

# An Interval Tree Based Feature Reduction Method for Cancer Classification using High-Throughput DNA Copy Number Data

**Siling Wang**

Department of Computer Science and Engineering  
Southern Methodist University  
Dallas, TX 75205, USA  
swang@engr.smu.edu

**Yuhang Wang**

Department of Computer Science and Engineering  
Southern Methodist University  
Dallas, TX 75205, USA  
yuhangw@engr.smu.edu

**Luc Girard**

Hamon Center for Therapeutic Oncology Research  
UT Southwestern Medical Center at Dallas  
Dallas, TX 75390, USA  
Luc.Girard@utsouthwestern.edu

**Young Kim**

Department of Pathology  
Stanford University School of Medicine  
269 Campus Drive, CCSR 3245A  
Stanford, CA 94305, USA  
yhkim100@stanford.edu

**Jonathan R. Pollack**

Department of Pathology  
Stanford University School of Medicine  
269 Campus Drive, CCSR 3245A  
Stanford, CA 94305, USA  
pollack1@stanford.edu

**John D. Minna**

Hamon Center for Therapeutic Oncology Research  
UT Southwestern Medical Center at Dallas  
Dallas, TX 75390, USA  
John.Minna@utsouthwestern.edu

*Abstract* Cancer classification using DNA copy number data is an important bioinformatics problem. Effective machine learning models for this task can be useful not only for cancer diagnosis, but also for discovering novel tumor suppressor genes and oncogenes. The recent array-based assays that detect DNA copy numbers contain very large numbers of probes and thus generate data of extremely high dimensionality. Therefore, the use of appropriate feature reduction methods is called for. In this paper, we proposed an efficient interval tree based feature reduction method for cancer classification using DNA copy number data. Instead of using probes as features, our approach extracts intervals as features from the original probe data. Experiment re-

sults on two real data sets showed that our approach led to statistically significantly better classification accuracies as compared to the based line approach where the DNA copy number data at probe loci were used as features directly.

*Keywords:* Feature reduction, interval tree, cancer classification, DNA copy number, array CGH

## 1 Introduction

Recently, high-throughput array-based assays [1, 2, 3, 4, 5, 6] have been developed to detect DNA copy num-

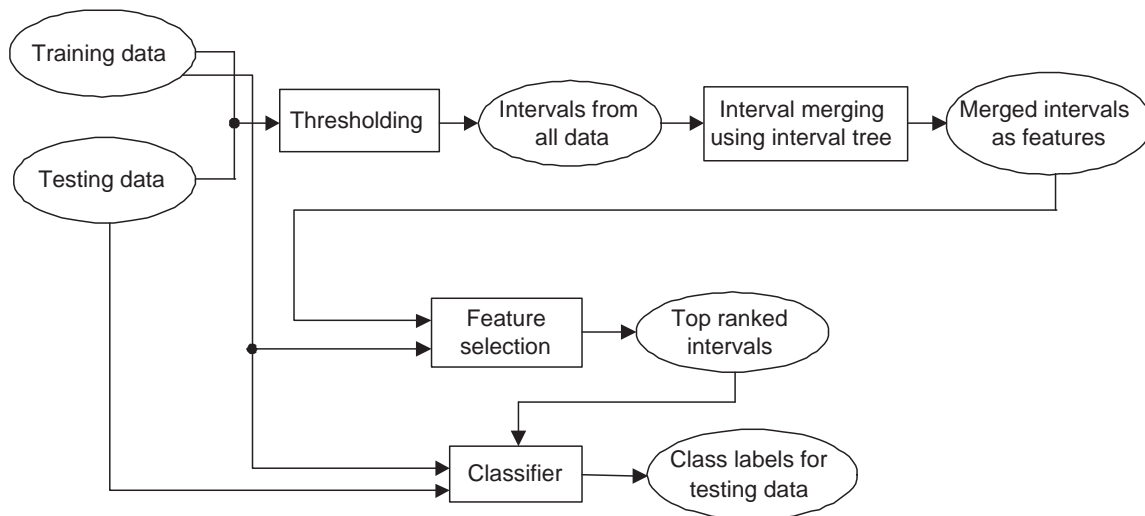


Figure 1: System diagram of the complete cancer classification process using DNA copy number data.

ber aberrations in tumor samples. DNA copy number aberrations are often associated with the development and progression of cancer. For example, amplification of oncogenes or deletion of tumor suppressor genes can lead to cancer development [7]. Both deletion and amplification change the copy numbers of tumor DNA.

