IE 512 Graphs, Networks, and Algorithms

Final

This is a 24-hours take home final exam. Read *all* the questions before starting to write down a solution for one of the problems. There are 4 problems, so try to be very efficient in solving and writing the solutions. Contact `swoh@illinois.edu` if you have any questions.

Problem 1 (matrix completion) Consider a ± 1-vector $x \in \{-1, +1\}^n$ and a symmetric rank-one ± 1-matrix $M = xx^T$. Some of the entries of the matrix M has been erased and we want to recover those missing entries. Notice that both x and $-x$ give the same matrix $M = xx^T = (-x)(-x)^T$, and therefore x is not uniquely determined from M. In particular, there is always a correct solution that sets one of the entries of x to $+1$, for example we can fix $x_1 = +1$. Let $E = \{(i,j)|$ either M_{ij} or M_{ji} is revealed $\}$ denote the set of pairs of indices such that the corresponding entry M_{ij} (or M_{ji}) is revealed. Note that since the matrix is symmetric, we know that $M_{ij} = M_{ji}$ and we only need one of those two entries. We use M^E to denote the matrix with revealed entries, which we filled in the missing entries with zeros. For example,

$$x = \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix} \ , \ M = \begin{bmatrix} 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 \end{bmatrix} \ , \ M^E = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & -1 \\ 0 & 1 & -1 & 1 \end{bmatrix}$$

(a) Consider a greedy algorithm:

- choose an index $i \in [n]$ at random
- set $x_i = +1$
- set $V = \{i\}$
- for all $j \in N(V)$

  let k be an index in V such that M_{kj} is revealed

  set $x_j = x_k \times M_{kj}$

  $V = V \cup \{j\}$
- end for
- output $\hat{M} = xx^T$

where $N(V)$ is defined as a set of indices j such that $(i,j) \in E$ for some $i \in V$.

Prove that this greedy algorithm correctly recovers the entries of M for the connected components of i, that is for the sub-matrix of M corresponding to a connected component of a graph formed by $G = ([n], E)$ that includes the index (or the node) i. In the example above, the subset of nodes $\{1\}$ and $\{2,3,4\}$ each form connected components. If we start the algorithm with $i = 1$, the algorithm does not recover any missing entries. If we start the algorithm with $i \in \{2,3,4\}$, the algorithm correctly recovers the missing entry M_{23}. In particular, if the whole graph is connected, the greedy algorithm correctly recovers all the entries.

(b) Consider a spectral algorithm that compute the leading eigen vector of M^E that corresponds to the largest eigen value. Let u denote this vector. Then, the spectral algorithm computes

$$x_i = \text{sign}(u_i) \ ,$$

and outputs $\hat{M} = xx^T$. For the parts (b) and (c), suppose that $x = \mathbf{1}$ which is all ones vector, such that $M = \mathbf{1}\mathbf{1}^T$ is the all ones matrix. Prove that the non-negative matrix M^E is *regular* if and only if the graph $G = ([n], E)$ is connected.

(c) Use Perron-Frobenius theorem to prove that if the graph $G = (V, E)$ is connected, then this spectral algorithm correctly recovers the original matrix $M = \mathbf{1}\mathbf{1}^T$.

(d) Notice that for general $x \in \{\pm 1\}^n$, which is not all ones, we can write $x = D\mathbf{1}$, where $D = \mathrm{diag}(x)$ is a diagonal matrix with the entries of x in the diagonal. Further, $M = D\mathbf{1}\mathbf{1}^T D$, and $DMD = \mathbf{1}\mathbf{1}^T$. Let u be the leading eigen vector of M^E and let u_0 be the leading eigenvector of $(\mathbf{1}\mathbf{1}^T)^E$, which is all ones matrix that has zeros wherever we have missing entries. Notice that $(\mathbf{1}\mathbf{1}^T)^E = DM^E D$ and $D(\mathbf{1}\mathbf{1}^T)^E D = M^E$.

