This homework is due on Sunday, July 9, 2017, at 23:59.
Self-grades are due on Monday, July 10, 2017, at 23:59.

Submission Format
Your homework submission should consist of **two** files.

- `hw3.pdf`: A single PDF file that contains all of your answers (any handwritten answers should be scanned) as well as your IPython notebook saved as a PDF.

  If you do not attach a PDF of your IPython notebook, you will not receive credit for problems that involve coding. Make sure that your results and your plots are visible.

- `hw3.ipynb`: A single IPython notebook with all of your code in it.

  In order to receive credit for your IPython notebook, you must submit both a "printout" and the code itself.

Submit each file to its respective assignment on Gradescope.

1. Determinant of A General 2×2 Matrix

Compute the determinant of the following 2×2 matrix.

$$\mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

2. Mechanical Problem

Compute the eigenvalues and eigenvectors of the following matrices.

(a) $\begin{bmatrix} 5 & 0 \\ 0 & 2 \end{bmatrix}$

(b) $\begin{bmatrix} 22 & 6 \\ 6 & 13 \end{bmatrix}$

(c) $\begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}$

(d) $\begin{bmatrix} \frac{\sqrt{3}}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix}$ (What special matrix is this?)

(e) $\begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \end{bmatrix}$

3. Image Compression

In this question, we explore how eigenvalues and eigenvectors can be used for image compression. A grayscale image can be represented as a data grid. Say a symmetric, square image is represented by a symmetric matrix $\mathbf{A}$, such that $\mathbf{A}^T = \mathbf{A}$. We can transform the images to vectors to make it easier to process them as data, but here, we will understand them as 2D data. Let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of $\mathbf{A}$ with corresponding eigenvectors $\vec{v}_1, \ldots, \vec{v}_n$. Also, let these eigenvectors be normalized (unit norm). Then, the matrix can be represented as the expansion

$$\mathbf{A} = \lambda_1 \vec{v}_1 \vec{v}_1^T + \lambda_2 \vec{v}_2 \vec{v}_2^T + \cdots + \lambda_n \vec{v}_n \vec{v}_n^T.$$

Note that the eigenvectors must be normalized for this expansion to be valid because we know that if $\vec{v}_i$ is an eigenvector, then any scalar multiple $\alpha \vec{v}_i$ is also an eigenvector. If we scaled every eigenvector on the right hand side of the equation by α, then the left hand side would change from $\mathbf{A}$ to $\alpha^2 \mathbf{A}$.

The previous expansion shows that the matrix $\mathbf{A}$ can be synthesized by its n eigenvalues and eigenvectors. However, $\mathbf{A}$ can also be *approximated* with the k largest eigenvalues and the corresponding eigenvectors. That is,

$$\mathbf{A} \approx \lambda_1 \vec{v}_1 \vec{v}_1^T + \lambda_2 \vec{v}_2 \vec{v}_2^T + \cdots + \lambda_k \vec{v}_k \vec{v}_k^T.$$

(a) Construct appropriate matrices $\mathbf{V}$, $\mathbf{W}$ (using $\vec{v}_i$'s as rows and columns) and a matrix Λ with the eigenvalues λ_i as components such that

$$\mathbf{A} = \mathbf{V} \Lambda \mathbf{W}.$$

(b) Use the IPython notebook `prob3.ipynb` and the image file `pattern.npy`. Use the `numpy.linalg.eig` command to find the $\mathbf{V}$ and Λ matrices for the image. Note that `numpy.linalg.eig` returns normalized eigenvectors by default. Mathematically, how many eigenvectors are required to fully capture the information within the image?

(c) In the IPython notebook, find an approximation for the image using the 100 largest eigenvalues and eigenvectors.

(d) Repeat part (c) with $k = 50$. By further experimenting with the code, what seems to be the lowest value of k that retains most of the salient features of the given image?

