relevant biological information from research papers is presented. The summarizer can also be exploited to validate the experimental results obtained with other data analyses.

The thesis is organized as follows. Background knowledge about data mining techniques applied to microarray data is provided in Chapter 2, while Chapter 3 introduces the problem of document collection analysis. Chapter 4 introduces the definitions exploited for gene expression analysis. Chapter 5 describes the methods to select the minimum relevant set of genes to improve classification accuracy on training data, while Chapter 6 introduces a filter feature selection method to select the most relevant genes for multiclass classification problems. Moreover, in Chapter 7 a new similarity measure to group similar genes integrated in a hierarchical clustering is presented. A summarization technique to extract the knowledge related to genes and proteins interaction with biological processes is described in Chapter 8. Experimental designs and results are reported in each chapter. Finally, Chapter 9 draws conclusions and presents future developments for each technique.

2

Microarray data analysis

In the last years, with the developing of new technologies and revolutionary changes in biomedicine and biotechnologies, there was an explosive growth of biological data. Genome wide expression analysis with DNA microarray technology has become a fundamental tool in genomic research. Since microarray technology was introduced, scientists started to develop informatics tools for the analysis and the information extraction from this kind of data. Due to the characteristics of microarray data (i.e. high levels of noise, high cardinality of genes, small samples size) data mining approaches became a suitable tool to perform any kind of analysis on these data.

Many techniques can be applied to analyze microarray data, which can be grouped in four categories: classification, feature selection, clustering and association rules.

Classification is a procedure used to predict group membership for data instances. Given a training set of samples with a specific number of attributes (or features) and a class label (e.g., a phenotype characteristic), a model of classes is created. Then, the model is exploited to assign the appropriate class label to new data. Model quality is assessed by means of the classification accuracy measure, i.e., the number of correct label predictions over the total number of unlabeled data. The classification of microarray data can be useful to predict the outcome of some diseases or discover the genetic behavior of tumors.

Since genetic data are redundant and noisy, and some of them do not contain useful information for the problem, it is not suitable to apply the classification directly to the whole dataset. Feature selection techniques are dimensional reduction methods usually applied before classification in order to reduce the number of considered features, by identifying and removing the redundant and useless ones. Moreover, feature selection algorithms applied to microarray data allow identifying genes which are highly correlated with

the outcome of diseases.

Another way to identify redundant genes is to group together sets of genes which show a similar behavior, and then select only a representative for the group. Furthermore, genes with similar expression pattern under various conditions or time course may imply co-regulations or relations in functional pathways, thus providing a way to understand functions of genes for which information has not been previously available.

In this chapter, we focus on the application of data mining techniques on microarray data, with the aim of making researchers aware of the benefits of such techniques when analyzing microarray data. The chapter is organized as follows. The first two sections provide a description of microarray data, to highlight the issues concerned with their analysis, and a brief discussion about the data cleaning approaches that can be exploited to prepare data before data mining. The following three sections provide a survey of classification, feature selection and clustering techniques based on their aims and characteristics. Finally, the last section describes new trends and provide some prospects of data mining application to microarray data.

2.1 Microarray datasets

A microarray dataset E can be represented in the form of a gene expression matrix, in which each row represents a gene and each column represents a sample. For each sample, the expression level of all the genes under consideration is measured. Element e_{ij} in E is the measurement of the expression level of gene i for sample j, where $i = 1, \ldots, N$, $j = 1, \ldots, M$ and usually $N \gg M$. Each sample is also characterized by a class label, representing the clinical situation of the patient or the biological condition of the tissue. The domain of class labels is characterized by C different values and label l_j of sample j takes a single value in this domain.

The format of a microarray dataset conforms to the normal data format of machine learning and data mining, where a gene can be regarded as a feature or attribute and a sample as an instance or a data point. However, the main characteristics of this data type are the high number of genes (usually tens of thousands) and the low number of samples (less then one hundred). This peculiarity causes specific challenges in analyzing microarray data (e.g., complex data interactions, high level of noisy, lack of biological absolute knowledge) which have to be addressed by data mining methods [109] .

In recent years an abundance of microarray datasets become public available due to the increase of publication in bioinformatics domain. A large collection of public microarray data is stored by the ArrayExpress archive (`http://www.ebi.ac.uk/microarray-as/ae/`). The datasets, stored in MIAME and MINSEQE format, are all preprocessed, but also the raw data (for a subset of the collection) can be downloaded. One of the best feature of this archive is the possibility to browse the entire collection or perform queries on experiment properties, submitter, species, etc. In the case of queries, the system retrieves summaries of experiments and complete data. Other datasets can be also downloaded from the author or tool websites (e.g., LibSVM software by [33] , GEMS software by [131]).

2.2 Data cleaning

With the term of data cleaning we refer to the task of detecting and correcting or removing corrupt or inaccurate records from a dataset, before applying a data mining algorithm. Microarray data cleaning includes the following issues.

Normalization. Normalization is needed to adjust the individual hybridization intensities to balance them appropriately so that meaningful biological comparisons can be made. It ensures that differences in intensities are due to differential expression and not some printing, hybridization or scanning artifacts. Several normalization methods have been proposed in literature [133] and some software package have been developed for the analysis of microarray data. One of the most popular and general purpose software packages for microarray data is Bioconductor (`http://www.bioconductor.org/`). Other software are distributed by the companies that produce the microarray technology, like Affymetrix and Agilent [165] .

Missing value estimation. Missing values in microarray data arise due to technical failures, low signal-to-noise ratio and measurement errors. For example, dust present on the chip, irregularities in the spot production and inhomogeneous hybridization all lead to missing values. It has been estimated that typically 1% of the data are missing affecting up to 95% of the genes [81]. To limit the effects of missing values several works addressed the problem of missing value estimation, and the most used approach is the k-nearest neighbors algorithm.

Outlier detection. The problem of outliers defined as "anomalous data points" often arises in large datasets. The aim of outlier detection methods is to detect and remove or substitute outliers. A broad survey of methods that have been found useful in the detection and treatment of outliers on microarray data analysis is presented in [107]. Usually outliers are detected by computing the mean and the standard deviation of values. The values outside the range $\mu \pm \sigma$ are considered outliers. Other techniques have been proposed by replacing the mean and the standard deviation values, for example by using 3σ instead of σ. An alternative specifically used in the microarray data analysis community is the Hampel identifier [39], which replaces the mean with the median and the standard deviation with the median absolute deviation.

After data have been cleaned through the previously discussed methods, the appropriate data mining technique can be applied.

2.3 Classification

An important problem in microarray experiments is the classification of biological samples using gene expression data, especially in the context of cancer research. Conventional diagnostic methods are based on subjective evaluation of the morphological appearance of the tissue sample, which requires a visible phenotype and a trained pathologist to interpret the view. In some cases the class is easily identified by cell morphology or cell-type distribution, but in many cases apparently similar pathologies can lead to very different clinical outcomes. Examples of diagnostic classes include cancer versus noncancer, different subtypes of tumor, and prediction of responses to various drugs or cancer prognosis. The prediction of the diagnostic category of a tissue sample from its expression array phenotype given the availability of similar data from tissues in identified categories is known as classification [158]. Firstly in [56], the feasibility of cancer classification based solely on gene expression monitoring is demonstrated.

2.3.1 Classification challenges

A critical issue in classifying microarray data is the limited number of samples that are available, thus it is difficult to asses the statistical significance of results. Moreover, the high number of genes could introduce noise affecting the classification model. Different algorithms were studied and proposed to

define classification models for microarray data. The most used are reported in the followings. In the next paragraph we will discuss which methods are demonstrated to deal better with the characteristics of microarray data. However, the comparison of results is another critical issue, because of the amount of different exploited experimental designs. In fact, the classification accuracy of an algorithm strongly depends on the exploited experimental design.

2.3.2 Classification algorithms

The most used classification algorithms exploited in the microarray analysis belong to four categories: decision tree, Bayesian classifiers, neural networks and support vector machines.

Decision Tree. Decision tree derives from the simple divide-and-conquer algorithm. In these tree structures, leaves represent classes and branches represent conjunctions of features that lead to those classes. At each node of the tree, the attribute that most effectively splits samples into different classes is chosen. To predict the class label of an input, a path to a leaf from the root is found depending on the value of the predicate at each node that is visited. The most common algorithms of the decision trees are ID3 [112] and C4.5 [113]. An evolution of decision tree exploited for microarray data analysis is the random forest [30], which uses an ensemble of classification trees. [41] showed the good performance of random forest for noisy and multi-class microarray data.

Bayesian classifiers and Naive Bayesian. From a Bayesian viewpoint, a classification problem can be written as the problem of finding the class with maximum probability given a set of observed attribute values. Such probability is seen as the posterior probability of the class given the data, and is usually computed using the Bayes theorem. Estimating this probability distribution from a training dataset is a difficult problem, because it may require a very large dataset to significantly explore all the possible combinations. Conversely, Naive Bayesian is a simple probabilistic classifier based on Bayesian theorem with the (naive) independence assumption. Based on that rule, using the joint probabilities of sample observations and classes, the algorithm attempts to estimate the conditional probabilities of classes given an observation. Despite its simplicity, the Naive Bayes classifier is known to be a robust method, which shows on average good performance in terms of classification accuracy, also when the independence assumption does not hold [96].

Artificial Neural Networks (ANN). An artificial neural network is a mathematical model based on biological neural networks. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. Neurons are organized into layers. The input layer consists simply of the original data, while the output layer nodes represent the classes. Then, there may be several hidden layers. A key feature of neural networks is an iterative learning process in which data samples are presented to the network one at a time, and the weights are adjusted in order to predict the correct class label. Advantages of neural networks include their high tolerance to noisy data, as well as their ability to classify patterns on which they have not been trained. In [91] a review of advantages and disadvantages of neural networks in the context of microarray analysis is presented.

Support vector machines (SVM). Support vector machines are a relatively new type of learning algorithm, originally introduced by [141]. Intuitively, SVM aims at searching for the hyperplane that best separates the classes of data. SVMs have demonstrated the ability not only to correctly separate entities into appropriate classes, but also to identify instances whose established classification is not supported by data. Although SVMs are relatively insensitive to the distribution of training examples in each class, they may still get stuck when the class distribution is too skewed.

Sometimes, a combination of the presented methods may outperform the single technique. For example, a method which combines a neural network classifier with a Bayesian one is proposed in [167]. In order to consider the correlations among genes, they build a neural network where the weights are determined by a Bayesian method. [163] proposed a Bayesian approach combined with SVM to determine the separating hyperplane of an SVM, once its maximal margin is determined in the traditional way.

2.3.3 Comparison for classification methods

Some works tried to compare classifier performances on microarray dataset. However, since a huge benchmark of microarray datasets is missing in literature, a comparison of results presented in different works is very difficult. Particularly, the main problem is the choice of the experimental design used to compute the classification accuracy. In fact, different types of experimental design (i.e., leave-one-out, k-fold cross-validation, bootstrap and re-substitution) exist and are exploited in different works.

In [92] the four classification techniques previously described were com-

pared on three microarray datasets and the accuracy is computed by applying a 10-fold cross-validation technique. The best performance is reached by SVM and ANN. For all datasets, the decision tree presents the worst performance. A similar result is presented in [110]. The robustness of SVMs is also remarked in [89] where the SVM outperforms all the other methods in the most of analyzed microarray datasets. On the other hand, the decision tree shows always the worst performance [146].

According to the experimental results presented in these studies, the best method for classifying microarray datasets seems to be the SVM, because it is the most powerful method to deal with the main characteristics of microarray data (i.e., few samples and high number of features). In fact, SVM reaches the best accuracy on almost every dataset. Therefore, the SVM represents the state-of-art for classification task on microarray. By regarding the experimental design, in [29] the authors conclude that the k-fold cross validation seems to be the best estimator of classification performance compared to the other methods, because bootstrap has a high computational cost and re-substitution tends to be biased.

2.4 Feature selection

Since the number of genes is usually significantly greater than the number of samples, and only a subset of the genes is relevant in distinguishing different classes, a feature selection is usually applied before classification. In this way, the performance of classifiers generally improves, because of the reduction in data dimensionality, the speed up of the learning process and the increasing in model interpretability [158]. Furthermore, when analyzing microarray datasets, feature selection helps in providing a deeper understanding of the molecular basis of diseases. In fact, by selecting only the most relevant genes, the biologists are allowed to investigate only a subset of genes which are strongly correlated with the considered classes (i.e., different diseases, different tumor types, and different relapse times).

Feature selection techniques can be divided in two high levels groups: supervised methods, which take into account the sample class information, and unsupervised methods, which analyze only the data distribution without using sample class labels. Among the first type, a further categorization can be done among filter, wrapper and embedded methods. In the following, first the common challenges of feature selection methods are described. Then, an overview of recent works in each category is presented, and finally

a comparison of their main characteristics is provided.

2.4.1 Feature selection challenges

A first challenge in the feature selection applications for microarray is that a ground truth of biological knowledge about the genes which are responsible of outcome diseases is missing. Thus, the validation of results is an open problem. Some ontologies, such as UMLS and GO, try to model the biological processes and the correlations among genes/proteins and the diseases. However, methods that integrate this heterogeneous knowledge are very few [111, 106].

Furthermore, some of the feature selection methods evaluate each gene in isolation, thus ignoring gene correlations. This problem is known as univariate approach, in contrast with the multivariate approach that considers the effects of groups of genes working together.

Finally, the evaluation of feature selection methods is highly dependent on classification task. Usually the experimental sections which are addressed to show the goodness of a method use the classification accuracy as measure of performance. A challenge in this direction should be the identification of a benchmark and a ground truth of biological processes to separate the gene list accuracy from the accuracy provided by a classifier.

2.4.2 Unsupervised feature selection methods

The unsupervised techniques do not require the class information on samples and can be applied when the information in biological datasets is incomplete. Since more effective supervised feature selection methods have been developed, there are only few unsupervised methods proposed in recent works.

The simplest unsupervised evaluation of the features is the variance. Higher the variance is, higher the gene relevance, because its expression varies among different conditions. On the contrary, if a gene expression does not vary very much, it can be considered irrelevant for the analysis. Although the data variance criteria finds features that are useful for representing data, it is not suited for selecting ones that must be useful for discriminating between samples in different classes. Thus, variance is generally used in addition to other methods [43].

In [142] the authors used the SVD decomposition to compute the SVD-entropy as a feature selection method. They propose several selection strate-

gies such as simple ranking (SR), forward selection (FS) and backward elimination (BE). However, the SVD-entropy is very expensive in term of computational cost in the case of large number of features as in microarray datasets.

Another unsupervised feature selection approach is the Laplacian score [63]. It is based on the observation that two data points are probably related to the same topic if they are close to each other. The assumption is that in many learning problem the local structure of the data space is more important than the global structure. The score computes how a feature respect the structure of a nearest neighbor graph of the dataset is built. An improvement of the Laplacian score is the LLDA-RFE [103]. While the Laplacian score is a univariate approach, the LLDA-RFE is multivariate allowing in this way to select features that contribute to the discrimination with other features. Also this approach has problem of complexity due to the computation of SVD and eigenvectors.

Other methods of unsupervised feature selection are based on clustering algorithms in order to identify group of similar genes and perform further analyses on this subset. For example, [99] presented a method based on measuring similarity between features in order to remove redundancy. The method partitions the dataset into distinct clusters using a new measure, called maximum compression index, and then selects a representative feature for each cluster.

2.4.3 Supervised feature selection methods

While the unsupervised methods analyze only the intrinsic characteristics of data (e.g., variance), the supervised techniques perform analyses considering the data distribution according to the sample classes. Among supervised feature selection methods, a further categorization can be done among filter methods, which assess the relevance of features by looking only at the data characteristics, wrapper methods, which use the model hypotheses to select the feature subset, and embedded methods, which search the optimal subset while the classifier model is built.

Filter methods. Filter methods aim at evaluating the differential expression of genes and rank them according to their ability to distinguish among classes. A gene is differentially expressed if it shows a certain distribution of expression levels under one condition and a significantly different distribution under the other conditions.

In literature many techniques have been proposed to address the problem

of detecting differentially expressed genes and define new ranking procedures [93]. Classic statistical approaches for detecting differences between two groups include t-test, Wilcoxon test, and Mann-Whitney test. For multiclass problems the statistical tests ANOVA, Kruskal-Wallis test, and Friedman test are exploited. These methods have the virtue of being easily and very efficiently computed. The disadvantage is that some of these methods require assumptions on the data distribution. For example, the t-test requires that expression levels are normally distributed and homogeneous within groups and may also require equal variances between them. These assumptions may be inappropriate for subsets of genes.

Other filter methods (e.g., information gain, gini index, max minority, sum minority, sum of variance, twoing rule) are implemented in the RankGene software [134]. These measures are widely used in literature for gene expression analysis. They attempt to quantify the best class predictability that can be obtained by dividing the full range of expression values of gene in two disjoint intervals (e.g. up-regulated, down-regulated). Each measure belonging to this category quantifies the error in prediction in a different manner.

A deficiency of ranking genes by assessing a score to each one (i.e., the univariate approach) is that the features could be correlated among themselves. For example, if two genes are top ranked but they are also highly correlated (i.e., they distinguish the same samples), their combination does not form a better feature. This raises the issue of redundancy within the feature set. The advantages of reducing the redundancy are that with the same number of features the subset is more representative of the targeted phenotypes and the same accuracy is reached by a smaller subset of features than larger conventional feature sets. In [42] the authors proposed a method to expand the space covered by the feature set by requiring the features to be maximally dissimilar to each other (e.g., by maximizing their mutual Euclidean distance or minimizing their pairwise correlations).

Wrapper methods. Feature selection using wrapper methods offers an alternative way to perform a multivariate gene subset selection, incorporating the classifier's bias into the search and thus offering an opportunity to construct more accurate classifiers. Since the features to analyze are generally tens of thousands, wrapper techniques can not be applied alone and require a further step to avoid the exhaustive search among all the possible solutions. In fact, the number of feature subsets grows exponentially with the number of features, making enumerative search infeasible. Wrapper methods typically require extensive computation to search the best features and de-

pend on the learning algorithm used [78]. Furthermore, they do not always achieve better classification performance [84], depending on the quality of the heuristics applied to the huge subset space. For these reasons, usually filter methods are preferred.

Some works combine a filter and a wrapper approach to gain the advantages of both. For example, [102] proposed a hybrid gene selection method. The first step is a filter technique in which each gene is evaluated according to a proximity degree metric. In the second step a wrapper procedure is performed using a genetic algorithm to choose the optimized gene subsets from the top-ranked genes. In this way, a subset of top-ranked genes are preselected and then a classification algorithm is applied on these genes to further select only a subset of them. Another interesting hybrid filter-wrapper approach is introduced in [122], crossing a univariate pre-ordered gene ranking with an incrementally augmenting wrapper method.

Embedded methods. The embedded approaches have the advantage of including the interaction with the classification model, while at the same time being far less computationally intensive than wrapper methods. For example, the random forest can be used to compute the relevance of a single gene in classification task [76, 41]. Guyon et al. [60] proposed the SVM-RFE feature selection based on SVM. The approach iteratively trains the classifier optimizing the weights of the features, then compute the ranking criterion for all features. Finally, the features with smallest ranking criterion are eliminated from the model. The weights given by the linear classifiers show the relevance of a feature in a multivariate way allowing the removing of irrelevant features represented by low weights. Draminski et al. [44] proposed a feature selection algorithm based on Monte Carlo approach and tree classifier performance. The method considers a particular feature to be important, or informative, if it is likely to take part in the process of classifying samples into classes more often than not.

The embedded methods, as wrapper approaches, have a high level of complexity due to the high number of features in microarrays.

2.4.4 Comparison for feature selection methods

As for the classification task, a benchmark of microarray data for comparing feature selection techniques is missing. Some works try to find out which method is the best for a particular dataset or present a robust behavior on different datasets. Jeffery et al. [73] compared eleven feature selection methods on six datasets. The gene lists produced by analyzed methods are

very dissimilar and produce different discrimination performance. The authors noticed that the t-statistic methods performed relatively poorly. Since microarray data could present high levels of noise together with low samples sizes, computing a t-statistic can be problematic, because the variance estimation can be skewed by the genes which have a low variance. Thus, genes with a low variance present a large t-statistic but they could be falsely predicted to be differentially expressed.

An interesting result is presented in [85]. The authors illustrated that it is not always true that multivariate approach perform better than univariate ones, because the correlation structures, if present, are difficult to extract due to the small number of samples, and that consequently, overly-complex gene selection algorithms that attempt to extract these structures are prone to overtraining.

The comparison presented in [89] analyzes in more details the behavior of feature selection methods respect to the classifier employed for model construction. They compare eight feature selection algorithms implemented in RankGene [134] software on 9 microarray datasets changing the number of selected features in a range from 1 to 250. The conclusion is that the accuracy of classification is highly dependent on the choice of the classification method. This choice becomes more important than the choice of feature selection method when the number of selected genes is higher than 150 since little variation on accuracy are detected. Moreover, a clear winner seems not to exist. In fact, each method shows different performance behavior on different datasets. In some cases, when a high number of genes are selected, the majority of genes are shared by all the gene lists.

According to the results presented in [89, 146], the best classifier on which a feature selection method should be tested is the decision tree. In fact, the decision tree algorithm shows in most of analyzed microarray data the worst performance. Thus, a good feature algorithm with a low number of features may improve dramatically the performance of a decision tree that is not so robust to the noise present in gene expression data.

2.4.5 Feature extraction

Another way to improve classification accuracy instead of selecting relevant genes is to combine them to obtain new artificial features, usually called meta-genes. Meta-genes combine the characteristics of many genes, thus few of them could reach a high classification accuracy.

The most popular method belonging to this category is the Principal Component Analysis (PCA). PCA is a technique that transforms the original attribute space in a new space in which the attributes are uncorrelated and ranked based on the amount of variation in the original data that they account for. The PCA is an unsupervised approach, therefore it does not use the available class membership information for the samples. For this reason on microarray datasets the performance achieved by a classifiers applied on the new feature space are worst than supervised methods [73].

The Fischer's Linear Discriminant Analysis (FLDA) is another popular feature extraction technique [45]. Differently to the PCA, FLDA is a supervised algorithm, which maximizes the ratio between the inter-class and the intra-class variances. The FLDA is more accurate in multiclass problems with respect to the PCA, since there is no reason to assume that the principal components obtained by the PCA must be useful to discriminate between data in different classes. The FLDA was applied with good results on microarray data by [46]. However, in some cases the accuracy of FLDA decreases due to the small training set and a fairly large number of genes that bias the estimation of covariance matrices.

Hanczar et al. [61] proposed a reduction algorithm to identify classes of similarity among genes and create representative genes for each class by means of a linear combination of genes with a high degree of similarity. Then, they apply an SVM classifier and evaluate its accuracy. The aim of this kind of analysis is to reach the highest possible accuracy, instead of selecting the most relevant genes, because producing a correct prediction of a relapse or a patient response to specific treatments is more important.

The main critical issue of feature extraction methods is the meaning of meta-genes. Since they are a combination of genes, they are not useful for diagnostic test or biomarker development. Thus, this kind of techniques can be exploited only to improve classification accuracy without a real biological meaning, while feature selection methods can be used also to identify real genes responsible of disease outcome.

2.5 Clustering

The goal of clustering in microarray technology is to group genes or experiments into clusters according to a similarity measure [38]. For instance, genes that share a similar expression pattern under various conditions may imply co-regulations or relations in functional pathways. Thus, clustering could

provide a way to understand function of genes for which information has not been available previously process [75]. Furthermore, clustering can be used as a preprocessing step before a feature selection or a classification algorithm, to restrict the analysis to a specific category or to avoid redundancy by considering only a representative gene for each cluster. Many conventional clustering algorithms have been applied or adapted to gene expression data [52, 74, 138] and new algorithms, which specifically address gene expression data, have recently been proposed [59]. In the following, first the main challenges of clustering methods are described. Then, an overview of recent works which applied the clustering to microarray data is presented, and finally a comparison of their main characteristics is provided.

2.5.1 Clustering challenges

The main challenges regarding the application of clustering to microarray data are (i) the definition of the appropriate distance between objects, (ii) the choice of the clustering algorithm, and (iii) the evaluation of final results. Especially evaluating the results of clustering is a non-trivial task. Each article justifies a specific evaluation criterion, and in literature many criteria exist, such as measures which evaluate the obtained clusters without knowing the real class of objects (i.e., homogeneity and separation), measures which evaluate the agreement between the obtained clusters and the ground truth, and measures which involve the comparison with biological databases (i.e., GO) to measure the biological homogeneity of clusters. In addition, some works also highlight the problem of giving a user friendly representation of clustering results. Another evaluation criterion could be the clustering computational complexity, even if an evaluation of the complexity and efficiency of a method is very difficult to perform without resorting to extensive benchmark.

2.5.2 Clustering algorithms

The similarity between objects is defined by computing the distance between them. Gene expression values are continuous attributes, for which several distance measures (Euclidean, Manhattan, Chebyshev, etc.) may be computed, according to the specific problem. However, such distance functions are not always adequate in capturing correlations among objects because the overall gene expression profile may be more interesting than the individual magnitude of each feature [144]. Other widely used schemes for determining

the similarity between genes use the Pearson or Spearman correlation coefficients, which measure the similarity between the shapes of two expression patterns. However, they are not robust with respect to outliers. The cosine correlation has proven to be more robust to outliers because it computes the cosine of the angle between the expression gene value vectors. Other kinds of similarity measures include pattern based (which consider also simple linear transformation relationships) and tendency based (which consider synchronous rise and fall of expression levels in a subset of conditions).

Once the distance measure has been defined, the clustering algorithms are divided based on the approach used to form the clusters. A detailed description of clustering algorithms applied to microarray has been provided by [128]. Mainly, they can be grouped in two categories, partitioning and hierarchical algorithms.

Partitioning algorithms. This family of clustering algorithms works similarly to k-means [94]. K-means is one of the simplest and fastest clustering algorithms. It takes the number of clusters (k) to be calculated as an input and randomly divides points into k clusters. Then it iteratively calculates the centroid for each cluster and moves each point to the closest cluster. This procedure is repeated until no further points are moved to different clusters. Despite its simplicity, k-means has some major drawbacks, such as the sensibility to outliers, the fact that the number of clusters has to be known in advance and that the final results may change in successive runs because the initial clusters are chosen randomly.

Several new clustering algorithms have been proposed to overcome the drawbacks of k-means. For example, the genetic weighted k-means algorithm [153] is a hybridization of a genetic algorithm and a weighted k-means algorithm. Each individual is encoded by a partitioning table which uniquely determines a clustering, and genetic operators are employed. Authors show that it performs better than the k-means in terms of the cluster quality and the clustering sensitivity to initial partitions.

In [40] the authors described the application of the fuzzy c-means to microarray data, to overcome the problem that a gene can be associated to more than one cluster. The fuzzy c-means links each gene to all clusters via a real-valued vector of indexes. The values of the components of this vector lie between 0 and 1. For a given gene, an index close to 1 indicates a strong association to the cluster. Inversely, indexes close to 0 indicate the absence of a strong association to the corresponding cluster. The vector of indexes defines thus the membership of a gene with respect to the various clusters. However, also in this approach there is the problem of parameter estimation.

Au et al. [21] proposed the attribute cluster algorithm (ACA), which adopts the idea of the k-means to cluster genes by replacing the distance measure with the interdependence redundancy measure between attributes and the concept of mean with the concept of mode (i.e., the attribute with the highest multiple interdependence redundancy in a group).

Hierarchical algorithms. Hierarchical clustering typically uses a progressive combination (or division) of elements that are most similar (or different). The result is plotted as a dendrogram that represents the clusters and relations between the clusters. Genes or experiments are grouped together to form clusters and clusters are grouped together by an inter-cluster distance to make a higher level cluster. Hierarchical clustering algorithms can be further divided into agglomerative approaches and divisive approaches based on how the hierarchical dendrogram is formed. Agglomerative algorithms (bottom-up approach) initially regard each data object as an individual cluster, and at each step, merge the closest pair of clusters until all the groups are merged into one. Divisive algorithms (top-down approach) starts with one cluster containing all the data objects, and at each step splits a cluster until only singleton clusters of individual objects remain. For example, Eisen et al. [48] applied an agglomerative algorithm called UPGMA (Unweighted Pair Group Method with Arithmetic Mean) and adopted a method to graphically represent the clustered data set, while Alon et al. [19] split the genes through a divisive approach, called the deterministic-annealing algorithm.

A variation of the hierarchical clustering algorithm is proposed in [74]. The authors applied a density-based hierarchical clustering method (DHC) on two datasets for which the true partition is known. DHC is developed based on the notions of density and attraction of data objects. The basic idea is to consider a cluster as a high-dimensional dense area, where data objects are attracted with each other. At the core part of the dense area, objects are crowded closely with each other, and thus have high density. Objects at the peripheral area of the cluster are relatively sparsely distributed, and are attracted to the core part of the dense area. Once the density and attraction of data objects are defined, DHC organizes the cluster structure of the data set in two-level hierarchical structures, one attraction tree and one density tree. However, to compute the density of data objects, DHC calculates the distance between each pair of data objects in the data set, which makes DHC not efficient. Furthermore, two global parameters are used in DHC to control the splitting process of dense areas. Therefore, DHC does not escape from the typical difficulty to determine the appropriate value of parameters.

2.5.3 Comparison for clustering methods

Richards et al. [121] provided a useful comparison of several recent clustering algorithms by concluding that k-means is still one of the best clustering method because it is fast, does not require parallelization, and produces clusters with slightly high levels of GO enrichment. Despite this consideration, the hierarchical clustering algorithms are the most used in biological studies. The main advantage of hierarchical clustering is that it not only groups together genes with similar expression pattern but also provides a natural way to graphically represent the data set [75]. The graphic representation allows users to obtain an initial impression of the distribution of data. However, the conventional agglomerative approach suffers from a lack of robustness because a small perturbation of the data set may greatly change the structure of the hierarchical dendrogram. Another drawback of the hierarchical approach is its high computational complexity.

In general, microarray data are clustered based on the continuous expression values of genes. However, when additional information is available (e.g., biological knowledge or clinical information), it may be beneficial to exploit it to improve cluster quality [71]. Clinical information can be used to build models for the prediction of tumor progression. For example Wang et al. [147] used epigenetic data to determine tumor progression in cancer, and Bushel et al. [32] presented a method to incorporate phenotypic data about the samples.

Au et al. [21] presented a particular validation technique for clustering. They selected a subset of top genes from each obtained cluster to make up a gene pool, and then they run classification experiments on the selected genes to see whether or not the results are backed by the ground truth and which method performs the best. Thus, they exploit class information on samples to validate the results of gene clustering. The good accuracy reached by selecting few genes from the clusters reveals that the good diagnostic information existing in a small set of genes can be effectively selected by the algorithm. It is an interesting new way of clustering validation, by integrating clustering and feature selection.

2.5.4 Biclustering

Due to the high complexity of microarray data, in last years the scientists focused their attention on biclustering algorithms. The notion of biclustering was first introduced in [62] to describe simultaneous grouping of both row

and column subsets in a data matrix. It tries to overcome some limitations of traditional clustering methods. For example, a limitation of traditional clustering is that gene or an experimental condition can be assigned to only one cluster. Furthermore, all genes and conditions have to be assigned to clusters. However, biologically a gene or a sample could participate in multiple biological pathways, and a cellular process is generally active only under a subset of genes or experimental conditions. A biclustering scheme that produces gene and sample clusters simultaneously can model the situation where a gene (or a sample) is involved in several biological functions. Furthermore, a biclustering model can avoid those noise genes that are not active in any experimental condition.