There are currently two main high-throughput approaches for interrogating DNA copy numbers:

- Array-based comparative genomic hybridization (array CGH). This approach yields data consisting of  $\log_2$  transformed fluorescence intensity ratios of tumor and reference normal DNA samples. The intensity ratios provide information about DNA copy number aberrations. The resolution of the CGH arrays has improved over the years: The CGH arrays using BAC (Bacterial Artificial Chromosome) clones can provide resolution in the order of 1 Mb [1, 2], and have been widely used. More recently developed CGH arrays using cDNA [4] and long oligonucleotides (60–100 bp) [3] can offer resolution in the order of 35–100 kb.
- Single-Nucleotide Polymorphism (SNP) arrays. High-density Single-Nucleotide Polymorphism (SNP) array is a recently introduced high-throughput technology that genotypes up to 500,000 human SNPs on a single array [5, 6]. These arrays are typically used to provide genotype information.

Cancer classification using DNA copy number data has been recently investigated by several groups [8, 9, 10, 11]. The motivation is that effective machine learning models for cancer classification using DNA copy

number data would be useful not only for clinical cancer diagnosis, but also for discovering novel tumor suppressor genes and oncogenes. Typically, DNA copy numbers at probe loci were used directly as features. This poses a major challenge to machine learning models because of the following characteristics:

- The number of features greatly exceeds the number of instances (tissue samples).
- Most features are not related to the given cancer classification problem.

This “curse-of-dimensionality” problem is particularly severe for cancer classification using DNA copy number data (as compared to gene expression data) because of the extremely large number of probes (currently up to 500,000 to 1 million). Another complicating factor is that the DNA copy number data is often very noisy. This situation is particularly severe for high-density arrays with short probes (cDNA or oligonucleotides). For example, in cDNA array-CGH data, the signal to noise ratio is often approximately 1 [12].

Previously, the common approach taken by researchers was to perform feature selection on DNA copy number data prior to cancer classification. However, because DNA copy number data is noisy and spatially correlated, we hypothesize that applying feature selection to DNA copy number data directly will result in highly correlated features and can lead to sub-optimal classification performance. Indeed, Willenbrock *et al.* [13] applied a segmentation method as a feature reduction step on DNA copy number data and reported better classification accuracies as compared to the case without segmentation.

In this paper, we propose to apply a novel interval tree based feature reduction method to DNA copy number data prior to feature selection and classification. We also perform wavelet denoising [14] to reduce noise in the data.

The remainder of the paper is organized as follows: Section 2 gives an overview of the proposed system; Section 3 presents the details of our interval tree based feature reduction method and the whole classification process. Section 4 demonstrates the experiment results of our system on two real data sets. Section 5 concludes the paper with discussion.

## 2 System Overview

An overview of our proposed cancer classification system is illustrated in Fig. 1.

1. Thresholding is applied to all of the DNA copy number data to derived intervals for each chromosome in each sample.
2. After obtaining the initial set of intervals, an algorithm based on the interval tree is used to merge them and create the final set of intervals as features for classification.
3. DNA copy number values in the original data are used to derive feature values the final set of intervals.
4. Feature selection is applied to the current set of features (intervals with values) from training data.
5. Top ranked features (intervals) are used by a classifier to classify the test data.

## 3 Our Proposed Approach

### 3.1 Extraction of Intervals

We use two thresholds (a positive threshold and a negative threshold) to extract intervals from the DNA copy number data. Fig. 3 illustrates how intervals are extracted using thresholds. Because of the intersections between the data on a chromosome and thresholds, the chromosome will be divided into three types of regions:

- Positive regions: regions containing DNA copy number data above the positive threshold;
- Negative regions: regions containing DNA copy number data below the negative threshold;
- “Normal” regions: regions containing DNA copy number data between the two thresholds.