4. Sports Rank

Every year in college sports, specifically football and basketball, debate rages over team rankings. The rankings determine who will get to compete for the ultimate prize, the national championship. However, ranking teams is quite challenging in the setting of college sports for a few reasons: there is uneven paired competition (not every team plays each other), there is a sparsity of matches (every team plays a small subset of all the teams available), and there is no well-ordering (team A beats team B who beats team C who beats A). In this problem, we will come up with an algorithm to rank the teams with real data drawn from the 2014 Associated Press (AP) poll of the top 25 college football teams.

Given N teams we want to determine the rating r_i for the i^{th} team for $i = 1, 2, \ldots, N$, after which the teams can be ranked in order from highest to lowest rating. Given the wins and losses of each team, we can assign each team a score s_i.

$$s_i = \sum_{j}^{N} q_{ij} r_j,$$

where q_{ij} represents the number of times team i has beaten team j divided by the number [1] of games played by team i. If we define the vectors $\vec{s} = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_N \end{bmatrix}$, and $\vec{r} = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_N \end{bmatrix}$ we can express their relationship as a system of equations

$$\vec{s} = \mathbf{Q}\vec{r},$$

where $\mathbf{Q} = \begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1N} \\ q_{21} & q_{22} & \cdots & q_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ q_{N1} & q_{N2} & \cdots & q_{NN} \end{bmatrix}$ is an $N \times N$ matrix.

(a) Consider a specific case where we have three teams, team A, team B, and team C. Team A beats team C twice and team B once. Team B beats team A twice and never beats team C. Team C beats team B three times. What is the matrix $\mathbf{Q}$?

(b) Returning to the general setting, if our scoring metric is good, then it should be the case that teams with better ratings have higher scores. Let's make the assumption that $s_i = \lambda r_i$ with $\lambda > 0$. Show that $\vec{r}$ is an eigenvector of $\mathbf{Q}$.

To find our rating vector, we need to find an eigenvector of $\mathbf{Q}$ with all nonnegative entries (ratings can't be negative) and a positive eigenvalue. If the matrix $\mathbf{Q}$ satisfies certain conditions (beyond the scope of this course), the dominant eigenvalue λ_D, i.e. the largest eigenvalue in absolute value, is positive and real. In addition, the dominant eigenvector, i.e. the eigenvector associated with the dominant eigenvalue, is unique and has all positive entries. We will now develop a method for finding the dominant eigenvector for a matrix if it is unique.

(c) Given $\vec{v}$, an eigenvector of $\mathbf{Q}$ with eigenvalue λ, and a real nonzero number c, express $\mathbf{Q}^n c\vec{v}$ in terms of $\vec{v}$, c,n, and λ

(d) Now given multiple eigenvectors $\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_m$ of $\mathbf{Q}$, their eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_m$, and real nonzero numbers $c_1, c_2, \ldots, c_m$, express $\mathbf{Q}^n(\sum_{i=1}^m c_i\vec{v}_i)$ in terms of $\vec{v}$'s, λ's, and c's.

(e) Assuming that $|\lambda_1| > |\lambda_i|$ for $i = 2, \ldots, m$, argue or prove that

$$\lim_{n\to\infty} \frac{1}{\lambda_1^n} \mathbf{Q}^n \left(\sum_{i=1}^m c_i \vec{v}_i \right) = c_1\vec{v}_1.$$

Hints:

i. For sequences of vectors $\{\vec{a}_n\}$ and $\{\vec{b}_n\}$, $\lim_{n\to\infty}\left(\vec{a}_n + \vec{b}_n\right) = \lim_{n\to\infty}\vec{a}_n + \lim_{n\to\infty}\vec{b}_n$.

ii. For a scalar w with $|w| < 1$, $\lim_{n\to\infty} w^n = 0$.