Use these relationships to show that $u = Du_0$, and that if the graph is connected then the spectral method finds the correct solution for general $x \in \{\pm 1\}^n$.

(e) The MATLAB script `http://web.engr.illinois.edu/~swoh/courses/ie512/hw/matcomp.m` creates an n-dimensional ± 1-vector x and $M = xx^T$, and erases each entry with probability $p \in (0, 1)$. Complete the function in `http://web.engr.illinois.edu/~swoh/courses/ie512/hw/matrixcompletion.m` that recovers M from a subset of its entries using either the greedy algorithm of (a) or the spectral algorithm of (b) (or any other approach you want), and plot number of entries of M reconstructed correctly (averaged over 10 instances) ($\frac{1}{2}$`sum(sum(abs(`$M - \hat{M}$`))`) as a function of p for values of $p \in \{0.002, 0.004, 0.006, 0.008, 0.01, 0.015\}$ and $n = 1000$. Which value of p does the resulting graph (known as Erdös-Renyi graph) start to get connected? [hint: use `eigs(sparse(ME))` to speed up the spectral method.]

Solution 1

(a) First notice that $xx^T = (-x)(-x)^T$ and we can set one of the x_i's either $+1$ or -1 and recover the whole vector x or $-x$ that fits our choice of x_i. Then, by the construction of the algorithm it is clear that the algorithm recovers all entries of x that is connected to our initial choice i such that the connected components of x are correctly recovered. The reason is that the algorithm continues until all connected components of i has been included in V and no more. Also, when it recovers an entry, the entry is correctly recovered.

(b) If M^E is regular, then it means that $(M^E)^k > 0$ for some k. In graph terminology, it means that after k steps we can reach any node from any other node. This implies that the graph is connected.

The same is true that if the graph is then there exists a k such that $(M^E)^k > 0$ for the same reason.

(c) If the graph is connected, then M^E is regular. Then, by PF theorem, the leading eigenvector of M^E is strictly positive. Hence, $\mathrm{sign}(u_i) = 1$, which means that the spectral algorithm correctly recovers all entries of x.

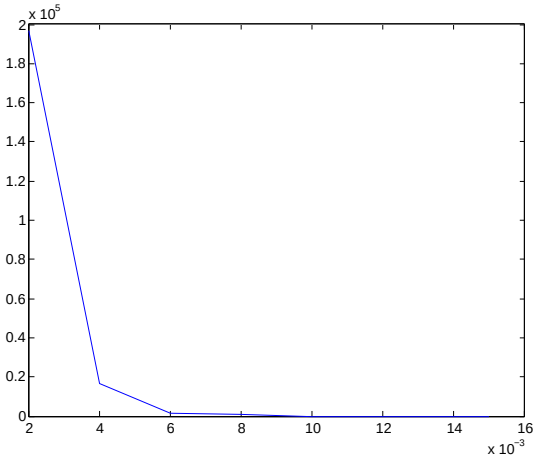
(d) We first show that $u = Du_0$. By definition,

$$\begin{aligned} u_0 &= \arg\max_{\|x\|=1} x^T(\mathbf{1}\mathbf{1}^T)^E x \ , \text{ and} \\ u &= \arg\max_{\|x\|=1} x^T D(\mathbf{1}\mathbf{1}^T)^E D x \ . \end{aligned}$$

This implies that we can change basis to get

$$\begin{aligned} u &= Dy, \text{ where} \\ y &= \arg \max_{\|Dx\|=1} (Dx)^T D(\mathbf{1}\mathbf{1}^T)^E D(Dx) \\ &= \arg \max_{\|Dx\|=1} x^T (\mathbf{1}\mathbf{1}^T)^E x \\ &= \arg \max_{\|x\|=1} x^T (\mathbf{1}\mathbf{1}^T)^E x \\ &= u_0 \ . \end{aligned}$$

Since, $(u_0)_i = D_{ii} u_i$, we have $\text{sign}(u_i) = D_{ii}\text{sign}((u_0)_i)$. Hence, whenever $\text{sign}(u_0) = \mathbf{1}$, we have $\text{sign}(u) = D\text{sign}(u_0) = x$.