Biclustering of microarray data was first introduced in [36]. They defined a residual score to search for submatrices as biclusters. This is a heuristic method and can not model the cases where two biclusters overlap with each other. Segal et al. [126] proposed a modified version of one-way clustering using a Bayesian model in which genes can belong to multiple clusters or none of the clusters. But it can not simultaneously cluster conditions/samples. Bergmann et al. [26] introduced the iterative signature algorithm (ISA), which searches bicluster modules iteratively based on two pre-determined thresholds. ISA can identify multiple biclusters, but is highly sensitive to the threshold values and tends to select a strong bicluster many times. Gu and Liu (2008) proposed a biclustering algorithm based on Bayesian model. The statistical inference of the data distribution is performed by a Gibbs sampling procedure. This algorithm has been applied to the yeast expression data, observing that majority of founded biclusters are supported by significant biological evidences, such as enrichments of gene functions and transcription factor binding sites in the corresponding promoter sequences.

2.6 New trends and applications

In the last years many studies were addressed to integrate microarray data with heterogeneous information. Since microarray experiments present few samples, the accuracy of the hypotheses extracted by means of data mining approaches could be low. Using different sources of information (e.g., ontologies, functional data, published literature), the biological conclusions achieve improvements in specificity. For example, multiple gene expression data sets and diverse genomic data can be integrated by computational methods to create an integrated picture of functional relationships between genes. These integrated data can then be used to predict biological functions or to aid in

understanding of protein regulations and biological networks modeling [65].

For feature selection approaches some works integrate Gene Ontology (GO) in the computation of most relevant genes. For example in [111] the authors proposed a method that combines the discriminative power of each gene using a traditional filtering method with the discriminative values of GO terms. Moreover, redundancy is eliminated using the ontology annotations. The results show an improvement of classification performance using fewer genes than the traditional filter methods.

The analysis of published literature on some specific topic could improve the results on DNA microarray data. With microarray experiments, hundreds of genes can be identified as relevant to the studied phenomenon by means of feature selection approaches. The interpretation of these gene lists is challenging as, for a single gene, there can be hundreds or even thousands of articles pertaining to the gene's function. Text-mining can alleviate this complication by revealing the associations between the genes that are apparent from literature [83].

Unfortunately, current works are focused on keyword search and abstract evaluation that limit the extraction of biological results done in previous studies, and requires the researchers to further filter the results [67].

The interpretations of microarray results can be improved by using ontologies such as MESH or GO [104]. For example, GOEAST [168] is a web-based user friendly tool, which applies appropriate statistical methods to identify significantly enriched GO terms among a given list of genes extracted by gene expression analysis.

Clustering is usually considered as an unsupervised learning approach because no a priori knowledge is assumed at the beginning of the process. However, in the case of gene expression data, some prior knowledge is often available (i.e., some genes are known to be functional related). Thus, integrating such knowledge can improve the clustering results. In recent years, some semi-supervised clustering methods have been proposed so that user-provided constraints or sample labels can be included in the analysis. For example, in [137] the authors proposed a semi-supervised clustering method called GO fuzzy c-means, which enables the simultaneous use of biological knowledge and gene expression data. The method is based on the fuzzy c-means clustering algorithm and utilizes the Gene Ontology annotations as prior knowledge to guide the process of grouping functionally related genes. By following the approach of using prior biological knowledge for the fuzzy c-means algorithm, other clustering algorithms such as hierarchical and k-means can be adapted to use prior biological knowledge as well.

3

Document collection analysis

Text mining or knowledge discovery from text (KDT) was mentioned for the first time in [50]. It deals with machine supported analysis of texts. It exploits techniques from information retrieval, information extraction as well as natural language processing (NLP) and connects them with the algorithms and methods of KDD, data mining, machine learning and statistics.

Text mining and data mining techniques are very often the same or quite similar. For this reason also the specialized literature in this field, for example [51, 69, 100, 148], jointly describe text mining and data mining algorithms. Anyway text mining and data mining are different in their nature. One remarkable difference between them is in the inputs that they expect [148]. Data mining methods expect a highly structured format for data, necessitating extensive data preparation. So there is the need either to transform the original data or to receive the data in a highly structured format. On the other hand text mining methods usually work on collections of documents in which the content is human readable. Despite this difference, there are a lot of common factors and techniques. Thus, normally, similar procedures are selected, whereby text documents and not data in general are in focus of the analysis.

It is possible to define text mining referring to related research areas [69]. For each of these areas, it is possible to give a different definition of text mining, which is motivated by the specific perspective of the area:

- **Information Extraction**. The first approach assumes that text mining essentially corresponds to information extraction, the extraction of facts from texts. The task of information extraction naturally decomposes into a series of processing steps, typically including tokenization, sentence segmentation, part-of-speech assignment, and the identification of named entities, i.e. person names, location names and names

of organizations. At a higher level phrases and sentences are parsed, semantically interpreted and integrated. Finally, information as "incoming person name" is entered into a database.

- **Text data mining**. Text mining can be defined similarly to data mining. In fact, it can be also defined as the application of algorithms and methods from the fields machine learning and statistics to texts with the goal of finding useful patterns. For this purpose, it is necessary to preprocess the texts accordingly. Normally, text mining techniques exploit information extraction methods, natural language processing or some simple preprocessing steps in order to extract data from texts. Data mining algorithms can be then applied to the extracted data.

- **KDD Process**. Following the knowledge discovery process model, text mining is often defined in literature as a process with a series of partial steps. Some of these steps also involve information extraction as well as the use of data mining or statistical procedures. According to this definition, text mining can be seen as a set of techniques oriented to discover information in large collections of texts.

Current research in the area of text mining tackles problems of text representation, classification, clustering, information extraction, automatic summarization or the search for and modeling of hidden patterns. In this context the selection of characteristics and also the influence of domain knowledge and domain-specific procedures plays an important role. Therefore, an adaptation of the known data mining algorithms to text data is, usually, necessary.

Many commercial tools and research projects tackle the problem of managing the huge mass of publications contained in the text repositories. Such projects face the problem from different points of view and with different techniques. In this chapter we describe some of the different approaches, focusing on automatic text summarization, but also giving an overview of other ways of dealing with textual information. The first section provides an overview of repositories for medical and biological research publications. The following section describes the approaches based on search engines to retrieve general purpose or domain specific information from document collections. The final section analyzes different summarization techniques oriented to produce a condensed representation of the information stored in text data.

3.1 Document repositories

Nowadays there are a lot of repositories for medical and biological texts. Some widely known ones are: PubMed Central [14], Internet Scientific Publications [8], PLoS [12], Canadian Breast Cancer Research Alliance Open Access Archive [3], Bioline International [1], DSpace Istituto Superiore Sanità [5], Archivo Aperto di Documenti per la Medicina Sociale [6], Wiley InterScience [17]. Moreover, there is also a great variety of search engines that provide access to this information in several ways. In the following a brief overview of some public repositories is presented.

3.1.1 PubMed Central

PubMed Central [14] is a free digital archive of biomedical and life sciences journal literature at the U.S. National Institutes of Health (NIH), developed and managed by NIH's National Center for Biotechnology Information (NCBI) in the National Library of Medicine (NLM). It is part of the NLM efforts in preserving and maintaining unrestricted access to the electronic literature. NLM has done similar efforts for decades with the printed biomedical literature. PubMed Central is not a journal publisher but a digital library that offers the access to many journals.

Participation by publishers in PubMed Central (PMC) is voluntary, although editorial standards must be met. Journals are encouraged to deposit all their content (and not just research papers) in PMC so that the archive becomes a true digital counterpart to NLM's extensive collection of printed journals. In line with this objective, NLM is digitizing earlier print issues of many of the journals already in PMC. The access to the journal content is not immediate in every case since a journal may delay release of its full text in PMC for some period of time after publication.

One relevant peculiarity of PubMed Central is that it stores in a common format and in a single repository data from diverse sources. This is for sure an advantage since it simplifies the searching task that can be performed in the same way independently from the data source. Another important characteristic is that it also makes it possible to integrate the literature with a variety of other information resources such as sequence databases and other factual databases that are available to scientists, clinicians and everyone else interested in the life sciences.

PubMed Central has also a FTP service which may be used to download

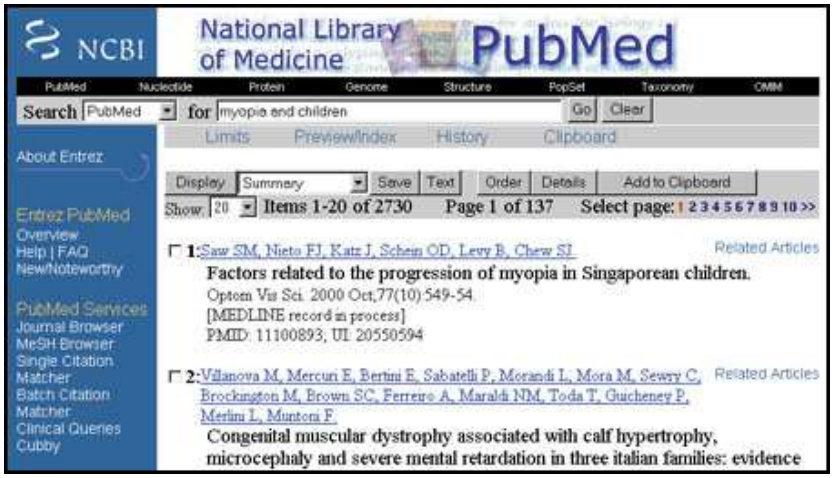


Figure 3.1: Search results with PubMed

the source files for any article in the PMC Open Access Subset. The source files for an article may include:

- A .nxml file, which is XML for the full text of the article.
- A PDF file of the article.
- Image files from the article, and graphics for display versions of mathematical equations or chemical schemes.
- Supplementary data, such as background research data or videos.

The data that can be downloaded through this service are particularly addressed to data mining analysis.

Together with PMC, the National Library of Medicine also offers PubMed. It is a free search engine for accessing databases of citations and abstracts of biomedical research articles. The search mask of PubMed can be seen in Figure 3.1. The core subject is medicine, and PubMed covers fields related to medicine, such as nursing and other allied health disciplines. It also provides full coverage of the related biomedical sciences, such as biochemistry and cell biology. NLM offers it at the National Institutes of Health as part of the Entrez information retrieval system.

3.1.2 PLoS

The Public Library of Science (PLoS) [12] is a nonprofit open-access scientific publishing project aimed at creating a library of open access journals and other scientific literature under an open content license. As of January 2008 it publishes PLoS Neglected Tropical Diseases, PLoS Biology, PLoS Medicine, PLoS Computational Biology, PLoS Genetics and PLoS Pathogens.

The history of the project helps in understanding its most important peculiarities. The Public Library of Science began in early 2001 as an on-line petition initiative by Patrick O. Brown, a biochemist at Stanford University and Michael Eisen, a computational biologist at the University of California, Berkeley and the Lawrence Berkeley National Laboratory. The petition called for all scientists to pledge that from September of 2001 they would discontinue submission of papers to journals which did not make the full-text of their papers available to all, free and unfettered, either immediately or after a delay of several months. Some now do this immediately, as open access journals, such as the BioMed Central stable of journals, or after a six-month period from publication (as what are now known as delayed open access journals) and some after 6 months or less, such as the Proceedings of the National Academy of Sciences.

Joined by Nobel-prize winner and former NIH-director Harold Varmus, the PLoS organizers next turned their attention to starting their own journal, along the lines of the UK-based BioMed Central which has been publishing open-access scientific papers in the biological sciences in journals such as Genome Biology and the Journal of Biology since late 1999.

As a publishing company, the Public Library of Science began full operation on October 13, 2003, with the publication of a peer reviewed print and on-line scientific journal, entitled PLoS Biology, and have since launched six more peer-reviewed journals. The PLoS journals are what they describe as “open access content”; all content is published under the Creative Commons “attribution” license (Lawrence Lessig, of Creative Commons, is also a member of the Advisory Board). The project states (quoting the Budapest Open Access Initiative) that: “The only constraint on reproduction and distribution, and the only role for copyright in this domain, should be to give authors control over the integrity of their work and the right to be properly acknowledged and cited”. To fund the journal, PLoS charges a publication fee to be paid by the author or the author’s employer or funder. In the United States, institutions such as the National Institutes of Health and the Howard Hughes Medical Institute have pledged that recipients of their grants will be

allocated funds to cover such author charges.

Differently from PubMed Central, PloS does not offer a service equivalent to the FTP one. It offers a search engine that interrogates the internal databases according to the keywords inserted by the user. This method is highly effective for the readers that want to retrieve a certain article. However, it is not very suitable as a starting point for text mining analysis.

3.2 Search Engines

Many research efforts have been devoted to automatically indexing the highly unstructured information contained in texts. Advanced search engines are frequently available on web sites that offer textual content and they often represent the front-end to access the information.

Is possible to make a coarse classification of the search engines based on the fact that they are "general purpose" or that they are specialized on a particular field or type of document.

From the point of view considered in this thesis, which is managing the information contained in text, they essentially have one intrinsic limitation. No matter whether they are general purpose or not, by nature they still require users to follow the hyperlinks, to read the documents and to locate the sentences that are more relevant for their information seeking goals. They rank the results, and provide a hyperlink to them, but not a description or a summary of the contents. For these reasons, they require a too relevant human intervention for scopes such as inferring knowledge and providing biological validation.

3.2.1 General purpose search engines

The most wide spread approach is the one of conventional "general purpose" information retrieval systems that include modern search engines. Their goal is to look for the documents that are mostly relevant for the keywords that a user inserted. Early search engines such as Alta Vista^{TM} mainly relied on techniques from traditional information retrieval such as maximizing the frequency of the keywords. However, this does not fit well for indexing the World Wide Web. In fact, with these search engines, webmasters could manipulate the rankings by changing the contents of their web pages. A person who wanted his web page to rank first for a search term could use

a "stuffing" technique, consisting in the repetition of a search term very often in its document. Even if search engines used techniques to find such manipulations, with simply content-based approaches the quality of search engine results became soon very poor.

A second generation of search engines came with the advent of Google™ [31]. These search engines use link-based approaches to determine the quality of documents. Even if still possible, ranking manipulation is more difficult with these techniques. Modern "general purpose" search engines use two different kinds of ranking factors [87]:

1. Query-dependent factors.
2. Query-independent factors.

Query-dependent are all ranking factors that are specific to a given query, while *query-independent* factors are attached to the documents, regardless of a given query. There are various kinds of query-dependent ranking factor. There are measures such as word documents frequency, the position of the query terms within the document or the Inverse Document Frequency, which were all used in traditional Information Retrieval. There are also measures more tied to HTML such as emphasis (that gives more importance to the terms with HTML tags like <b> or <i>) or anchor text (that gives higher scores when query terms appears in anchor text). Finally, some measures take into account the language of the document in relation to the language of the query or the "geographical" distance between the user and the document. They are not part of the classic information retrieval measures and they focus on finding the most relevant documents by comparing queries and documents.

The second group of measures used by search engines are query-independent factors. Such ranking factors are mainly used to to determine the quality of a given document. They are necessary due to the highly heterogeneous content of the World Wide Web that ranges from low quality to high-quality documents. The final goal of search engines is to provide the user with the highest possible quality and to omit low-quality documents.

Query-independent factors are used to determine the quality of documents regardless of a certain query. The most popular of these factors is PageRank [105]. It is the measure of link popularity used by the search engine Google ™. While early approaches to link popularity just counted the number of in-links to a page, PageRank and other link based ranking measures take into account the link popularity of the linking pages or measure the link popularity within a set of pages relevant to a given query.

Some search engines also count the number of clicks a document gets from the results pages and thereby count a measure of click popularity. Another query-independent factor considered by some search engines is the directory level of a given document, whereby documents on a higher (or a certain) level in the hierarchy are preferred. The document length can also be a ranking factor, because it is known that users prefer short documents in general, but not too short documents that consist of just a few words. Moreover, also the size of the website hosting the document can be used as a ranking factor. This factor assumes that a document on a larger website is more likely authoritative than another on a small website. Up-to-dateness of a document is another factor. It takes into account that for some queries, newer documents are more important than older ones. Finally, the file type can be used as a ranking factor as well. In fact, usually, search engines prefer regular HTML documents over PDF or Word files because the user can see these files in his browser without opening another program or plug-in.

3.2.2 Domain specific search engines

Some projects such as PubMed are specifically tailored for specific domains. Such engines perform searches in specialized repositories or using specific keywords only. Most of these project exploits the techniques used by general purpose search engines and apply them on a narrower domain. A project that introduces some novel concepts and that is addressed to biomedical articles is the iHop (Information Hyperlinked over Proteins) project [66].

It exploits the fact that biology and medicine are in the exceptional position of having a natural underlying topology. In fact, in most biological models and hypotheses, genes and proteins serve as basic units of information. PubMed and any other resource of biological knowledge can thus be seen as networks of concurrent genes and proteins that include relationships ranging from direct physical interaction to less direct associations with phenotypic observations and pathologies. The iHop project exploits this network concept as a mean of structuring and linking the biomedical literature and making it navigable in a way that is similar to the Internet. In other words, it clusters and hyperlinks the biomedical literature based on genes and proteins. This is an effective approach to rapidly have access to the documents that deals with a certain gene or group of genes.

The iHop project is very useful in web searching due to the usage of MeSH headings. MeSH [9] stands for Medical Subject Headings. It is the National Library of Medicine's controlled vocabulary thesaurus. It consists

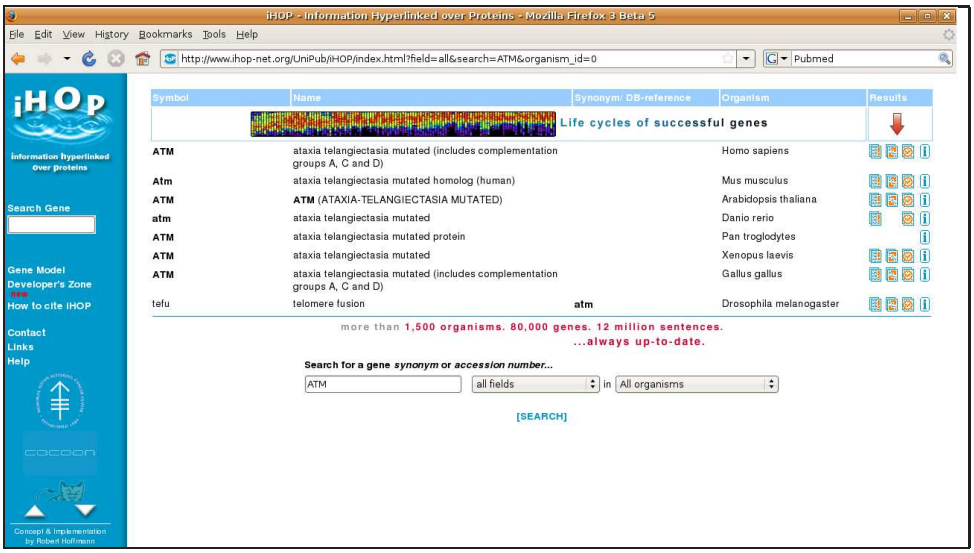


Figure 3.2: Output of a search with iHop

of sets of terms naming descriptors in a hierarchical structure that permits searching at various levels of specificity. MeSH terminology is used to assign topic headings (indexing) to every article that is entered into Medline. The usage of Mesh terminology significantly improves the quality of the search. In fact, searching by keyword can result in retrieving articles in which the term is incidental. In summary, a keyword search is likely to retrieve a high number of irrelevant citations, and miss many very useful ones. An example of search with iHop is in Figure 3.2.

Another interesting feature is that iHop also exploits a combination of natural language parsing and gene identifiers to extract declarative statements from PubMed abstracts and MeSH headings to provide a rapid overview of gene properties. However, for biological validation and knowledge inference, the fact of working only on abstracts and MeSH headings is a critical limitation. In fact, working only on this narrow fields, iHop normally extracts statements which provide a definition of the gene/protein, but is likely to miss statements that deals with protein-protein interactions (these pieces of information are written in the bodies of the scientific publications). Moreover, a physician's evaluation of randomly-selected papers [120] shows that the author's abstract does not always reflect the entire contents of the full-text and the crucial biological information. For these reasons working only on the abstracts is a critical limitation.

Considering its strengths and this limitation iHop is a very powerful do-

main specific search engine, but it still requires a lot of human intervention to fulfill the needs of inferring knowledge and providing biological validation.

3.3 Summarization based projects

All the projects and tools based on search engines have the intrinsic limitation of requiring an active human intervention to select the most relevant parts of the text. The approaches based on text summarization overcome this limitation. They provide to the user a more concise and compact version of the document. Thus, they better fulfill the need of reducing and organizing the huge amount of unstructured information contained in the texts. From the point of view of the thesis work, that is the one of knowledge inference and biological validation, this is an added value. In this section we present different classes of projects related to text summarization. We analyze "classical" summarizer, summarizer specifically tailored for the biomedical domain and approaches that exploits semantic information and ontology knowledge.

The reference point for summarization related issue is the Text Analysis Conference (TAC) [16] that is the most important international conference in this field. Most of the projects (not all) that we describe have been recently presented to this conference.

The summarization related projects have many advantages but a critical common limitation. The sentences extracted by all these summarizers are suitable to provide a human readable synthesis and to emphasize the main ideas of an article or of a group of articles. However, such sentences give only a general description of the major topics in the texts. The summarizers instead often discard the most domain-specific sentences (e.g., the ones listing genes and their interactions). The content of these sentences is very important for biological validation and knowledge inference.

3.3.1 Generic text summarizers

Projects based on "classical" text summarizers are the most wide spread. The summarizers are classified in single-document summarizers and multi-document summarizers depending on the fact that they work on one document or on a group of documents. Single-text summarizers, such as [115], are the eldest ones whereas, recently, the vast majority of the projects [88, 124, 151] are based on multi-document summarizers.

We focus on multi-document summarizers since they are the most recent branch in this field and since the thesis work is related to a multi-document summarizer. We briefly describe a project related to multi-document summarization by cluster/profile relevance and redundancy removal [124]. It presents many relevant peculiarities that are common to other projects in the same branch. Moreover, the techniques that it uses and the structure of the system are not an exception, but they represent a common way of operating.

The project takes as starting point a general purpose single document summarization system and then implements components to support multi-document summarization functionality. The basic single document system is a pipeline of linguistic and statistical components. The system supports "generic", query-based, and multi-language (English, Finish, Swedish, Latvian, Lithuanian) summarization. The inputs to the process are a single document, a compression rate specified as a percentage of the sentences of the document or as a number of words to extract, and an optional query. The document is automatically transformed by a text structure analyzer into a representation containing the "text" of the original input and a number of annotation sets. Linguistic analysis or summarization components add new information to the document in the form of new annotations or document features. Some summarization components compute numerical features for the purpose of sentence scoring. These features are combined in order to produce the final sentence score. Sentences are output until the compression rate is reached.

In a multi-document situation, there is the need of measuring not only the content of each sentence in relation to the other sentences in the same document but also across documents. In this situation, the system takes in consideration the relationship each sentence has to the set of documents (cluster) that constitutes the input to the process. A centroid representation of the cluster of related documents is constructed. It is a vector of pairs of terms and weights, where the weight w_i of term i in the centroid is obtained through:

$$w_i = \frac{\sum_{k=1}^{n} w_{i,k}}{n} \tag{3.1}$$

where $w_{i,k}$ is the weight of term i in document k. A cosine similarity value is computed between each sentence in the document set and the centroid. Each sentence in the cluster is scored using the features:

- Sentence cluster similarity

- Sentence lead-document similarity
- Absolute document position

These values are combined with appropriate weights to produce the sentences final score which is used to rank them. An ad hoc similarity metric is devoted to the detection of similar information in texts (redundancy detection).

For what concerns content evaluation, summaries were assessed by human assessors using model (reference) summaries. This is still a quite common approach in content evaluation. However, it is considered a weak method because different assessors may produce a different evaluation and also the same assessor may give a diverse judgment in different periods. Moreover, this evaluation technique is considered expensive since it requires a lot of human intervention. For all these reasons, automatic techniques are usually preferred.

3.3.2 Text summarizers in the Biomedical Domain

Recent studies explored the possibility of tailoring the summarization techniques to exploit the specificity of certain research domains. This is a quite new and promising branch of summarization. Currently, there is no complete project that specifically refers to the biological domain, that is the one analyzed in our thesis work. However, there is an approach called BioChain [120, 119] that deals with the biomedical domain. Even if the project is still in its early stages and it does not refer to the biological domain that we examined, we briefly describe it anyway since it takes into account many concepts useful to build a domain specific summarizer.

BioChain is an effort to summarize individual oncology clinical trial study publications into a few sentences to provide an indicative summary to medical practitioners or researchers. The summary is expected to allow the reader to gain a quick sense of what the clinical study has found. The work is being done as a joint effort between the Drexel University College of Information Science and Technology and College of Medicine. The College of Medicine has provided a database of approximately 1,200 oncology clinical trial documents that have been manually selected, evaluated and summarized. So far the approaches for summarizing single documents were developed. The ultimate goal of the project is summarizing documents into a single integrated summary in order to reduce the information overload burden on practicing

physicians. The part of the system that exploits the single-document summarizer is consolidated whereas the multi-document part is still in progress.

Lexical chaining is a crucial part of this project. It is a method for determining lexical cohesion among terms in text. Lexical cohesion is a property of text that causes a discourse segment to "hang together" as a unit. This property is important in computational text understanding for two major reasons:

1. Providing term ambiguity resolution
2. Providing information for determining the meaning of text.

Lexical chaining is useful for determining the "aboutness" of a discourse segment, without fully understanding the discourse. A basic assumption is the text must explicitly contain semantically related terms identifying the main concept. Lexical chains are an intermediate representation of source text, and are not used directly by an end-user. They are, instead, applied internally in some application. In the BioChain project the application is text summarization for document understanding.

Lexical chains that involves terms are quite common in text summarization. The added value of the BioChain project is that it uses concept chaining rather than lexical chaining. Concept chaining operates at the level of concepts rather than terms. The specificity of the biomedical domain makes this shift possible. In fact, the Unified Medical Language System (UMLS) provides tools for mapping biomedical text into concepts and semantic types. This semantic mapping allows to chain together related concepts based on each concept's semantic type. The UMLS semantic network types are used as the head of chains, and the chains are composed of concept instances generated from noun phrases in the biomedical text.

The usage of concept chains is important for summarization since it must identify sentences most likely capture the main ideas of text. BioChain uses the sentence extraction method to generate a summary. The top-n sentences in text are extracted, using n as an upper bound on the number of sentences to select. Top sentences are identified by sorting chains into ascending order based on their score, and then identifying the most frequent concepts within each chain. The score of the chain is obtained multiplying the frequency of the more frequent concept by the number of distinct concepts. Then sentences that include the most frequent concepts are extracted to make the summary. Multiple concepts having the same frequency count are considered equal, and sentences from each concept are extracted.

This project has interesting aspects, but it has the limitation of working on clinical trial documents which are a very sectorial set of inputs. In fact, these documents share a common vocabulary and they are also very often structured in the same way. Moreover, the project heavily relies on the UMLS tools that are specific for this domain.

3.3.3 Semantic and Ontology based summarizers

In this Section we describe semantic analysis and ontologies together because they work at the same abstraction level and, at least in the domain of summarization, they are converging.

The vast majority of current summarizers base their sentence ranking functions on traditional Information Retrieval techniques such as Word Documents Frequency or the Inverse Document Frequency. However, semantic based approach have been studied too. Such works are based on Latent Semantic Analysis (LSA) [86].

LSA is an approach for extracting and representing the contextual-usage meaning of words by statistical computations applied to a large corpus of text. The underlying idea is that the totality of information about all the word contexts in which a given word does and does not appear provides a set of mutual constraints that largely determines the similarity of meaning of words and set of words to each other.

LSA is based on singular value decomposition (SVD), a mathematical matrix decomposition technique closely akin to the factor analysis applicable to databases approaching the volume of relevant language experienced by people. Word and discourse meaning representations derived by LSA are capable of simulating a variety of human cognitive phenomena, ranging from acquisition of recognition vocabulary to sentence-word semantic priming and judgments of essay quality. LSA differs from other statistical approaches mainly because it uses as its initial data not just the summed contiguous pairwise (or tuple-wise) co-occurrences of words, but the detailed patterns of occurrences of words over very large numbers of local meaning-bearing contexts, such as sentences or paragraphs, treated as unitary wholes.

Totally independent research projects in the last years were focused on Ontologies [58]. In the context of computer and information sciences, an ontology defines a set of representational primitives with which to model a domain of knowledge or discourse. The representational primitives are typically classes (or sets), attributes (or properties), and relationships (or relations

among class members). The definitions of the representational primitives include information about their meaning and constraints on their logically consistent application. In the context of database systems, ontology can be viewed as a level of abstraction of data models, analogous to hierarchical and relational models, but intended for modeling knowledge about individuals, their attributes, and their relationships to other individuals. Due to their independence from lower level data models, ontologies are used for integrating heterogeneous databases, enabling interoperability among disparate systems, and specifying interfaces to independent, knowledge-based services.

Very often, due to their expressive power, ontologies are said to work at the "semantic" level. So, quite naturally, in the domain of summarization semantic analysis and ontologies are converging. A recent research project is focused on a semantic free-text summarization system that uses ontology knowledge [143]. The used summarization technique is knowledge-rich and user query-based. The original document is represented with a semantically connected concept network. Then a subset of sentences is selected from the original document as its summary. The approach is totally term-based. This means that the system recognizes and processes only terms defined in Wordnet for general documents (UMLS for medical documents) and ignore all other words. The approach was presented at the Text Analysis Conference (TAC) in 2007 and although very promising it is in its early stages. Moreover, many details were not presented and remain unclear.

This summarization technique seeks to obtain the quality and the readability of a summary written by human beings. However, the produced summaries usually contain descriptive sentences which give a generic description of the most important topics of a document. Very often, such sentences miss the more technical and specific concepts that are mostly needed for knowledge inference and biological validation. This is a critical limitation for the objectives of the thesis.

3.3.4 Clustering-Summarization Integration and Data Representation

This last class of projects covers two important aspects:

1. Integration of clustering and summarization to improve the quality of the summary [161].
2. Sentence representation using graphs [114, 161] both for internal evaluations and output presentation.

The two aspects appear to be related since they often occur together. More precisely, the second problem is tackled only by projects that manage the first one.

The combination of clustering an summarization techniques is required when heterogeneous collections of texts are analyzed. In fact, normally, such inputs have multiple topics. For this reason text summarization does not yield high-quality summary without document clustering. On the other hand, document clustering is not very useful for users to understand a set of documents if the explanation for document categorization or the summaries for each document cluster is not provided. In other words, document clustering and text summarization are complementary. This is the main motivation for which projects such as CSUGAR (Clustering and SUmmarization with GrAphical Representation for documents) [161] involve the integration between document clustering and text summarization.