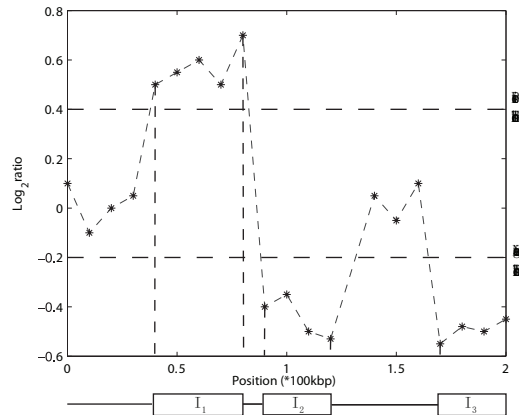


Figure 2: Extraction of intervals from DNA copy number data. The dashed star line shows the DNA copy number data. Interval  $I_1$  will be assigned feature value +1, whereas Intervals  $I_2$  and  $I_3$  will be assigned feature value -1.

The intervals defined by the the positions regions and negative regions are the initial set of intervals we derive from the chromosome. We use this method to find all of the intervals (possibly overlapping) on all chromosomes in all samples.

Clearly, different thresholds can lead to different sets of intervals. In order to determine the optimal pair of positive and negative thresholds, we select positive thresholds from  $[0.1, 2.0]$  and negative thresholds from  $[-0.1, -2.0]$  with a step size of 0.1. Therefore, there are a total number of 400 pairs of thresholds to examine. Then we select the best pair of thresholds heuristically using cross-validation as follows: Each pair of thresholds is applied to the training DNA copy number data to obtain the merged intervals as features. Then a feature selection algorithm is used to select the top 50, 100, 150, ..., 400 features (intervals), which are then fed to a classifier to obtain 10-fold cross-validation classification accuracies. Finally, the pair of thresholds that leads to the best average classification accuracy is chosen as the optimal thresholds.

### 3.2 Merging of Interval

Because there are always more than one sample in data set and the intervals from different samples are likely to be different, these intervals cannot be used directly as features. We need to process the intervals from different samples to construct a set of merged intervals that can be used as features for classification purposes. Fig. 3 shows an example of merging intervals for a chromosome in three samples.

In order to merge intervals efficiently, we developed an algorithm based on the interval tree. An interval tree is a red-black binary tree with each node representing an interval[15]. The key of each node is the low endpoint of an interval. Therefore an in-order walk of the tree lists the nodes in sorted order by low endpoint.

The main procedure of our algorithm:

- 1: **for** each chromosome  $Ch_j$  **do**
- 2:   build an interval tree  $T_j$  from all of the intervals on chromosome  $Ch_j$  in the first sample
- 3:   **for** each interval  $i$  on  $Ch_j$  in all other samples **do**
- 4:      $mergeinterval(root[T_j], i)$
- 5:   **end for**
- 6: **end for**

Pseudocode for the function  $mergeinterval$ :

- ```

mergeinterval( $x, i$ ) :
1: if  $i$  is overlapped with  $int[x]$  then
2:    $int[x] \leftarrow$  common part between  $int[x]$  and  $i$ 
3:   if  $low[i] < low[int[x]]$  then
4:     if  $left[x] \neq nil$  then
5:        $mergeinterval(left[x], [low[i], low[int[x]]])$ 
6:     else
7:        $int[left[x]] \leftarrow [low[i], low[int[x]]]$ 
8:        $update\_tree$ 
9:     end if
10:  else if  $low[i] > low[int[x]]$  then
11:   if  $left[x] \neq nil$  then
12:      $mergeinterval(left[x], [low[int[x]], low[i]])$ 
13:   else
14:      $int[left[x]] \leftarrow [low[int[x]], low[i]]$ 
15:      $update\_tree$ 
16:   end if
17:  end if
18:  if  $high[i] < high[int[x]]$  then
19:   if  $right[x] \neq nil$  then
20:      $mergeinterval(right[x], [high[i], high[int[x]]])$ 
21:   else
22:      $int[right[x]] \leftarrow [high[i], high[int[x]]]$ 
23:      $update\_tree$ 
24:   end if
25:  else if  $high[i] > high[int[x]]$  then
26:   if  $right[x] \neq nil$  then
27:      $mergeinterval(right[x], [high[int[x]], high[i]])$ 
28:   else
29:      $int[right[x]] \leftarrow [high[int[x]], high[i]]$ 
30:      $update\_tree$ 
31:   end if
32:  end if
33: else if  $high[i] \leq low[int[x]]$  then
34:   if  $left[x] \neq nil$  then
35:      $mergeinterval(left[x], i)$ 
36:   else