(e)

Problem 2 (maximum bipartite matching) Consider an undirected weighted bipartite graph $G = (U, V, E)$. The maximum cardinality bipartite matching problem can be formulated as the following integer programmming:

$$\begin{aligned} \text{maximize} \quad & \sum_{(i,j)\in E} x_{ij} \\ \text{subject to} \quad & \sum_{j:(i,j)\in E} x_{ij} \le 1 \,, \quad \text{for all } i \in U \\ & \sum_{i:(i,j)\in E} x_{ij} \le 1 \,, \quad \text{for all } j \in V \\ & x_{ij} \in \{0, 1\} \,, \quad \text{for all } (i,j) \in E \end{aligned}$$

(a) Write the LP relaxation of the above IP, and explain why we do not need upper bounds on x_{ij}'s.

We will show that this LP always has an integer optimal solution. Therefore, there is no loss in LP relaxation: we can (efficiently) solve the LP relaxation to get the optimal solution of the original IP. We will prove this by showing that, if the LP relaxation of (a) has a non-integral optimal solution, then we can always find an integral solution with the same value of $\sum_{(i,j)\in E} x_{ij}$.

(b) Suppose there is an optimal, feasible and non-integral solution $\tilde{x}$. Then, show that you can always find either

- a cycle of non-integral solution edges; or
- a path of non-integral solution edges whose two end nodes i and j satisfy the constraints with strict inequality such that $\sum_{k:(i,k)\in E} x_{ik} < 1$ and $\sum_{k:(j,k)\in E} x_{jk} < 1$.

(c) For each of the above cases, find x^+ and x^- satisfying the following conditions:

- $\tilde{x} = \frac{1}{2}(x^+ + x^-)$;
- both x^+ and x^- are feasible; and
- either x^+ or x^- has at least one less non-integral edge than $\tilde{x}$.

Then we can conclude that we can continue the process until all edges are integral to obtain an integer optimal solution.

Solution 3

(a) if any one of the non-negative X_{ij} is strictly larger than one, then at least one of the constraint is giong to be violated, assuming that the there is no node with degree zero. Hence, any feasible solution will satisfy $x_{ij} \le 1$ even if it is not explicitly stated.

$$\begin{aligned} \text{maximize} \quad & \sum_{(i,j)\in E} x_{ij} \\ \text{subject to} \quad & \sum_{j:(i,j)\in E} x_{ij} \le 1 \,, \quad \text{for all } i \in U \\ & \sum_{i:(i,j)\in E} x_{ij} \le 1 \,, \quad \text{for all } j \in V \\ & x_{ij} \ge 0 \,, \quad \text{for all } (i,j) \in E \end{aligned}$$

We will show that this LP always has an integer optimal solution. Therefore, there is no loss in LP relaxation: we can (efficiently) solve the LP relaxation to get the optimal solution of the original IP. We will prove this by showing that, if the LP relaxation of (a) has a non-integral optimal solution, then we can always find an integral solution with the same value of $\sum_{(i,j)\in E} x_{ij}$.

(b) Suppose there is an optimal, feasible and non-integral solution $\tilde{x}$. Then, we will show that you can always find either

(1) a cycle of non-integral solution edges; or

(2) a path of non-integral solution edges whose two end nodes i and j satisfy the constraints with strict inequality such that $\sum_{k:(i,k)\in E} x_{ik} < 1$ and $\sum_{k:(j,k)\in E} x_{jk} < 1$.

Start from an arbitrary node i whose one of the edge $\tilde{x}_{ij}$ is fractional. Move to node j and see if it has any other edge that is fractional (note that it cannot have an edge with value $\tilde{x}_{jk} = 1$ since that violates one of the constraions. If there is no such edge, then we go back to the original node i. If $\tilde{x}_{ij}$ was the only fractional edge, then the path $i - j$ satisfy (2) since both node i and j have only one edge with non-zero value and the value is not one. The same is true if the path was longer than two nodes. If the node has another edge with fractional value, we continue along the path until we come to a node that has only one fractional edge. This defines a path satisfying (2)

Let's say all the nodes in our path have at least two fractional valued edges, then we can continue this search until we come back to the original node i, since the graph is finite and the proacess can only visit each edge once. In this case we found a cycle satisfying (1).