The other added value of this class of works is the graphical representation of the sentences obtained by using Scale Free Graph Clustering [161]. Networks and graphs have been extensively used to represent a heterogeneous range of phenomena. Traditionally, those networks were interpreted with random graph theory, in which nodes are randomly distributed and two nodes are connected randomly and uniformly (i.e. Gaussian distribution). However, researchers have recently observed that the graph connecting words in English text follows a Scale-Free Network instead of the random graph theory. Thus, the graphical representation of documents belongs to a highly heterogeneous family of scale-free networks.

A scale-free network is a network whose degree distribution follows a power law, at least asymptotically. That is, the fraction $P(k)$ of nodes in the network having k connections to other nodes goes for large values of k as:

$$P(k) \sim k^{\gamma} \tag{3.2}$$

where γ is a constant whose value is typically in the range $2 < \gamma < 3$.

As with all systems characterized by a power law distribution, the most notable characteristic in a scale-free network is the relative commonness of vertexes with a degree that greatly exceeds the average. The highest-degree nodes are often called “hubs”, and are thought to serve specific purposes in their networks, although this depends greatly on the domain.

The Scale Free Graph Clustering (SFGC) algorithms are based on the scale-free nature of the graphical representation of documents. In fact, in this

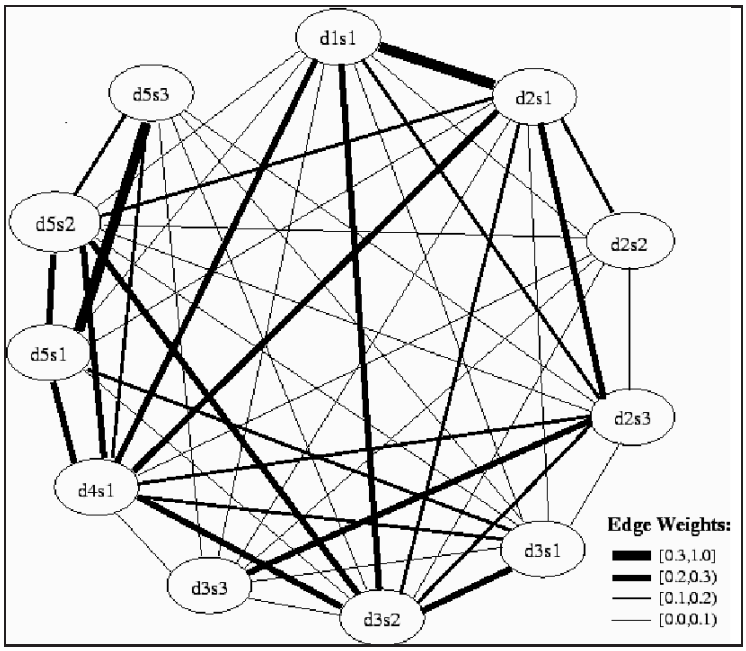


Figure 3.3: Graphical representation of documents

problem is possible to observe the existence of a few hub vertexes (concepts). SFGC starts detecting k hub vertex sets (HVSs) as the centroids of k graph clusters and then assigns the remaining vertexes to graph clusters based on the relationships between the remaining objects and k hub vertex sets.

The result is a graph that is then used to select significant text content for the summary. In order to identify "important" nodes (vertexes) in networks (or graphs), the degree centrality based approaches are used. They assume that nodes that have more relationships with others are more likely to be regarded as important in the network because they can directly relate to more other nodes. In other words, the more relationships the nodes in the network have, the more important they are. An example of graph representation of a document is in Figure 3.3.

The proper management of clustering-summarization integration is, from our point of view, crucial because it is the only way of addressing multi-topic inputs. Moreover, the graphical representation is an interesting approach to manage the information contained in text. However, also this class of works has the limitation described for all the summarization approaches since it tends to build up "generic" summaries discarding very useful information. Another limitation is they may be applied only in specific conditions. These

systems, even if multi-topic require a certain consistency between the analyzed documents. Thus such approaches only deal with texts all related to a common background. For example, the experimental sets presented in [161] were obtained by using "MajorTopic" tag along with the disease-related MeSH terms as queries to Medline. This way of operating limits the scope of such works that, for this reason, are not suitable for biological journal articles that, normally, deal with several diseases.

4

Gene Mask representation

Genome wide expression analysis with DNA microarray technology has become a fundamental tool in genomic research [56, 75, 139]. An important goal of bioinformatics is the development of algorithms that can accurately analyze microarray data sets. Since microarray data are noisy and are highly dependent on the technology employed in the gene expression measurement, we need to define some basic concepts to deal with microarray data characteristics. The aim of this chapter is to provide a new representation, named *gene mask*, which captures the capability of a gene in distinguishing the sample classes (i.e., classification power). Thus, a preprocessing analysis on the expression values is exploited in order to define the expression interval in which a gene may be measured.

In general, microarray data E are represented in the form of a gene expression matrix, in which each row represents a gene and each column represents a sample. For each sample, the expression level of all the genes under consideration is measured. Element e_{is} in E is the measurement of the expression level of gene i for sample s, where $i = 1, \ldots, N$ and $s = 1, ..., S$. Each sample is also characterized by a class label, representing the clinical situation of the patient or tissue being analyzed. The domain of class labels is characterized by C different values and label k_s of sample s takes a single value in this domain.

$$E = \begin{bmatrix} e_{11} & e_{12} & \ldots & e_{1S} \\ e_{21} & e_{22} & \ldots & e_{2S} \\ \ldots & \ldots & \ldots & \ldots \\ e_{N1} & e_{N2} & \ldots & e_{NS} \end{bmatrix} \tag{4.1}$$

According to this representation we define:

- **Core expression interval.** Definition of the range of expression values for a given gene in a given class. Two different approaches are exploited in the core expression interval definition.
- **Gene mask.** Definition of the *gene mask* as representatives of gene classification power, where the classification power is the capability of a gene in discriminating the sample classes. The gene mask is generated by analyzing the gene core expression intervals.

These definitions will be used in the following chapters to identify the genes which have a high discriminative power among classes in order to improve classification accuracy and to evaluate the similarity among groups of genes under different experimental conditions (i.e., sample classes).

4.1 Core expression interval definition

The core expression interval of a gene in a class represents the range of gene expression values taken by samples of the considered class. Different approaches can be applied to compute the core expression interval for each gene in each class. For example, the MAD estimator first computes the median of the data and defines the set of absolute values of differences between each data value and the median. Then, the median of this set is computed. By multiplying this value by 1.4826, the MAD unbiased estimate of the standard deviation for Gaussian data is obtained. The MAD estimator smooths the effect of values far from the median value, independently of their density. We considered two different approaches described in the following.

The first approach consider the minimum and the maximum values measured for each gene in each class. By considering the gene expression matrix in (4.1), with a class label for each sample, the class expression intervals for each gene are defined. Let i be an arbitrary gene with S samples belonging to C classes. For each gene i we define C class expression intervals (one for each class), each of which contains the entire expression value range for the corresponding class. The class expression interval for gene i and class k is expressed in the form:

$$I_{i,k} = [min_{i,k}, max_{i,k}] \tag{4.2}$$

where $min_{i,k}$ and $max_{i,k}$ are the minimum and the maximum gene expression values for class k.

Since microarray data may be noisy, we propose a density based approach to reduce the effect of outliers on the core expression interval definition, the *Weighted Mean Deviation*.

4.1.1 Weighted Mean Deviation

Weighted Mean Deviation (or WMD) is a variation of the MAD estimator. In WMD the mean is replaced by the weighted mean and the standard deviation by the weighted standard deviation. The weights are computed by means of a density estimation. A higher weight is assigned to expression values with many neighbors belonging to the same class and a lower weight to isolated values.

Consider an arbitrary sample s belonging to class k and its expression value e_{is} for an arbitrary gene i. Let the expression values be independent and identically distributed (i.i.d) random variables and $\sigma_{i,k}$ be the standard deviation for the expression values of gene i in class k. The density weight w_{is} measures, for a given expression value e_{is}, the number of expression values of samples of the same class which belong to the interval $\pm\sigma_{i,k}$ centered in e_{is}.

The density weight for the expression value e_{is} for a gene i and a sample s belonging to class k is defined as

$$w_{is} = \sum_{m=1, m\neq s}^{S} \delta_{im} \tag{4.3}$$

where δ_{im} is a function defined as

$$\delta_{im} = \begin{cases} 1 & \text{if sample } m \text{ belongs to class } k \wedge \\ & e_{im} \in [e_{is} - \sigma_{i,k}; e_{is} + \sigma_{i,k}] \\ 0 & \text{otherwise} \end{cases} \tag{4.4}$$

If an expression value is characterized by many neighboring values belonging to the same class, its density weight is higher. For example, in Figure 4.1 the expression values of an arbitrary gene i with four samples of class 1 (labeled as w, x, y, and z) and seven of class 2 (labeled as a, b, c, d, e, f, and g) are shown. For sample a, the expression level (denoted as e_{ia} in Figure 4.1) is characterized by a density weight w_{ia} equal to 0, because for gene i there are no other expression values of class 2 in the interval $e_{ia} \pm \sigma_{i,2}$ (represented by a curly bracket). For sample b, the expression value (e_{ib}) is characterized

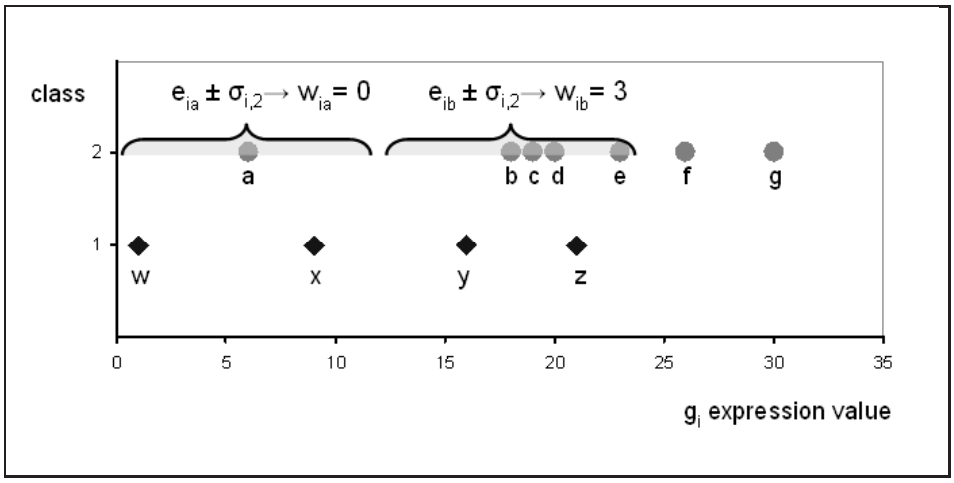


Figure 4.1: Gene i: Density weight computation for samples a and b.

instead by a density weight w_{ib} equal to 3, because three other samples of class 2 belong to the interval $e_{ib} \pm \sigma_{i,2}$.

The core expression interval of an arbitrary gene i in class k is given by

$$I_{i,k} = \hat{\mu}_{i,k} \pm (2 \cdot \hat{\sigma}_{i,k}) \tag{4.5}$$

where the weighted mean $\hat{\mu}_{i,k}$ and the weighted standard deviation $\hat{\sigma}_{i,k}$ are based on the density weights and are computed as follows[1].

The weighted mean $\hat{\mu}_{i,k}$ is defined as

$$\hat{\mu}_{i,k} = \frac{1}{W_{i,k}} \sum_{s=1}^{S} \delta_{is} \cdot w_{is} \cdot e_{is} \tag{4.6}$$

where δ_{is} is a function defined as

$$\delta_{is} = \begin{cases} 1 & \text{if sample } s \text{ belongs to class } k \\ 0 & \text{otherwise} \end{cases} \tag{4.7}$$

and $W_{i,k}$ is the sum of density weights for gene i in class k (i.e., $\sum_{s=1}^{S} \delta_{is} \cdot w_{is}$).

[1]The term $2 * \hat{\sigma}_{i,k}$ covers about 95% of expression values. Higher (or lower) values of the weighted standard deviation multiplicative factor may increase (or decrease) the number of included values.

The weighted standard deviation $\hat{\sigma}_{i,k}$ is given by

$$\hat{\sigma}_{i,k} = \sqrt{\frac{1}{W_{i,k}} \sum_{s=1}^{S} \delta_{is} \cdot w_{is} \cdot (e_{is} - \hat{\mu}_{i,k})^2} \tag{4.8}$$

The core expression interval is less affected by outliers, as shown in the upper part of Figure 4.2. Since the first sample of class 2 (i.e., sample a) has a low density weight (equal to zero), its value provides no contribution to the weighted mean and standard deviation computation.

4.2 Gene Mask computation

For each gene we define a gene mask, which is an array of S bits, where S is the number of samples. It represents the capability of the gene to classify correctly each sample, i.e., its classification power. Consider an arbitrary gene i. Bit s of its mask is set to 1 if the corresponding expression value e_{is} belongs to the core expression interval of a single class, otherwise it is set to 0. Formally, given two arbitrary classes $c_1, c_2 \in C = \{1, \ldots, k\}$, bit s of gene mask i is computed as follows.

$$mask_{is} = \begin{cases} 1 & \text{if } (e_{is} \in I_{i,c_1}) \ \wedge \ \nexists c_2 \neq c_1 \mid e_{is} \in I_{i,c_2} \\ 0 & \text{otherwise} \end{cases} \tag{4.9}$$

A sample might not belong to any core expression interval (i.e., it is an outlier). In this case, the value of the corresponding bit is set to 0 according to (4.9). Figure 4.2 shows the gene mask associated to an arbitrary gene i after the computation of its core expression intervals $I_{i,1}$ and $I_{i,2}$.

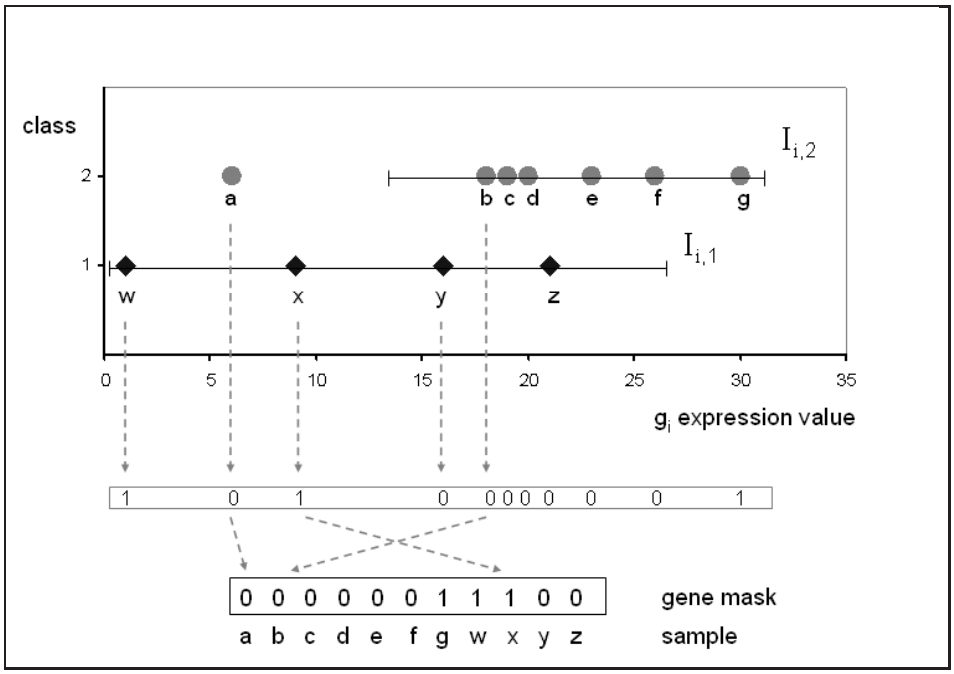


Figure 4.2: Gene i: Core expression interval computation for classes 1 and 2 and gene mask computation.

5

Minimum number of genes

A fundamental problem in microarray analysis is to identify relevant genes from large amounts of expression data. Feature selection aims at identifying a subset of features for building robust learning models. Since only a small number of genes among tens of thousands show strong correlation with the targeted disease, some works address the problem of defining which is the appropriate number of genes to select [108]. However, finding the optimal number of genes, is a difficult goal. While an excessively conservative estimate of the number of relevant genes may cause an information loss, an excessively liberal estimate may increase the noise in the resulting dataset.

This chapter describes a method which automatically selects the minimum number of genes to reach the best classification accuracy on the training set. Moreover, the genes belonging to the minimum subset can be used as genetic markers for further biomedical investigations. Thus, according to the gene mask representation introduced in Chapter 4, two strategy to select the minimum set of genes which maximize the training samples coverage are exploited. The performance of our method has been tested on publicly available datasets and experimentally compared to other feature selection algorithms.

5.1 Method

Our method aims at identifying the minimum number of genes useful to correctly classify the samples in the training set.

The approach is based on the following main phases.

- **Class interval definition.** For each gene the class interval of expression values is defined according to (4.2). Since the aim of this phase is

to evaluate the capability of a gene to distinguish sample classes, the WMD approach is not exploited.

- **Gene mask computation.** A gene mask is computed for each gene according to (4.9). The gene mask shows which training samples the gene can unambiguously assign to the correct class. It is a string of 0s and 1s, generated by analyzing the overlaps among the class expression intervals (i.e., the range of expression values of samples belonging to the same class) for each gene.

- **Minimum gene subset selection.** The minimum number of genes needed to provide the best training set sample coverage is selected by analyzing the gene masks and exploiting the overlap scores. Different searching algorithms (i.e., greedy and set covering) are exploited.

The minimum subset of genes for classifying the maximum set of samples in the training set (i.e., for providing the best sample coverage) and avoiding redundant information is defined by analyzing the gene masks.

Let $\mathcal{G}$ be a set of genes. We define a *global mask* as the logic OR between all the gene masks belonging to genes in $\mathcal{G}$. The objective is the definition of the minimum set of genes $\mathcal{G}$ that holds enough discriminating power to classify the maximum number of samples in the training set. Thus, given the gene mask of each gene, we search for the global mask with the maximum number of ones. To this aim, we propose two different techniques: a greedy approach and a set covering approach. The two approaches are experimentally compared in Section 5.2.

5.1.1 Greedy approach

The greedy approach identifies at each step the gene with the best complementary gene mask with respect to the current global mask. Thus, it adds at each step the information for classifying most currently uncovered samples.

The pseudo-code of the Greedy approach is reported in Algorithm 1. It takes as input the set of gene masks ($\mathcal{M}$), the set of scores ($\mathcal{OS}$) and produces as output the minimum subset of genes ($\mathcal{G}$). The scores associated to each gene can be computed by different feature selection techniques. In Section 5.2 the variance of expression values is exploited as score. The first step is initializing $\mathcal{G}$ at $\emptyset$ (line 2), the candidate set ($\mathcal{C}$) at $\emptyset$ (line 3), and the global mask with all zeros (line 4). Then the following steps are iteratively performed.

Algorithm 1 Minimum gene subset - Greedy approach

Input: set $\mathcal{M}$ of all the $mask_i$, set $\mathcal{OS}$ of score os_i for each gene i
Output: set $\mathcal{G}$ of genes
1: */*Initialization*/*
2: $\mathcal{G} = \emptyset$
3: $C = \emptyset$ */*candidate gene set at each iteration*/*
4: *global_mask* = all_zeros() */*vector with only 0s*/*
5: */*Control if the global mask contains only 1s*/*
6: **while** not global_mask_all_ones() **do**
7: */*Determine the candidate set of genes with most ones*/*
8: C = max_ones_genes()
9: **if** C != $\emptyset$ **then**
10: */*Select the candidate with the best score (e.g., the minimum)*/*
11: $c = C[1]$
12: **for all** j in $C[2:]$ **do**
13: **if** OS_j is better OS_c **then**
14: $c = j$
15: **end if**
16: **end for**
17: */*Update sets and global_mask*/*
18: $\mathcal{G} = \mathcal{G} + c$
19: $global\_mask = global\_mask$ OR $mask_c$
20: $\mathcal{M} = \mathcal{M} - mask_c$
21: */*Update the masks belongs to $\mathcal{M}$*/*
22: **for all** $mask_i$ in $\mathcal{M}$ **do**
23: $mask_i = mask_i$ AND $\overline{global\_mask}$
24: **end for**
25: **else**
26: break
27: **end if**
28: **end while**
29: **return** $\mathcal{G}$

1. The gene mask with the highest number of bits set to 1 is chosen (line 8). If more than one gene mask exists, the one associated to the gene with the best score is selected (lines 9-16). The best score depends on the range values produced by the technique exploited. For example, if the variance is used as score the gene with the highest variance is selected.

2. The selected gene is added to set $\mathcal{G}$ (line 18) and the global mask is updated by performing the logical OR between the gene mask and the global mask (line 19).

3. The gene masks of the remaining genes (gene mask set $\mathcal{M}$, line 20) are updated by performing the logical AND with the negated global mask (lines 21-24). In this way, only the ones corresponding to the classification of still uncovered samples are considered.

4. If the global mask has no zeros (line 6) or the remaining genes have no ones (line 9), the procedure ends.

5.1.2 Set covering approach

The set covering approach considers the set of gene masks as a matrix of $N \times S$ bits and performs the following three steps.

1. *Sample reduction.* Each sample (i.e., column) that contains all 0 or 1 over the N gene masks is removed, because it is uninformative for the searching procedure.

2. *Gene reduction.* Each gene (i.e., row) whose gene mask is a subsequence of another gene mask is removed from the matrix. If two or more genes are characterized by the same gene mask, only the gene with the best score is kept in the matrix. At the end of these two steps a reduced matrix is obtained.

3. *Reduced matrix evaluation.* The reduced matrix is evaluated by an optimization procedure that searches the minimum set of rows necessary to cover the binary matrix. Since it is a min-max problem, it can be converted to the following linear programming problem.

$$\begin{aligned} min \quad & \sum_{i=1}^{N} g_i \\ & \sum_{i=1}^{N} mask_{ij} \cdot g_i \geq 1, j = 1, ..., S \\ & g_i \in \{0, 1\} \end{aligned}$$

   The branch and bound implementation provided by the Symphony library [116] has been exploited to find the optimum solution.

At the end of this phase, the minimum set of genes required to provide the best sample coverage of the training set is defined. The genes in the minimum subset are ordered by decreasing number of 1s in the gene mask.

5.2 Experimental results

We validated our method by comparison with other feature selection techniques on public gene expression datasets. Classification accuracy is used as the performance metric for evaluation, while biological relevance of the selected genes is discussed in Section 5.2.3.

Dataset	*Samples*	*Features*	*Classes*
Brain1	90	5920	5
Brain2	60	10364	4
SRBCT	83	2308	2
DLBCL	77	5469	2

Table 5.1: Dataset characteristics

5.2.1 Experimental setting

We evaluated the performance of our algorithm on four microarray datasets, publicly available on [131]. Two of them are multi-class (Brain1 and Brain2), and the other two are binary (SRBCT and DLBCL). Table 5.1 summarizes their characteristics. We compared the performance of our method with the following supervised feature selection methods implemented in RankGene software [134]: Information Gain (IG), Twoing Rule (TR), Sum Minority (SM), Max Minority (MM), Gini Index (GI), Sum of Variance (SV).

For each of these methods we performed experiments with a number of selected features in the range from 1 to 12 to allow the comparison with the number of features selected by our method. We exploited the libSVM classifier [33] with a 4-fold cross validation, similarly to [97]. Samples were randomly partitioned in a stratified manner into four folds of equal size. Three folds become the training set and the fourth the test set. Classification is repeated for four times, each time with a different fold as test set. To avoid the selection bias, feature selection algorithms were applied only to the training set, and accuracy was computed by applying the classifier to the test set. We repeated the cross-validation 50 times, changing the seed for the split generation.

5.2.2 Classification accuracy

As shown in Table 5.2, the gene reduction step significantly reduces the number of considered features. In the second column the average value of remaining features over 50 repetitions of the 4-fold cross validation is reported. The reduction rate (i.e. the number of discarded features over the total number of features) in the third column highlights that there are between 60% and 90% genes with a gene mask which is a subsequence of another gene mask. This affects the number of genes selected by our algorithm. For example, Brain2 and DLBCL, which have the highest reduction rates, also have the

minimum number of selected genes, as shown in the fourth column of the table. This average value is always slightly less than the average number of features selected by the greedy approach reported in the fifth column of the table.

The average accuracy over 50 repetitions of the 4-fold cross validation is reported in Figures 5.1, 5.2, 5.3 and 5.4, where the performance of the mask covering algorithm and the six RankGene methods are compared. Also the accuracy of the greedy approach is reported. For the mask covering and the greedy algorithm a single point is shown on the graph, because these methods automatically select the best number of features for the dataset. Instead, for the other methods, the behavior varying the gene number may be analyzed, as they require the gene number as input parameter.

For the DLBCL dataset (Figure 5.1) our method needs only 3 or 4 genes, depending on the number of samples in the split (the diagram reports the average gene number value over all repetitions), to reach an accuracy higher than 88%, while other methods do not reach such accuracy even when using 10 or more genes. Also the greedy approach performs slightly worse than our method, even if the number of selected genes is higher. For the Brain2 dataset (Figure 5.2) our method reaches a high accuracy (about 63%) with less than 5 genes, while the other methods reach a lower accuracy also with a higher number of genes.

For the SRBCT dataset (Figure 5.3) our method selects the best small subset of genes. However classification accuracy may be further improved by increasing the cardinality of the gene set. Finally, in the Brain1 dataset (Figure 5.4) our method is less effective in detecting a very small set of high quality features and shows a behavior closer to other feature selection techniques. These differences in the accuracy levels depend on the dataset characteristics.

The genes selected by each method are mostly different from the ones selected by the other methods. The percentage of common genes in about 10-20%. It means that there are a lot of genes which classify same samples and each method select different genes to reach a similar accuracy.

We performed the Student's t-test to asses the statistical significance of the results. We compared our method with each of the RankGene methods by setting as gene cardinality the integer number nearest to the mean value of genes selected by our method. We obtained a p-value less than 0.01 in 3 over 4 datasets (Brain2, SRBCT, DLBCL). In the case of Brain1 the p-value value is less than 0.05 for some methods.

Dataset	*Remaining features*	*Reduction rate*	*Mask Covering*	*Greedy*
Brain1	1874	68%	6.76	7.80
Brain2	847	92%	4.62	5.05
SRBCT	660	71%	5.28	5.75
DLBCL	1245	77%	3.50	3.79

Table 5.2: Reduction rate and average number of selected features

5.2.3 Biological discussion

We investigated the biological meaning of our selected genes. For the DLBCL dataset, the genes selected by the mask covering algorithm include the T-cell chemoattractant SLC and the DNA replication licensing factor CDC47 homolog, which are known to be related to lymphoma [129]. Furthermore, the genes selected by the greedy approach include the DNA replication licensing factor CDC47 homolog, the Cancellous bone osteoblast mRNA for GS3955 and the Chloride channel (putative) 2163bp, which are listed as relevant for lymphoma [129].

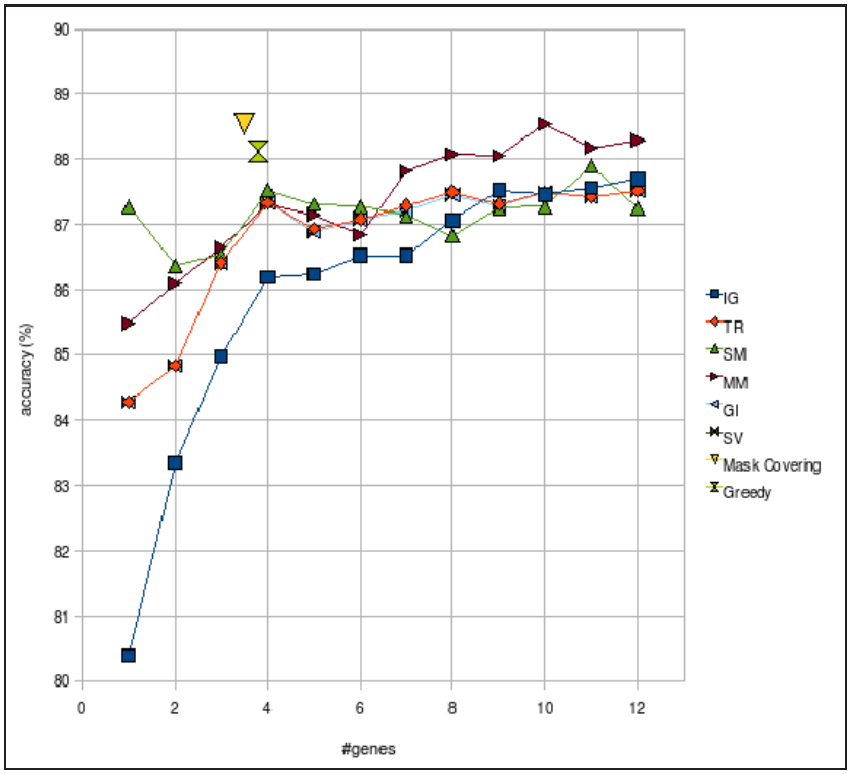


Figure 5.1: Mean classification accuracy of six RankGene methods, Mask Covering and Greedy on DLBCL dataset.

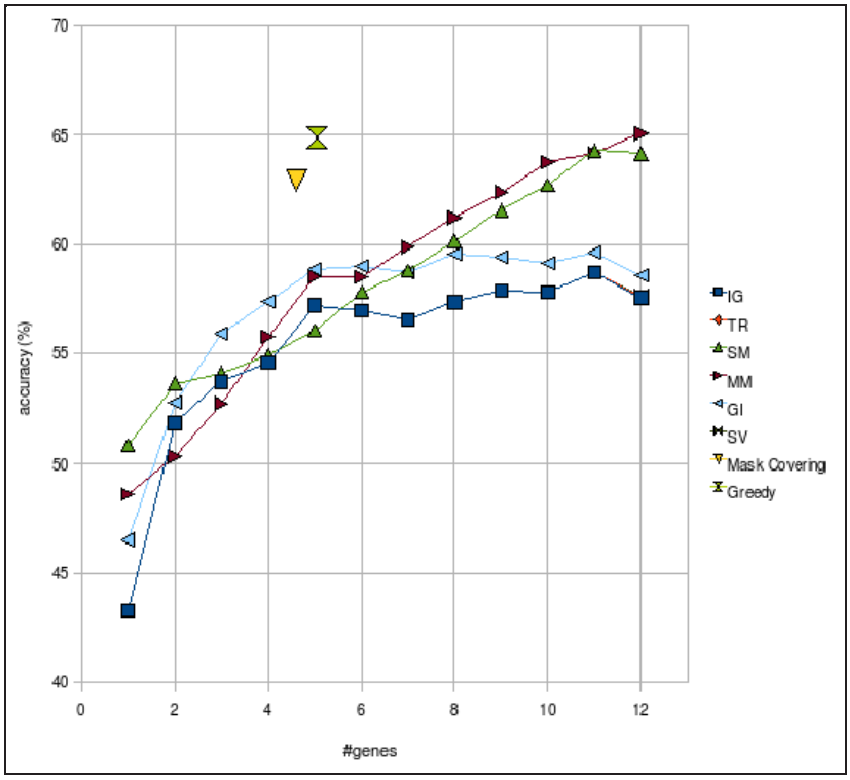


Figure 5.2: Mean classification accuracy of six RankGene methods, Mask Covering and Greedy on Brain2 dataset.