```

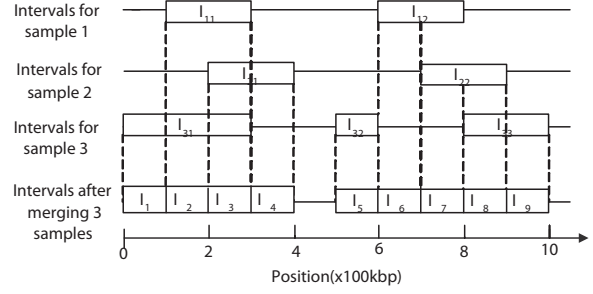


Figure 3: Example of merging intervals for a chromosome in three samples.  $I_1, I_2, \dots, I_9$  are the final set of merged intervals.

- ```

37:    $int[left[x]] \leftarrow i$ 
38:    $update\_tree$ 
39:  end if
40:  else if  $low[i] \geq high[int[x]]$  then
41:   if  $right[x] \neq nil$  then
42:      $mergeinterval(right[x], i)$ 
43:   else
44:      $int[right[x]] \leftarrow i$ 
45:      $update\_tree$ 
46:   end if
47:  end if

```

In the above pseudocode,  $x$  represents a node in the interval tree.  $i$  represents the interval to be merged.  $int[x]$  represents the interval corresponding to node  $x$ .  $low[i]$  represents the low endpoint of interval  $i$ , and  $high[i]$  represents the high endpoint of interval  $i$ .  $left[x]$  represents the left child of node  $x$ , and  $right[x]$  represents the right child of node  $x$ . The function  $update\_tree$  is the RB-INSERT-FIXUP function defined in Chapter 13.3 in [15]. It is used to update the tree to preserve the red-black properties after we changed the structure of the interval tree each time.

Clearly, the running time for merging intervals on one chromosome in all samples is  $O(M \log M)$ , where  $M = \max(N, n)$ ,  $N$  is the total number of intervals to be merged on the chromosome from all samples,  $n$  is the number of probes on the chromosome from one sample. The space complexity is  $O(M)$ . Fig. 4 illustrates the step-by-step process of merging intervals in Fig. 3 using interval trees.

After obtaining the final set of merged intervals for all of the chromosomes, we still need to assign feature values to these intervals for each sample. The feature value of an interval  $[p, q]$  on chromosome  $k$  for a sample is determined by the sample's probe feature value (the  $\log_2$ -ratio at the probe locus in array CGH data) at any probe that falls in the interval  $[p, q]$ . There are three cases:

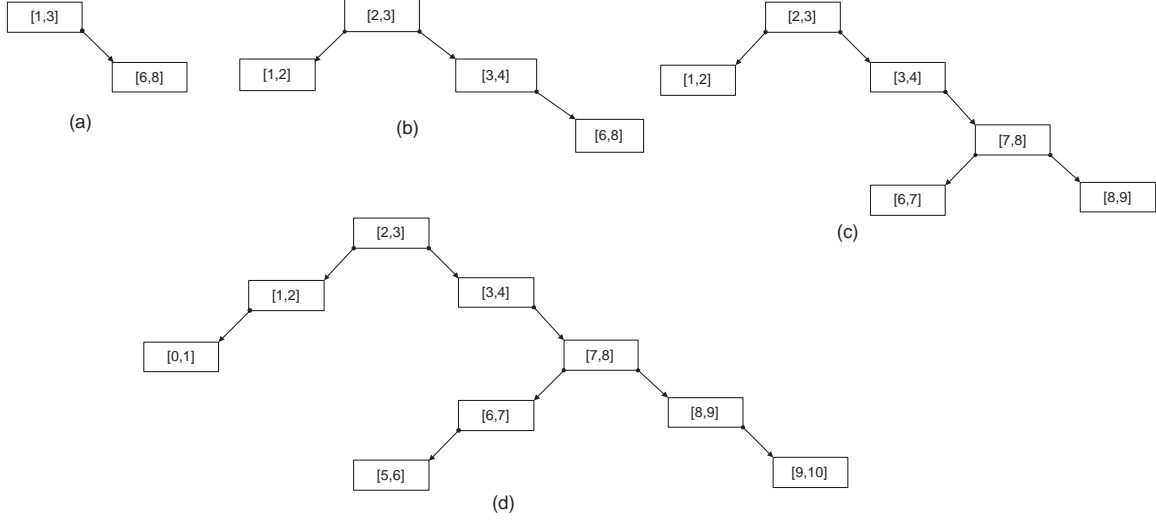


Figure 4: Example of interval trees when merging intervals. (a) The interval tree built from sample 1 in Fig. 3; (b) The interval tree after merging intervals from sample 1 and the interval  $I_{21}$  from sample 2; (c) The interval tree after merging all intervals from sample 1 and sample 2; (d) The interval tree after merging all intervals from all three samples in Fig. 3.

- If the probe feature value is equal to or above the positive threshold, the feature value for the interval will be +1;
- If the probe feature value is equal to or below the negative threshold, the feature value for the interval will be -1;
- If the probe feature value falls between the negative threshold and the positive threshold, the feature value for the interval will be 0;

Clearly, the running time for this step is  $O(kM)$  on one chromosome, where  $k$  is the number of samples, and  $M$  is defined above.

### 3.3 Feature Selection

In this study, we used the Information Gain [16] method to perform feature selection on the interval features. Information Gain is a well-known and empirically proven method for high-dimensional feature selection. It measures the number of bits of information obtained for class prediction by knowing the value of a feature. Let  $\{c_i\}_{i=1}^m$  denote the set of classes. Let  $V$  be the set of possible values for feature  $f$ . The information gain of a feature  $f$  is defined to be:

$$G(f) = - \sum_{i=1}^m P(c_i) \log P(c_i) + \sum_{v \in V} \sum_{i=1}^m P(f=v) P(c_i|f=v) \log P(c_i|f=v)$$

The Information Gain feature selection method is particularly suitable for features with categorical values. In our case, the interval features only have three possible values: +1 (gain), -1 (loss) and 0 (normal). Therefore, Information Gain is appropriate in our case.

For numeric features, such as gene expressional levels and raw DNA copy number data, Information Gain requires that numeric features be discretized. It has been shown that mean-entropy discretized features [17] are effective for classification using gene expression data [18]. Here, we also use the entropy-based discretization method [17] implemented in Weka [19] in our experiments when Information Gain is applied to the original DNA copy number data.

### 3.4 Classification

After feature selection, we train a linear Support Vector Machine (SVM) [20] to classify test data. A linear SVM aims to find the separating hyperplane with the largest margin, defined as the sum of the distances from a hyperplane (implied by a linear classifier) to the closest positive and negative exemplars. The expectation is that the larger the margin, the better the generalization of the classifier. In a non-separable case, a linear SVM seeks a trade-off between maximizing the margin and minimizing the number of errors.

The SVM classifier has been commonly used in cancer classification using microarray data and is considered one of the best classifiers.

