Proceedings of the 6th INFORMS Workshop on Data Mining and Health Informatics (DM-HI 2011)

P. Qian, Y. Zhou, C. Rudin, eds.

Finding the most discriminating sets of biomarkers by maximum weighted clique [1]

Seyed Javad Sajjadi[a], **Amin Ahmadi Adl**[b], **Bo Zeng**[a], **Xiaoning Qian**[b]

[a]Department of Industrial and Management Systems Engineering,

[b]Department of Computer Science and Engineering, University of South Florida, Tampa, FL

Abstract

In order to find the most discriminating and reliable subsets of biomarkers for a given disease, we consider a node and edge weighted network to represent the set of biomarkers and their interactions with respect to their influence on disease development. Then, we formulate a Maximum Weighted Clique Problem (MWCP) to identify the most important and fully connected subsets. To compute its solution, we develop an efficient algorithm that can deal with real size biological networks. A set of computational experiments will be presented along with discussion of insight on our solution strategy in solving biomarker identification problems.

Keywords: Biomarkers, Integer Programming, Maximum Weighted Clique, Synergy

Introduction

With unprecedentedly massive amounts of genome-wide genomic data generated by modern high-throughput biomolecular technologies, including microarray data for studying complex diseases, such as cancer, one pressing challenge is to analyze these high-dimensional data to identify effective biomarkers that are differentially expressed across different phenotypes so that they can be used to improve disease prognosis and diagnosis [2, 4, 14, 16, 19]. As analyzing high-throughput genomic data faces infamous "curse of dimensionality" with limited sample size, a subset of genetic markers have to be selected from thousands of genes to improve predictive accuracy based on the given data set while reducing dimensionality to avoid overfitting [14, 16, 19]. Due to the high computational complexity of finding the optimal gene set as biomarkers, previous filter and wrapper methods typically first filter genes to reduce dimensionality based on their individual predicting performance [6, 19]. However, these greedy feature selection methods may miss the genes whose interactions produce high predictive accuracy [14, 3], especially, when we consider the fact that complex diseases including cancer are systems impairments due to the failure of the coordinated operations among genes and gene products [6, 9].

In this paper, we propose to develop a novel network-based approach to identify highly discriminating biomarkers by capturing both individual and interactive effects among genes based on an information theoretic framework [7, 3]. We construct an *interaction network* whose nodes represent all the genes in the study, whose predictive values, measured by the mutual information between their expression profiles in microarray data and disease outcome, are assigned to each node as the node prize. Significant interactive effects between pairs of genes, defined by their

[1]**Acknowledgments**: The project was supported by Award Number R21DK092845 from the National Institute Of Diabetes And Digestive And Kidney Diseases. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institute Of Diabetes And Digestive And Kidney Diseases or the National Institutes of Health.

synergistic predictive values [3], are represented by the edge weights in the network. To identify highly discriminating biomarkers, we present a general graphical model based on this interaction network with an objective function for feature selection considering their global effects. Mathematically, our network-based biomarker identification can be solved as a Maximum (Node and Edge) Weighted Clique Problem (MWCP) to select subsets of nodes with the largest sum of both node and edge weights. We develop a new fast algorithm to solve the MWCP in this brief paper.

Maximum Weighted Clique Problem (MWCP)

Given a network $G(V, E)$, in which $V = \{v_1, ..., v_n\}$ is the set of nodes and $E \subseteq V \times V$ is the set of edges with e_{ij} denoting the edge connecting v_i to v_j, a clique C is a subset of V such that every pair of its nodes are connected by an edge in E. Let functions $p : V \to \mathbb{R}$ and $w : E \to \mathbb{R}$ associate node prizes and edge weights respectively. The weight of a clique C can be computed as

$$W(C) = \sum_{v_i \in C} p(v_i) + \sum_{e_{ij} \in E_C} w(e_{ij}), \tag{1}$$

where $E_C \subseteq E$ is the set of edges in C induced from E. For a MWCP, we search for C with the largest $W(C)$.