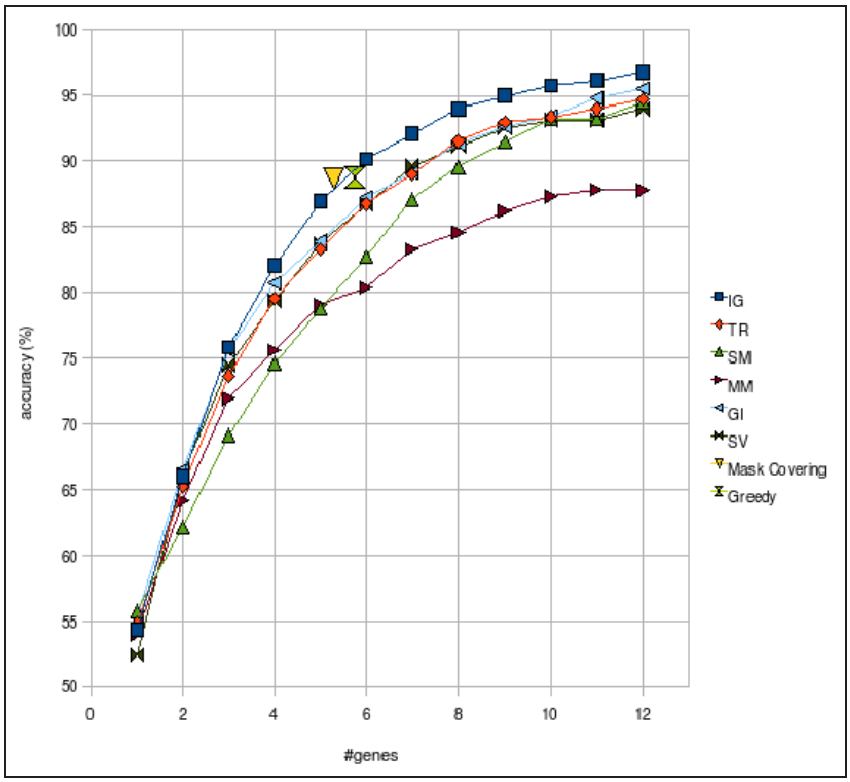


Figure 5.3: Mean classification accuracy of six RankGene methods, Mask Covering and Greedy on SRBCT dataset.

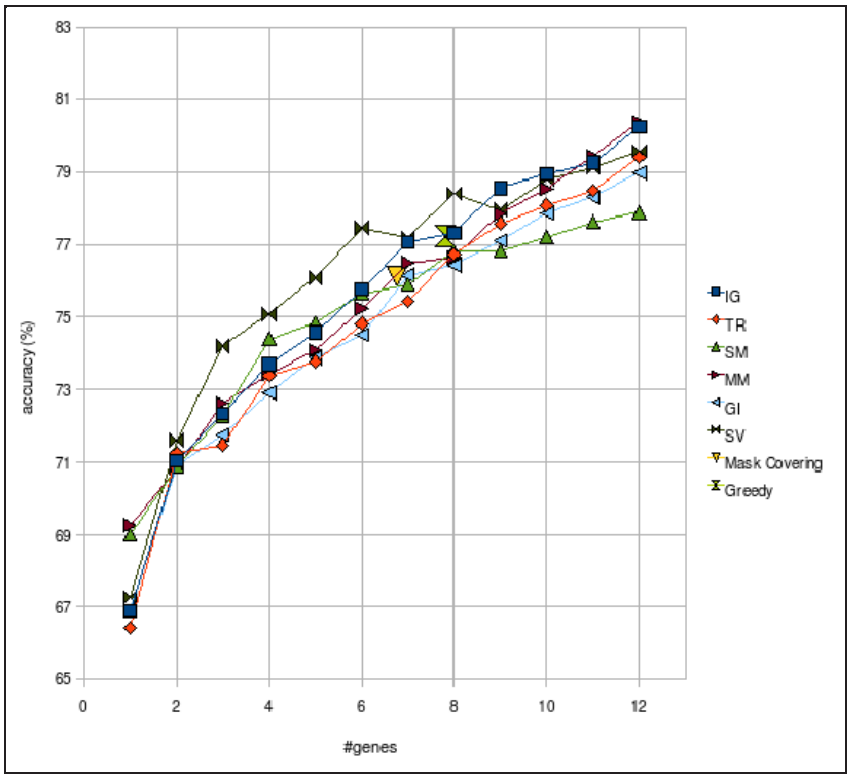


Figure 5.4: Mean classification accuracy of six RankGene methods, Mask Covering and Greedy on Brain1 dataset.

6

MaskedPainter feature selection

Finding the optimal number of features is a challenging problem, as it is a trade off between information loss when pruning excessively and noise increase when pruning is too weak. The approach described in the previous chapter was addressed to identify a minimum subset of genes to improve classification accuracy on training data. Thus, the approach stops the research of best features when the coverage of training samples is complete. The aim of this chapter is to provide a filter feature selection method that, based on the previous approach, selects the most discriminative genes according to their expression value distribution. Moreover, the method allows the user to select the number of retrieved features in order to reach the best classification accuracy and study a larger set of high relevant genes for the target disease.

In this chapter we present the *MaskedPainter* feature selection method. It adopts the filter feature selection model, hence it is independent of the adopted classifier. The MaskedPainter method provides two main contributions: (i) it identifies the minimum number of genes that yield the best coverage of the training data (i.e. that maximize the correct assignment of the training samples to the corresponding class), and (ii) it ranks the genes according to a quality score. A density based technique is exploited in the quality score computation to smooth the effect of outliers. The minimum gene subset and the top ranked genes are then combined to obtain the final feature set, thus defining a general set of features aimed at obtaining a high classification accuracy.

The name "MaskedPainter" originates from the Painter's algorithm used in computer graphics. The Painter's algorithm assigns a priority to the objects to be painted that is based on their overlaps. Similarly, the MaskedPainter assigns a priority to genes that is based on the overlaps in their expression intervals. The term masked is due to the fact that the information carried by a gene is represented in a particular format, named gene

mask.

We validated our method on different microarray datasets. We mainly focused on multi-category datasets, because classification problems with multiple classes are generally more difficult than binary ones and give a more realistic assessment of the proposed methods [73]. We compared the MaskedPainter performance with different feature selection techniques by measuring the classification accuracy provided by a classifier taking as input the selected features. In almost all circumstances, the MaskedPainter yields statistically significant higher accuracy than the other techniques. All experiments have been performed for different gene set cardinalities and with different classifiers. Finally, the biological relevance of the genes selected by the proposed approach has been assessed.

6.1 Method

The MaskedPainter method is based on the following idea. Certain genes can identify samples belonging to a class, because their expression interval in that class is not overlapped with the expression intervals of other classes (i.e., all the samples for which the expression value of the gene is in a given range belong to a single class). For example, Figure 6.1(a) shows the expression values of a gene with 12 samples belonging to 3 different classes. The same information can be represented as shown in Figure 6.1(b). With the latter representation, it is easy to see that the gene is relevant for class 3, because the expression values of this class are concentrated in a small range, which is different from the range of expression values associated with the other classes. Instead, the same gene is not useful to distinguish between class 1 and class 2, because the values for such classes have mostly overlapping ranges.

The MaskedPainter initially characterizes each gene by means of a *gene mask*, which represents the gene's capability of unambiguously assigning training samples to the correct class. Next, the method assigns to each gene two values that are then combined in the ranking phase: the *overlap score* and the *dominant class*. The overlap score is a quality index that describes the overlap degree of the expression intervals for different classes. Genes with less overlapping intervals are more important because they can unambiguously assign the samples to the correct class. The dominant class of a gene is the class to which the majority of samples without overlapping expression intervals belong.

By exploiting these elements, the MaskedPainter defines (i) the minimum

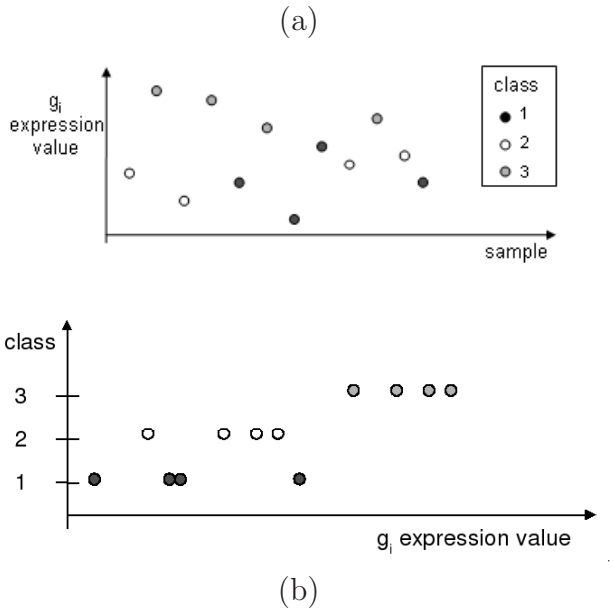


Figure 6.1: Two different representations of a gene with 12 samples belonging to 3 classes.

set of genes needed to provide the best sample coverage on training data and (ii) a sort of genes by dominant class and increasing value of overlap score. The final gene set is obtained by combining the minimum gene subset and the top ranked genes in the sort.

The building blocks of the MaskedPainter approach are presented in Figure 6.2. The approach is based on the following main phases.

- **Gene mask computation.** A gene mask is computed for each gene according to (4.9). The gene mask shows which training samples the gene can unambiguously assign to the correct class. It is a string of 0s and 1s, generated by analyzing the overlaps among the class expression intervals (i.e., the range of expression values of samples belonging to the same class) for each gene. The class expression interval exploited in the gene mask computation is defined by (4.2).

- **Overlap score computation and dominant class assignment.** An overlap score is assigned to each gene. It assesses the overlap degree among its core expression intervals. The core expression interval of a gene, separately defined for each class, is the expression interval obtained by smoothing the effect of outliers. To compute the core ex-

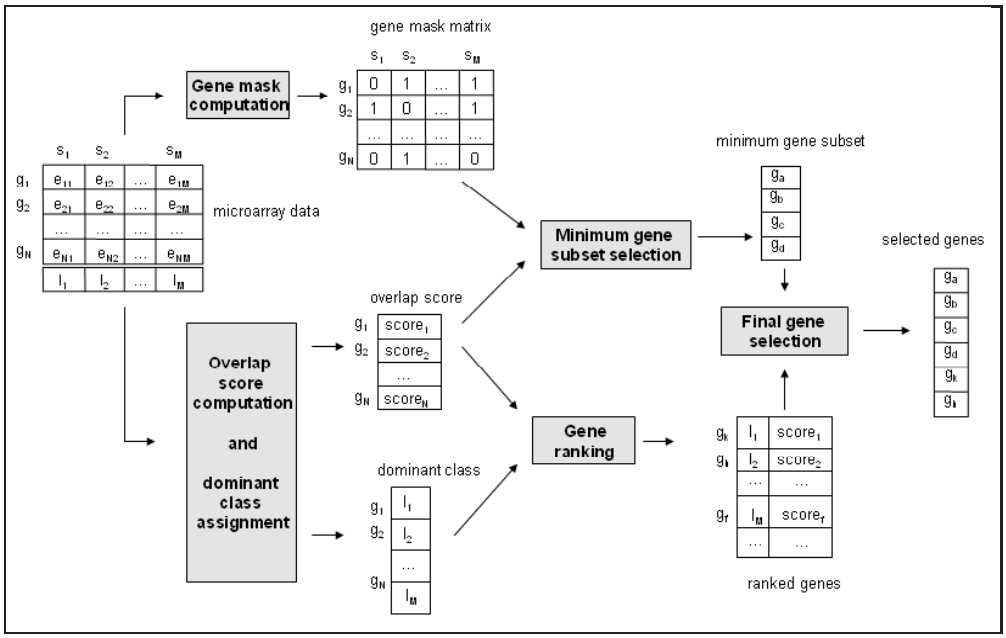


Figure 6.2: Building blocks of the MaskedPainter method.

pression intervals, the density based approach WMD is exploited (see Section 4.1.1 for details). A dominant class, i.e., the best distinguished class, is also assigned to each gene. This information is exploited to reduce redundancy when genes are selected.

- **Minimum gene subset selection.** The minimum number of genes needed to provide the best training set sample coverage is selected by analyzing the gene masks and exploiting the overlap scores. Different searching algorithms (i.e., greedy and set covering) are exploited. The approaches are described in Section 5.1. The overlap score of each gene is used as discriminant score in the cases discussed in the previous chapter.

- **Gene ranking.** Genes that do not belong to the minimum subset are ranked according to increasing values of overlap score, separately for each dominant class. The final gene rank is composed by selecting the topmost gene from each dominant class in a round robin fashion.

- **Final gene selection.** Selected top ranked genes are added to the minimum gene subset, thus providing the final gene set.

In the following sections the phases which are not covered by the previous chapters are discussed in details.

6.1.1 Overlap score computation and dominant class assignment

An overlap score is assigned to each gene, depending on the amount of overlapping expression intervals among classes. Differently from the gene mask, which is based on the min-max expression intervals (as discussed in Section 4.1), the overlap score aims at modeling the discrimination power of genes and needs to handle noise and outliers to avoid overfitting. Hence, to better model the expected values in an unseen test set, the overlap score computation is based on core expression intervals computed by WMD approach.

The dominant class, i.e., the class with the highest number of samples in non-overlapping intervals, is also assigned to each gene.

This phase can be further divided into three steps: (i) core expression interval definition, (ii) overlap score computation, and (iii) dominant class assignment.

The details of the core expression interval definition can be founded in Section 4.1.1. The other steps are described in the following sections.

Overlap score computation

For each gene we define an overlap score (denoted as *os* in the following) that measures the degree of overlap among core expression intervals. Since overlapping intervals may lead to misclassifications due to insufficient discriminative power of the considered gene, the overlap score is exploited for ranking genes. The score is higher for less important genes with many overlapping intervals among different classes. On the contrary, lower scores denote genes with higher discriminating power, because they have few overlaps among their intervals.

The *os* depends on the following characteristics of the gene expression values

1. the number of samples associated to different classes which are in the same range,

2. the number of overlapping classes,

3. the overlapping interval length.

We compute the overlap score os_i for each gene i. To ease readability, we will omit the i subscript in the following formulas.

We define the total expression interval of a gene as the range given by the minimum and maximum among its core expression interval boundaries. We denote such interval as W, and its amplitude as $|W|$. For example, in Figure 6.3, the total expression interval of a gene with samples belonging to three classes (and thus with three core expression intervals) is shown. We divide W in subintervals, where each subinterval is characterized by a different set of overlapping classes with respect to the adjacent subintervals. More specifically, the subinterval w_t is defined as the interval delimited by two consecutive extremes of core expression intervals, as shown in Figure 6.3. The amplitude of subinterval w_t is denoted as $|w_t|$.

The idea of the overlap score is to assign higher scores to genes that are characterized by more overlaps among expression intervals of different classes. The score is based on both the number of samples of different classes that belong to the same subinterval and the amplitude of the subinterval itself. Thus, subintervals with larger class overlaps provide a higher contribution to the os. According to this intuition, the overlap score for an arbitrary gene is defined as follows.

$$os = \sum_{t=1}^{T} c_t \frac{m_t}{M} \frac{|w_t|}{|W|} \tag{6.1}$$

where T is the number of subintervals, c_t is the number of classes which overlap in subinterval t, m_t is the number of samples expressed in subinterval t, and M is the total number of samples. Subintervals covered by a single class (e.g., w_1 in Figure 6.3) provide no contribution to the overlap score, because the number of overlapping classes is 0. In the case of subintervals without values (i.e., gaps such as w_4 in Figure 6.3), the number of overlapping classes is 0. Thus, also in this case, no contribution is added to the overlap score.

Consider the example in Figure 6.3. The total expression interval of the gene is divided into five subintervals and the c_t components (number of overlapping classes in each subinterval) take the following values: $c_1 = 0$, $c_2 = 2$, $c_3 = 0$, $c_4 = 0$, $c_5 = 0$.

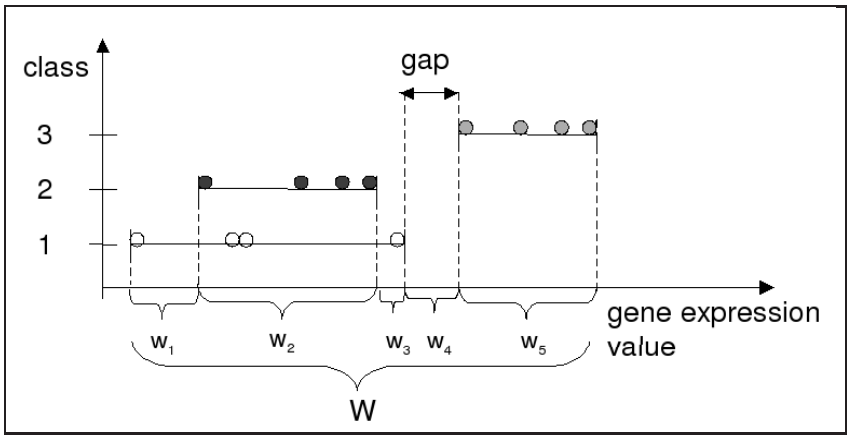


Figure 6.3: Subintervals for the computation of the overlap score (*os*) of a gene.

The *os* value ranges from 0, when there is no overlap among class intervals, to C (i.e., the number of classes), when all intervals are completely overlapped. For example, in Figure 6.4, two illustrative cases for a binary problem are reported. Figure 6.4(a) shows a gene that correctly distinguishes two classes, because its core expression intervals are not overlapped. The *os* of this gene is equal to 0, because $c_1 = 0$, $c_2 = 0$, and $c_3 = 0$. Instead, Figure 6.4(b) shows a gene unable to distinguish two classes, because the expression intervals associated with the two classes are almost completely overlapped. In this case, the overlap score is close to 2.

Dominant class assignment

Once the overlap score is computed, we associate each gene to the class it distinguishes best, i.e., to its *dominant class*. To this aim, we consider the subintervals where expressed samples belong to a simple class and evaluate the number of samples for the considered class in these subintervals. The class with the highest number of samples is the dominant class of the gene. The gene is assigned to the class with the highest number of samples to take into account the a priori probability of the classes in improving classification accuracy.

For example, in Figure 6.4(b) the gene dominant class is class 1, because its samples for the two non-overlapping subintervals w_1 and w_3 are labeled with class 1. Instead, in Figure 6.4(a), since both classes have completely

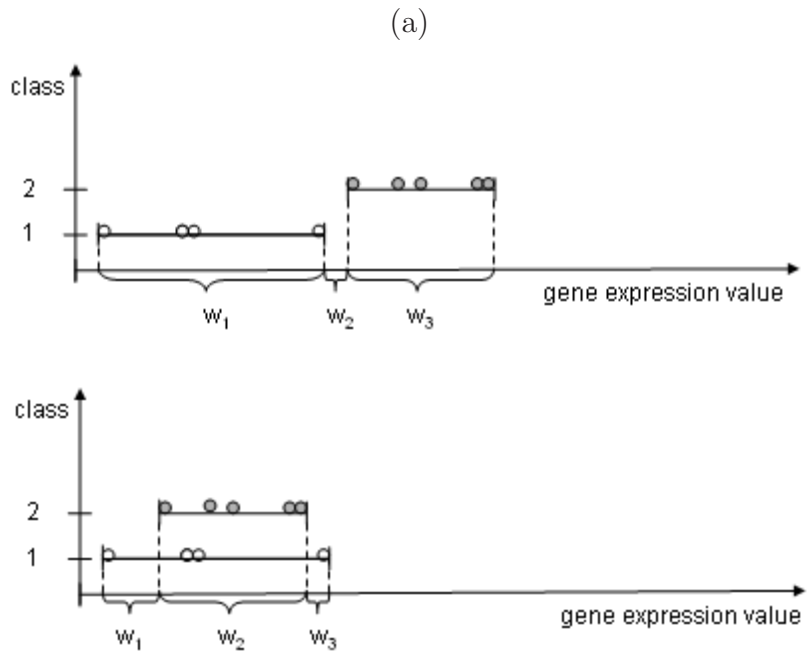


Figure 6.4: Overlap score computation for two core expression intervals: (a) a gene with an overlap score equal to 0 and (b) a gene with an overlap score close to 2.

non-overlapping intervals, the gene dominant class is class 2, according to the number of samples.

Associating a gene with the class it distinguishes best will allow us to balance the number of selected genes per class (see Section 6.1.3). A gene could exceptionally have more than one candidate dominant class, i.e., two or more classes with the same number of samples in non-overlapping intervals. A thorough experimental evaluation showed that this situation is very rare, because it reported no cases of multiple candidate dominant classes among the first 20 selected genes. Hence, to guarantee a deterministic behavior to the algorithm, we simply selected the dominant class as the first in the lexicographical order of the class labels.

6.1.2 Gene ranking

The gene rank is defined by considering both the overlap score and the dominant class. Genes that do not belong to the minimum subset are ranked by increasing value of overlap score separately for each dominant class. The final rank is composed by selecting the topmost gene from each dominant class rank in a round-robin fashion. Classes are considered in lexicographical order.

A feature selection method considering only a simplified version of the overlap score was presented in [20]. The overlap score alone ranks high genes with few overlaps, without considering the class they distinguish best. Hence, high-ranked genes may all classify samples belonging to the same class, thus biasing gene selection (typically by disregarding less populated classes). The round-robin gene selection by dominant class allows mitigating this effect.

6.1.3 Final gene selection

The minimum gene subset includes the minimal number of genes that provide the best sample coverage on the training set, ordered by decreasing number of 1s in the gene mask. However, a larger set of genes may be either beneficial to improve the classification accuracy on unseen (test) data, or directly requested by the user. In this case, the minimum gene subset is extended by including the top k ranked genes in the gene ranking, where k is set by the user. Observe that genes in the minimum subset are inserted in the final gene set independently of their overlap score, because these features allow the classifier to cover the maximum set of training samples. The effect of this choice is discussed in Sections 6.2.3 and 6.2.4.

6.1.4 Example

As a summarizing example, consider the set of genes represented in Figure 6.5(a). Each gene is associated with its overlap score (os), its gene mask (string of 0 and 1), and its dominant class (dc). For example, gene $g1$ has a mask of 0100101 (i.e., it classifies unambiguously the second, the fifth and the seventh samples), an overlap score of 0.11, and its dominant class is class 1. For convenience, genes are pre-sorted by increasing overlap score value.

The first gene selected by the greedy method in the minimum subset is $g4$, because it is characterized by the highest number of bits set to 1

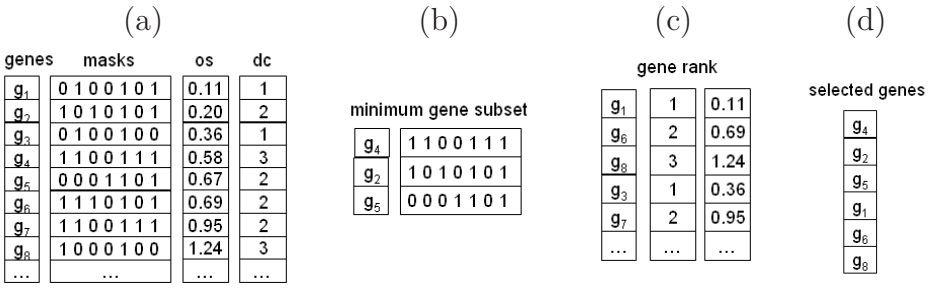


Figure 6.5: An example of the MaskedPainter method: (a) genes with their mask, overlap score, and dominant class; (b) minimum gene subset obtained by applying the greedy algorithm; (c) gene ranked by dominant class and overlap score; (d) selected genes at the end of the process.

(the same as $g6$ and $g7$) and the lowest overlap score. Then, genes with the best complementary masks are $g2$, $g5$, and $g6$, which all have the same number of bits set to 1. Again, $g2$ is selected because of its lower overlap score. Eventually, the only gene with a complementary mask, which is $g5$, is chosen. In this case the minimum number of genes is three. In Figure 6.5(b) the genes in the minimum gene subset are reported.

The remaining genes are divided by dominant class and sorted by ascending overlap score. The gene rank is composed by selecting the topmost gene from each dominant class in a round robin fashion (e.g., g_1 for class 1, g_6 for class 2, g_8 for class 3, g_3 for class 1, etc..) as shown in Figure 6.5(c). Suppose that six genes are required by the user for its biological investigation. Then, the three top ranked genes are added to the three genes of the minimum gene subset. The final gene set is shown in Figure 6.5(d).

6.2 Experimental results

We validated the MaskedPainter method by comparison with other feature selection techniques on public gene expression datasets. Classification accuracy is used as the performance metric for evaluation, while biological relevance of the selected genes is discussed in Section 6.3. We performed a large set of experiments addressing the following issues.

1. **Classification accuracy.** The accuracies yielded by MaskedPainter and several other feature selection approaches on seven public datasets

are compared.

2. **Cardinality of the selected feature set.** The impact on classification accuracy of different numbers of selected genes (from 2 to 20) is analyzed.

3. **Minimum gene subset definition.** The two proposed subset search techniques are compared by considering both accuracy and computational cost. Their performance is also evaluated against the fixed size subset.

4. **Classifier bias.** The effect of the peculiarities of different classification techniques on the gene set selected by MaskedPainter has been analyzed by comparing classification experiments performed with three different classifiers.

We also analyzed the computational cost of our approach. We compared the time required by each approach to extract a high number of features (i.e., 1000 features) from the considered datasets. The MaskedPainter algorithm proved to be as efficient as the competing feature selection methods. In particular, on a Pentium 4 at 3.2 GHz with 2 GByte of RAM, the time required to extract the top 1000 genes on any complete dataset is in the order of few seconds (e.g., less than 1 second on the Alon dataset, 3 seconds on the Brain2 dataset) and very similar to the time required by the other methods.

The experiments addressing each issue are reported in the following subsections.

6.2.1 Experimental setting

We validated our feature selection approach on 7 multicategory microarray datasets, publicly available on [131], [149], and [19]. Table 6.1 summarizes the characteristics of the datasets. Five are characterized by 3 to 9 classes, while two are bi-class datasets. Most contain between 60 and 90 samples, whereas one has 34 samples. The number of features ranges from 2 thousands to more than 10 thousands.

For each dataset we selected the best subset of genes according to the following 6 feature selection methods available in RankGene [134], besides our approach.

Dataset	*Samples*	*Genes*	*Classes*
Alon	62	2000	2
Brain1	90	5920	5
Brain2	50	10367	4
Leukemia	72	5327	3
Srbct	83	2308	4
Tumor9	60	5727	9
Welsh	34	7129	2

Table 6.1: Dataset characteristics on which MaskedPainter method was applied.

1. Information Gain (IG)
2. Twoing Rule (TR)
3. Sum Minority (SM)
4. Max Minority (MM)
5. Gini Index (GI)
6. Sum of Variance (SV)

These feature selection methods are widely used in machine learning [89]. Furthermore, they are used as comparative methods in many feature selection studies on microarray data [123, 70]. The experimental design exploits 50 repetitions of 4-fold stratified cross validation for each parameter set (i.e., given number of features, feature selection algorithm, classifier, and dataset), changing the split seed for each repetition. The average classification accuracy on the 50 repetitions is then computed. Similar experimental designs have been applied in [46, 89, 169]. Feature selection algorithms have been applied only on the training set to avoid selection bias. The statistical significance of the results has been assessed by computing the Student's t-test for each set of repetitions. In the reported tables, statistically relevant values with respect to us (i.e., p-value ≤ 0.05) are followed by a * sign, while best absolute values for each row are in bold.

All experiments have been performed by using small sets of features (from 2 to 20) to focus on the capability of the selected features to improve the classification performance. Using large sets of features allows the classifier to compensate for possible feature selection shortcomings by automatically pruning or giving low weights to the least relevant features.

6.2.2 Classification accuracy

We computed the classification accuracy provided by the MaskedPainter approach and by the six feature selection techniques reported in Section 6.2.1 with different cardinalities of the selected feature set. The experiments have been performed on all datasets in Table 6.1 with the J48 decision tree classifier [152] and the greedy subset search. The decision tree classifier is less capable to compensate for possible feature selection shortcomings by weighting the most/least relevant features (see Section 6.2.5). Hence, it allowed us to focus more effectively on the actual contribution of the feature selection. Different choices for these settings are discussed in the following subsections.

Tables 6.2, 6.3 and 6.4 show the classification accuracy yielded by the 7 feature selection methods on the Alon, Leukemia, and Srbct datasets. The results obtained on the other four datasets are reported in Appendix A. Each row reports the accuracy for a specific cardinality of the selected feature set (reported in Column 1). The average accuracy value, the maximum value and the standard deviation for each method are reported in the last three rows[1].

The MaskedPainter (MP) approach provides a very good accuracy on all datasets. In particular, on the Alon, Brain1, Brain2, Leukemia, and Welsh datasets, the accuracy is statistically better than all other feature selection techniques. On the Tumor9 dataset, the MP method shows a performance comparable with the best techniques (IG and SV). Eventually, on the Srbct dataset it is outperformed by the SV technique for larger sets of features (18 and 20). However, its overall average performance is statistically better than all other methods.

Averaging the improvement in accuracy with respect to the second best method on all cardinalities of the feature set leads to a +5.65% on the Welsh dataset, which is the highest average accuracy improvement.

6.2.3 Cardinality of the selected feature set

We analyzed the behavior of the MaskedPainter approach when varying the cardinality of the selected feature set. The average improvement of the proposed approach over the second best method is computed across all datasets,

[1]The max row never has the * (statistical significance), because the maximum value can be obtained by different methods for different numbers of genes.

#	MP	IG	TR	SM	MM	GI	SV
2	**78.94**	74.20*	74.68*	74.84*	75.81*	74.68*	74.68*
4	**76.95**	73.88*	73.24*	74.01*	75.23*	73.24*	73.24*
6	**77.37**	73.73*	73.75*	74.01*	75.11*	73.75*	73.75*
8	**77.05**	73.79*	73.68*	74.38*	74.83*	73.68*	73.68*
10	**76.85**	73.94*	73.77*	74.15*	74.99*	73.77*	73.77*
12	**76.02**	74.07*	73.99*	74.22*	74.12*	73.99*	73.99*
14	**76.15**	73.39*	73.80*	74.24*	74.31*	73.80*	73.80*
16	**75.26**	73.07*	73.19*	73.75*	74.11	73.19*	73.19*
18	**75.65**	73.00*	73.05*	73.50*	75.23	73.05*	73.05*
20	**75.63**	73.13*	73.08*	73.25*	75.29	73.08*	73.08*
avg	**76.59**	73.62*	73.63*	74.03*	74.90*	73.63*	73.63*
max	**78.94**	74.20	74.68	74.84	75.81	74.68	74.68
dev	1.03	0.42	0.48	0.43	0.53	0.48	0.48

Table 6.2: Accuracy yielded by the J48 classifier on the Alon dataset.