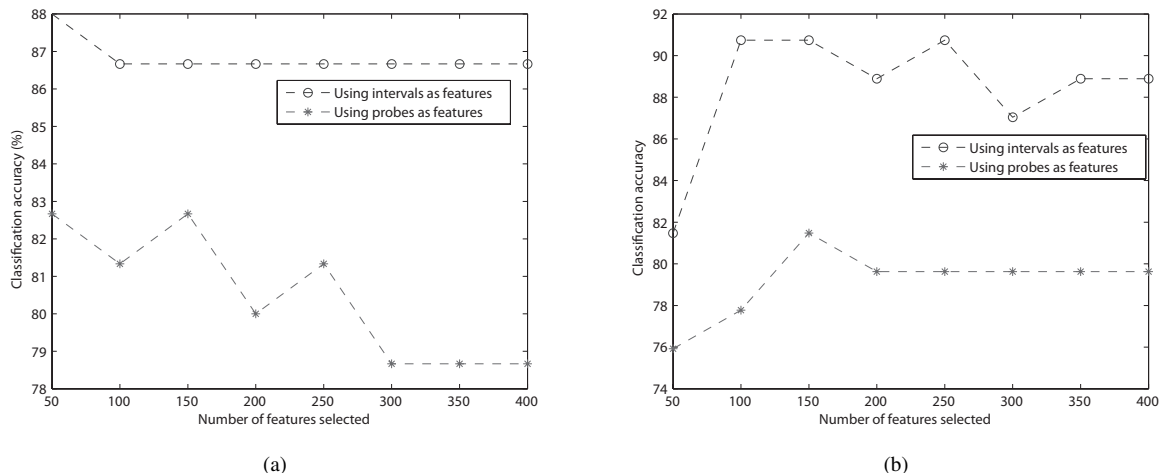


Figure 5: Comparison of 10-fold cross-validation classification accuracies on (a) the TP53 data set and (b) the lung cancer data set. The x-axis represents the number of top ranked features selected. The y-axis represents the classification accuracy.

## 4 Experiment Results

In this section, we present experiment results on two real data sets. Details about the data, preprocessing, experimental parameters, and results are provided in sections below.

### 4.1 Data Source

The first DNA copy number data set we used was published in [21] by Snijders *et al.* It can be downloaded at the following URL: <http://www.cbs.dtu.dk/~hanni/aCGH/>. The original data set contains 75 oral squamous cell carcinoma samples, which can be divided into 14 TP53 mutants and 61 wild-type samples.

The second data set is a lung cancer data set which contains DNA copy number data from 21 small-cell lung cancer (SCLC) cell lines and 33 non-small-cell lung cancer (non-SCLC) cell lines. This data set is from the Hamon Center for Therapeutic Oncology Research at UT Southwestern Medical Center at Dallas.

### 4.2 Experimental Settings

First, we applied our recently developed stationary wavelet denoising method [14] to reduce noise in the data. Then we compared the 10-fold cross-validation classification accuracies for the following two cases:

- using the original DNA copy number data at probe loci as features;

- using the interval features derived with our proposed method as features.

In both cases, the Information Gain feature selection algorithm was used to select the top 50, 100, 150, ..., and 400 features before classification with the linear SVM classifier.

Our system is implemented in Perl and deployed on an Intel Core Duo 1.6GHz computer with 1GB RAM.

### 4.3 Results

For the TP53 data set, the optimal pair of thresholds chosen by our system was (1.1, -0.4). The classification results on the TP53 data are shown in Fig. 5(a). For the lung cancer data set, the optimal pair of thresholds chosen by our system was (0.4, -0.5). The classification results on the lung cancer data are shown in Fig. 5(b).

We can observe from the results that the performance of our interval based approach is better than the baseline approach on both data sets. To evaluate the statistical significance of the difference, we used the paired T-test to calculate the P-values. For the TP53 data set, the P-value was  $5.05 \times 10^{-5}$ . For the lung cancer data set, the P-value was  $6.99 \times 10^{-6}$ .

