Math 218D-1: Homework #13

due Wednesday, November 30, at 11:59pm

1. For each quadratic form $q(x_1, x_2)$ of HW12#15**(a,b)**, first **i)** draw the solutions of $q(x_1, x_2) = 1$, being sure to draw the shortest and longest solutions, and then **ii)** find the maximum and minimum values of $\|x\|^2$ subject to the constraint $q(x) = 1$, and at which points (x_1, x_2) these values are attained.

What happens if you try to extremize $\|x\|^2$ subject to

$$q(x_1, x_2) = x_1^2 - 6x_1x_2 + x_2^2 = 1?$$

(This is the form from part **(c)** of HW12#15.)

2. For the quadratic form

$$q(x_1, x_2, x_3) = 7x_1^2 + 6x_2^2 + 5x_3^2 + 4x_1x_2 + 4x_2x_3$$

of HW12#16, find the maximum and minimum values of $\|x\|^2$ subject to the constraint $q(x) = 1$, along with the points (x_1, x_2, x_3) at which these values are attained.

3. **a)** Consider the quadratic form

$$q(x_1, x_2, x_3) = 7x_1^2 + 6x_2^2 + 5x_3^2 + 4x_1x_2 + 4x_2x_3,$$

of HW12#16. Find the *smallest* value of $q(x)$ subject to the constraints $\|x\| = 1$ *and* $x \perp \frac{1}{3}(1, -2, 2)$. At which vectors x is this minimum attained?

b) Consider the quadratic form

$$q(x_1, x_2, x_3) = x_1^2 + x_2^2 + 7x_3^2 - 16x_1x_2 + 8x_1x_3 + 8x_2x_3.$$

of HW12#17. Find the *largest* value of $q(x)$ subject to the constraints $\|x\| = 1$ *and* $x \perp \frac{1}{\sqrt{5}}(0, 1, 2)$. At which vectors x is this maximum attained?

4. For each matrix A, find the minimum and maximum values of $\|Ax\|^2$ subject to the constraint $\|x\| = 1$. At which vectors are these extrema achieved? Check your work by choosing a unit vector x maximizing $\|Ax\|^2$, computing $b = Ax$, and verifying that $\|b\|^2$ is equal to the maximum.

a) $\begin{pmatrix} 3 & -1 \\ 2 & 3 \\ 1 & -1 \end{pmatrix}$ **b)** $\begin{pmatrix} 1 & -3 \\ 0 & 0 \end{pmatrix}$ **c)** $\begin{pmatrix} 2 & 0 & -1 \\ 1 & 1 & 2 \\ 1 & -1 & 0 \\ 1 & 2 & 0 \end{pmatrix}$

5. Consider the matrix

$$A = \begin{pmatrix} 3 & 2 & -1 & 4 & -3 \\ 1 & 7 & -2 & 3 & -5 \\ 2 & 0 & 8 & -1 & 1 \\ 1 & 2 & 0 & 3 & 9 \end{pmatrix}.$$

a) Find a unit vector u_1 maximizing $\|Ax\|^2$ subject to $\|x\| = 1$.

b) Find the maximum value of $\|Ax\|^2$ subject to $\|x\| = 1$ and $x \perp u_1$.

c) Find the minimum value of $\|Ax\|$ subject to $\|x\| = 1$ without doing any work.

You'll need to use a computer algebra system. With the Sage cell on the course webpage, you'd want something like this:

```
A = Matrix([[ 3., 2.,-1., 4.,-3.],
            [ 1., 7.,-2., 3.,-5.],
            [ 2., 0., 8.,-1., 1.],
            [ 1., 2., 0., 3., 9.]])
pprint((A.transpose()*A).eigenvects())
```

(Entering numbers as "`3.`" instead of "3" forces SymPy to perform a floating-point computation instead of a symbolic one.)

6. Show that the maximum value of $\|Ax\|$ subject to $\|x\| = 1$ is the same as the maximum value of $\|Ax\|/\|x\|$ subject to $x \neq 0$.

Remark: This gives an equivalent definition of the *matrix norm* $\|A\|$.

7. In this problem, we will touch on the role of quadratic optimization in *spectral graph theory*. Spectral graph theory is the study of graphs using linear algebra, and is widely applied to problems in networking and partitioning. (Google's PageRank algorithm can be formulated as a spectral graph theory problem.)