#	MP	IG	TR	SM	MM	GI	SV
2	**82.72**	81.67	80.00*	77.83*	78.25*	81.89	82.47
4	**86.50**	84.36*	81.75*	83.72*	82.50*	84.44*	85.78
6	**86.69**	85.17*	84.06*	85.44*	83.81*	85.42*	85.53
8	**86.44**	85.53	85.25	85.53	83.78*	86.14	85.06*
10	**86.86**	85.39*	85.22*	85.89	84.42*	85.75*	84.94*
12	**86.83**	85.14*	85.14*	85.69*	85.56*	85.56*	85.11*
14	**86.72**	84.97*	84.92*	85.11*	85.42*	85.25*	85.50*
16	**86.58**	84.92*	84.89*	85.11*	85.28*	85.11*	85.69
18	**86.67**	84.69*	84.72*	84.97*	85.03*	84.94*	86.17
20	**87.22**	84.86*	84.86*	85.03*	85.36*	84.97*	86.44
avg	**86.32**	84.67*	84.08*	84.43*	83.94*	84.95*	85.27*
max	**87.22**	85.53	85.25	85.89	85.56	86.14	86.44
dev	1.21	1.05	1.68	2.27	2.11	1.11	1.04

Table 6.3: Accuracy yielded by the J48 classifier on the Leukemia dataset.

#	MP	IG	TR	SM	MM	GI	SV
2	**71.50**	65.37*	63.76*	59.51*	62.41*	65.79*	63.63*
4	**81.73**	75.60*	72.97*	69.23*	69.89*	74.57*	74.00*
6	**81.92**	78.18*	75.17*	75.06*	72.72*	75.96*	78.05*
8	**82.07**	78.94*	76.75*	76.63*	75.18*	77.32*	80.61*
10	**82.07**	79.52*	78.06*	78.21*	77.18*	78.29*	81.02
12	**82.09**	80.63*	78.99*	78.68*	79.02*	79.64*	80.93*
14	**81.48**	80.85	79.80*	78.37*	80.75	80.54*	81.23
16	81.07	81.11	80.48	78.10*	81.13	81.20	**82.10**
18	81.16	81.18	81.01	78.23*	81.71	81.51	**82.71***
20	80.61	81.06	81.25	78.28*	82.48*	81.79*	**82.78***
avg	**80.57**	78.24*	76.82*	75.03*	76.25*	77.66*	78.71*
max	82.09	81.18	81.25	78.68	82.48	81.79	**82.78**
dev	3.06	4.60	5.04	5.85	6.06	4.59	5.60

Table 6.4: Accuracy yielded by the J48 classifier on the Srbct dataset.

separately for cardinalities of the feature set ranging in the interval 2-20. Table 6.5 shows the obtained results.

The MaskedPainter approach yields the highest improvements for low numbers of selected features, when the quality of the selected features more significantly affects classifier performance. Since the first few selected features typically belong to the minimum gene subset (see Section 6.2.4), these results highlight the quality of this small subset with respect to the features selected by all other methods. For increasing cardinality of the selected feature set the performance difference decreases, but the MaskedPainter algorithm still yields higher accuracy.

Furthermore, the second best feature selection algorithm is not always the same for all datasets. Hence, our approach can self adapt to the dataset characteristics better than the other methods, whose performance is more affected by the data distribution.

6.2.4 Minimum gene subset

To evaluate the effectiveness of the minimum gene subset we compared (a) the classification accuracy and execution time of the greedy and the set covering techniques, and (b) the accuracy provided by the minimum gene subset and the maximum accuracy obtained by the MaskedPainter method for an arbitrary gene subset with fixed size in the range 2-20 genes.

features	accuracy improvement
2	+3.08%
4	+2.84%
6	+2.81%
8	+1.87%
10	+1.79%
12	+1.91%
14	+1.79%
16	+1.41%
18	+1.11%
20	+1.08%
average	+2.14%

Table 6.5: Average accuracy improvement over the second best method on all datasets.

In Table 6.6 the average accuracy and the average size[2] of (i) the greedy (Columns 2 and 3) and (ii) the set covering (Columns 5 and 6) minimum subsets are reported for all datasets. Values are averaged over the 50 repetitions of the 4-fold cross validation. The average execution time for one fold, which corresponds to an estimate of the time needed by the final user to perform the feature selection, is also reported for both techniques (greedy in Column 4 and set covering in Column 7). The last two columns in Table 6.6 contain the highest classification accuracy (Column 9) and the corresponding number of genes (Column 8) leading to it.

The minimum gene subset, already provides good performance for most datasets. For instance, on the Leukemia dataset, an almost maximum accuracy (87.00% vs 87.22%) is reached by the greedy selection using as few as 4.15 genes on average, whereas the maximum accuracy with a fixed subset is obtained by considering 20 genes.

Independently of the dataset, the greedy minimum subset size is always larger than the set covering size. The greedy approach selects the gene maximizing the number of covered samples at each iteration. The set covering approach, instead, exploits a global optimization procedure to select the minimum number of genes that cover the samples. Hence, the greedy approach may need a larger number of genes to reach the best coverage of the training samples. This larger gene set provides a higher accuracy on most datasets,

[2]The average size (i.e., the number of genes) of the minimum subset is not an integer, because it depends on the samples included in the considered fold. To cover different folds (i.e., sample sets), a different number of genes may be needed.

	Greedy			Set covering			Max accuracy with fixed #genes	
Dataset	#genes	acc.	time [sec.]	#genes	acc.	time [sec.]	#genes	acc.
Alon	5.09	71.82%	0.137	4.68	70.33%	16.255	2	78.94%
Brain1	6.33	70.23%	1.229	5.59	70.11%	938.556	20	74.49%
Brain2	5.07	57.49%	1.927	4.62	56.52%	35.142	16	60.04%
Leukemia	4.15	87.00%	0.529	3.82	85.89%	46.098	20	87.22%
Srbct	6.51	81.09%	0.246	5.95	79.55%	43.956	12	82.09%
Tumor9	10.11	27.57%	2.552	9.08	28.03%	86.382	20	33.03%
Welsh	1.83	89.00%	0.163	1.83	86.06%	11.488	8	90.24%

Table 6.6: Performance of the minimum gene subset selection on all datasets

because it yields a more general model which may be less prone to overfitting. For instance, on the Leukemia dataset the average accuracy is 85.89% for the set covering approach and 87.00% for the greedy approach.

The greedy algorithm is also characterized by a lower execution time with respect to the set covering algorithm. For example, considering the Brain2 dataset, the set covering completed in 35 seconds, whereas the greedy took less than 2 seconds. Since the greedy technique reaches higher classification accuracy with lower execution time, we have selected it as the method of choice both for the MaskedPainter feature selection approach and for all the other experiments in Section 6.2.

6.2.5 Classifier bias

Different classification techniques may exploit differently the same feature set and yield different classification performance. To analyze the effect of the peculiarities of different classification techniques on the gene set selected by the MaskedPainter we compared the classification accuracy obtained by different classifiers. Three classifiers have been chosen as representatives of different classification techniques: (a) for decision trees, the J48 classifier of Weka [152], (b) the Support Vector Machine implemented in LibSVM [33], and, (c) for the K-Nearest Neighbors approach, the IBk implementation in Weka [152] with K=3. For LibSVM the provided script for automatically tuning the training phase (from data scaling to parameter selection) has been exploited, while for the other approaches the default parameter values have been set.

The experiments have been performed on the Leukemia dataset for different numbers of selected features. Table 6.7 and Table 6.8 show the results for the KNN and SVM classifiers respectively, while the results for the decision tree are reported in Table 6.3. The MaskedPainter approach and the SV technique show a similar behavior, always providing the best performance.

#	MP	IG	TR	SM	MM	GI	SV
2	84.39	82.89*	80.78*	82.08*	82.14*	83.39	**84.83**
4	**90.81**	88.67*	85.42*	88.83*	86.17*	89.81*	90.47
6	90.75	90.89	89.31*	91.11	87.03*	**91.19**	90.83
8	91.64	92.03	91.25	91.56	88.47*	**92.33**	91.64
10	92.08	92.78*	91.56	92.36	89.22*	**92.83***	92.25
12	92.75	92.86	92.03	92.81	90.42*	93.00	**93.19**
14	93.28	92.78	92.44*	92.47*	90.53*	92.89	**93.42**
16	93.94	92.58*	92.53*	92.31*	90.58*	92.44*	**94.03**
18	**94.67**	92.22*	92.33*	92.33*	90.44*	92.08*	94.19
20	**94.69**	92.47*	92.44*	92.31*	90.61*	92.31*	94.50
avg	91.90	91.02*	90.01*	90.82*	88.56*	91.23*	**91.94**
max	**94.69**	92.86	92.53	92.81	90.61	93.00	94.50
dev	2.85	2.97	3.72	3.11	2.63	2.77	2.72

Table 6.7: Accuracy obtained using KNN classifier on Leukemia dataset.

The accuracy provided by KNN and SVM, as expected, is considerably higher than the accuracy of the decision tree. The first two classifiers build more robust models, which may make up for the selection of less interesting features by weighting them less in the model. Thus, decision trees allows better highlighting the effectiveness of different feature selection methods, because the quality of the selected feature set has a stronger impact on the accuracy obtained by the classifier. For this reason, we chose the decision tree to evaluate the quality of our feature selection method in the previous sections.

6.3 Discussion

We analyzed the biological information presented in literature for the genes selected by the MaskedPainter technique. In Table 6.9 we report the first twenty genes selected by our algorithm on the entire Alon dataset, related to colon cancer and commonly used for biological validation [35, 53]. Column 4 shows references to published works on colon cancer discussing the genes reported in Column 1. The majority of genes deemed as relevant by the MaskedPainter feature selection technique have been identified and discussed in previous biological studies.

For example, gene Z50753, named GUCA2B, related to uroguanylin precursor, is shown to be relevant in [127]. Lowered levels of the uroguanylin

#	MP	IG	TR	SM	MM	GI	SV
2	84.42	84.19	82.44*	82.14*	82.47*	**84.64**	84.61
4	**90.89**	88.42*	86.33*	88.75*	86.81*	89.14*	90.36
6	**91.31**	90.08*	89.11*	90.06*	86.72*	90.39	90.56
8	**91.47**	90.92	90.67	90.56*	87.69*	91.14	91.11
10	**92.28**	90.75*	90.81*	91.58	88.92*	90.72*	91.94
12	**92.89**	91.17*	91.31*	91.39*	89.94*	91.22*	92.69
14	93.22	91.08*	91.58*	91.14*	90.92*	91.44*	**93.25**
16	93.39	91.64*	91.67*	91.97*	91.19*	91.69*	**94.03**
18	93.97	92.00*	91.47*	92.08*	91.33*	92.06*	**94.75***
20	93.83	92.44*	91.67*	91.89*	91.39*	92.33*	**94.72***
avg	91.77	90.27*	89.71*	90.16*	88.74*	90.48*	**91.80**
max	93.97	92.44	91.67	92.08	91.39	92.33	**94.75**
dev	2.66	2.28	2.89	2.85	2.72	2.13	2.84

Table 6.8: Accuracy obtained using SVM classifier on Leukemia dataset.

may interfere with renewal and removal of epithelial cells. This could result in the formation of polyps, which can progress to malignant cancers of the colon and rectum [127]. As a second example, the downregulation of H06524, the GSN gene (gelsolin), combined with that of PRKCB1, may concur in decreasing the activation of PKCs involved in phospholipid signalling pathways and inhibit cell proliferation and tumorigenicity [27].

Rank	Gene ID	Gene Name	References
1	Z50753	GUCA2B	[127, 34]
2	H06524	GSN	[27, 34]
3	J02854	MYL9	[156, 150, 82, 154, 34]
4	K03474	AMH	[145]
5	L07032	PRKCQ	[21]
6	M63391	DES	[156, 150, 82, 154, 18, 34]
7	M36634	VIP	[136, 154, 34]
8	R87126	MYH9	[77, 154, 34]
9	M76378	CSRP1	[156, 150, 82, 154, 34]
10	H43887	CFD	[156, 150, 82, 34]
11	M22382	HSPD1	[156, 150, 82, 34]
12	X63629	CDH3	[18, 34]
13	H40095	MIF SLC2A11	[18, 34]
14	X74295	ITGA7	[82]
15	T71025	MT1G	[34]
16	H77597	MT1G MT1H	[55]
17	J05032	DARS	[154]
18	X86693	SPARCL1	[82, 154, 34]
19	M26697	NPM1	[18, 34]
20	H08393	OVGP1 WDR77	[79, 154, 18, 34]

Table 6.9: Top 20 genes on the Alon dataset (colon cancer) and related references.

7

Gene similarity measure

While feature selection approaches are addressed to identify the most relevant genes related to a target disease, clustering algorithms are often used to detect functionally related genes by grouping together genes with similar patterns of expression [38]. Many works consider the application or the adaptation of conventional clustering algorithms to gene expression data (see [75] and [138] for a review) and new algorithms have recently been proposed [52, 59]. All clustering algorithms need to define the notion of similarity between elements.

The common characteristics of most used clustering approaches applied on microarray data is that they cluster genes only by analyzing their continuous expression values. These approaches are appropriate when there is no information about sample classes and the aim of clustering is to identify a small number of similar expression patterns among samples. However, when additional information is available (e.g., biological knowledge or clinical information), it may be beneficial to exploit it to improve cluster quality [71].

We address the problem of measuring gene similarity by combining the gene expression values and the sample class information. To this aim, we define the concept of *classification power* of a gene, that specifies which samples are correctly classified by a gene. A gene classifies correctly a sample if, by considering the sample expression level, it assigns the sample unambiguously to the correct class. Thus, instead of discovering genes with similar expression profiles, we identify genes which play an equivalent role for the classification task (i.e., genes that give a similar contribution for sample classification). Two genes are considered equivalent if they classify correctly the same samples. The classification power of a gene is represented by a string of 0 and 1, that denotes which samples are correctly classified. This string is named *gene mask*.

To measure gene similarity, we define a novel distance measure between

genes, the *classification distance*, which computes the distance between gene masks. The classification distance has been integrated in a hierarchical clustering algorithm, which iteratively groups genes or gene clusters through a bottom up strategy [135]. To allow the computation of inter-cluster distance by means of the classification distance, the concept of *cluster mask* (i.e., the total classification power of genes in a cluster) was also defined. Besides hierarchical clustering, the classification distance measure may be integrated in clustering algorithms based on different approaches (e.g., DBSCAN [49], or PAM [80]).

We validated our method on different microarray datasets by comparing our distance measure with the widely used Euclidean distance, Pearson correlation and cosine distance measures. The experimental results confirm the intuition of the proposed approach and show the effectiveness of our distance measure in clustering genes with similar classification behavior.

7.1 Method

We propose a method to define the similarity between genes by measuring their classification power (i.e., their capability to correctly classify samples), which performs the following steps.

- **Core expression interval definition.** Definition of the range of expression values for a given gene in a given class. To address the problem of outliers, the density based approach (WDM) described in Chapter 4 is exploited in the core expression interval definition.
- **Gene mask and cluster mask generation.** Definition of the *gene mask* and the *cluster mask* as representatives of gene and cluster classification power. The gene mask is generated by analyzing the gene core expression intervals, while the cluster mask is generated by analyzing the gene masks of genes in the cluster. The gene mask computation is done according to the definition discussed in Section 4.2. Moreover, the notion of classification power may be extended to clusters of genes. Given an arbitrary gene cluster, its *cluster mask* is the logical OR between the masks of the genes in the cluster. It represents the total classification power of the cluster, i.e., the samples that can be correctly classified by considering all the genes in the cluster.
- **Classification distance computation.** Definition of the *classification distance* measure to evaluate the dissimilarity between the classifi-

cation power of genes (or clusters). The Hamming distance is exploited to measure the distance between masks.

In the following section the classification distance computation and the integration of the new similarity measure in a hierarchical clustering algorithm are described in details. The other steps are based on the definitions of Chapter 4.

7.1.1 Classification distance computation

The classification distance measure captures the dissimilarity between genes (or clusters) by analyzing their masks. It evaluates the classification power of each object, represented by its mask, and allows the identification of objects which provide similar information for classification.

Given a pair of objects (i, j), the classification distance between them is defined as follows

$$d_{ij} = \frac{1}{S} \sum_{s=1}^{S} mask_{is} \oplus mask_{js} \tag{7.1}$$

where S is the number of samples (bits) of the mask, $mask_{is}$ is bit s of mask i, and $\oplus$ is the EX-OR operator which yields 1 if and only if the two operands are different. Hence, the classification distance is given by the Hamming distance between masks.

When two genes (or clusters) classify in the same way the same samples, their distance is equal to 0 because their masks are identical. On the other extreme, if two objects have complementary masks, their distance d_{ij} is maximum and equal to 1, because the sum of complementary bits is equal to the number of samples S.

The classification distance is a symmetric measure that assesses gene similarity by considering both correct and uncertain classification of samples. We also considered, as an alternative, an asymmetric distance measure similar to the Jaccard coefficient [135]. This asymmetric measure considered the contribution of correctly classified samples (i.e., both 1 in the mask) and disregarded the contribution of samples for which classification is uncertain, due to interval overlap (i.e., both 0 in the mask). An experimental evaluation (not reported in the paper) of this alternative showed a worse performance, thus highlighting that also the similarity for uncertain classifications is important to group genes with similar behavior.

7.1.2 Integration in clustering algorithms

The classification distance measure may be integrated in various clustering approaches. To validate its effectiveness, we integrated it into a hierarchical clustering algorithm [135]. Agglomerative hierarchical clustering iteratively analyses and updates a distance matrix to group genes or gene clusters through a bottom up strategy.

Consider an arbitrary set G of N genes. The triangular distance matrix D can be computed on G by means of the classification distance measure defined in (7.1). An arbitrary element d_{ij} in D represents the distance between two objects i and j, which may be either genes or gene clusters. Matrix D is iteratively updated each time a new cluster is created by merging genes or gene clusters. The process is repeated $N - 1$ times, until only one single element remains.

At each iteration, the two objects to be merged are selected by identifying in D the element with the lowest value d_{ij}, which represents the most similar pair of objects (genes or clusters) i and j. If more object pairs are characterized by the same minimum distance, the element with the maximum average variance is selected, because variance is the simplest unsupervised evaluation method for gene ranking [63]. In particular, genes with high variance are usually ranked higher because their expression values significantly change over conditions [63]. Average variance of an element is given by the average over the variance of the expression levels of all genes belonging to the two objects i and j concurring to the new (cluster) element.

The classification distance measure may be integrated in other clustering approaches. For example, density-based clustering methods, such as DBSCAN [49], consider the Euclidean distance among elements to compute the reachability relationship needed to define each element neighborhood. The proposed distance measure may replace the Euclidean distance, while ϵ may be defined in terms of the maximum number of mismatching bits between the two masks (i.e., the maximum number of bits set to 1 after the EX-OR computation). Similar considerations hold for partition-based clustering algorithms (e.g., PAM [80]).

7.2 Experimental results

We validated our method on 9 microarray datasets, publicly available on [131] and [19]. Table 7.1 summarizes their characteristics. The data distribution

Table 7.1: Dataset characteristics: name, number of samples, number of genes, and number of classes

Dataset	*Samples*	*Genes*	*Classes*
Brain1	90	5920	5
Tumor9	60	5726	9
Leuk1	72	5327	3
Leuk2	72	11225	3
Lung	203	12600	5
Colon	62	2000	2
Prostate	102	10509	2
SRBCT	83	2308	2
DLBCL	77	5469	2

and cardinality of these datasets are rather diverse and allowed us to validate our approach under different experimental conditions.

We performed a set of experiments addressing the following issues.

- **Classification distance evaluation.** To evaluate the effectiveness of the classification distance in measuring the classification power of genes we compared the classification accuracy provided by neighboring genes. Furthermore, the biological relevance of our results has been assessed by verifying if neighboring genes are reported with similar biological meaning in tumor literature.
- **Core expression interval comparison.** The Weighted Mean Deviation (WMD) and the Hampel identifier (MAD) for detecting the core expression intervals have been compared in terms of both accuracy and interval characteristics.
- **Cluster characterization.** The characteristics of the clusters yielded by hierarchical clustering exploiting the classification distance have been investigated.

7.2.1 Classification distance evaluation

The effectiveness of the classification distance in detecting genes with similar classification power is evaluated by measuring the classification accuracy

Table 7.2: Differences between the accuracy of the original subset and the modified ones on the Brain1 dataset for different feature selection methods and distance measures

Method	*Distance*	1	2	3	4	5	6	7	8	9	10	*Mean ± Std*
ANOVA 81.11	Euclidean	-1.11	2.22	2.22	3.33	-2.22	-1.11	2.22	-1.11	-2.22	1.11	0.33±2.10
	Pearson	0.00	1.11	-1.11	2.22	-3.33	2.22	1.11	0.00	-3.33	-2.22	-0.33±2.04
	Cosine	1.11	-1.11	-2.22	3.33	-2.22	-1.11	1.11	1.11	-3.33	-1.11	-0.44±1.34
	Classification	-2.22	4.44	-1.11	2.22	1.11	1.11	3.33	1.11	-2.22	-2.22	**0.56±2.41**
BW 74.45	Euclidean	2.22	-2.22	-4.44	-7.78	-2.22	-4.44	-5.56	-5.56	-3.33	-2.22	-3.56±2.71
	Pearson	-8.89	-3.33	-3.33	-4.45	-5.56	-6.67	-5.56	-5.56	-3.33	-7.78	-5.44±1.55
	Cosine	-3.33	-3.33	-1.11	0.00	-3.33	-4.44	-3.33	-3.33	-3.33	-5.56	-3.11±2.20
	Classification	-1.11	-1.11	-5.56	-1.11	-3.33	-5.56	-4.45	-1.11	-2.22	-3.33	**-2.89±1.83**
OVO 74.45	Euclidean	2.22	0.00	3.33	-4.45	3.33	-1.11	1.11	3.33	-2.22	2.22	0.78±2.67
	Pearson	2.22	-1.11	5.55	5.55	1.11	1.11	0.00	2.22	-1.11	2.22	1.78±2.35
	Cosine	1.11	0.00	6.67	4.44	0.00	1.11	1.11	2.22	-1.11	3.33	**1.89±2.69**
	Classification	0.00	3.33	2.22	5.56	3.33	1.11	0.00	-1.11	-3.33	5.56	1.67±2.88
OVR 73.34	Euclidean	-6.67	-10.00	-5.56	-3.33	-3.33	-5.56	-1.11	-7.78	-5.56	-1.11	-5.00±2.83
	Pearson	-6.67	-6.67	-3.33	-4.45	-4.45	-3.33	1.11	-4.45	-2.22	-5.56	-4.00±3.01
	Cosine	-7.78	-7.78	-5.56	-2.22	-4.45	0.00	1.11	-3.33	-5.56	-5.56	-4.11±2.29
	Classification	-4.44	-5.56	0.00	-3.33	-2.22	-4.45	0.00	-2.22	-2.22	-8.89	**-3.33±2.67**

provided by similar genes. Accuracy is defined as the number of samples correctly associated to their class over the total number of samples.

In the context of tumor classification, to which the datasets in Table 7.1 are devoted, the most interesting genes are those which play a role in the disease. We focused our analysis on these genes, which are commonly selected by means of feature selection techniques [101]. In our experiments, we computed the accuracy provided by the set of top ranked genes selected by means of a supervised feature selection technique. Then, we substituted in turn a single gene with the most similar gene according to various distance metrics. We computed the new accuracies and we compared the obtained results to the previous accuracy value.

In particular, to avoid biasing our analysis by considering a single feature selection technique, we performed supervised feature selection by means of the following popular techniques [131]: (i) Analysis of variance (ANOVA), (ii) signal to noise ratio in one-versus-one fashion (OVO), (iii) signal to noise ratio in one-versus-rest fashion (OVR), (iv) ratio of variables between categories to within categories sum of squares (BW). Feature selection has been performed separately for each dataset. We considered the first ten genes ranked by each feature selection technique. These small gene subsets only contain genes which are relevant for discriminating among sample classes.

In each of the 10-gene sets obtained from feature selection, we substituted in turn a single gene with the most similar gene according to a distance

measure. In particular, we considered the Euclidean distance, the Pearson correlation, the cosine correlation, and the classification distance. Thus, for each 10-gene set and for each distance measure, we created ten new different gene sets, each of which with one substituted gene. The accuracy provided by these new sets has finally been computed and compared.

Classification has been performed by means of the LibSVM classifier [33], with parameters optimized by using the grid search in the scripts downloaded with the LibSVM package. Ten fold cross-validation has been exploited to avoid selection bias. The reported accuracy is the overall value computed on all the splits. The considered feature selection methods are available in the GEMS software [131].

Table 7.2 shows the results of the experiments on the Brain1 dataset. Similar results hold for the other datasets. The accuracy of the original setting (i.e., the ten original genes selected by the feature selection methods) is reported in the first column. Columns labeled 1-10 report the accuracy difference between the original set and each of the modified sets (each one with a different substituted gene). The last column reports the average value over the 10 modified settings.

For three out of four feature selection methods the classification distance selects the best substituted gene with respect to the other distance measures. In the case of OVO and ANOVA, the substitution even improves accuracy with respect to the original setting (i.e., it selects a better gene with respect to that selected by the supervised feature selection method).

The different overall accuracy increase/decrease depends on the intrinsic nature of each feature selection method. For the ANOVA and OVO methods, the original gene masks are characterized by more bits set to 1 (on average 20 over 90 samples) than the other two methods (on average 8). The highly selective genes (i.e., with few 1 in their mask) chosen by BW and OVR may be more difficult to replace appropriately. In this context, the classification distance selects a gene with a classification behavior more similar to the gene to be substituted than the other distance measures. Finally note that highly selective genes do not necessarily imply high accuracy as shown by the accuracy values in column 1.

Experiments performed with larger gene sets (i.e., 50 genes) showed a similar behavior. The original accuracy is higher (for example, it is 77.78% for BW when a set of 50 genes is considered) and the average difference in accuracy is lower (about 0.5% for the classification distance and -0.3% for the cosine distance). When the number of considered genes increases, the effect of a single gene on the classification performance becomes less evident.

Hence, these experiments are less effective in evaluating the characteristics of the classification distance.

To assess the biological meaning of similar genes, we focused on the Colon dataset, which has been widely studied in previous works. Two genes that are known to play a role in the tumor progression are J02854 (Myosin regulatory light chain 2, smooth muscle isoform) and M76378 (Cysteine-rich protein gene). According to the classification distance, the genes nearest to J02854 are M63391, T92451, R78934, and T60155. Gene M63391 is listed in the top relevant genes for colon cancer in [34, 164, 28, 25], while gene T60155 is cited in [25] and [164]. Furthermore, the genes nearest to M76378 are M63391 and J02854, both relevant for colon cancer.

We also analyzed the performance of other distance measures. The cosine correlation shows a similar behavior. For example, in the case of gene J02854, it detects as nearest three of the genes detected by the classification distance (R78934, T60155, T92451). On the contrary, there is no intersection between the nearest genes yielded by the classification and Euclidean distances. For example, for the Euclidean distance, the nearest to gene J02854 are genes R87126, X12369, R46753 and R67358. Among them, only gene X12369 shows a correlation to the colon cancer [155].

These results show that our distance metric groups genes with both comparable classification accuracy and similar biological meaning. Hence, our method can effectively support further investigation in biological correlation analysis.

7.2.2 Core expression interval comparison

As introduced in Section 4.1, the MAD estimator smooths the effect of values far from the median value, independently of their density. Hence, the core expression intervals defined by MAD are usually narrower than those defined by WMD. Figure 7.1 reports the comparison between the number of ones in the masks generated by MAD and WMD interval definitions. The number of ones in the masks is generally larger for MAD, because the intervals defined by MAD are smaller and yield less overlaps. Figure 7.2 shows the similarity between the gene masks obtained by using the two outlier detection methods. The masks agree in roughly 90% of cases (i.e., gene/class pairs).

We also analyzed the classification accuracy yielded by the gene mask representations provided by the MAD and the WMD methods. The same experimental design described in Section 7.2.1 has been used for these ex-

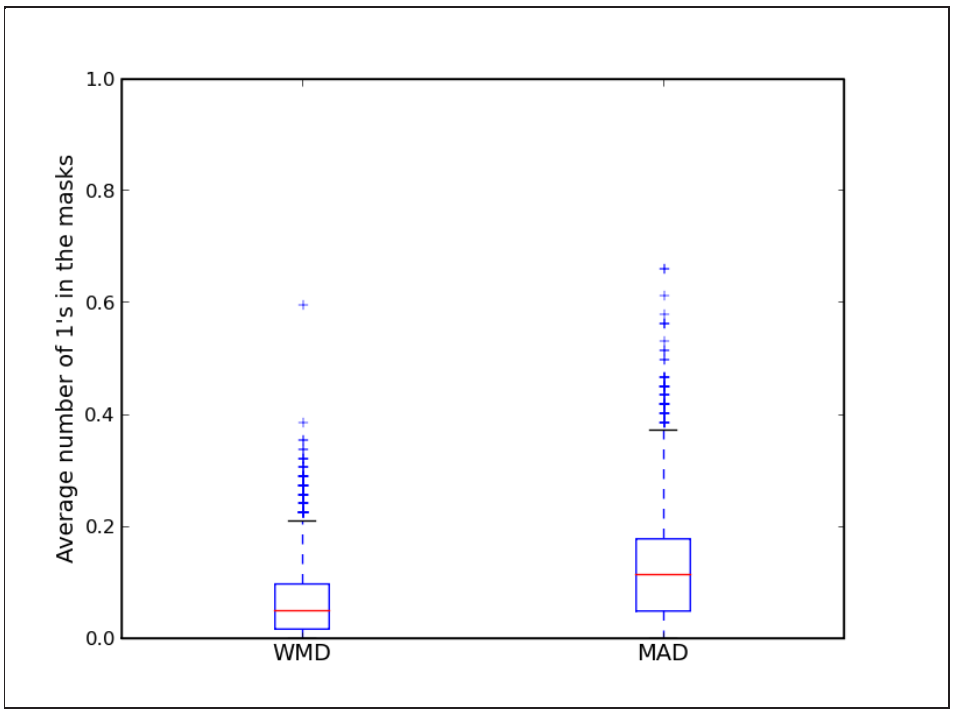


Figure 7.1: Distribution of ones in the gene masks created by using the WMD (left) and MAD (right) methods for outlier detection.