Note that the classical Maximum Clique Problem (MCP) is a special case of the MWCP with $p(\cdot) = 1$ and $w(\cdot) = 0$. Given that MCP is NP-hard, it follows directly that MWCP also is an NP-hard problem. Existing research on MWCP in the literature [11, 5, 20, 8, 17, 13, 1, 15] typically considers either weights on nodes only or weights on edges only [13]. Such a focus enables efficient algorithms which fully explore network structures. Especially, MWCP with weights only on nodes is closely related to maximum weighted independent set problem (MWISP). Hence, algorithms for either problem can be directly benefited from insights and understanding on the other one [13]. To the best of our knowledge, MWCP with weights on both nodes and edges has been a less investigated problem. An important reason is that much less structural information can be utilized by just focusing on network topology, especially for the cases with non-restricted weights. Note that, from the mathematical optimization point of view, MWCP with weights on both nodes and edges can be formulated as a (unconstrained) binary quadratic optimization problem, which shares the same form of MWCP with weights on edges only. Therefore, heuristic algorithms [1, 15] for the latter one could be modified to solve the former problem. In addition, solution procedures [12, 10] to (unconstrained) binary quadratic optimization problems can be applied directly. However, those algorithms either cannot guarantee the quality of the solution or fail to solve large-scale instances, especially those arising in biomedical applications.

Integer Programing (IP) Formulation

MWCP can be formulated as the following binary quadratic programming model – $\mathbf{MWCP-IP}$. Let A denote the adjacency matrix of $G(V, E)$, i.e. $A_{ij} = 1$ if $e_{ij} \in E$ and 0 otherwise. We have

$$\mathbf{MWCP-IP}: \quad max \quad \sum_i p(v_i)X_i + \sum_i \sum_{j>i} w(e_{ij})X_iX_j \tag{2}$$

$$s.t. \quad X_iX_j \leq A_{ij} \quad \forall i < j \tag{3}$$

$$X_i \in \{0, 1\} \quad \forall i, \tag{4}$$

where $X_i = 1$ if v_i is selected to be in the clique and 0 otherwise. The goal is to maximize the objective function (2), equivalently $W(C)$ in (1), under the constraints (3) and (4), which guarantee that the solution forms a clique. One can linearize $\mathbf{MWCP-IP}$ by introducing new variables and constraints, and then solve it using commercial mixed integer programming solvers. However, it will drastically increase the size and the complexity of the model, which will be computationally prohibitive for large-scale networks, such as the one described in our experiments. We here seek to develop more efficient algorithms that are suitable for large-scale problems.

Exact Algorithm

To deal with MWCP with weights on both nodes and edges, we have designed an exact algorithm that extends the backtrack search procedure in [11] that considered edge weights only. We include a new pruning strategy to enhance the efficiency of our algorithm.

The idea of the backtrack search in [11] is as follows. First we sort all nodes according to a specific criterion to obtain an ordering. Note that ordering will not affect the quality of the final solution. Let $V = \{v_1, ..., v_n\}$ denote the set of ordered nodes. Also, let $G_S(S, E_S)$ denote the sub-network induced by $S \subseteq V$. The algorithm then considers a

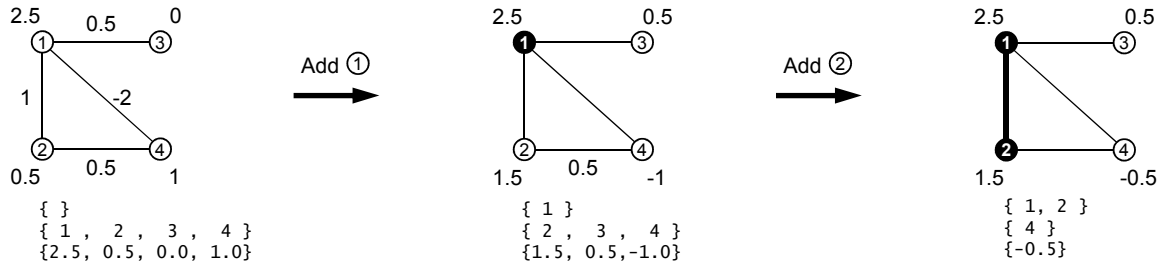

Figure 1: Search for a candidate clique. Three sets are respectively the forming clique, U, and P.

series of subsets $V_i = \{v_i, ..., v_n\}$ starting with $i = n$. It finds the maximum weighted clique in G_{V_i} that includes v_i. Note that the best clique in G_{V_i} that does not include v_i has already been discovered in $G_{V_{i+1}}$. By keeping track of the best clique found so far, from V_n to V_1, the maximum weighted clique of G can be obtained when $V_i = V$.