A *graph* is a set of *vertices*, or points, connected by a set of *edges*. For simplicity, we will assume that each edge has distinct endpoints (i.e., there are no loop edges), and that there is at most one edge connecting any two vertices: such a graph is called *simple*. Under these assumptions, an edge is determined by the two vertices it connects, so we can write $e = (1,2)$ for the edge connecting vertices 1 and 2. We also write $i \sim j$ if (i,j) is an edge of G. The *degree* of a vertex is the number of edges connected to it; the degree of vertex i is written $\deg(i)$.

Let G be a graph with n vertices labeled $1,2,\ldots,n$. We consider a vector $x \in \mathbf{R}^n$ as a way to assign a real number to each vertex: the ith coordinate x_i is the number attached to the ith vertex. The *Laplacian* of G is the $n \times n$ matrix L whose (i,j) entry is

$$L_{ij} = \begin{cases} \deg(i) & \text{if } i = j \\ -1 & \text{if there is an edge from vertex } i \text{ to vertex } j \\ 0 & \text{otherwise.} \end{cases}$$

Note that L is symmetric. Let $x \in \mathbf{R}^n$ and let $y = Lx$. Then the ith coordinate of y is

$$(\star) \qquad y_i = x_i \deg(i) - \sum_{j\sim i} x_j = \sum_{j \sim i}(x_i - x_j).$$

In other words, y is the vector that assigns the number $\sum_{j\sim i}(x_i - x_j)$ to vertex i.

The eigenvalues of the graph Laplacian contain important information about the structure of the graph.

a) Show that the vector $\mathbf{1} = (1,1,\ldots,1) \in \mathbf{R}^n$ is in the null space of L.

It follows that 0 is always an eigenvalue of L.

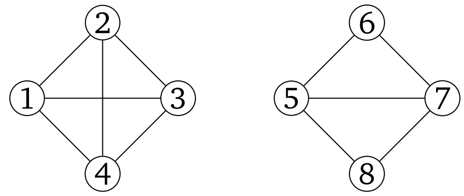
b) Show that $x^T L x = \sum_{j\sim i}(x_i - x_j)^2$. Explain why L is positive-semidefinite.

Since L is positive-semidefinite, all of its eigenvalues are *nonnegative*, so 0 is the smallest eigenvalue of L. The fact that 0 is an eigenvalue gives us no information about the graph, so we wish to "rule it out" by imposing the constraint $x \perp \mathbf{1}$.

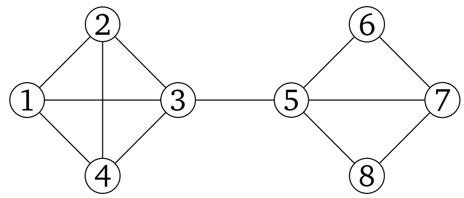
According to **b)**, minimizing $q(x) = x^T L x$ subject to the constraints $\|x\|^2 = 1$ and $x \perp \mathbf{1}$ amounts to finding a way to assign a number to each vertex such that *neighboring vertices have similar values*, but such that the sum of the values is zero ($x \perp \mathbf{1}$) and the sum of their squares is 1 ($\|x\| = 1$).

For each of the following graphs, **i)** compute the Laplacian matrix L and **ii)** minimize $x^T L x$ subject to $x \perp \mathbf{1}$ and $\|x\| = 1$. **iii)** For a (unit) vector x achieving this minimum, draw the number x_i next to vertex i on the graph. **iv)** What does the second-smallest eigenvalue say about the graph? (This is open-ended.)

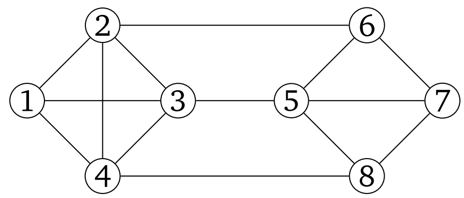
c)



d)



e)



You should feel free to use a computer algebra system to compute the eigenvalues and eigenvectors. For instance, you can use SymPy in the Sage cell on the course webpage. Finding the eigenvalues and eigenvectors of a matrix in SymPy is done as follows: if your matrix is

$$A = \begin{pmatrix} 7 & 2 & 0 \\ 2 & 6 & 2 \\ 0 & 2 & 5 \end{pmatrix}$$

then you would type:

```
A = Matrix([[7.,2.,0.],[2.,6.,2.],[0.,2.,5.]])
pprint(A.eigenvects())
```

(Entering numbers as "`3.`" instead of "3" forces SymPy to perform a floating-point computation instead of a symbolic one.) The output is a list of tuples of the form (eigenvalue, multiplicity, eigenspace basis)—note that the eigenspace basis will not necessarily be orthonormal.