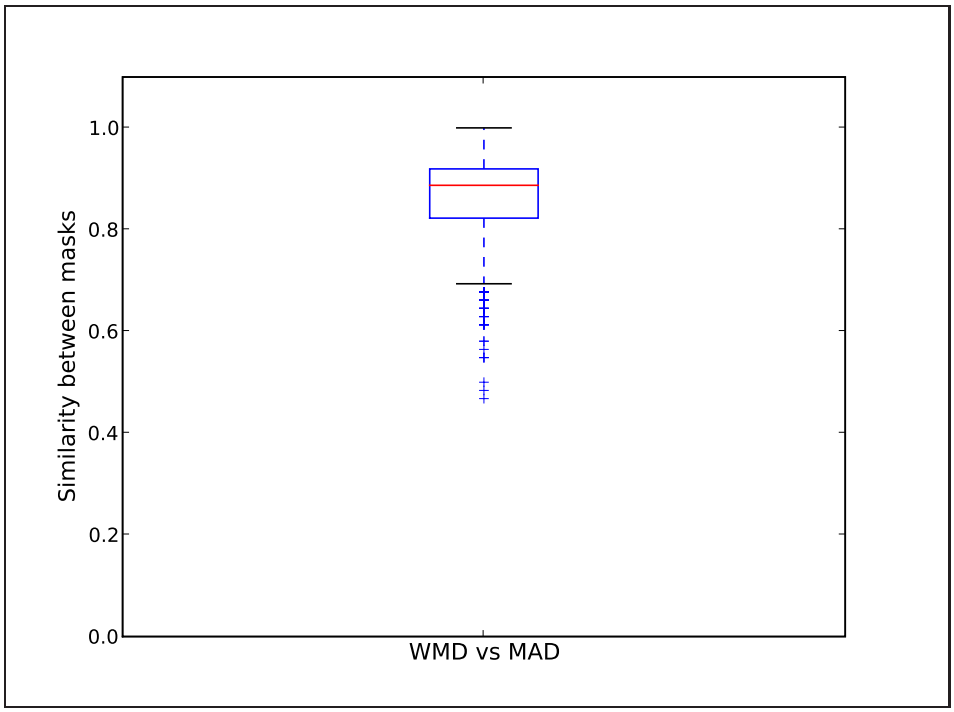


Figure 7.2: Similarity between the gene masks created by using the WMD and MAD methods for outlier detection.

periments. In most cases WMD provided a better accuracy than MAD. For example on the Brain1 dataset, the difference in accuracy between the original subset and the modified subset obtained by exploiting the MAD technique is -0.22±1.74 with ANOVA, 3±3.07 with BW, 1.56±2.24 with OVO, and -6.33±1.74 with OVR. Thus, for ANOVA, OVO and OVR, WMD accuracy (see Table 7.2) is higher than MAD accuracy. Furthermore, the standard deviation of the accuracy difference of MAD is, on average, larger than the standard deviation of WMD, thus showing a less stable behavior. Similar results are obtained for the other datasets.

This behavior may be due to an overestimation of the gene classification power when intervals are defined by means of MAD. In particular, since the core expression intervals defined by MAD are narrower, intervals are less overlapped. Hence, masks are characterized by a larger number of ones. Thus, the discriminating capability of a gene may be overestimated. The WMD method, by taking into account sample density, defines larger intervals which smooth this effect.

7.2.3 Cluster characterization

We evaluated the characteristics of the hierarchical clustering algorithm presented in Section 7.1.2, which integrates the classification distance measure. Since sample class labels are available, but gene class labels are unknown, the result of gene clustering cannot be straightforwardly validated. To evaluate the characteristics of our approach, we (i) compared by means of the Rand Index [117] the clustering results obtained by using our measure, the cosine, and the Euclidean metrics and (ii) evaluated the homogeneity of the clusters by analyzing the classification behavior of genes included into the same cluster.

To measure the agreement between the clustering results obtained with different metrics, we computed the Rand Index [117]. It measures the number of pairwise agreements between a clustering K and a set of class labels C over the same set of objects. It is computed as follows

$$R(C,K) = \frac{a+b}{\binom{N}{2}} \tag{7.2}$$

where a denotes the number of object pairs with the same label in C and assigned to the same cluster in K, b denotes the number of pairs with a different label in C that were assigned to a different cluster in K and N is the

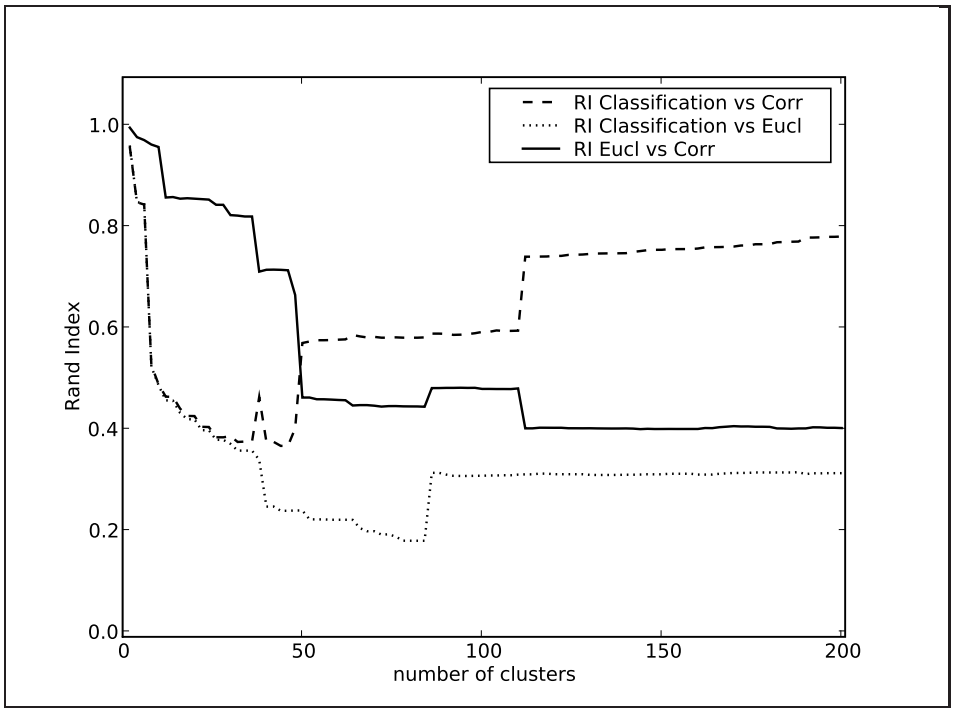


Figure 7.3: Pairwise Rand index evaluation between classification, Euclidean, and cosine distance metrics on the Colon dataset.

number of objects. The values of the index are in the range 0 (totally distinct clusters) to 1 (exactly coincident clusters). The Rand Index is meaningful for a number of clusters in the range $[2; N-1]$, where N is the number of objects. Clusters composed by a single element provide no contribution to the Rand Index evaluation [117].

To perform a pairwise comparison of the clustering results obtained by different distance metrics, we selected one metric to generate the clustering K and used as labels C the cluster identifiers obtained by clustering with the same hierarchical algorithm and a different distance metric. We repeated the process to perform the pairwise comparison of all three metrics. The results for the Colon dataset are shown in Figure 7.3. Similar results are obtained on the other datasets. Hierarchical clustering based on the classification distance shows a good agreement (ca. 70%) with cosine correlation clustering. Instead, the Rand Index between classification distance clustering and Euclidean distance clustering is very low. This last behavior is similar to that between Euclidean distance clustering and cosine correlation clustering.

To evaluate cluster homogeneity, we compared the classification accuracy

N	*Rapp.*	*Brain1*	*Leuk1*	*Lung*	*Tumor9*	*Leuk2*	*SRBCT*	*Prostate*	*DLBCL*	*Colon*	*Mean±Std*
10	original	74.45	94.44	86.21	54.89	93.06	93.98	93.14	85.71	81.97	
	diff_central	0.00	0.00	-1.97	7.72	-1.39	-1.21	0.00	2.60	0.00	0.64±2.96
	diff_border	0.00	-1.38	-4.93	1.54	-2.78	-7.23	0.00	1.30	0.00	-1.50±2.96
50	original	85.56	97.22	94.09	70.12	94.44	100.00	91.18	94.81	86.89	
	diff_central	2.22	0.00	0.00	1.78	0.00	0.00	0.00	0.00	0.00	0.44±0.89
	diff_border	0.00	2.17	0.98	-3.33	0.00	0.00	0.00	0.00	0.00	-0.02±1.45
100	original	84.45	95.83	97.04	66.40	93.06	100.00	92.16	96.10	86.89	
	diff_central	2.22	0.00	-1.47	-1.11	2.77	0.00	0.00	1.30	0.00	0.41±1.42
	diff_border	1.11	0.00	0.00	-1.11	1.38	0.00	0.98	1.30	0.00	0.41±0.83

Table 7.3: Differences from the original OVO rank accuracy on all datasets by using the central and the border genes

N	*Brain1*	*Leuk1*	*Lung*	*Tumor9*	*Leuk2*	*SRBCT*	*Prostate*	*DLBCL*	*Colon*
10	4.20	6.20	17.00	20.90	3.60	124.90	1.00	6.30	1.00
50	8.00	15.10	2.06	1.92	1.58	1.90	1.00	1.00	1.00
100	1.48	1.25	1.24	1.06	7.98	1.38	1.54	5.45	1.00

Table 7.4: Average cluster size for the experiment reported in Table 7.3

of genes belonging to the same cluster. To this aim, we defined two genes as representatives of each cluster, i.e., the one with the minimum (named central) and the one with the maximum (named border) classification distance to the cluster mask.

We only considered informative clusters, i.e., clusters containing relevant information for classification purposes, thus ignoring noise clusters. Informative clusters are selected by (i) identifying relevant genes, denoted as original genes in the following, by means of feature selection methods, (ii) selecting clusters such that each cluster contains a single original gene. More specifically, for the ANOVA, BW, OVO, and OVR feature selection methods, we selected the 10, 50 and 100 top ranked genes in a given dataset. For each original gene (i.e., gene in the rank), the largest cluster containing this gene and no other original gene is selected. In this way, three subsets of clusters are defined: (i) with 10 clusters, (ii) with 50 clusters, and (iii) with 100 clusters. For a larger number of clusters, the cluster size became too small and the analysis was not relevant.

Three different classification models have been built by considering (a) all original genes, (b) the substitution of each original gene with the central gene in its cluster, and (c) the substitution of each original gene with the border gene in its cluster. Classification accuracy has been computed in all three settings for each dataset, each feature selection method and each gene subset (i.e., 10, 50, and 100 genes).

Table 7.3 reports the original accuracy values (setting (a)) and the difference with respect to settings (b) and (c) for the OVO feature selection method on all datasets. The average size of the pool from which equivalent genes are drawn (i.e., the average cluster size) is reported in Table 7.4. Similar results have been obtained for the other feature selection methods.

Differences from the original classification accuracy are low. Clusters formed by a single gene (e.g., for the Colon and Prostate datasets) are not significant, because obviously the difference in accuracy is equal to zero. For larger clusters the differences are always limited to few percentage points. For example, for the ten cluster case on the Brain1, Leuk1, Leuk2 and DLBCL (cluster size range from about 3 to 6 genes) the difference in accuracy varies from -2.78 to 2.60. Always in the ten cluster case, the bad performance of SRBCT is due to the fact that one of the selected genes is located in a big cluster (average cluster size 124.90 genes). Thus, the border gene might be very different from the original gene.

On average, the obtained clusters provide a good quality gene pool from which equivalent genes may be drawn. The substitution with the central gene usually provides better results with respect to the substitution with the border gene. This difference is more significant for the larger clusters obtained for the 10 gene subset, than for the smaller, more focused clusters obtained in the case of the 50 or 100 gene subsets.

8

BioSumm biological summarizer

Analyzing the most relevant parts of research papers and performing on demand data integration for inferring new knowledge and for validation purposes is a fundamental problem in many biological studies. The growing availability of large document collections has stressed the need of effective and efficient techniques to operate on them (e.g., navigate, analyze, infer knowledge). Given the huge amount of information, it has become increasingly important to provide improved mechanisms to detect and represent the most relevant parts of textual documents effectively.

Initially, analyzing and extracting relevant and useful information from research papers was manually performed by molecular biologists [64]. This approach has become unfeasible in recent years, due to the huge amount of information that is constantly generated and contributed by a vast research community spread all over the world. Repositories like PubMed [13], the U.S. National Institutes of Health free digital archive of biomedical and life sciences journal literature, nowadays contain billions of documents and are constantly growing.

this chapter provides a new summarization approach. The extracts produced by our summarizer are intended to maximize the quantity of information related to the specific domain of biology. To obtain these "special" summaries we developed the BioSumm (Biological Summarizer) framework that analyzes large collections of unclassified biomedical texts and exploits clustering and summarization techniques to obtain a concise synthesis, explicitly addressed to emphasize the text parts that are more relevant for the disclosure of genes (and/or proteins) interactions. The framework is designed to be flexible, modular and to serve the purposes of knowledge inference and biological validation of the interactions discovered in independent ways (e.g., by means of data mining techniques).

We validated our approach on different text collections by comparing our

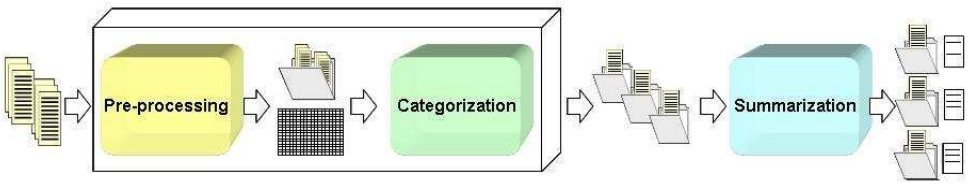


Figure 8.1: BioSumm framework architecture

summaries with a general purpose summarizer. The experimental results confirm the intuition of the proposed approach and show the effectiveness of our grading function in weighting sentences with a biological content relevant for gene/protein relationships. We also performed experiments for the identification of the better configuration of summarizer parameters. Finally, the capability of preprocessing block to separate in document clusters with similar topics is also discussed.

8.1 Method

The research papers stored in public repositories like PubMed [13], contain a huge amount of information about gene and protein relationships. This kind of information can be useful for biological validation in many research activities. The BioSumm framework exploits a summarization based approach to manage this mass of information. It analyzes unclassified biomedical texts to produce a good quality summary targeted to a specific goal such as, in our case, inferring knowledge of gene/protein relationships. However, the concepts and the techniques used, make the framework "general" and customizable also for other domains.

The framework is characterized by a flexible and modular structure, shown in Figure 8.1 and composed by the following blocks:

- **Preprocessing and Categorization.** It extracts relevant parts of the original document, produces a matricial representation of the sources and divides unclassified and rather diverse texts into homogeneous groups.
- **Summarization.** It produces a summary oriented to gene/protein relationships for each document group.

This blocks are described in details in the following paragraphs.

8.1.1 Preprocessing and Categorization

Since the document collections are unlabeled and are written in natural language, we exploit a preprocessing and categorization step to build a common representation of each article and identify groups of documents with similar topics. To achieve this goal, we performed two steps to prepares the document collections:

1. *Preprocessing* to build a matricial representation of the documents
2. *Categorization* to group documents which share the major topics

The preprocessing step parses the input collections and is designed to be flexible. Many different biological document sources are available [12, 14]. Among the various alternatives, we focused on PubMed Central, a well known repository for research articles. This repository offers all its articles in a format tailored for data mining analysis. Another important feature is that the articles are publicly available from [14]. These articles are in the form of a .nxml file [10], which is XML for the full text of the article, containing several tags (e.g., "journal" or "date of publication").

BioSumm may parse documents in different formats like xml, nxml and plain unstructured text files in order to produce a uniform text output. If the document collection is provided in nxml format, the preprocessing step extracts relevant parts of research papers, namely title, abstract, body and, when available, the keywords that describe the content of the article. Since only some parts of a research paper can be available or interesting for the analysis, the user may select which should be used by the summarizer tool.

Given a document collection D, we build a matricial representation W in which each row is a document and each column corresponds to a feature (stemmed word) of the documents.

Each element of matrix W is the TF-IDF (Term Frequency - Inverse Document Frequency) value for a term, computed as follows:

$$w_{ij} = tf_{ij} \cdot idf_j \tag{8.1}$$

where tf_{ij} is the term frequency of word j in document i and idf_j is the inverse document frequency of the term j. The tf_{ij} term in (8.1) is defined as:

$$tf_{ij} = \frac{n_{ij}}{\sum_k n_{kj}} \tag{8.2}$$

where n_{ij} is the number of occurrences of the considered term in document i and the denominator is the number of occurrences of all terms in document i. Moreover, the idf_j term is defined as:

$$idf_j = \log \frac{|D|}{|\{d : j \in d\}|} \tag{8.3}$$

where $|D|$ is the cardinality of document collection and $|\{d : j \in d\}|$ is the number of documents $d \in D$ in which term j appears.

The matrix W is generated by means of the text plug-in of RapidMiner [98]. This tool stems the documents exploiting the Porter stemming algorithm [140]. Since in most cases the generated matrix is still characterized by a high dimensionality, a further filtering step is applied. It eliminates "useless features", i.e., very frequent words that tend to be non discriminative for successive analysis.

The matricial representation of the document collection can be used to explore in a more efficient way the whole collection and find groups of documents which share similar topics. To accomplish this goal we apply a clustering algorithm. This step divides unclassified texts, belonging to document collections made of specialized journals, into more homogeneous subsets. The categorization phase is very important to detect documents which share a common topic without any a priory knowledge of their content.

Clustering is performed by means of the CLUTO software package [4]. CLUTO's algorithms have been optimized for operating on very large datasets both in terms of the number of objects as well as the number of dimensions. BioSumm is based on a partitional one, the repeated-bisecting method , which produces a globally optimized solution. This method reaches a suitable trade off between the quality of the results and the scalability guaranteed by partitional algorithms [132, 160].

Since a partitional algorithm is used, the number of clusters is required as input. BioSumm allows the user to select it. The effect of different value selection is explored in Section 8.2.4.

The similarity measure used is the cosine similarity function, which further improves the scalability of the approach. The combination of cosine correlation and repeated bisecting method is the most scalable in terms of time and space complexity [159]. Its time complexity is $O(NNZ * log(k))$ and its space complexity is $O(NNZ)$, where NNZ is the number of non-zero values in the input matrix and k is the number of clusters.

The criterion function used in the clustering part is $\mathcal{I}_2$:

$$\max \sum_{i=1}^{k} \sqrt{\sum_{v,u \in P_i} Sim(v,u)} \tag{8.4}$$

where k is the total number of clusters, P_i is the set of objects assigned to cluster i, v and u represent two objects and $Sim(v,u)$ is the similarity between the two objects. This criterion function is suitable in cases of high dimensionality and demanding scalability issues [159]. Moreover, according to [166] the $\mathcal{I}_2$ criterion function leads to very good clustering solutions where clusters tend to be internally homogeneous and well separated among each others. This is crucial for BioSumm because it means that in general the articles in the clusters tend to deal with similar topics. In this scenario the summarizer is particularly effective.

8.1.2 Summarization

This block is the core of the BioSumm framework. It provides, separately for each cluster, an ad hoc summary, containing the sentences that are potentially more useful for inferring knowledge of gene and protein relationships. The summary is in the form of an extract where the sentences of the original document collection with highest scores are reported.

BioSumm is a multi-document summarizer based on the Open Text Summarizer [11], a single-document summarizer whose implementation was proved to be particularly efficient by recent studies [157]. Our summarizer scans each document and gives a score to each sentence based on a grading function. The sentences with the highest scores are selected to build a summary, containing a given percentage of the original text. This percentage, which is set by the user, is called summarization ratio [54].

The BioSumm summarizer is designed to look for the presence of some domain specific words. Therefore, we define a dictionary G which stores the domain word (i.e. gene and protein names). We built the dictionary by querying the BioGrid publicly available database [130]. The Biological General Repository for Interaction Datasets (BioGrid) is a curated biological database of protein-protein interactions. It provides a comprehensive resource of protein-protein interactions for all major species while attempting to remove redundancy to create a single mapping of protein interactions. It currently contains over 198000 interactions from six different species, as derived from both high-throughput studies and conventional focused studies.

The grading function establishes a score for each sentence. The BioSumm grading function takes into account the presence of the domain specific words contained in the dictionary G. Let be K the words of the document collection which are identified by the preprocessing block and G the gene and protein dictionary defined above.

The grading function Γ for sentence j in document i is given by:

$$\Gamma_j = \delta_j \cdot \sum_{k \in K} \omega_k \cdot \varphi_k \tag{8.5}$$

where ω_k is the number of occurrences of term k of set K and φ_k is the *Key words* factor of the Edmundson statistic-based summarization method [47]. The factor φ_k is used to give higher scores to statistically significant words in the document

The factor δ_j is a weighting factor which considers the number of occurrences of gene/protein dictionary G in sentence j. It is defined by:

$$\delta_j = \begin{cases} 1 & \text{if } \omega_g = 0, \\ & \forall g \in G \\ \alpha + \beta \cdot \sum_{g \in G} \hat{\omega}_g + \gamma \cdot \sum_{g \in G} (\omega_g - 1) & \text{otherwise} \end{cases} \tag{8.6}$$

where $\hat{\omega}_g$ represents the number of distinct occurrences in sentence j of term g belonging to G dictionary. Instead, $(\omega_g - 1)$ counts all the occurrences of g, duplicates included, starting from the second one. This means that in a sentence in which there is five times the same entry of the dictionary $\hat{\omega}_g$ is equal to 1 and $(\omega_g - 1)$ is equal to 4. We used $(\omega_g - 1)$ so that no dictionary entry is considered twice.

The parameters α, β and γ are three constant factors. The coefficient α belongs to the range $[1, +\infty)$ and its role is to favor the sentences that contain terms in G, disregarding their number. The coefficient β is instead in the range $[0, 1]$ and weights the occurrences of distinct words of G. Finally, the coefficient γ is in the range $[0, \beta]$ and weights the "repetitions" of words of G.

With $\alpha = 1$, $\beta = 0$ and $\gamma = 0$ the summarizer ignores terms in G, thus disregarding the dictionary. In this case the only contribution to the sentence score is the one given by ω_k. By increasing α, the presence of a gene or protein of G raises the score of the sentence, but sentences with a different number of gene references are weighted identically. To weight the occurrences of distinct terms of G, β should be different from 0. The closer

β is to 1, the more different gene occurrences in the sentence are deemed relevant. In case of β different from 0, when γ is equal to 0 only the distinct dictionary entries are taken into account, while setting β and γ equal, all the entries are equitably evaluated. The upper bound of the coefficient γ is β because is meaningless to weight the multiple occurrences of a certain entry more than the entry itself.

The coefficient α has no upper bound, but when it becomes high the importance of the ω_k part decreases significantly. In other words, with such configuration the BioSumm grading functions tends to be an extractor of sentences containing dictionary entries rather than a summarizer that favors domain specific sentences.

Since our summarizer works with a group of documents, the grading function scores are normalized to have a common scale. The scaling factor is proportional to the maximum score obtained for the i-th document.

Finally, the top-N sentences are selected to build the summary of the document group analyzed. The number of sentences selected is given by the summarization ratio selected by the user.

8.2 Experimental results

We validated our approach on a subset of the PubMed [14] text collections composed by the articles of the following journals:

- *Breast_Cancer* (911 papers) focused on the genetic, biochemical, and cellular basis of breast cancer
- *Arthritis_Res* (996 papers) focused on the mechanisms of localized and systemic immune-inflammatory and degenerative diseases of the musculoskeletal system
- *J_Key* (872 papers) composed by the articles of the previous journals that have keywords field

We performed a set of experiments addressing the following issues:

- **Summary analysis.** To evaluate the effectiveness of our approach in extracting sentences with biological content, we compared the contents of the summaries obtained with BioSumm and with a general purpose summarizer.

- **Summarizer configuration.** The most suitable BioSumm parameter configurations is validated by analyzing (i) the sentence recall and (ii) the gene score of the summaries generated.
- **Summarization performance**. The output of our summarizer is automatically evaluated using the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) [90].
- **Categorization evaluation.** The capability of the clustering block to group similar documents is evaluated by using the Rand Index analysis to compare the results of the clustering on keywords and on different part of the documents.

8.2.1 Summary analysis

In this section we analyze the ability of the BioSumm summarizer to capture the sentences potentially more useful for biological validation and knowledge inference. We compared the biological content of our summary with respect to a general purpose one (i.e. OTS).

We report the analysis on one cluster of Breast_Cancer. Similar consideration can be done on other clusters of other collections. The "keywords" of analyzed cluster, namely its most descriptive and discriminative words, are *proband*, *Ashkenazi*, and *Jewish*. The cluster contains 10 documents. The documents contain 310 gene/protein dictionary entries. In Table 8.1 we report the six sentences graded with the highest scores by the two summarizers.

We obtained the BioSumm summary using as coefficients of (8.6) α equal to 2.0, β equal to 0.8 and γ equal to 0.0. We will discuss the most suitable configuration of BioSumm in next section.

BioSumm, generally, gives a high score to the sentences containing genes, which are very likely selected for the summary. More specifically, most of the top sentences contain at least a reference to BRCA1 or BRCA2, that are human genes belonging to a class known as tumor suppressors [24, 57]. Some others mention TP53 , PTEN , CHK2 and ATM which are also human genes of the class of tumor suppressors [68]. Finally, one sentence refers to CYP17 that is a gene whose mutations may be associated to prostate cancer and male breast cancer [162].

Among the sentences that contain these genes, the summarizer prefers those referencing the highest number of them. These sentences are more relevant for knowledge inference and biological validation, because they may

	BioSumm sentences	OTS sentences
1	In contrast to studies on North and East European populations the present results indicate a lack of relevant founder effects for BRCA1- and BRCA2 -related disease in the sample of patients analyzed, which is in agreement with other Italian studies and with ethnical and historical data.	In contrast to studies on North and East European populations the present results indicate a lack of relevant founder effects for BRCA1- and BRCA2 -related disease in the sample of patients analyzed, which is in agreement with other Italian studies and with ethnical and historical data.
2	Initially there was evidence that a TtoC variant in the CYP17 gene (allele frequency about 0.4) played a role in serum oestrogen and progesterone levels, and was associated with an increased risk of advanced disease.	This is a low proportion compared with studies that suggested that BRCA1 and BRCA2 are responsible for the large majority of breast/ovarian cancer families, with the greater proportion due to BRCA1.
3	Conclusions Considering the reported higher frequency of BRCA1 and BRCA2 germline mutations related to breast and ovarian cancer among Ashkenazi women in different countries, the results presented in this study were interpreted as showing a relatively lower than expected breast cancer mortality pattern among Ashkenazi women in the studied Brazilian cities.	Third, we let i Y = log(2 i p) if the i i th woman was a carrier and log[2(1-p)] otherwise, i E1 = n log2 + p log(ip) + (1-ip) log(1-ip) and i O1 = Y.
4	Two studies have estimated that mutations in the BRCA1 and BRCA2 genes only account for approximately 15% of the excess familial risk of the disease, while the contribution of the other known breast cancer susceptibility genes (TP53 , PTEN , CHK2 and ATM) is even smaller.	Furthermore, the tumor genes and their mutations also appear to be responsible for an important, but still debated proportion of male breast cancers.
5	We also compared and combined parameter estimates from our new analyses with those from our logistic regression analyses of data on unaffected women from the Washington study, and derived a simple algorithm to estimate the probability that an Ashkenazi Jewish woman carries one of the three ancestral mutations in BRCA1 and BRCA2.	The statistic i Z1 = (O1-E1)/[var(E1)] 1/2 , where var(E1) = p(1-ip)log[ip/(1-ip)] i 2 has a standard normal distribution under the null hypothesis, and deviations test whether the predicted values were too clustered or too dispersed.
6	Mutations in TP53, and possibly in the ATM and CHK2 genes, seem to confer moderately increased risks of breast cancer but might explain only a very small proportion of familial aggregation.	These mutations were already reported in the literature or in the Breast Cancer Information Core electronic database.

Table 8.1: Sentences with the highest score in cluster *proband*, *Ashkenazi*, *Jewish*

describe the relationship between several genes/proteins. For example, by considering the third sentence, is possible to learn that BRCA1 and BRCA2 are involved in breast/ovarian cancer. It is possible to have the biological evidence of the correctness of this information in [23] and [37], which are scientific papers not belonging to our collections. Another example is the fourth sentence. It confirms the findings on BRCA1 or BRCA2 and it states that other genes, such as TP53, PTEN, CHK2 and ATM, are breast cancer susceptibility genes. The biological evidence of this information can be found in [118].

The sentences selected by BioSumm are all closely related to the keywords of the cluster. In fact, most sentences describe gene interactions discovered in statistical analysis on different populations. Moreover, two out of the six most important sentences explicitly refer to the Ashkenazi Jewish, the major topic of the cluster. Hence, the BioSumm grading function, although strongly gene and protein oriented, is still capable of detecting the most relevant sentences for the topics of the cluster, which deals with genetic studies on populations.

This BioSumm peculiarity is also confirmed by considering the second column of Table 1, which contains the sentences selected by OTS.

In fact, the top sentence is the same for both summarizers. This means that BioSumm is still able to extract the sentences that are also important from the point of view of a traditional summarizer. BioSumm also extracts the second sentence of OTS but it does not put it in its top six (it is the tenth and it is not reported here). Thus, this sentence is not discarded, but the framework favors other sentences which contain more dictionary entries or more information. Such sentences are potentially more useful for our final goals.

The third and the fifth sentences selected by OTS are long sentences that introduce new paragraphs and deal with statistical analysis, but not directly with biological topics. For this reason BioSumm discards them, while OTS selects them because of their position in the text.

Finally, the sixth sentence is particularly important to understand the difference between BioSumm and a traditional summarizer. In fact, it is a quite short and technical sentence that tends to be pruned by most summarizers. BioSumm selects a sentence which is really meaningful for our purposes, because it describes the roles and the interactions of three different dictionary entries and of their mutations (this information is also confirmed by [118]). The sentence extracted by OTS, instead, albeit addressing the same issue, is more general. In fact, it also deals with gene mutations, but misses all the dictionary entries. The BioSumm framework was able to capture this crucial

piece of knowledge. This comparison explains how a traditional summarization approach favors generic and descriptive sentences whereas BioSumm is able to extract domain specific content.

By analyzing all the six sentences is possible to observe that BioSumm extracts many different dictionary entries. Using a traditional summarization approach only two genes are mentioned. From the point of view of biological validation and knowledge inference, this is an important difference.

A crucial aspect of the BioSumm summary is that its sentences do not simply list domain specific terms. They generally describe such entries, their relationships, the research fields in which they are involved and their mutations. They have the descriptive power of the sentences extracted by traditional summarizers, but they focus on domain specific information.

8.2.2 Summarizer configuration

Since our summarizer is based on the three parameters in (8.6), we evaluated their most suitable configuration to obtain summaries oriented to gene and protein interactions. The vast majority of the projects in the summarization field evaluate their results with a two step approach [95]. First a *Golden summary* which represents the ideal output of the summarizer is build, second a comparison using similarity measures between the produced summary and the Golden one is performed.

In most cases the Golden summary is made by human beings. However, this widely used approach is considered a weak validation technique. The reasons are that in this way is very difficult to obtain an objective evaluation and that the reference summaries made by two different individuals may differ significantly. Moreover, building a multi-document summary may require a lot of time to a person. The stronger and cheaper approach that we used involves the automatic generation of the golden summary.

Before discussing the procedure used to generate the golden summary, is necessary to carefully define what the golden summary should be. The ideal output of BioSumm summarizer must find a balance between the sentences that mostly contain domain specific words and the sentences normally extracted by traditional summarizer. BioSumm must not be neither a traditional summarizer nor a tool for extracting sentences containing the entries of a dictionary. It must build an extract mostly containing domain specific information.

For these reasons, we did not use a unique summary, but two texts as

golden summary. The first one is made by extracting from the documents the sentences that contain more entries of the dictionary, listed in descending order of number of entries. The second one is the output of a general purpose summarizer, in our case OTS.

The second step requires to compare the reference summary with the produced one. Most projects involve a manual comparison of the two summaries done by a human judge which reads both the texts and compares them [95]. Even if still used, the solution is considered too expensive and not very objective. A preferred approach exploits an automatic evaluation based on similarity measures.

For our evaluation we used two similarity measures: *Sentence Recall* and *Sentence Rank*. They are not mutually exclusive since they tackle different aspects. Sentence Recall compares the set of extracted sentences and the golden summary disregarding the order in which the sentences are presented. Sentence Rank is instead focused on the order.