We have tested different ordering strategies and the one made by increasing *total weight* (node and edge) seems to be a good one. Total weight of node v_i is $p(v_i) + \sum_{v_j \in N(v_i)} w(e_{ij})$ where $N(v_i)$ is the set of nodes adjacent to v_i. In G_{V_i}, to find its maximum weighted clique, we maintain three sets, the *forming clique*, i.e. a subset of nodes already selected to be in the clique, the *working set* denoted by U, i.e. a subset of nodes that could be added to the forming clique, and P, the set of current node weights. Initially, the forming clique set is empty, U is equal to V_i and P is the set of original node weights. When a node is selected to be added to the forming clique, we update U and P. The new node and all other nodes that are not connected to it will be removed from U. The members of P corresponding to the removed members of U will be removed as well. Then each member of P will be incremented by the weight of edge connecting it to the new node (See Figure 1). We keep adding the nodes until there is no node with positive weight in U, indicating the weight of the forming clique can not be increased.

function $find(U, P, CliqueWeight)$
  if $CliqueWeight > BestWeight$
    $BestWeight := CliqueWeight$
    record the forming clique in $BestClique$
    $found = true$
  if $U = \emptyset$
    delete the last member of the forming clique
    return
  while $U \neq \emptyset$ **do**
    if $Pruned(U, P, CliqueWeight, BestWeight)$
      return
    add u_1 **to the forming clique**
    $NewCliqueWeight := CliqueWeight + p_1$
    for $i = 1$ **to** $|U|$ **do**
      $p'_i := p_i + w(u_1, u_i)$
    $U := U \setminus \{u_1\}$
    $P := P \setminus \{p_1\}$
    $find(U \cap N(u_1), P', NewCliqueWeight)$
    if $found$
      return
return

To enhance the computational performance in our new algorithm, for any working set U, we find an upper bound for the weight of the maximum weighted clique. If the weight of the forming clique together with the upper bound is less than the weight of the best clique found so far, it is not necessary to explore U, i.e. U is *pruned*. We have devised two upper bounds. The first upper bound, UB_1, is simply the summation of all positive members of P and positive edge weights in G_U. If UB_1 fails to prune U, we apply the second upper bound which is tighter but takes more time to compute. Let n, m denote the number of positive node weights and positive edge weights in G_U respectively. Also, let s be $\max\left\{n, \{i \in \mathbb{N} : \frac{i(i-1)}{2} \leq m\}\right\}$, i.e. the maximum cardinality of the maximum weighted clique in G_U. We introduce the second upper bound as $UB_2 = \max_{0 \leq i \leq s} \{\sum_{j=1}^{i} p_{(j)} + \sum_{j=1}^{i(i-1)/2} w_{(j)}\}$, where $p_{(j)}$ and $w_{(j)}$ are the j-th largest node and edge weights in G_U. UB_2 is basically the maximum of the weights of all cliques that could be made by the largest node and edge weights.

Algorithm – MWCP
function $main$
  read V, E, V^W, E^W
  $BestWeight := -\infty$
  for $i = |V|$ **to** 1 **do**
    $U = \{v_i, ..., v_{|V|}\}$
    $P = \{v_i^W, ..., v_{|V|}^W\}$
    $found = false$
    $find(U, P, 0)$
return $BestClique, BestWeight$

Our algorithm contains three major functions. The *main* function reads in data, calls other two subroutines—*find* and *pruned*—iteratively to find the maximum weighted clique. Function $find(U, P, CliqueWeight)$ finds the

Table 1: Experimental results for random networks.

$\|V\|$	Density%	Iterations	Pruned%	CPU sec
50	95	55225267	49.9	117.9
50	80	44838	50.0	0.08
100	80	17138065	49.9	65.6
100	70	759825	50.0	2.2
200	70	80567335	50.0	564.5
200	50	486956	50.0	2.2
500	50	72552815	49.9	922.9
500	40	4681656	49.9	57.7
500	30	586762	49.8	5.6
1000	30	10447162	49.9	218.5
1000	20	1005971	49.5	17.0
1000	10	77288	49.2	1.2
2000	15	3552043	49.0	108.4
2000	10	470788	48.6	18.7
2000	5	132665	46.5	3.1

maximum weighted clique in the working set U with node weights P when the current weight of the forming clique is $CliqueWeight$. V, E, V^W, E^W are the ordered node set, the edge set, the corresponding node weight set and the edge weight set of the original network, respectively. Function $pruned(U, P, CliqueWeight, BestWeight)$ returns $true$ if the current working set U with set of node weights P can be pruned by an upper bound, and returns $false$ otherwise. P' is the set of node weights corresponding to $U \cap N(u_1)$. Variables $found, BestClique, BestWeight, V, E, V^W$ and E^W are global variables to carry information or data.

Experimental Results

1. Random Networks

To evaluate the efficiency of the algorithm, we have created a set of random networks. In random networks, nodes are connected randomly with node weights uniformly distributed between -1 and 1 while edge weights are randomly distributed between -0.5 and 0.5. Table 1 gives the running time of our solution algorithm with random networks of different size and density. These values are based on a C++ implementation on a standard PC with a 2.2 GHz CPU and 2Gb RAM. They are considered as rough indices to show that the algorithm is efficient and scalable with the size of the networks with the reasonable density.