(f) Now further assuming that λ_1 is positive, show that

$$\lim_{n\to\infty} \frac{\mathbf{Q}^n(\sum_{i=1}^m c_i\vec{v}_i)}{\|\mathbf{Q}^n(\sum_{i=1}^m c_i\vec{v}_i)\|} = \frac{c_1\vec{v}_1}{\|c_1\vec{v}_1\|}$$

Hints:

[1] We normalize by the number of games played to prevent teams from getting a high score by just repeatedly playing against weak opponents.

i. Divide the numerator and denominator by λ_1^n and use the result from the previous part.

ii. For the sequence of vectors $\{\vec{a}_n\}$, $\lim_{n\to\infty} \|\vec{a}_n\| = \|\lim_{n\to\infty} \vec{a}_n\|$.

iii. For a sequence of vectors $\{\vec{a}_n\}$ and a sequence of scalars $\{\alpha_n\}$, if $\lim_{n\to\infty} \alpha_n$ is not equal to zero then the $\lim_{n\to\infty} \frac{\vec{a}_n}{\alpha_n} = \frac{\lim_{n\to\infty} \vec{a}_n}{\lim_{n\to\infty} \alpha_n}$.

Let's assume that any vector $\vec{b}$ in $\mathbb{R}^N$ can be expressed as a linear combination of the eigenvectors of any square matrix $\mathbf{A}$ in $\mathbb{R}^{N\times N}$, i.e. $\mathbf{A}$ has N rows and N columns.

Let's tie it all together. Given the eigenvectors of $\mathbf{Q}$, $\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_N$, we arbitrarily choose the dominant eigenvector to be $\vec{v}_1 = \vec{v}_D$. If we can find a vector $\vec{b} = \sum_{i=1}^{m} c_i \vec{v}_i$, such that c_1 is nonzero, then [2]

$$\lim_{n\to\infty} \frac{\mathbf{Q}^n \vec{b}}{\|\mathbf{Q}^n \vec{b}\|} = \frac{c_1 \vec{v}_D}{\|c_1 \vec{v}_D\|}.$$

This is the idea behind the power iteration method, which is a method for finding the unique dominant eigenvector (up to scale) of a matrix whenever one exists. In the IPython notebook, we will use this method to rank our teams.

Note: For this application we know the rating vector (which will be the dominant eigenvector) has all positive entries, but c_1 might be negative resulting in our method returning a vector with all negative entries. If this happens, we simply multiply our result by -1.

(g) From the method you implemented in the IPython notebook, name the top five teams, the fourteenth team, and the seventeenth team.

5. The Dynamics of Romeo and Juliet's Love Affair

In this problem, we will study a discrete-time model of the dynamics of Romeo and Juliet's love affair—adapted from Steven H. Strogatz's original paper, *Love Affairs and Differential Equations*, Mathematics Magazine, 61(1), p.35, 1988, which describes a continuous-time model.

Let $R[n]$ denote Romeo's feelings about Juliet on day n, and let $J[n]$ quantify Juliet's feelings about Romeo on day n. If $R[n] > 0$, it means that Romeo loves Juliet and inclines toward her, whereas if $R[n] < 0$, it means that Romeo is resentful of her and inclines away from her. A similar interpretation holds for $J[n]$, which represents Juliet's feelings about Romeo.

A larger $|R[n]|$ represents a more intense feeling of love (if $R[n] > 0$) or resentment (if $R[n] < 0$). If $R[n] = 0$, it means that Romeo has neutral feelings toward Juliet on day n. Similar interpretations hold for larger $|J[n]|$ and the case of $J[n] = 0$.

We model the dynamics of Romeo and Juliet's relationship using the following coupled system of linear evolutionary equations:

$$R[n+1] = aR[n] + bJ[n], \quad n = 0, 1, 2, \ldots$$

and

$$J[n+1] = cR[n] + dJ[n], \quad n = 0, 1, 2, \ldots,$$

which we can rewrite as

$$\vec{s}[n+1] = \mathbf{A}\,\vec{s}[n],$$

[2] If we select a vector at random, c_1 will be almost certainly non-zero.

where

$$\vec{s}[n] = \begin{bmatrix} R[n] \\ J[n] \end{bmatrix}$$

denotes the state vector and

$$\mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

the state transition matrix for our dynamic system model.

The parameters a and d capture the linear fashion in which Romeo and Juliet respond to their own feelings, respectively, about the other person. It's reasonable to assume that $a, d > 0$, to avoid scenarios of fluctuating day-to-day mood swings. Within this positive range, if $0 < a < 1$, then the effect of Romeo's own feelings about Juliet tend to fizzle away with time (in the absence of influence from Juliet to the contrary), whereas if $a > 1$, Romeo's feelings about Juliet intensify with time (in the absence of influence from Juliet to the contrary). A similar interpretation holds when $0 < d < 1$ and $d > 1$.