Let define D_b the output summary of BioSumm D_{OTS} the golden summary produced by the general purpose summarizer OTS and D_g the golden document that maximizes the number of gene/protein dictionary entries. We separately evaluated the Sentence Recall between BioSumm output and each one of these sets. Thus, the Sentence Recall is given by the number of sentences in common between two texts divided by their total number of sentences.

Since we have two golden summaries (one produced by a general purpose summarizer and one by a gene extractor), we modified the original definition of Sentence Rank. Thus, we defined two rank based scores: Score_Gene (S_g) and Score_ots (S_o).

Score_Gene is related to D_g golden summary. We defined the Sentence Rank based on genes S_g as:

$$S_g = \sum_{j \in D_g} \sum_{g} \hat{\omega}_{g,j} \tag{8.7}$$

where $\hat{\omega}_{g,j}$ represents the number of distinct occurrences, in sentence j, of term g belonging to the gene/protein dictionary. The highest is the score, the most the summarizer has taken sentences with the domain specific terms (i.e genes and proteins). A high S_g also means that the summarizer is able to take the sentences with the highest rank in the golden document D_g.

The S_o score is related to D_{OTS} golden summary. Since OTS for each

sentence retrieves a score which depends on Edmudson method, a normalization preprocessing step is performed. The aim of all the following operations is to build a 20-point scale of similarity that is independent from the the document statistics. For each document we normalized OTS score by dividing it for the maximum score for that document. Thus, we obtained a value in the range $[0, 1]$. Then the range $[0, 1]$ is divided in 20 buckets with the same width (i.e. 0.05). Depending on in which bucket the normalized score lays, we assign to it a score s_j in the range $[1, 20]$. The result is that the sentences with the highest score receive 20 points whereas the sentences with the lowest scores receives 1 point. The scale of similarity is a standard concept, normally used in the Sentence Rank [95]. Therefore, S_o is defined as the sum of all sentence scores of the BioSumm summary.

In order to have a common scale between the two scores, we normalized them in the range $[0, 1]$. The normalization is done by dividing them for their theoretical maximum.

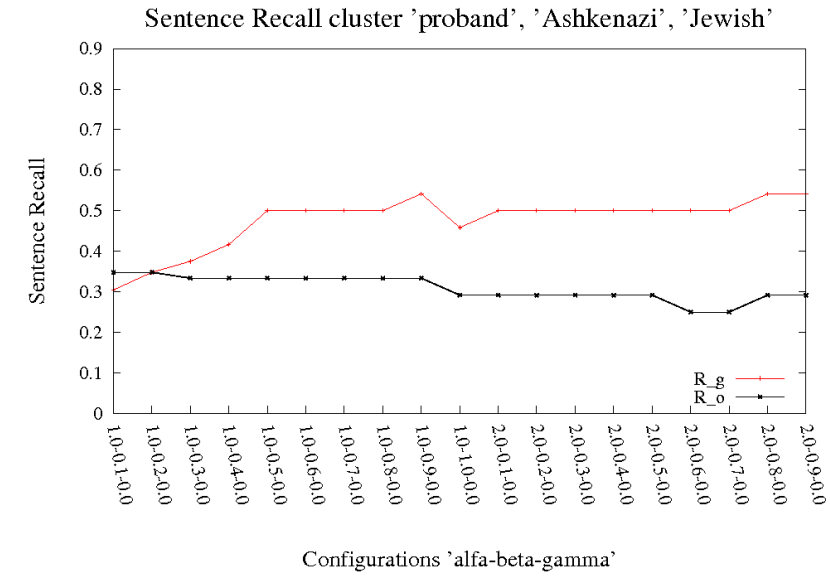
Our ideal summary must have $S_g > S_o$ since domain specific content must be favored. Anyway their values must be close. If they are very different it means that BioSumm is favoring one of the two parts by selecting the sentences with high scores only for one criteria. This may not happen since the goal is to find a balance.

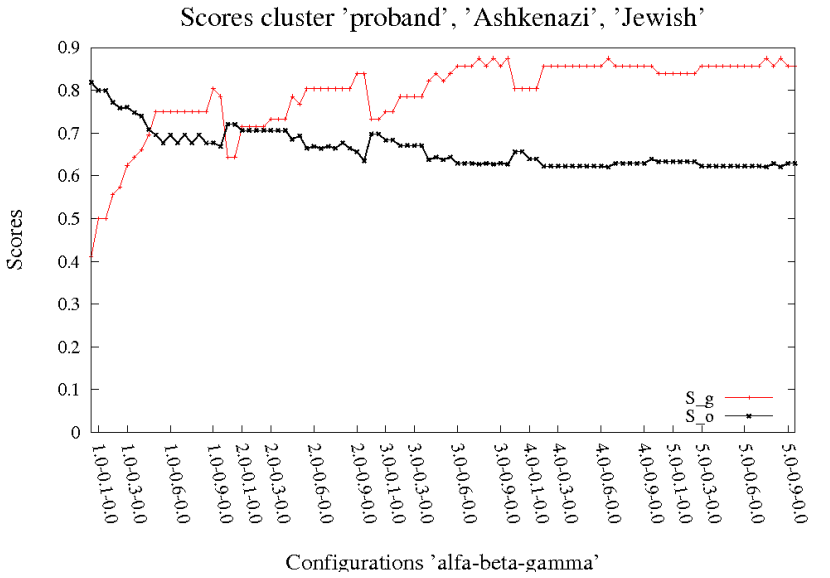
We performed experiments on all the datasets. We set the number of clusters to 80. This setting produces a good clustering quality as discussed in next paragraph. Both abstract and body have been considered. The summarization ratio is set to 20%. We report the analysis on the cluster of Breast_Cancer considered in the previous paragraph. Similar results are obtained on other document clusters. In Figure 8.2(a) we report, for the various configurations, the two recalls R_g and R_o, and in Figure 8.2(b) the two normalized Rank-based scores, S_g and S_o.

Both the plots must be read in this way. On the x axes there are the various configurations of α, β, γ with the three values of the coefficients separated by a '-'. Practically "1.0-0.0-0.0" means $\alpha = 1.0$, $\beta = 0.0$, $\gamma = 0.0$. Two consecutive configurations have different values of γ. This means that after "1.0-0.1-0.0" there is "1.0-0.1-0.1". When γ reaches β, the next configuration will have higher β. This means that after "1.0-0.1-0.1" there is "1.0-0.2-0.0". When β and γ reaches 1.0, the next configuration will have higher α. Namely, after "1.0-1.0-1.0" there is "2.0-0.0-0.0".

The parameter γ has a limited impact and it behaves like an "on/off" value. The decision of setting it to its maximum β or to 0.0 depends on if the user needs to take into account duplicated dictionary entries or not.

(a)





(b)

Figure 8.2: Sentence Recall R_f (a) and Normalized scores S_g and S_o (b) on cluster *proband*, *Ashkenazi*, *Jewish*

The configuration on which the grading function introduced start to extract sentences which contain more gene/protein information is around α equal to 1.0 and β equal to 0.3. Moreover, the ranges that maximize the combination of the two Sentence Recalls have α equal or higher than 1.0 and β greater than 0.6. All the other configurations that have α equal or higher than 3.0 and β higher than 0.5, have poorer results a part from some isolated peaks.

In Figure8.2(b) is possible to see that the configurations with α greater than 3.0 have a very high S_g (the one related to the dictionary entries) but low values of S_o (the one related to the "traditional" summary). These configurations are not the ideal ones since the BioSumm summarizer in this case tends to behave like a dictionary entries extractor. When α is equal to 1.0 the situation is the exact opposite. S_o is very high whereas S_g is low and in some cases even lower than S_o. This situation must be avoided too since in this case BioSumm behaves similarly to a traditional summarizer.

When α is equal to 2.0 (especially when β is high) the situation is closer to the ideal output. In fact, the values of S_g and S_o are not very distant. Moreover, S_g is not very far from the configurations that maximizes it and this means that BioSumm is performing quite well in extracting the information useful for knowledge inference and biological validation. An acceptable result is also the one with α equal to 3.0 and β greater than 0.7. In such configurations BioSumm tends to favor the dictionary entries but the contribution of the traditional summary is not minimal.

Therefore, α equal to 2.0 produces an output close to the ideal one. When β is high (greater than 0.5) the situation is optimal since S_g is high and S_o is not low. Other two ranges are also acceptable: α equal to 1.0, β greater than 0.7 and α equal to 3.0 with β greater than 0.7. We prefer the second one since in the first one the contribution of the traditional summary is too high. BioSumm must behave differently than a traditional summarizer and favor domain specific information, so S_o must not be too high.

8.2.3 Summarization performance

Since a summarizer has to capture also the main concepts (N-grams) of a document, we compared the performance of our grading function respect to the general purpose summarizer OTS. To compare summaries generated automatically from systems, we used the abstract of each paper (author's summary) as model. We suppose that the abstract is focused also on the biological content and not only on the method proposed in the paper. The systems (e.g., BioSumm and OTS) worked only on the body of the paper

	BioSumm			OTS		
	Precision	Recall	F-score	Precision	Recall	F-score
Breast_Cancer	0.082	0.225	0.114	0.080	0.218	0.111
Arthritis_Res	0.090	0.253	0.125	0.088	0.244	0.121

Table 8.2: ROUGE-2 scores

	BioSumm			OTS		
	Precision	Recall	F-score	Precision	Recall	F-score
Breast_Cancer	0.100	0.281	0.140	0.098	0.275	0.138
Arthritis_Res	0.110	0.317	0.154	0.109	0.308	0.151

Table 8.3: ROUGE-SU4 scores

disregarding the abstract. For BioSummthe parameters was set to the optimal condition according to the previous analyses (i.e. $\alpha = 2.0$, $\beta = 0.8$ and $\gamma = 0$). The summary size was set at 20%. Finally, the output of each summarizer was automatically compared using an automated tool called ROUGE [90].

The ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [90] is an automated tool which compares a generated summary from an automated system (e.g., BioSumm ,OTS) with one or more ideal summaries. The ideal summaries are called models. ROUGE uses N-grams to determine the overlap between a summary and the models. ROUGE was used in the 2004 and 2005 Document Understanding Conferences (DUC) (National Institute of Standards and Technology (NIST), 2005) as the evaluation tool. We used parameters from the DUC 2005 conference. ROUGE-2 and ROUGE-SU4 precision, recall and f-measure scores are used to measure each summarizer. ROUGE-2 evaluates bigram co-occurrence while ROUGE-SU4 evaluates "skip bigrams" which are pairs of words (in sentence order) having intervening word gaps no larger than four words.

The average results of ROUGE-2 and ROUGE-SU4 precision, recall and f-measure between BioSumm and OTS are shown in Table 8.2.3 and 8.2.3 . BioSumm outperforms always OTS on all the document collections. Combining the statistical analysis of Edmudson method with the presence of gene and proteins in the sentences allows our system to identify sentences which describe the major topic of the article but also the biological aspect (gene and protein relationships). Since the analyzed collections come from journal which are addressed to the biological and biomedical aspects of diseases, the abstract usually is addressed more to resume the biological information than

the method employed to achieve the results. Therefore, the idea of using a grading function based on the presence of gene/protein to capture the biological content and interactions seems to retrieve also better summaries on biological and biomedical papers.

8.2.4 Categorization evaluation

The role of the clustering block is to divide a collection in small subsets, maximizing the internal similarity and cohesion of each generated cluster, without any a priori knowledge of the document contents. Therefore, a good cluster is a group of documents sharing similar topics.

To measure the agreement between topics and clustering results we computed the Rand Index [117]. It measures the number of pairwise agreements between a clustering K and a set of class labels C over the same set of objects. It is computed as follows

$$R(C,K) = \frac{a+b}{\binom{N}{2}} \tag{8.8}$$

where a denotes the number of object pairs with the same label in C and assigned to the same cluster in K, b denotes the number of pairs with a different label in C that were assigned to a different cluster in K and N is the number of items.

The values of the index are in the range 0 (totally distinct clusters) and 1 (exactly coincident clusters). The Rand Index is meaningful for a number of cluster in the range $[2; N-1]$, where N is the number of objects. Outside the range it produces degenerate results. Moreover, clusters with only one element are penalized giving no contribution to Rand Index analysis.

The most important difficulty to face is that we do not have any a priori knowledge of the content of the texts. Moreover, the labels are not available in the PubMed Central repository. We tackled the problems by analyzing collections in which some keywords are available for all articles. The keywords provide an objective way to define the topics of the articles. In fact, they must be inserted by the author or the editors of the article and so they are very often representative of the content of the document.

We clustered the keyword descriptors of the articles and we used the resulting clusters as class labels C for the Rand Index. Separately, we clustered the abstracts, the bodies, and the abstracts plus bodies of the same

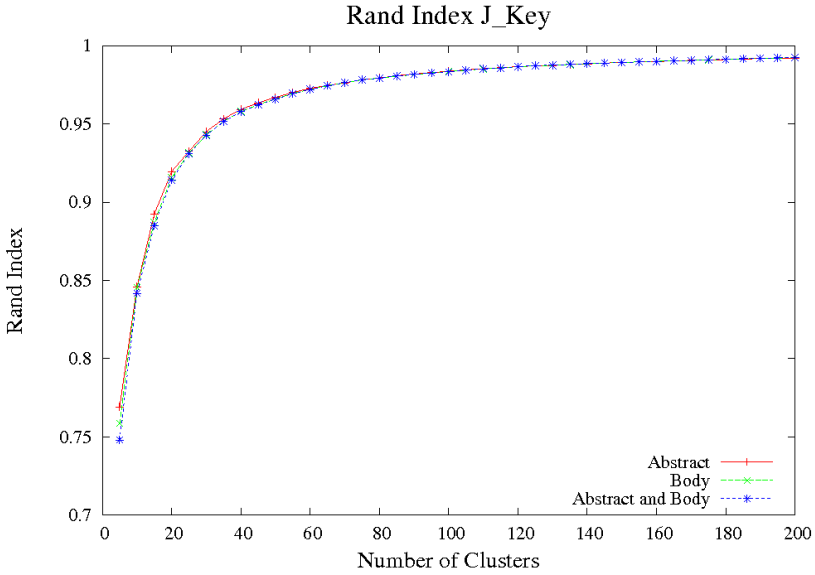


Figure 8.3: Rand Index for J_Key collection

documents discarding the keywords field. We repeated the experiment with several values of the cluster number parameter. We performed the experiment on all the document collections. We report the results obtained on J_key collection. Similar results are obtained on the others.

The J_key collection groups only the articles with keywords belonging to the collections *Breast_Cancer* and *Arthritis Res.* From the point of view of the clustering part, this collection is important because its articles, which belong to different journals, are heterogeneous. In such condition the clustering role is fundamental to identify common topics and a suitable division of the documents.

The Rand Index is generally high (greater than 0.7) and becomes very close to 1 for more than 40 clusters as shown in Figure 8.3. In fact, smaller clusters (containing around 10-20 documents) tend to include more homogeneous documents. The clustering result of the keywords and the result obtained using the other parts of documents are very similar. Hence, the clustering block clusters the documents according to the topics they actually deal with. For this collection a common choice for the clustering number parameter is 80. In fact Rand Index with 80 clusters achieves 0.98. This result proves that this value is able to provide a good quality of the clustering block.

8.3 BioSumm tool

We also developed a tool based on the summarization approaches described in the previous sections. BioSumm tool is a summarization environment that allows the users to query online repositories (e.g., PubMed) and extract small subsets of documents relevant to the search. BioSumm enhances keyword document search by (a) grouping a (possibly large) set of retrieved documents in focused clusters, and (b) providing a multidocument summarization of each cluster. Browsing the small number of generated summaries will allow the user to select the subset(s) of documents of interest.

8.3.1 Demonstration Plan

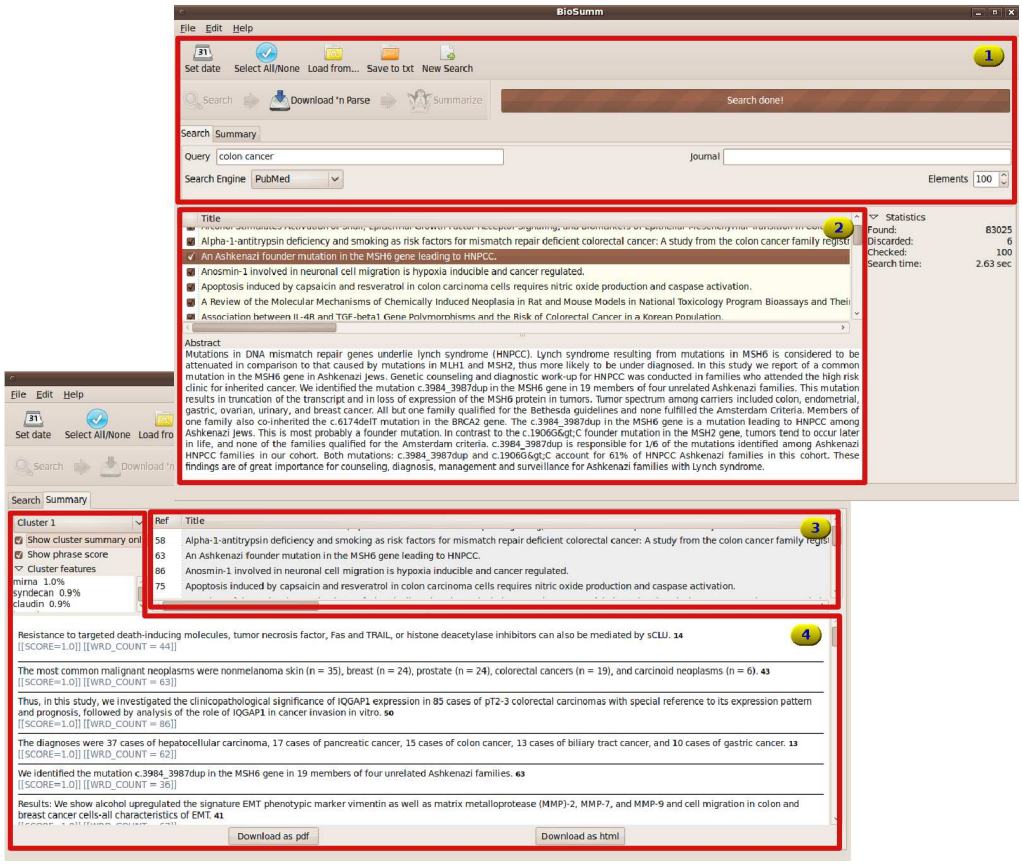


Figure 8.4: BioSumm graphical interface

The interaction with the BioSumm system occurs through a GUI interface. The screenshots of the main windows are shown in Figure 8.4. The user may perform the following activities.

- **Document search** (block (1) in Fig. 8.4). The user can set the search parameters (e.g., keywords, journal, date of publication) for the web search or load local repositories to create the relevant document collection. For the demo, either documents will be dynamically gathered through the web, or, in case of limited connection, local repositories will be exploited to build the collection.
- **Cluster browsing**. When the returned document collection is large, documents are clustered and the obtained clusters are characterized by the BioSumm summarizer. Each cluster can be analyzed by the user from two points of view: (a) the cluster abstract description, and (b) the cluster content. For point (a) (block (4) in Fig. 8.4), the cluster summary can be browsed. Each sentence in the summary is scored by relevance with respect to the considered application domain. Furthermore, the cluster is described by means of the relevant keywords appearing in its documents. For point (b) (block (3) in Fig. 8.4), the list of documents belonging to the cluster is available, together with a BibTex description of the documents.
- **Document browsing** (block (2) in Fig. 8.4). A more detailed exploration of specific cluster contents may be performed by directly browsing documents. The available descriptive information on the selected document is shown. Furthermore, a summarization of the document to extract and highlight the most relevant sentences with respect to the domain of interest can be performed. Finally, documents may be individually excluded from the document collection on which both clustering and summarization are performed, thus further refining the analysis.

The system is available at [2].

8.3.2 System architecture

BioSumm architecture is shown in Figure 8.5. It is fully modular and allows the user to integrate plugins addressed to a specific task (e.g., clustering, web search, text summarization). Furthermore, by selecting the appropriate

domain dictionary, the grading function may be effectively tailored to the application domain of interest. In the following the main components of the framework are described.

Online search & local repository. Given a keyword query and a target search engine (or publication repository), this module executes the query on the selected engine and returns the set of retrieved documents. The demonstrated system integrates the plugins to translate the user keyword search for Google Scholar [7], PubMed Central (PMC) [15], and PubMed [13]. PubMed is a free search engine for accessing a number of databases of citations, abstracts and some full text articles on life sciences and biomedical topics. It is part of the Entrez information retrieval system, maintained by the U.S. National Library of Medicine (NLM). As of July 2009, PubMed contains more than 19 millions of citations. PubMed Central is a free digital database including only full text scientific literature in biomedical and life sciences, grown from the Entrez PubMed search system. Finally, Google Scholar is a freely-accessible Web search engine that indexes the full text of scholarly literature across an array of publishing formats and disciplines. Other repositories may be straightforwardly integrated to provide access to different, domain specific document sources.

Alternatively, the system also allows the user to select locally stored documents, possibly produced by previous search sessions. The local repository supports the pdf and nxml (i.e., xml for article full text, encoded in the NLM Journal Archiving and Interchange DTD [10]) formats.

Semantic extractor. The documents returned by a search session are parsed to extract the available components (e.g., title, authors, journal, abstract, body, keywords). The documents are then locally stored in a common representation in XML format.

Clustering. To reduce the heterogeneity of the retrieved documents, a clustering step can be optionally performed. The purpose of this block is to agglomerate documents covering related topics into homogeneous clusters. The document collection is represented as a matrix whose rows are the documents represented in the vector space model. Each cell of the vector contains the term frequency in the document as tf-idf (term frequency-inverse document frequency) [125]. The Bisecting K-means clustering method [132] is then applied to this representation. However, a different clustering technique may be easily integrated in the framework.

Dictionary evaluator. For each sentence of each paper, the semantic weights, which will bias the grading function of the summarizer, are computed according to the terms in the domain dictionary [22]. In the demo,

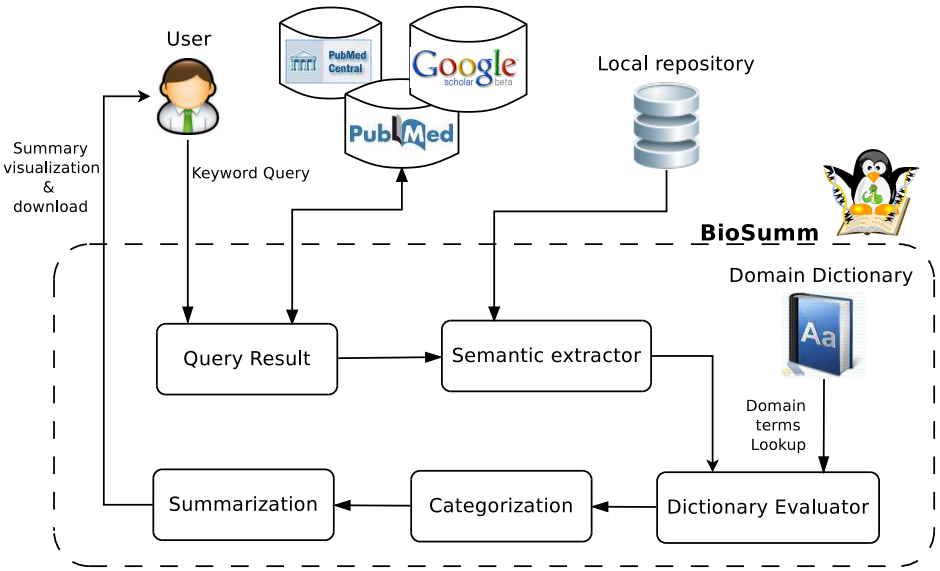


Figure 8.5: BioSumm system architecture

the provided domain dictionary contains terms describing human genes and proteins. The terms have been derived from the BioGrid database [130]. By modifying the entries in the domain dictionary, the user may customize the summarization for different domains (e.g., the financial domain).

Summarization. For each document cluster, a summary is extracted. The proposed summarization technique provides an ad-hoc multidocument summary based on a traditional statistical single-document summarizer [157], whose grading function has been tailored by means of the domain specific information stored in the domain dictionary. More details on the summarization approach, the configuration of the grading function parameters, and the evaluation of the summary quality are available in [22].

The BioSumm framework is developed in the C++ and Python languages. It provides a visual interface that allows a friendly interaction with the system without requiring specific technological competence or expert knowledge of the selected application domain (e.g., biology).

9

Conclusions

Analyzing different data sources in order to extract relevant information is a fundamental task in the bioinformatics domain in order to understand biological and biomedical processes. The aim of the thesis work was to provide data mining techniques in order to extract biological knowledge from different data sources (i.e., microarray data and document collection). Different data mining techniques were exploited with different aims. Feature selection analysis is a well-known approach to identify relevant genes for biological investigation (e.g., tumor diseases). Feature selection techniques have proved to be helpful in tumor classification and in understanding the genes related to clinical situations. Similarity measures and clustering approaches are powerful analyses to identify set of genes which have a similar behavior under different experimental conditions. Finally, summarization techniques allow to extract and present relevant information stored in documents which can be used for the validation of data analysis results.

In the thesis work, to analyze gene expression values measured for each gene under different experimental conditions (e.g., disease outcome), we introduced two representations: (i) the core expression interval and (ii) the gene mask. The core expression interval defines the interval in which the unseen expression values for a gene should lie according to the experimental condition. Since microarray data are noisy, a density based method, named WMD, was exploited in order to smooth outlier effect. The gene mask is a string representation of the capability of a gene in distinguishing sample classes. The computation of the gene mask associated to a gene depends on the definition used for the core expression interval. According to these representations we analyzed microarray data in order to: (i) identify a subset of gene that maximize the classification accuracy yielded on the training data, (ii) select the most relevant genes for classification task and (ii) group genes that share a similar behavior according to the gene mask representation.

The method presented in Chapter 5 automatically selects the minimum number of genes needed to reach a good classification accuracy. The minimum set of genes is obtained by applying two different approaches (i.e., greedy approach, set covering algorithm) to the gene mask representation. Experimental results show that our method reaches a very good accuracy with a low number of genes. These few genes can be used for further biological investigations.

In Chapter 6 we proposed a new method for feature selection on microarray data, the MaskedPainter. It allows (a) defining the minimum set of genes that provides a complete coverage of the training samples, and (b) ranking genes by decreasing relevance, thus allowing the user to customize the final size of the feature set. The MaskedPainter method has been compared with six other feature selection techniques on both binary and multiclass microarray datasets. In most experimental settings it yields the best accuracy, while its computational cost is similar to the other feature selection techniques. On the Alon dataset, the identified relevant genes are consistent with the literature on tumor classification. Hence, the MaskedPainter approach may provide a useful tool both to identify relevant genes for tumor diseases and to improve the classification accuracy of a classifier.

In Chapter 7 we proposed a new similarity measure between genes, the *classification distance*, that exploits additional information which may be available on microarray data (e.g., tumor or patient classification). The discrimination ability of each gene is represented by means of a gene mask, which describes the gene classification power, i.e., its capability to correctly classify samples. The classification distance measures gene similarity by analyzing their masks, i.e., their capability of correctly classifying the same samples. The classification distance measure can be integrated in different clustering approaches. We have integrated it into a hierarchical clustering algorithm, by introducing the notion of cluster mask as representative of a cluster and defining as inter-cluster distance the distance between cluster masks. We validated our method on both binary and multiclass microarray datasets. The experimental results show the ability of the classification distance to group genes with similar classification power and similar biological meaning in the tumor context.

Since documents are unstructured data, a different technique was proposed to extract biological knowledge. The aim of this work was also to provide an automatic tool useful to researchers that discover gene correlations by means of analysis tools(e.g., feature selection, clustering). In Chapter 8, we proposed the BioSumm approach. In contrast to general purpose

summarizer, BioSumm generates ad hoc document summaries oriented to biological content, in particular to gene and protein information. It is a multi-document summarizer based on a single document summarizer and on statistical methods. We defined a new grading function to give more weight to the sentences which contain more biological information, especially gene and protein interactions. We validated our approach on PubMed document collections and the experimental results show the ability of BioSumm to summarize large collections of unclassified data by extracting the sentences that are more relevant for knowledge inference and biological validation of gene/protein relationships. Although focused on a specific subject, its capability to detect the sentences that better cover the major topics of a group of documents is still preserved. BioSumm is able to find a suitable balance between descriptive information and domain specific content. Moreover, our system achieves better performance on single documents than a general purpose summarizer. Thus, researchers may exploit this framework to effectively support the biological validation of their results.

Currently, we are investigating to improve the techniques presented in the thesis work under different aspects. Feature selection approaches should be improved in term of classification accuracy and biological meaning of selected genes by means of the integration of the knowledge store in biological ontologies like UMLS. Similarity measures based on the classification distance and the ontology knowledge should be applied on microarray data to improve the understanding of genes and proteins under different experimental conditions.

Moreover, we believe that the MaskedPainter method may be applied to any dataset characterized by noisy and continuously valued features and the classification distance measure may be applied also in other application domains with the same characteristics (e.g., user profiling, hotel ranking, etc.), to improve the clustering results by exploiting additional information available on the data being clustered.

Finally, we are planning to improve each block of BioSumm framework. We are studying a more efficient categorization block oriented to biological content and the integration of semantic analysis for the summarization block. Moreover, we are developing research engines to query the summary generated for each sub-collection in order to help researchers to find their topics of interest.

A

MaskedPainter experimental results

We report the results of the classification accuracy yielded by the MaskedPainter presented in Chapter 6 and the other 6 feature selection methods on the Brain1, Brain2, Tumor9, and Welsh data sets, which were not reported in Section 6.2.2.

#	MP	IG	TR	SM	MM	GI	SV
2	70.27	68.89*	67.62*	**70.65**	68.88*	68.42*	69.01*
4	**72.19**	70.45*	69.99*	71.36	69.93*	69.72*	71.01
6	**73.22**	71.41*	70.76*	71.25*	70.50*	70.84*	71.25*
8	**73.11**	72.35	71.09*	71.13*	70.95*	70.78*	72.43
10	**73.25**	72.38	71.48*	71.16*	71.25*	71.27*	72.61
12	**73.25**	72.79	72.06	71.64*	71.97	71.77*	72.74
14	**73.39**	73.19	72.55	72.16	71.76*	71.60*	72.97
16	**73.54**	73.46	72.95	72.77	71.70*	72.22	73.25
18	**74.11**	73.71	73.58	73.03	72.20*	72.79	73.42
20	**74.49**	73.80	73.65	73.22*	71.72*	73.05*	73.37
avg	**73.08**	72.24*	71.57*	71.84*	71.09*	71.25*	72.21*
max	**74.49**	73.80	73.65	73.22	72.20	73.05	73.42
dev	1.10	1.50	1.74	0.85	0.99	1.33	1.32

Table A.1: Accuracy yielded by the J48 classifier on the Brain1 dataset.