(c) Let C be the cycle or the path found in the above problem. Define $\varepsilon = \min_{(i,j)\in C} \min\{|\tilde{x}_{ij}|, |1 - \tilde{x}_{ij}|\}$ and let $x^+ = \tilde{x} + \varepsilon I_C$ and $x^- = \tilde{x} - \varepsilon I_C$, where $I_C \in \{0, +1, -1\}^{|E|}$ has $+1$ for the odd edges in C and -1 for the even edges in C. Notice that both x^+ and x^- are in $[0, 1]^{|E|}$ and $\tilde{x} = (1/2)(x^+ + x^-)$, and all the constraints are satisfied for x^+ and x^-. Further, be the definition of ε, ar least one of the edges is integral for either x^+ or x^-.

Problem 3 (minimum vertex cover for bipartite graphs) Consider an undirected weighted bipartite graph $G = (U, V, E)$ with non-negative weights w_{ij}'s on the edges. This can be formulated minimum weight vertex cover problem as the following integer programmming:

$$\begin{aligned} \text{minimize} \quad & \sum_{i \in U} y_i + \sum_{j \in V} z_j \\ \text{subject to} \quad & y_i + z_j \geq 1 \ , \quad \text{for all } (i,j) \in E \\ & y_i, z_j \in \{0,1\} \ , \quad \text{for all } i \in U \text{ and } j \in V \end{aligned}$$

(a) Write the LP relaxation of the above IP, and explain why we do not need upper bounds on y_i's and z_j's.

We will show that this LP always has an integer optimal solution. Therefore, there is no loss in LP relaxation: we can (efficiently) solve the LP relaxation to get the optimal solution of the original IP. We will prove this by showing that, if the LP relaxation of (a) has a non-integral optimal solution, then we can always find an integral solution with the same value of $\sum_i y_i + \sum_j z_j$.

(b) Suppose there is an optimal, feasible and non-integral solution $\tilde{y}$ and $\tilde{z}$. Notice that all entries satisfy $\tilde{y}_i \leq 1$ and $\tilde{z}_j \leq 1$. Let Q be the set of nodes with fractional (non-integral) values in U and R be the set of nodes with fractional values in V. Assume without loss of generality that $|Q| \leq |R|$ (since we can change the role of Q and R in the other case), and let $\epsilon \triangleq \min\{\tilde{y}_i | i \in Q\}$. Define a new solution y^- from $\tilde{y}$ by subtracting ϵ for all nodes in Q. Define a new solution z^+ from $\tilde{z}$ by adding ϵ for all nodes in R. Prove that the new solution y^- and z^+ is

- feasible;
- satisfy $\sum_{i \in U} y_i^- + \sum_{j \in V} z_j^+ \leq \sum_i \tilde{y}_i + \sum_j \tilde{z}_j$; and
- has at least one less non-integral valued node than $\tilde{y}$ and $\tilde{z}$.

[hint: to prove feasibility, divide the constraints into four cases: constraints corresponding to the edges between Q, R; edges between $Q, V \setminus R$; edges between $U \setminus Q, R$; and edges between $U \setminus Q, V \setminus R$. Then prove feasibility for each case separately.]

Then we can conclude that we can continue the process until all nodes have integral solution to obtain an integer optimal solution.

(c) Prove that the LP relaxation of the maximum matching in Problem 2 part (a) is dual of the LP relaxation of the minimum vertex cover in Problem 3 part (a).

(d) Prove that for bipartite graphs, the cardinality of the minimum cardinality vertex cover is equal to the cardinality of the maximum cardinality matching.

Solution 3.