The parameters b and c capture the linear fashion in which the other person's feelings influence $R[n]$ and $J[n]$, respectively. These parameters may or may not be positive. If $b > 0$, it means that the more Juliet shows affection for Romeo, the more he loves her and inclines toward her. If $b < 0$, it means that the more Juliet shows affection for Romeo, the more resentful he feels and the more he inclines away from her. A similar interpretation holds for the parameter c.

All in all, each of Romeo and Juliet has four romantic styles, which makes for a combined total of sixteen possible dynamic scenarios. The fate of their interactions depends on the romantic style each of them exhibits, the initial state, and the values of the entries in the state transition matrix $\mathbf{A}$. In this problem, we'll explore a subset of the possibilities.

(a) Consider the case where $a + b = c + d$ in the state-transition matrix

$$\mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}.$$

i. Show that

$$\vec{v}_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

is an eigenvector of $\mathbf{A}$, and determine its corresponding eigenvalue λ_1. Also determine the other eigenpair $(\lambda_2, \vec{v}_2)$. Your expressions for λ_1, λ_2, and $\vec{v}_2$ must be in terms of one or more of the parameters a, b, c, and d.

ii. Consider the following state-transition matrix:

$$\mathbf{A} = \begin{bmatrix} 0.75 & 0.25 \\ 0.25 & 0.75 \end{bmatrix}$$

i. Determine the eigenpairs for this system.

ii. Determine all the *fixed points* of the system. That is, find the set of points such that if Romeo and Juliet start at, or enter, any of those points, they'll stay in place forever: $\{\vec{s}_* \mid \mathbf{A}\vec{s}_* = \vec{s}_*\}$. Show these points on a diagram where the x and y-axes are $R[n]$ and $J[n]$.

iii. Determine representative points along the state trajectory $\vec{s}[n]$, $n = 0, 1, 2, \ldots$, if Romeo and Juliet start from the initial state

$$\vec{s}[0] = \begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

iv. Suppose the initial state is $\vec{s}[0] = \begin{bmatrix} 3 & 5 \end{bmatrix}^T$. Determine a reasonably simple expression for the state vector $\vec{s}[n]$. Find the limiting state vector

$$\lim_{n\to\infty} \vec{s}[n].$$

(b) Consider the setup in which

$$\mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}.$$

In this scenario, if Juliet shows affection toward Romeo, Romeo's love for her increases, and he inclines toward her. The more intensely Romeo inclines toward her, the more Juliet distances herself. The more Juliet withdraws, the more Romeo is discouraged and retreats into his cave. But the more Romeo inclines away, the more Juliet finds him attractive and the more intensely she conveys her affection toward him. Juliet's increasing warmth increases Romeo's interest in her, which prompts him to incline toward her—again!

Predict the outcome of this scenario before you write down a single equation.

Then determine a complete solution $\vec{s}[n]$ in the simplest of terms, assuming an initial state given by $\vec{s}[0] = \begin{bmatrix} 1 & 0 \end{bmatrix}^T$. As part of this, you must determine the eigenvalues and eigenvectors of the $\mathbf{A}$.

Plot (by hand, or otherwise without the assistance of any scientific computing software package), on a two-dimensional plane (called a *phase plane*)—where the horizontal axis denotes $R[n]$ and the vertical axis denotes $J[n]$—representative points along the trajectory of the state vector $\vec{s}[n]$, starting from the initial state given in this part. Describe, in plain words, what Romeo and Juliet are doing in this scenario. In other words, what does their state trajectory look like? Determine $\|\vec{s}[n]\|^2$ for all $n = 0, 1, 2, \ldots$ to corroborate your description of the state trajectory.

6. Finding Null Spaces

(a) Consider the column vectors of any 3×5 matrix. What is the maximum possible number of linearly independent vectors you can pick from these column vectors?