2. Colon Cancer Classification

For a preliminary test of our network-based method for disease diagnosis, we have applied the method to identify biomarkers for a public colon cancer data set, which consists of expression profiles of 2000 genes in total using an Affymetrix oligonucleotide array from 22 normal and 40 colon tumor tissues [2]. We first construct the interaction network, in which each gene is represented by a network node v_i. We compute the mutual information between the corresponding gene expression x_i and the sample label (normal vs. tumor) c: $p(v_i) = I(c; x_i) = H(c) - H(c|x_i)$ [7] as the node weight. The edge weights in the network are computed based on the concept of synergy first given in [3]. For an edge e_{ij} between two genes v_i and v_j, $w(e_{ij}) = I(x_i, x_j; c) - [I(x_i; c) + I(x_j; c)]$. Note that the synergy measures the amount of information which is due to the cooperative interaction of two genes with respect to the disease. To compute these weights, we discretize the expression level of a gene into two levels with the cut-off threshold 0 so that a gene can be either up-regulated or down-regulated. Furthermore, to avoid having spurious weights due to the small sample size of the cancer data set, we have only kept the significant individual and synergistic weights with p-values smaller than 0.01 and set the other weights to be 0 based on a permutation test, which randomly permutes the sample labels 100 times to obtain a background distribution of the corresponding weights.

With the computed interaction network, we implement our MWCP solution algorithm to sequentially identify cliques with the largest clique weights. We also consider a varying weighting coefficient λ and reformulate (2) as $\max \{\lambda \sum_i p(v_i) X_i + (1 - \lambda) \sum_i \sum_{j>i} w(e_{ij}) X_i X_j\}$. To analyze the cancer data set, we set λ to vary from 0.1 to 1.0. When $\lambda = 1.0$, only individual effects of genes are used for biomarker identification.

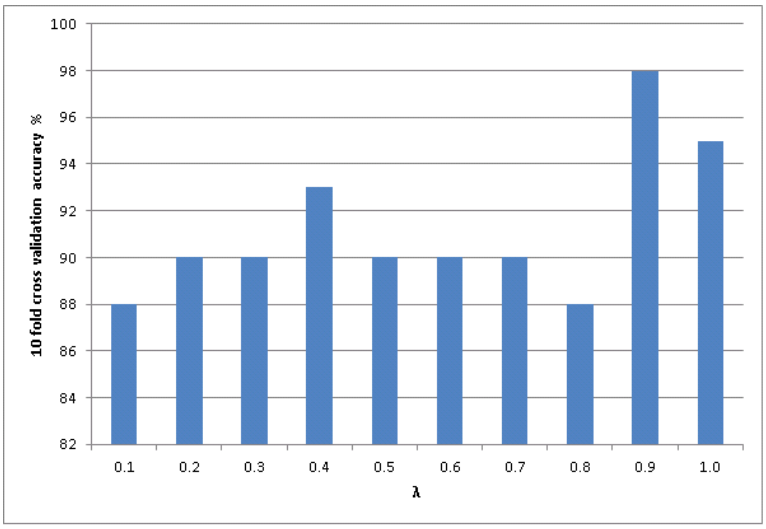

Figure 2: The trend of classification accuracy with different λ for the colon cancer data set.

We compare the classification performance of the genetic biomarkers selected by our network-based method with different λ using support vector machines (SVM). First, when $\lambda = 1.0$, i.e., only individual effects are considered, we rank all 2000 genes in a descending order of node weights so that the genes that have a larger correlation with the sample label are ranked higher. Forward feature selection algorithm, similar as in [4, 16, 18], has been applied to select a subset of genes that give the best classification accuracy. Specifically, during the training of SVM, we sequentially add genes one by one into the subset based on the ranked list if adding it improves the classification accuracy based on 10-fold cross validation. At the end, seven genes are selected which achieve 95.2% accuracy rate estimated by 10-fold cross validation. We consider this as the best accuracy based on the individual effects of genes only.