(a)

$$\begin{aligned} \text{minimize} \quad & \sum_{i \in U} y_i + \sum_{j \in V} z_j \\ \text{subject to} \quad & y_i + z_j \geq 1 \ , \quad \text{for all } (i,j) \in E \\ & y_i, z_j \geq 0 \ , \quad \text{for all } i \in U \text{ and } j \in V \end{aligned}$$

Since we are minimizing non-negative variables y_i's and z_j's and we only need the sum to satisfy $y_i + z_j \geq 1$, this can be always satisfied for either $y_1 = 1$ or $z_j = 1$ and the variables will never be larger than one at global minimum.

(*b*) We want to show

- feasible;
- satisfy $\sum_{i \in U} y_i^- + \sum_{j \in V} z_j^+ \leq \sum_i \tilde{y}_i + \sum_j \tilde{z}_j$; and
- has at least one less non-integral valued node than $\tilde{y}$ and $\tilde{z}$.

To prove feasibility we divide the caonstraints into four cases.

- edges between Q, R: $y_i^- = \tilde{y}_i - \epsilon$ and $z_j^+ = \tilde{z}_j + \epsilon$, hence feasibility still holds.
- edges between $Q, V \setminus R$: $z_j^+ = \tilde{z}_j = 1$ and the constraint is satisfied regardless of y_i^-.
- edges between $U \setminus Q, R$: $y_i^- = \tilde{y}_i = 1$ and the constraint is satisfied regardless of z_j^+.
- edges between $U \setminus Q, V \setminus R$: $y_i^- = \tilde{y}_i$ and $z_j^+ = \tilde{z}_j$, hence feasiliity still holds.

Since the overall change in the objective value is $-\epsilon(|Q| - |R|)$ and we suppose $|Q| \geq |R|$, the objective value decreases as expected.

By definition of ϵ, U has at least one more integral solution, and the number of integral solution for V doaes not decrease.

(*c*) This follows from dual formulation of LP.

(*d*) Let $MinVertexCover_{LP}$ be the otimal value of the LP relaxation of minimum vertex cover problem, $MinVertexCover_{IP}$ be the optimal value of minimum vertex cover problem, $MaxMatching_{LP}$ be the optimal value of the LP relaxation of the maximum bipartite matching problem, and $MaxMatching_{IP}$ be the optimal value of the maximum bipartite matching problem. Then, we proved in previous steps that $MinVertexCover_{LP} = MinVertexCover_{IP}$ and $MaxMatching_{LP} = MaxMatching_{IP}$. It follows from the strong LP duality theorem that $MinVertexCover_{LP} = MaxMatching_{LP}$. Hence, max biaprtite matching is equal to min vertec cover.

Problem 4 (minimum/maximum spanning tree)
Download the MATLAB file `http://web.engr.illinois.edu/~swoh/courses/ie512/hw/treeoflife.m`, which has the names of 11 species stored in `name` and genetic distances between each pair stored in `dist`. Use your solution from previous homework on finding the minimum spanning tree to find the minimum spanning tree in the complete graph of 11 nodes and weights as the genetic distances. Use this to cluster the species into three disjoint sets C_1, C_2, and C_3 such that the minimum inter-cluster distance is maximized, i.e.

$$\text{maximize} \quad \min\{D(C_1, C_2), D(C_1, C_3), D(C_2, C_3)\}$$

where $D(C_i, C_j) = \min_{u \in C_i, v \in C_j} d(u, v)$, and $d(u, v)$ is as defined in the variable `dist`. Write the optimal choice of three sets C_1, C_2 and C_3.

Solution 4. From the lecture notes we know that we need to cluster the set into three sets by first partitioning into two sets by cutting the edge that has the largest weight in the minimum spanning tree. We further partition one of the previously divided sets, by again cutting the largest weighted edge in the MST. By cutting two edges with the largest weight in the MST, we get three disconnected components as follows:
$C_1 = \{Opossum\}$,
$C_2 = \{Gallus\}$,
$C_3 = \{Mouse, Goat, Rat, Bovine, Lemur, Chimpanzee, Rabbit, Gorilla, Human\}$.