(b) Suppose we have the following 3×5 matrix after row reduction:

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 0 & -2 & 3 \\ 0 & 0 & 2 & -2 & 2 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

What is the minimum number of vectors spanning the range of $\mathbf{A}$. Find a set of such vectors.

(c) Recall that for every vector $\vec{x}$ in the null space of $\mathbf{A}$, $\mathbf{A}\vec{x} = \vec{0}$. The dimension of a the null space is the minimum number of vectors needed to span it. Find vectors that span the null space of $\mathbf{A}$ (the matrix in the previous part). What is the dimension of the null space of $\mathbf{A}$?

(d) Find vector(s) that span the null space of the following matrix:

$$\mathbf{B} = \begin{bmatrix} 2 & -4 & 4 & 8 \\ 1 & -2 & 3 & 5 \\ 2 & -4 & 5 & 9 \\ 3 & -6 & 7 & 13 \end{bmatrix}$$

7. Traffic Flows

Your goal is to measure the flow rates of vehicles along roads in a town. However, it is prohibitively expensive to place a traffic sensor along every road. You realize, however, that the number of cars flowing into an intersection must equal the number of cars flowing out. You can use this "flow conservation" to determine the traffic along all roads in a network by only measuring flow along only some roads. In this problem, we will explore this concept.

(a) Let's begin with a network with three intersections, A, B and C. Define the flows t_1 as the rate of cars (cars/hour) on the road between B and A, t_2 as the rate on the road between C and B and t_3 as the rate on the road between C and A.

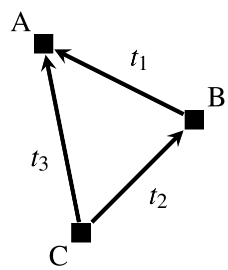


Figure 1: A simple road network.

(Note: The directions of the arrows in the figure are only the way that we define the flow by convention. If there were 100 cars per hour traveling from A to C, then $t_3 = -100$.)

We assume the "flow conservation" constraints: the total number of cars per hour flowing into each intersection is zero. For example at intersection B, we have the constraint $t_2 - t_1 = 0$. The full set of constraints (one per intersection) is:

$$\begin{cases} t_1 + t_3 = 0 \\ t_2 - t_1 = 0 \\ -t_3 - t_2 = 0 \end{cases}$$

As mentioned earlier, we can place sensors on a road to measure the flow through it, but we have a limited budget, and we would like to determine all of the flows with the smallest possible number of sensors.

Suppose for the network above we have one sensor reading, $t_1 = 10$. Can we figure out the flows along the other roads? (That is, the values of t_2 and t_3).

(b) Now suppose we have a larger network, as shown in Figure 2.

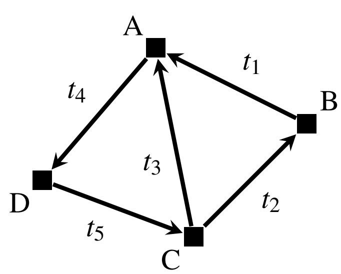


Figure 2: A larger road network.

We would again like to determine the traffic flows on all roads, using measurements from some sensors. A Berkeley student claims that we need two sensors placed on the roads AD and BA. A Stanford student claims that we need two sensors placed on the roads CB and BA. Is it possible to determine all traffic flows with the Berkeley student's suggestion? How about the Stanford student's suggestion?

(c) Suppose we write the traffic flow on all roads as a vector $\vec{t} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \end{bmatrix}$. Show that the set of valid flows (which satisfy the conservation constraints) form a subspace. Then, determine the subspace of traffic flows for the network of Figure 2. Specifically, express this space as the span of two linearly independent vectors.

(Hint: Use the claim of the correct student in the previous part.)

(d) We would like a more general way of determining the possible traffic flows in a network. As a first step, let us try to write all the flow conservation constraints (one per intersection) as a matrix equation. Find a 4×5 matrix $\mathbf{B}$ such that the equation $\mathbf{B}\vec{t} = \vec{0}$

$$\begin{bmatrix} & & \\ & \mathbf{B} & \\ & & \end{bmatrix} \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

represents the flow conservation constraints for the network in Figure 2.