We select genetic biomarkers for other values of λ in a similar way as when $\lambda = 1.0$. When $\lambda < 1.0$, we implement a fast heuristic to sequentially search for the maximum weighted cliques from iteratively induced networks (with previously identified cliques removed) based on the weighted objective function. For different λ, our algorithm consistently identifies around 600 cliques in less than 30 seconds. Based on the order of the identified cliques from our method, we derive the ranked gene list by sorting the genes within these cliques based on the node weights in the descending order. With that, we implement the same forward selection algorithm to select genes for the best accuracy. For a fair comparison, we limit the maximum number of selected genes to the same number as when $\lambda = 1.0$, which is seven. The best classification accuracy based on a 10-fold cross validation for different λ is shown in Figure 2. We note that with $\lambda = 0.9$, we have achieved the highest classification accuracy 98.4%, indicating that our network-based method can find a better set of biomarkers for colon cancer classification by considering interactive effects among genes. To further check the reliability of the selected genetic markers, especially with the limited sample size in this study, we further test the accuracy based on 2-fold and 5-fold cross validation. The trend of the best accuracy for different λ remains the same and the selected seven genes with $\lambda = 0.9$ consistently achieve the highest accuracy (data not shown), which demonstrates again that the proposed MWCP model and algorithm have the potential to identify biomarkers which give more accurate disease diagnosis than identified biomarkers by individual gene-based ranking.

Conclusion

We have proposed a new network-based method for biomarker identification, which solves the MWCP with both node and edge weights. Our preliminary experiments have shown that the solution algorithm to MWCP is exact and scales well with the size of networks with a reasonable density. We have also applied our algorithm for finding genetic markers of colon cancer based on a public data set. The results have shown that integrating synergistic interactive effects among genes with individual discriminating power may improve the accuracy of disease prognosis and diagnosis.

References

[1] B. Alidaee, F. Glover, G. Kochenberger, and H. Wang. Solving the maximum edge weight clique problem via unconstrained quadratic programming. *European journal of operational research*, 181(2):592–597, 2007.

[2] U. Alon, N. Barkai, D.A. Notterman, et al. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc Natl Acad Sci USA*, 96:6745–50, 1999.

[3] D. Anastassiou. Computational analysis of the synergy among multiple interacting genes. *Molecular Systems Biology*, 3(83), 2007.

[4] T. Bo and I. Jonassen. New feature subset selection procedures for classification of expression profiles. *Genome Biol*, 3:RESEARCH0017, 2002.

[5] S. Busygin. A new trust region technique for the maximum weight clique problem. *Discrete applied mathematics*, 154(15):2080–2096, 2006.

[6] H. Y. Chuang, E. Lee, Y. T. Liu, D. Lee, and T. Ideker. Network-based classification of breast cancer metastasis. *Molecular Systems Biology*, 3:140, 2007.

[7] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. New York: Wiley-Interscience, 1991.

[8] M. Hunting, U. Faigle, and W. Kern. A lagrangian relaxation approach to the edge-weighted clique problem. *European Journal of Operational Research*, 131(1):119–131, 2001.

[9] T. Ideker and R. Sharan. Protein networks in disease. *Genome Research*, 18:644–652, 2008.

[10] G.R. Mauri and L.A.N. Lorena. A column generation approach for the unconstrained binary quadratic programming problem. *European Journal of Operational Research*, 2011.

[11] P.R.J. Östergård. A new algorithm for the maximum-weight clique problem. *Nordic Journal of Computing*, 8(4):424–436, 2001.

[12] P.M. Pardalos, O.A. Prokopyev, O.V. Shylo, and V.P. Shylo. Global equilibrium search applied to the unconstrained binary quadratic optimization problem. *Optimisation Methods and Software*, 23(1):129–140, 2008.

[13] P.M. Pardalos and J. Xue. The maximum clique problem. *Journal of Global Optimization*, 4(3):301–328, 1994.

[14] H.C. 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.

[15] W. Pullan. Approximating the maximum vertex/edge weighted clique using local search. *Journal of Heuristics*, 14(2):117–134, 2008.

[16] Y. Saeys, I. Inza, and P. Larrañaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507–2517, 2007.

[17] M.M. Sorensen. New facets and a branch-and-cut algorithm for the weighted clique problem. *European Journal of Operational Research*, 154(1):57–70, 2004.

[18] J. Su, B.J. Yoon, and E.R. Dougherty. Accurate and reliable cancer classification based on probabilistic inference of pathway activity. *PLoS ONE*, 4:e8161, 12 2009.

[19] Z.Q. Tang, L.Y. Han, H.H. Lin, J. Cui, J. Jia, B.C. Low, B.W. Li, and Y.Z. Chen. Derivation of stable microarray cancer-differentiating signatures using consensus scoring of multiple random sampling and gene-ranking consistency evaluation. *Cancer Res*, 67:9996–10003, 2007.

[20] K. Yamaguchi and S. Masuda. A new exact algorithm for the maximum weight clique problem. In *Proceedings fo the International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), Shimonoseki, Japan*, 2008.