Besides comparing classification accuracies, we also performed a Gene Ontology (GO) analysis of the top-ranked discriminating intervals from the lung cancer data set. The question we are trying to answer here is: what are the GO biological process terms that are statistically significantly associated with the genes contained by or intersecting with the top-ranked discriminating intervals. We first selected the top 100 intervals for the op-

Table 1: Gene Ontology terms significantly associated with copy number differences in the lung cancer data set

GO ID	GO Annotation	Genes
GO:0007586	Digestion	TFF1, ATP8B1, MEP1B, TFF2
GO:0006508	Proteolysis	LNPEP, MALT1, TMPRSS3, ZMPSTE24, MEP1B, CNDP2, SERPINB13, PSMD6, SERPINB4, KIAA1815, CNDP1
GO:0030162	Regulation of Proteolysis	SERPINB13, SERPINB4

timal thresholds. Then the genes that are contained by or intersecting with these intervals were obtained based on the human genome build hg16. Finally, a list of statistically significant GO terms (P-value threshold 0.05) were obtained using GoMiner [22], which uses the hypergeometric distribution to evaluate the statistical significance of the GO terms. Several GO terms were found to be highly associated with the genes in the top 100 intervals, which are shown in Table 1. The results suggest that the GO biological processes digestion, proteolysis and regulation of Proteolysis are related to differences between small-cell lung cancer cell lines and non-small-cell lung cancer (non-SCLC) cell lines.

## 5 Conclusion and Discussion

Cancer classification using DNA copy number data is an important problem because effective machine learning models for this task would be useful not only for clinical cancer diagnosis, but also for discovering novel tumor suppressor genes and oncogenes. However, the number of probes on current CGH arrays and SNP arrays is extremely large (up to 500,000 to 1 million), thus necessitating the use of appropriate feature reduction and selection methods. In this paper, we proposed an efficient interval tree based feature reduction method for cancer classification using DNA copy number data. Our approach extracts intervals as features from the DNA copy number data. Experiment results on two real data sets showed that our approach led to statistically significantly better classification accuracies as compared to the based line approach where the DNA copy number data at probe loci were used as features directly. The top-ranked informative intervals may contain tumor suppressor genes and oncogenes that are specific to a certain type of tumor. Gene ontology based analysis also suggested the biological relevance of the genes in these intervals.

## 6 Acknowledgement

This work was supported in part by NCI Lung Cancer SPORE grant P50CA70907 and the Longenbaugh and Anderson Charitable Foundations.

## References

- [1] D. Pinkel, R. Seagraves, D. Sudar, S. Clark, I. Poole, D. Kowbel, C. Collins, W. L. Kuo, C. Chen, Y. Zhai, S. H. Dairkee, B. M. Ljung, J. W. Gray, and D. G. Albertson. High resolution analysis of dna copy number variation using comparative genomic hybridization to microarrays. *Nat Genet*, 20(2):207–211, 1998.
- [2] A. M. Snijders, N. Nowak, R. Seagraves, S. Blackwood, N. Brown, J. Conroy, G. Hamilton, A. K. Hindle, B. Huey, K. Kimura, S. Law, K. Myambo, J. Palmer, B. Ylstra, J. P. Yue, J. W. Gray, A. N. Jain, D. Pinkel, and D. G. Albertson. Assembly of microarrays for genome-wide measurement of dna copy number. *Nat Genet*, 29(3):263–264, 2001.
- [3] C. Brennan, Y. Zhang, C. Leo, B. Feng, C. Cauwels, A. J. Aguirre, M. Kim, A. Protopopov, and L. Chin. High-resolution global profiling of genomic alterations with long oligonucleotide microarray. *Cancer Res*, 64(14):4744–4748, 2004.
- [4] J. R. Pollack, C. M. Perou, A. A. Alizadeh, M. B. Eisen, A. Pergamenschikov, C. F. Williams, S. S. Jeffrey, D. Botstein, and P. O. Brown. Genome-wide analysis of dna copy-number changes using cDNA microarrays. *Nat Genet*, 23(1):41–46, 1999.
- [5] H. Matsuzaki, H. Loi, S. Dong, Y. Y. Tsai, J. Fang, J. Law, X. Di, W. M. Liu, G. Yang, G. Liu, J. Huang, G. C. Kennedy, T. B. Ryder, G. A. Marcus, P. S. Walsh, M. D. Shriver, J. M. Puck, K. W. Jones, and R. Mei. Parallel genotyping of over 10,000 snps using a one-primer assay on a high-density oligonucleotide array. *Genome Res*, 14(3):414–25, 2004. 1088-9051 Journal Article.
- [6] Affymetrix Inc. Mapping 500k assay manual. <http://www.affymetrix.com>, 2006.
- [7] D. Pinkel and D. G. Albertson. Array comparative genomic hybridization and its applications in cancer. *Nat Genet*, 37 Suppl:S11–S17, 2005.