#	MP	IG	TR	SM	MM	GI	SV
2	**57.64**	46.80*	46.80*	46.13*	49.03*	49.35*	46.80*
4	**58.23**	46.11*	46.11*	46.15*	51.78*	49.38*	46.11*
6	**58.83**	45.84*	45.84*	48.45*	53.12*	49.01*	45.84*
8	**59.19**	46.77*	46.77*	49.02*	54.66*	49.33*	46.77*
10	**59.43**	48.16*	48.16*	51.29*	55.51*	50.29*	48.16*
12	**59.27**	49.65*	49.53*	54.30*	56.41*	51.11*	49.65*
14	**59.73**	50.00*	49.96*	55.87*	56.61*	51.50*	49.92*
16	**60.04**	50.58*	50.26*	56.77*	57.15*	52.73*	50.33*
18	**59.85**	50.94*	51.20*	56.96*	57.33*	53.48*	50.58*
20	**59.62**	51.27*	50.90*	57.24*	57.75	54.23*	50.91*
avg	**59.18**	48.61*	48.55*	52.22*	54.93*	51.04*	48.51*
max	**60.04**	51.27	51.20	57.24	57.75	54.23	50.91
dev	0.72	2.00	1.95	4.31	2.68	1.80	1.89

Table A.2: Accuracy yielded by the J48 classifier on the Brain2 dataset.

#	MP	IG	TR	SM	MM	GI	SV
2	**25.40**	24.30	23.03*	21.47*	18.63*	21.13*	24.77
4	29.33	28.77	28.30	24.00*	20.80*	23.43*	**29.77**
6	30.97	30.43	30.47	25.83*	20.73*	24.33*	**31.27**
8	31.03	**32.30**	31.07	28.00*	21.40*	24.67*	32.17
10	31.57	**32.77**	31.63	28.27*	22.37*	26.60*	31.50
12	32.03	**32.60**	31.40	29.80*	21.87*	26.97*	31.60
14	31.97	**32.77**	30.97	29.77*	21.13*	27.33*	31.80
16	31.97	**32.83**	30.97	28.50*	22.23*	26.90*	31.17
18	32.07	**32.87**	31.23	29.60*	23.27*	27.63*	30.33
20	33.03	**33.07**	30.80*	29.63*	23.43*	27.70*	30.90*
avg	30.94	**31.27**	29.99*	27.49*	21.59*	25.67*	30.53
max	33.03	**33.07**	31.63	29.80	23.43	27.70	32.17
dev	2.06	2.67	2.48	2.70	1.33	2.08	2.03

Table A.3: Accuracy yielded by the J48 classifier on the Tumor9 dataset.

#	MP	IG	TR	SM	MM	GI	SV
2	**89.72**	85.81*	85.81*	85.81*	85.81*	85.81*	85.81*
4	**90.21**	85.74*	85.74*	85.74*	85.74*	85.74*	85.74*
6	**90.03**	84.72*	84.72*	84.72*	84.72*	84.72*	84.72*
8	**90.24**	85.08*	85.08*	85.08*	85.08*	85.08*	85.08*
10	**89.56**	84.26*	84.26*	84.26*	84.26*	84.26*	84.26*
12	**90.10**	83.58*	83.58*	83.58*	83.58*	83.58*	83.58*
14	**89.97**	83.26*	83.26*	83.26*	83.26*	83.26*	83.26*
16	**90.08**	83.30*	83.30*	83.30*	83.30*	83.30*	83.30*
18	**89.56**	83.31*	83.31*	83.31*	83.31*	83.31*	83.31*
20	**89.30**	83.23*	83.23*	83.23*	83.23*	83.23*	83.23*
avg	**89.88**	84.23*	84.23*	84.23*	84.23*	84.23*	84.23*
max	**90.24**	85.81	85.81	85.81	85.81	85.81	85.81
dev	0.30	0.99	0.99	0.99	0.99	0.99	0.99

Table A.4: Accuracy yielded by the J48 classifier on the Welsh dataset.

Index

List of Figures

List of Tables

Bibliography

[1] Bioline international. http://www.bioline.org.br/.

[2] BioSumm. https://dbdmg.polito.it/twiki/bin/view/Public/BioSumm.

[3] Canadian breast cancer research alliance open access archive. https://researchspace.library.utoronto.ca/handle/1807.1/1.

[4] Cluto - software for clustering high-dimensional datasets. http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview.

[5] Dspace iss. http://dspace.iss.it/dspace/.

[6] E-ms - archivo aperto di documenti per la medicina sociale. http://e-ms.cilea.it/.

[7] Google Scholar. http://scholar.google.com.

[8] Ispub - internet scientific publications. http://www.ispub.com.

[9] Mesh - medical subject headings. http://www.nlm.nih.gov/mesh/.

[10] Nlm journal archiving and interchange tag suite. http://dtd.nlm.nih.gov/.

[11] Open text summarizer. http://libots.sourceforge.net/.

[12] Plos - public library of science. http://www.plos.org/.

[13] PubMed. http://www.ncbi.nlm.nih.gov/pubmed/.

[14] Pubmed central ftp service. http://www.pubmedcentral.nih.gov/about/ftp.html#Source_files.

[15] PubMed Medical Center. http://www.ncbi.nlm.nih.gov/pmc/.

[16] Tac - text analysis conference. http://www.nist.gov/tac/.

[17] Wiley interscience. http://www3.interscience.wiley.com/.

[18] S.M. Alladi, P. Shinde Santosh, V. Ravi, and U.S. Murthy. Colon cancer prediction with genetic profiles using intelligent techniques. *Bioinformation*, 3(3):130, 2008.

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

[20] D. Apiletti, E. Baralis, G. Bruno, and A. Fiori. The Painter's Feature Selection for Gene Expression Data. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, pages 4227–4230, 2007.

[21] W.H. Au, K.C.C. Chan, A.K.C. Wong, and Y. Wang. Attribute clustering for grouping, selection, and classification of gene expression data. *IEEE/ACM transactions on computational biology and bioinformatics*, pages 83–101, 2005.

[22] E. Baralis, A. Fiori, and L. Montrucchio. Biosumm: a novel summarizer oriented to biological information. *Proceedings of 8th IEEE International Conference on BioInformatics and BioEngineering (BIBE08)*, 2008.

[23] G. Barnett and C. Friedrich. Recent developments in ovarian cancer genetics. *Curr Opin Obstet Gynecol*, 16(1):79–85, 2004.

[24] M. Bella, R. Camisa, and S. Cascinu. Molecular profile and clinical variables in brca1 positive breast cancers. a population based study. *Tumori*, 91:505–512, 2005.

[25] A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, and Z. Yakhini. Tissue classification with gene expression profiles. *Journal of Computational Biology*, 7(3-4):559–583, 2000.

[26] S. Bergmann, J. Ihmels, and N. Barkai. Iterative signature algorithm for the analysis of large-scale gene expression data. *Physical review E*, 67(3):31902, 2003.

[27] F. Bertucci, S. Salas, S. Eysteries, V. Nasser, P. Finetti, C. Ginestier, E. Charafe-Jauffret, B. Loriod, L. Bachelart, and J. Montfort. Gene expression profiling of colon cancer by DNA microarrays and correlation with histoclinical parameters. *Oncogene*, 23(7):1377–1391, 2004.

[28] T. Bo and I. Jonassen. New feature subset selection procedures for classification of expression profiles. *Genome biology*, 3(4):0017, 2002.

[29] U.M. Braga-Neto and E.R. Dougherty. Is cross-validation valid for small-sample microarray classification? *Bioinformatics*, 20(3):374, 2004.

[30] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[31] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.

[32] P.R. Bushel, R.D. Wolfinger, and G. Gibson. Simultaneous clustering of gene expression data with clinical chemistry and pathological evaluations reveals phenotypic prototypes. *BMC Systems Biology*, 1(1):15, 2007.

[33] C.C. Chang and C.J. Lin. Training v-support vector classifiers: theory and algorithms, 2001. `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[34] J.J. Chen, C.A. Tsai, S.L. Tzeng, and C.H. Chen. Gene selection with multiple ordering criteria. *BMC bioinformatics*, 8(1):74, 2007.

[35] J.L. Chen and B.W. Futscher. Optimal search-based gene subset selection for gene array cancer classification. *IEEE Transactions on Information Technology in Biomedicine*, 11(4):398–405, 2007.

[36] Y. Cheng and G.M. Church. Biclustering of expression data. In *Proc Int Conf Intell Syst Mol Biol*, volume 8, pages 93–103, 2000.

[37] D. Daniel. Highlight: Brca1 and brca2 proteins in breast cancer. *Microsc Res Tech*, 59(1):68–83, 2002.

[38] S. Datta and S. Datta. Evaluation of clustering algorithms for gene expression data. *BMC bioinformatics*, 7(Suppl 4):S17, 2006.

[39] L. Davies and U. Gather. The identification of multiple outliers. *Journal of the American Statistical Association*, 88(423):782–792, 1993.

[40] D. Dembele and P. Kastner. Fuzzy C-means method for clustering microarray data. *Bioinformatics*, 19(8):973, 2003.

[41] R. Díaz-Uriarte and A. de Andrés. Gene selection and classification of microarray data using random forest. *BMC bioinformatics*, 7(1):3, 2006.

[42] C. Ding and H. Peng. Minimum redundancy feature selection from microarray gene expression data. *Journal of Bioinformatics and Computational Biology*, 3(2):185–206, 2005.

[43] C.H.Q. Ding. Unsupervised feature selection via two-way ordering in gene expression analysis. *Bioinformatics*, 19(10):1259, 2003.

[44] M. Draminski, A. Rada-Iglesias, S. Enroth, C. Wadelius, J. Koronacki, and J. Komorowski. Monte Carlo feature selection for supervised classification. *Bioinformatics*, 24(1):110, 2008.

[45] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern classification*. Citeseer, 2001.

[46] S. Dudoit, J. Fridlyand, and T.P. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, 97(457):77–88, 2002.

[47] H. P. Edmundson and R. E. Wyllys. Automatic abstracting and indexing survey and recommendations. *Commun. ACM*, 4(5):226–234, 1961.

[48] M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863, 1998.

[49] M. Ester, H. Kriegel, S. Jörg, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pages 226–231, 1996.

[50] R. Feldman and I. Dagan. Knowledge discovery in textual databases (kdt). In *Knowledge Discovery and Data Mining*, pages 112–117, 1995.

[51] R. Feldman and J. Sanger. *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, 2007.

[52] L. Fu and E. Medico. FLAME, a novel fuzzy clustering method for the analysis of DNA microarray data. *BMC bioinformatics*, 8(1):3, 2007.

[53] C. Furlanello, M. Serafini, S. Merler, and G. Jurman. Entropy-based gene ranking without selection bias for the predictive classification of microarray data. *BMC bioinformatics*, 4(1):54, 2003.

[54] M.K. Ganapathiraju. Relevance of cluster size in MMR based summarizer: a report. *Advisors: Carbonell, J. and Yang, Y*, 2002.

[55] C.P. Giacomini, S.Y. Leung, X. Chen, S.T. Yuen, Y.H. Kim, E. Bair, and J.R. Pollack. A gene expression signature of genetic instability in colon cancer, 2005.

[56] TR Golub, DK Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, JP Mesirov, H. Coller, ML Loh, JR Downing, MA Caligiuri, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531, 1999.

[57] E. Greenwood. Tumour suppressors: Unfold with brca1. *Nature Reviews Cancer*, 2(8), January 2002.

[58] R. Gruber. A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2):199–220, 1993.

[59] J. Gu and J. Liu. Bayesian biclustering of gene expression data. *BMC genomics*, 9(Suppl 1):S4, 2008.

[60] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1):389–422, 2002.

[61] B. Hanczar, M. Courtine, A. Benis, C. Hennegar, K. Clément, and J.D. Zucker. Improving classification of microarray data using prototype-based feature selection. *ACM SIGKDD Explorations Newsletter*, 5(2):23–30, 2003.

[62] JA Hartigan. Direct clustering of a data matrix. *Journal of the American Statistical Association*, pages 123–129, 1972.

[63] X. He, D. Cai, and P. Niyogi. Laplacian score for feature selection. *Advances in Neural Information Processing Systems*, 18:507, 2006.

[64] T. Hernandez and S. Kambhampati. Integration of biological sources: current systems and challenges ahead. *SIGMOD Rec*, 33(3):51–60, 2004.

[65] WR Hersh. *Information retrieval: a health and biomedical perspective*. Springer Verlag, 2008.

[66] R. Hoffmann and A. Valencia. A gene network for navigating the literature. *Nature Genetics*, 36:664, 2004. `http://www.ihop-net.org/`.

[67] R. Hoffmann and A. Valencia. Implementing the iHOP concept for navigation of biomedical literature. *BIOINFORMATICS-OXFORD-*, 21(2), 2005.

[68] E. Honrado, A. Osorio, J. Palacios, and J. Benitez. Pathology and gene expression of hereditary breast tumors associated with brca1, brca2 and chek2 gene mutations. *Oncogene*, 25:5837–5845, 2006.

[69] A. Hotho, A. Nürnberger, and G. Paaß. A brief survey of text mining. *GLDV Journal for Computational Linguistics and Language Technology*, 20(1):19–62, may 2005.

[70] J. Hua, W.D. Tembe, and E.R. Dougherty. Performance of feature-selection methods in the classification of high-dimension data. *Pattern Recognition*, 42(3):409–424, 2009.

[71] D. Huang and W. Pan. Incorporating biological knowledge into distance-based clustering analysis of microarray gene expression data. *Bioinformatics*, 22(10):1259, 2006.

[72] J. Jäger, R. Sengupta, and W.L. Ruzzo. Improved gene selection for classification of microarrays. In *Biocomputing 2003: Proceedings of the Pacific Symposium Hawaii, USA 3-7 January 2003*, page 53. World Scientific Pub Co Inc, 2002.

[73] I.B. Jeffery, D.G. Higgins, and A.C. Culhane. Comparison and evaluation of methods for generating differentially expressed gene lists from microarray data. *BMC bioinformatics*, 7(1):359, 2006.

[74] D. Jiang, J. Pei, and A. Zhang. DHC: A density-based hierarchical clustering method for time series gene expression data. In *Proceedings of the 3rd IEEE Symposium on BioInformatics and BioEngineering*, page 393. IEEE Computer Society Washington, DC, USA, 2003.

[75] D. Jiang, C. Tang, and A. Zhang. Cluster analysis for gene expression data: A survey. *IEEE Transactions on Knowledge and Data Engineering*, pages 1370–1386, 2004.

[76] H. Jiang, Y. Deng, H.S. Chen, L. Tao, Q. Sha, J. Chen, C.J. Tsai, and S. Zhang. Joint analysis of two microarray gene-expression data sets to select lung adenocarcinoma marker genes. *BMC bioinformatics*, 5(1):81, 2004.

[77] W. Jiang, X. Li, S. Rao, L. Wang, L. Du, C. Li, C. Wu, H. Wang, Y. Wang, and B. Yang. Constructing disease-specific gene networks using pair-wise relevance metric: application to colon cancer identifies interleukin 8, desmin and enolase 1 as the central elements. *BMC Systems Biology*, 2(1):72, 2008.

[78] T. Juliusdottir, E. Keedwell, D. Corne, and A. Narayanan. Two-phase EA/k-NN for feature selection and classification in cancer microarray datasets. In *Computational Intelligence in Bioinformatics and Computational Biology, 2005. CIBCB'05. Proceedings of the 2005 IEEE Symposium on*, pages 1–8, 2005.

[79] G. Karakiulakis, C. Papanikolaou, SM Jankovic, A. Aletras, E. Papakonstantinou, E. Vretou, and V. Mirtsou-Fidani. Increased type IV collagen-degrading activity in metastases originating from primary tumors of the human colon. *Invasion and metastasis*, 17(3):158, 1997.

[80] L. Kaufman and P.J. Rousseeuw. *Finding groups in data: An introduction to cluster analysis*. Wiley, New York, 1990.

[81] H. Kim, G.H. Golub, and H. Park. Missing value estimation for DNA microarray gene expression data: local least squares imputation. *Bioinformatics*, 21(2):187, 2005.

[82] H. Kishino and P.J. Waddell. Correspondence analysis of genes and tissue types and finding genetic links from microarray data. *GENOME INFORMATICS SERIES*, pages 83–95, 2000.

[83] M. Krallinger, A. Valencia, and L. Hirschman. Linking genes to literature: text mining, information extraction, and retrieval applications for biology. *Genome biology*, 9(Suppl 2):S8, 2008.

[84] C. Lai, M. Reinders, and LFA Wessels. Multivariate gene selection: Does it help? In *IEEE Computational Systems Biology Conference, Stanford*. Citeseer, 2005.

[85] C. Lai, M.J.T. Reinders, L.J. van't Veer, L.F.A. Wessels, et al. A comparison of univariate and multivariate gene selection techniques for classification of cancer datasets. *BMC bioinformatics*, 7(1):235, 2006.

[86] T. K. Landauer, P. W. Foltz, and D. Laham. Introduction to latent semantic analysis. *Discourse Processes*, 25:259–284, 1998.

[87] D. Lewandowski. Web searching, search engines and information retrieval. *Information Services and Use*, 25(3), 2005.

[88] S. Li, Y. Ouyang, W. Wang, and B. Sun. Multi-document summarization using support vector regression. *Proceedings of Document Understanding Conference (DUC 07)*, 2007.

[89] T. Li, C. Zhang, and M. Ogihara. A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression. *Bioinformatics*, 20(15):2429, 2004.

[90] C.Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*, pages 25–26, 2004.

[91] R. Linder, T. Richards, and M. Wagner. Microarray data classified by artificial neural networks. *METHODS IN MOLECULAR BIOLOGY-CLIFTON THEN TOTOWA-*, 382:345, 2007.

[92] N. Ling and Y. Hasan. Classification on microarray data. *Proceedings of the 2nd IMT-GT Regional Conference on Mathematics, Statistics and Applications*, 2006.

[93] L. Loo, S. Roberts, L. Hrebien, and M. Kam. New criteria for selecting differentially expressed genes. *IEEE Engineering in Medicine and Biology Magazine*, 26(2):17, 2007.

[94] J. Macqueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1967, pages 281–297, 1967.

[95] I. Mani. Summarization evaluation: An overview. In *Proceedings of the Second NTCIR Workshop on Research in Chinese & Japanese Text Retrieval and Text Summarization*, 2001.

[96] R.S. Michalski and K. Kaufman. Learning patterns in noisy data: The AQ approach. *Machine Learning and its Applications, Springer-Verlag*, pages 22–38, 2001.

[97] S. Michielis, S. Koscielny, and C. Hill. Prediction of cancer outcome with microarrays: a multiple random validation strategy. *Lancet*, 365:488–492, 2005.

[98] I. Mierswa, M. Wurst, R. Klinkenberg, M. Scholz, and T. Euler. Yale: Rapid prototyping for complex data mining tasks. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-06)*, 2006. `http://rapid-i.com/`.

[99] P. Mitra, CA Murthy, and S.K. Pal. Unsupervised feature selection using feature similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 301–312, 2002.

[100] M. Moty Ben-Dov and R. Feldman. Text mining and information extraction. In *The Data Mining and Knowledge Discovery Handbook*, pages 801–831. 2005.

[101] S. Mukkamala, Q. Liu, R. Veeraghattamand, and A. Sung. *Feature Selection and Ranking of Key Genes for Tumor Classification: Using Microarray Gene Expression Data*. Springer Berlin/Heidelberg, 2006.

[102] B. Ni and J. Liu. A hybrid filter/wrapper gene selection method for microarray classification. In *Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on*, volume 4, 2004.

[103] S. Niijima and Y. Okuno. Laplacian linear discriminant analysis approach to unsupervised feature selection. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 99(1), 2007.

[104] J.D. Osborne, L. Zhu, S.M. Lin, and W.A. Kibbe. Interpreting microarray results with gene ontology and MeSH. *METHODS IN MOLECULAR BIOLOGY-CLIFTON THEN TOTOWA-*, 377:223, 2007.

[105] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[106] G. Papachristoudis, S. Diplaris, and P.A. Mitkas. SoFoCles: Feature filtering for microarray classification based on Gene Ontology. *Journal of Biomedical Informatics*, 2009.

[107] R.K. Pearson, G.E. Gonye, and J.S. Schwaber. Outliers in microarray data analysis. *Proc. Critical Assessment of Microarray Data Analysis, CAMDA-02*.

[108] H. Peng, F. Long, and C. 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.

[109] G. Piatetsky-Shapiro and P. Tamayo. Microarray data mining: facing the challenges. *ACM SIGKDD Explorations Newsletter*, 5(2):1–5, 2003.

[110] M. Pirooznia, J.Y. Yang, M.Q. Yang, and Y. Deng. A comparative study of different machine learning methods on microarray gene expression data. *BMC genomics*, 9(Suppl 1):S13, 2008.

[111] J. Qi and J. Tang. Integrating gene ontology into discriminative powers of genes for feature selection in microarray data. In *Proceedings of the 2007 ACM symposium on Applied computing*, page 434. ACM, 2007.

[112] J.R. Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

[113] J.R. Quinlan. C4. 5: Programming for machine learning. *Morgan Kauffmann*, 1993.

[114] D.R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22, 2004.

[115] D.R. Radev, S. Blair-Goldensohn, and Z. Zhang. Experiments in single and multi-document summarization using mead. *Proceedings of Document Understanding Conference (DUC 01)*, 2001.

[116] T. K. Ralphs and M. GÃ$\frac{1}{4}$zelsoy. The symphony callable library for mixed integer programming. *The Next Wave in Computing, Optimization, and Decision Technologies*, 29:61–76, 2006. Software available at `http://http://www.coin-or.org/SYMPHONY`.

[117] W.M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, pages 846–850, 1971.

[118] K. Rapakko, K. Heikkinen, S. Karppinen, H. Erkko, and R. Winqvist. Germline alterations in the 53bp1 gene in breast and ovarian cancer families. *Cancer Letters*, 245:337–340, 2003.

[119] L. Reeve, H. Han, and A.D. Brooks. BioChain: lexical chaining methods for biomedical text summarization. In *Proceedings of the 2006 ACM symposium on Applied computing*, pages 180–184. ACM New York, NY, USA, 2006.

[120] L. H. Reeve, H. Han, and A. D. Brooks. The use of domain-specific concepts in biomedical text summarization. *Information Processing and Management: an International Journal*, 43(6), 2007.

[121] A.L. Richards, P. Holmans, M.C. O'Donovan, M.J. Owen, and L. Jones. A comparison of four clustering methods for brain expression microarray data. *BMC bioinformatics*, 9(1):490, 2008.

[122] R. Ruiz, J.C. Riquelme, and J.S. Aguilar-Ruiz. Incremental wrapper-based gene selection from microarray data for cancer classification. *Pattern Recognition*, 39(12):2383–2392, 2006.

[123] Y. Saeys, I. Inza, and P. Larranaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507, 2007.

[124] H. Saggion and R. Gaizauskas. Multi-document summarization by cluster/profile relevance and redundancy removal. *Proceedings of Document Understanding Conference (DUC 04)*, 2004.

[125] G. Salton, A. Wong, and CS Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):620, 1975.

[126] E. Segal, A. Battle, and D. Koller. Decomposing gene expression into cellular processes. In *Biocomputing 2003: Proceedings of the Pacific Symposium Hawaii, USA 3-7 January 2003*, page 89. World Scientific Pub Co Inc, 2002.

[127] K. Shailubhai, H.H. Yu, K. Karunanandaa, J.Y. Wang, S.L. Eber, Y. Wang, N.S. Joo, H.D. Kim, B.W. Miedema, and S.Z. Abbas. Uroguanylin treatment suppresses polyp formation in the Apc Min/+ mouse and induces apoptosis in human colon adenocarcinoma cells via cyclic GMP. *Cancer research*, 60(18):5151–5157, 2000.

[128] R. Shamir and R. Sharan. Algorithmic approaches to clustering gene expression data. *Current Topics in Computational Biology*, 2002.

[129] M. Shipp and al. Diffuse large b-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. *Nature Medicine*, 8(1):68,74, Jan. 2002.

[130] C. Stark, B. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers. Biogrid: a general repository for interaction datasets. *Nucleic Acids Research*, 34, 2006. `http://www.thebiogrid.org/`.

[131] A. Statnikov, C.F. Aliferis, I. Tsamardinos, D. Hardin, and S. Levy. A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. *Bioinformatics*, 21(5):631, 2005.

[132] M. Steinbach, G. Karypis, and V. Kumar. A comparison of document clustering techniques. *Proceedings of KDD Workshop on Text Mining*, 2006.

[133] D. Stekel. *Microarray bioinformatics*. Cambridge Univ Pr, 2003.

[134] Y. Su, TM Murali, V. Pavlovic, M. Schaffer, and S. Kasif. RankGene: identification of diagnostic genes based on expression data. *Bioinformatics*, 19(12):1578, 2003.

[135] P. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining*. Addison Wesley, 2005.

[136] Q. Tao, J. Ren, and J. Li. Vasoactive Intestinal Peptide Inhibits Adhesion Molecule Expression in Activated Human Colon Serosal Fibroblasts by Preventing NF-κB Activation. *Journal of Surgical Research*, 140(1):84–89, 2007.

[137] L. Tari, C. Baral, and S. Kim. Fuzzy c-means clustering with prior biological knowledge. *Journal of Biomedical Informatics*, 42(1):74–81, 2009.

[138] A. Thalamuthu, I. Mukhopadhyay, X. Zheng, and G.C. Tseng. Evaluation and comparison of gene clustering methods in microarray analysis. *Bioinformatics*, 22(19):2405, 2006.

[139] R.C. Thompson, M. Deo, and D.L. Turner. Analysis of microRNA expression by in situ hybridization with RNA oligonucleotide probes. *Methods*, 43(2):153–161, 2007.

[140] C.J. van Rijsbergen, S.E. Robertson, and M.F. Porter. New models in probabilistic information retrieval. *British Library Research and Development Report*, (5587), 1980. `http://tartarus.org/~martin/PorterStemmer/`.

[141] V. Vapnik. Statistical learning theory. 1998.

[142] R. Varshavsky, A. Gottlieb, M. Linial, and D. Horn. Novel unsupervised feature filtering of biological data. *Bioinformatics*, 22(14):e507, 2006.

[143] R. Verma, P. Chen, and W. Lu. A semantic free-text summarization system using ontology knowledge. *Proceedings of Document Understanding Conference (DUC 07)*, 2007.

[144] H. Wang, W. Wang, J. Yang, and P.S. Yu. Clustering by pattern similarity in large data sets. In *Proceedings of the 2002 ACM SIGMOD international conference on Management of data*, page 405. ACM, 2002.

[145] S. Wang, H. Chen, and S. Li. Gene Selection Using Neighborhood Rough Set from Gene Expression Profiles. In *Proceedings of the 2007 International Conference on Computational Intelligence and Security*, pages 959–963. IEEE Computer Society Washington, DC, USA, 2007.

[146] Y. Wang, F.S. Makedon, J.C. Ford, and J. Pearlman. HykGene: a hybrid approach for selecting marker genes for phenotype classification using microarray gene expression data. *Bioinformatics*, 21(8):1530, 2005.

[147] Z. Wang, P. Yan, D. Potter, C. Eng, T.H. Huang, and S. Lin. Heritable clustering and pathway discovery in breast cancer integrating epigenetic and phenotypic data. *BMC bioinformatics*, 8(1):38, 2007.

[148] S. Weiss, N. Indurkhya, T. Zhang, and F. Damerau. *Text Mining: Predictive Methods for Analyzing Unstructured Information*. SpringerVerlag, 2004.

[149] J.B. Welsh, L.M. Sapinoso, A.I. Su, S.G. Kern, J. Wang-Rodriguez, C.A. Moskaluk, H.F. Frierson, and G.M. Hampton. Analysis of gene expression identifies candidate markers and pharmacological targets in prostate cancer, 2001.

[150] L. Wessels, M. Reinders, T. van Welsem, and P. Nederlof. Representation and classification for high-throughput data. In *Proceedings of SPIE*, volume 4626, page 226, 2002.

[151] R. Witte, R. Krestel, and S. Bergler. Context-based multi-document summarization using fuzzy coreference cluster graphs. *Proceedings of Document Understanding Conference (DUC 06)*, 2006.

[152] I.H. Witten and E. Frank. Data Mining: Practical machine learning tools and techniques, 2005. `http://www.cs.waikato.ac.nz/ml/weka/`.

[153] F.X. Wu. Genetic weighted k-means algorithm for clustering large-scale gene expression data. *BMC bioinformatics*, 9(Suppl 6):S12, 2008.

[154] M. Xiong, X. Fang, and J. Zhao. Biomarker identification by feature wrappers, 2001.

[155] P. Yang and Z. Zhang. Hybrid Methods to Select Informative Gene Sets in Microarray Data Classification. *Lecture Notes in Computer Science*, 4830:810, 2007.

[156] Y.L. Yap, X.W. Zhang, M.T. Ling, X.H. Wang, Y.C. Wong, and A. Danchin. Classification between normal and tumor tissues based on the pair-wise gene expression ratio. *BMC cancer*, 4(1):72, 2004.

[157] V. A. Yatsko and T. N. Vishnyakov. A method for evaluating modern systems of automatic text summarization. *Automatic Documentation and Mathematical Linguistics*, 41(3):93–103, 2007.

[158] K.Y. Yeung and R.E. Bumgarner. Multiclass classification of microarray data with repeated measurements: application to cancer. *Genome Biology*, 4(12):83–83, 2003.

[159] I. Yoo and X. Hu. Empirical and theoretical comparisons of selected criterion functions for document clustering. *Machine Learning*, 55:365–371, 2004.

[160] I. Yoo and X. Hu. A comprehensive comparison study of document clustering for a biomedical digital library medline. *Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries*, 2006.

[161] I. Yoo, X. Hu, and I. Song. Integrating biomedical literature clustering and summarization approaches using biomedical ontology. *Proceedings of the 1st international workshop on Text mining in bioinformatics*, 2006.

[162] E. Young. A polymorphism in the cyp17 gene is associated with male breast cancer. *Br J Cancer*, 1999.

[163] J. Yu, F. Cheng, H. Xiong, W. Qu, and X. Chen. A Bayesian approach to support vector machines for the binary classification. *Neurocomputing*, 72(1-3):177–185, 2008.

[164] L.T.H. Yu, F. Chung, S.C.F. Chan, and S.M.C. Yuen. Using emerging pattern based projected clustering and gene expression data for cancer detection. In *Proceedings of the second conference on Asia-Pacific bioinformatics-Volume 29*, pages 75–84. Australian Computer Society, Inc. Darlinghurst, Australia, Australia, 2004.

[165] M. Zahurak, G. Parmigiani, W. Yu, R.B. Scharpf, D. Berman, E. Schaeffer, S. Shabbeer, and L. Cope. Pre-processing Agilent microarray data. *BMC bioinformatics*, 8(1):142, 2007.

[166] Y. Zhao and G. Karypis. Criterion functions for document clustering: Experiments and analysis. *Machine Learning*, 2001.

[167] G. Zheng, E.O. George, and G. Narasimhan. Neural network classifiers and gene selection methods for microarray data on human lung adenocarcinoma. *Methods of Microarray Data Analysis IV*, 2005.

[168] Q. Zheng and X.J. Wang. GOEAST: a web-based software toolkit for Gene Ontology enrichment analysis. *Nucleic acids research*, 36(Web Server issue):W358, 2008.

[169] X. Zhou and D.P. Tuck. MSVM-RFE: extensions of SVM-RFE for multiclass gene selection on DNA microarray data. *Bioinformatics*, 23(9):1106, 2007.