*(Hint: Each row is the constraint of an intersection. You can construct **B** using only* $0, 1,$ *and* -1 *entries.)*

This matrix's transpose is called the **incidence matrix.** What does each row of $\mathbf{B}$ represent? What does each column of $\mathbf{B}$ represent?

(e) Notice that the set of all vectors $\vec{t}$ that satisfy $\mathbf{B}\vec{t} = \vec{0}$ is exactly the null space of the matrix $\mathbf{B}$. That is, we can find all valid traffic flows by computing the null space of $\mathbf{B}$. Use Gaussian elimination to determine the dimension of the null space of $\mathbf{B}$ and compute a basis for the null space. (You may use a calculator to compute the reduced row echelon form.) Does this match your answer to part (c)? Can you interpret the dimension of the null space of $\mathbf{B}$ for the road networks of Figure 1 and Figure 2?

(f) **PRACTICE:** Now let us analyze general road networks. Say there is a road network graph G, with incidence matrix $\mathbf{B}_G$. If $\mathbf{B}_G^T$ has a k-dimensional null space, does this mean measuring the flows along *any* k roads is always sufficient to recover the exact flows? Prove or give a counterexample.

(Hint: Consider the Stanford student.)

(g) **PRACTICE:** Let G be a network of n roads, with incidence matrix $\mathbf{B}_G$ whose transpose has a k-dimensional null space. We would like to characterize exactly when measuring the flows along a set of k roads is sufficient to recover the exact flow along all roads. To do this, it will help to generalize the problem and consider measuring *linear combinations* of flows. If $\vec{t}$ is a traffic flow vector, assume we can measure linear combinations $\vec{m}_i^T\vec{t}$ for some vectors $\vec{m}_i$. Then making k measurements is equivalent to observing the vector $\mathbf{M}\vec{t}$ for some $k \times n$ "measurement matrix" $\mathbf{M}$ (consisting of rows $\vec{m}_i^T$).

For example, for the network of Figure 2, the measurement matrix corresponding to measuring t_1 and t_4 (as the Berkeley student suggests) is:

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Similarly, the measurement matrix corresponding to measuring t_1 and t_2 (as the Stanford student suggests) is:

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

For general networks G and measurements M, give a condition for when the exact traffic flows can be recovered, in terms of the null space of $\mathbf{M}$ and the null space of $\mathbf{B}_G^T$.

(Hint: Recovery will fail iff there are two valid flows with the same measurements. Can you express this in terms of the null spaces of $\mathbf{M}$ and $\mathbf{B}_G^T$?)

(h) **PRACTICE:** Express the condition of the previous part in a way that can be checked computationally. For example, suppose we are given a huge road network G of all roads in Berkeley, and we want to find if our measurements M are sufficient to recover the flows.

(Hint: Consider a matrix $\mathbf{U}$ whose columns form a basis of the null space of $\mathbf{B}_G^T$. Then $\{\mathbf{U}\vec{x} : \vec{x} \in \mathbb{R}^k\}$ is exactly the set of all possible traffic flows. How can we represent measurements on these flows?)

(i) **PRACTICE:** If the incidence matrix's transpose $\mathbf{B}_G^T$ has a k-dimensional null space, does this mean we can always pick a set of k roads such that measuring the flows along these roads is sufficient to recover the exact flows? Prove or give a counterexample.

8. Homework Process and Study Group

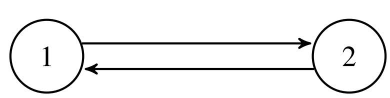
Who else did you work with on this homework? List names and student ID's. (In case of homework party, you can also just describe the group.) How did you work on this homework?
Working in groups of 3-5 will earn you credit for your participation grade.

9. (PRACTICE) Counting The Paths Of A Random Surfer

In class, we discussed the behavior of a random web surfer who jumps from webpage to webpage. We would like to know how many possible paths there are for a random surfer to get from a page to another page. To do this, we represent the webpages as a graph. If page 1 has a link to page 2, we have a directed edge from page 1 to page 2. This graph can further be represented by what is known as an "adjacency matrix", $\mathbf{A}$, with elements a_{ij}. We define $a_{ji} = 1$ if there is link from page i to page j. Matrix operations on the adjacency matrix make it very easy to compute the number of paths to get from a particular webpage i to webpage j.