- [8] Y. Wang, F. Makedon, and J. Pearlman. Tumor classification based on dna copy number aberrations determined using snp arrays. *Oncol Rep*, 15 Spec no.:1057–9, 2006.
- [9] B. C. Bastian, A. B. Olshen, P. E. LeBoit, and D. Pinkel. Classifying melanocytic tumors based on dna copy number changes. *Am J Pathol*, 163(5):1765–70, 2003. 0002-9440 Journal Article.
- [10] T. Mattfeldt, H. W. Gottfried, H. Wolter, V. Schmidt, H. A. Kestler, and J. Mayer. Classification of prostatic carcinoma with artificial neural networks using comparative genomic hybridization and quantitative stereological data. *Pathol Res Pract*, 199(12):773–84, 2003. 0344-0338 Journal Article Validation Studies.
- [11] R. C. O’Hagan, C. W. Brennan, A. Strahs, X. Zhang, K. Kannan, M. Donovan, C. Cauwels, N. E. Sharpless, W. H. Wong, and L. Chin. Array comparative genome hybridization for tumor classification and gene discovery in mouse models of malignant melanoma. *Cancer Res*, 63(17):5352–6, 2003. 0008-5472 Journal Article.
- [12] S. Bilke, Q. R. Chen, C. C. Whiteford, and J. Khan. Detection of low level genomic alterations by comparative genomic hybridization based on cDNA micro-arrays. *Bioinformatics*, 21(7):1138–1145, 2005.
- [13] H. Willenbrock and J. Fridlyand. A comparison study: applying segmentation to array cgh data for downstream analyses. *Bioinformatics*, 21(22):4084–91, 2005.
- [14] Yuhang Wang and Siling Wang. A novel stationary wavelet denoising algorithm for array-based dna copy number data. *International Journal of Bioinformatics Research and Applications*, In press, 2007.
- [15] Thomas H. Cormen and Thomas H. Cormen. *Introduction to algorithms*. MIT Press, Cambridge, Mass., 2nd edition, 2001.
- [16] Tom M. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.
- [17] U.M. Fayyad and K.B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, pages 1022–1027, 1993.
- [18] Jinyan Li, Huiqing Liu, and Limsoon Wong. Mean-entropy discretized features are effective for classifying high-dimensional biomedical data. In *Proceedings of the 3rd ACM SIGKDD Workshop on Data Mining in Bioinformatics*, 2003.
- [19] I. H. Witten and Eibe Frank. *Data mining : practical machine learning tools and techniques with Java implementations*. Morgan Kaufmann, San Francisco, Calif., 1999.
- [20] V. N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998.
- [21] A. M. Snijders, B. L. Schmidt, J. Fridlyand, N. Dekker, D. Pinkel, R. C. Jordan, and D. G. Albertson. Rare amplicons implicate frequent deregulation of cell fate specification pathways in oral squamous cell carcinoma. *Oncogene*, 24(26):4232–42, 2005.
- [22] B. R. Zeeberg, W. Feng, G. Wang, M. D. Wang, A. T. Fojo, M. Sunshine, S. Narasimhan, D. W. Kane, W. C. Reinhold, S. Lababidi, K. J. Bussey, J. Riss, J. C. Barrett, and J. N. Weinstein. Gominer: a resource for biological interpretation of genomic and proteomic data. *Genome Biol*, 4(4):R28, 2003.