This path counting aspect actually is an implicit part of the how the "importance scores" for each webpage are described. Recall that the "importance score" of a website is the steady-state frequency of the fraction of people on that website.

Consider the following graphs.

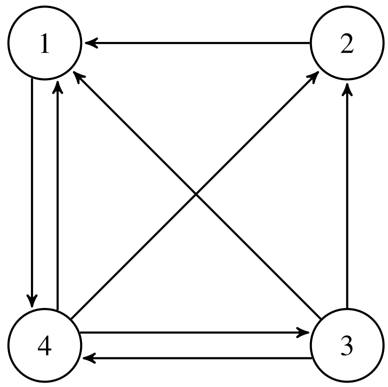


Graph A

(a) Write out the adjacency matrix for graph A.

(b) For graph A: How many one-hop paths are there from webpage 1 to webpage 2? How many two-hop paths are there from webpage 1 to webpage 2? How about three-hop paths?

(c) For graph A: What are the importance scores of the two webpages?

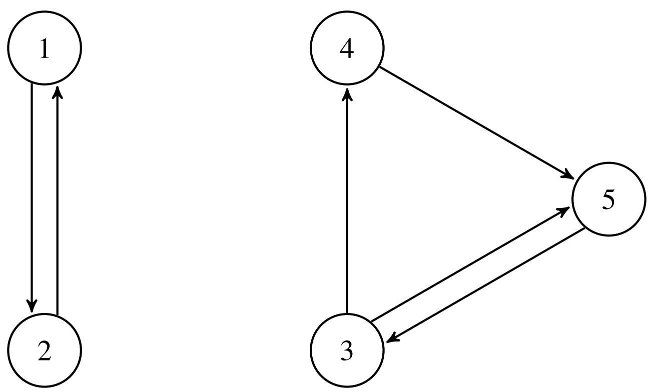


Graph B

(d) Write out the adjacency matrix for graph B.

(e) For graph B: How many two-hop paths are there from webpage 1 to webpage 3? How many three-hop paths are there from webpage 1 to webpage 2?

(f) For graph B: What are the importance scores of the webpages? You may use your IPython notebook for this.



Graph C

(g) Write out the adjacency matrix for graph C.

(h) For graph C: How many paths are there from webpage 1 to webpage 3?

(i) For graph C: What are the importance scores of the webpages? How is graph (c) different from graph (b), and how does this relate to the importance scores and eigenvalues and eigenvectors you found?

10. (PRACTICE) Can You Hear the Shape of a Drum?

This problem is inspired by a popular problem posed by Mark Kac in his article "Can you hear the shape of a drum?"[3] Kac's question was about different shapes of drums. Here's what he wanted to know: if the shape of a drum defines the sound that's made when we strike it, can we listen to the drum and automatically infer its shape? Deep down, this is really a question about eigenvalues and eigenvectors of a matrix. The vibrational dynamics of a particularly shaped drum membrane can be captured by a system of linear equations represented by a matrix. The eigenvalues and eigenvectors of this matrix reveal interesting properties about the drum that will help us answer the question: can we hear its shape?

[3] Marc Kac, Can one hear the shape of a drum?, Amer. Math. Monthly 73 (1966), 1-23.

We'll use a model of vibration given by the equation,

$$\nabla^2 u(x,y) + \lambda u(x,y) = 0$$

Where u is the amount of displacement of the drum membrane at a particular location (x,y), and λ is a parameter (which will turn out to be an eigenvalue, as you will see). The "∇^2" is an operator called the "Laplacian," and just stands for taking the 2nd x-partial-derivative and adding it to the 2nd y-partial-derivative:

$$\nabla^2 u(x,y) = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \approx \frac{u(x+h,y) + u(x,y+h) - 4u(x,y) + u(x,y-h) + u(x-h,y)}{h^2}$$

I've given you an approximation for the Laplacian above, which is the key to formulating this problem as a matrix equation. This equation is known as the "5-point finite difference equation" because it uses five points (the point at x,y and each of its nearest neighbors) to approximate the value of the Laplacian. The last thing you'll need before we start is the 1D version of this equation, to start:

$$\frac{d^2 u}{dx^2} \approx \frac{u(x+h) - 2u(x) + u(x-h)}{h^2}$$

(Note: for 1D the Laplacian simplifies to a regular 2nd derivative; the factor on the $u(x)$ is 2 instead of 4; and there are only 3 points!)

(a) First we'll do a simple model: a violin string. Write the finite difference matrix problem for a 1×5 1D violin string as shown in Figure 3. Use the model shown above to derive your matrix. You can make the assumption that the ends of the string (points 0 and 4) are anchored, so they always have a displacement of zero. Assume that the length of the string is 1 meter (even though that's kind of long for a violin...) (Note: there are only 3 unknowns here!)

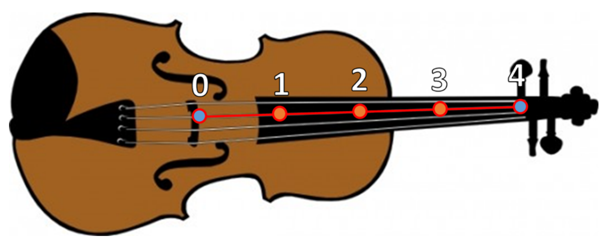


Figure 3: A 5-point model of a violin string.

(b) For our vibrating string, find the 3 eigenvalues (λ) of the matrix $\mathbf{A}$.

(c) For the vibrating string, find the 3 eigenvectors $\vec{u}$ that correspond to the λ's from part (b). What do these vectors look like?

(d) What do you think the eigenvalues mean for our vibrating string? (Hint: what does a larger eigenvalue seem to indicate about the corresponding eigenvector?)

Using what you know from part (a) of this problem, we will write down the 5-point finite difference equation for a 5×5 square drum in the form of a matrix problem so that it has the same form as

$$-\lambda\vec{u} = \mathbf{A}\vec{u}$$

In this formulation, as in the 1D formulation, each row of $\mathbf{A}$ will correspond to the equation of motion for one point on the model. In our 5×5 grid, we will be modeling the motion of the inner 3×3 grid, since we will assume the membrane is fixed on the outer border. Since there are 9 points that we are modeling, this corresponds to 9 equations and 9 unknowns, so $\mathbf{A}$ should be 9×9.

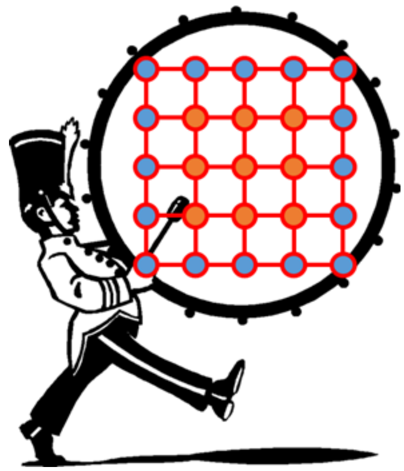


Figure 4: A 25-point model of a drum membrane.

(e) Based on our intuition from the 1D problem, what do the eigenvalues and eigenvectors correspond to in the 2D problem?

(f) Write down the 9×9 matrix, $\mathbf{A}$, for the drum in Figure 4. It should have some symmetry, but be careful with the diagonals.

(g) In the IPython Notebook, implement a function to solve the finite difference problem for a square drum of any side-length (though keep the side-length short at first, so that you don't run into memory problems!). What are the eigenvalues of the 5×5 drum?

(h) Using some of the built-in functionality, you can construct a drum with any polygonal shape. There are two shapes already implemented, with the shapes shown below. The code already included will construct the $\mathbf{A}$ matrix given a polygon and a grid. Find the first 10 vibrational modes of each drum, and the associated eigenvalues (this is analogous to finding the first 10 eigenvectors of each $\mathbf{A}$ matrix, and the associated eigenvalues). Plot the 0th, 4th, and 8th modes using a contour plot.

(i) These two drums are different shapes. Do they sound the same? Why or why not? Can you hear the shape of a drum?