This homework is due December 6, 2016, at 13:00.

1. **Homework process and study group**

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

2. **Mechanical Problem**

   Compute the eigenvalues and eigenvectors of the following matrices.

   (a) $\begin{bmatrix} 3 & 0 \\ 0 & 5 \end{bmatrix}$

   (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?)

3. **Mechanical Diagonalization**

   All calculations in this problem are intended to be done by hand, but you can use a computer to check your work.

   Diagonalize the matrix

$$A = \begin{bmatrix} 1/2 & 1/2 & -1/2 \\ -1/2 & 3/2 & 1/2 \\ -1 & 1 & 1 \end{bmatrix} \tag{1}$$

   given that A has eigenvalues 1, 2, and 0.

4. **Spectral Mapping and the Fibonacci Sequence**

   One of the most useful things about diagonalization is it allows us to easily compute polynomial functions of matrices. This in turn lets us do far more, including solving many linear recurrence relations. This problem shows you how this can be done for the Fibonacci numbers, but you should notice that the same exact technique can apply far more generally.

   Suppose we have a matrix A that can be diagonalized as

$$A = PDP^{-1} = \begin{bmatrix} | & & | \\ \vec{p}_1 & \cdots & \vec{p}_n \\ | & & | \end{bmatrix} \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & & \vdots \\ 0 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} | & & | \\ \vec{p}_1 & \cdots & \vec{p}_n \\ | & & | \end{bmatrix}^{-1} \tag{2}$$

where D is a diagonal matrix with the eigenvalues $\lambda_1, \dots, \lambda_n$ on the diagonal and P is a matrix whose columns $\vec{p}_1, \dots, \vec{p}_n$ are the eigenvectors of A.

(a) **Write out A^N in terms of P,P^{-1}, and D and simplify it as much as you can.** You should be able to show that you can write A^N as

$$A^N = PD^N P^{-1} = \begin{bmatrix} | & & | \\ \vec{p}_1 & \cdots & \vec{p}_n \\ | & & | \end{bmatrix} \begin{bmatrix} \lambda_1^N & \cdots & 0 \\ \vdots & & \vdots \\ 0 & \dots & \lambda_n^N \end{bmatrix} \begin{bmatrix} | & & | \\ \vec{p}_1 & \cdots & \vec{p}_n \\ | & & | \end{bmatrix}^{-1} \tag{3}$$

What does this say about any polynomial function of A?

(b) This idea that for diagonalizable matrices you can raise a matrix to any power by simply raising it's eigenvalues to that power is part of the **spectral mapping theorem**. We will now illustrate the power of this theorem to compute analytical expressions for numbers in the famous Fibonacci sequence. **Take a look at the Wikipedia article and find a cool fact about Fibonacci numbers to report!**

(c) The Fibonacci sequence can be constructed according to the following relation. The Nth number in the Fibonacci sequence, F_N is computed by adding the previous two numbers in the sequence together

$$F_N = F_{N-1} + F_{N-2} \tag{4}$$

We select the first two numbers in the sequence to be $F_1 = 0$ and $F_2 = 1$ and then we can compute the following numbers as

$$0,\ 1,\ 1,\ 2,\ 3,\ 5,\ 8,\ 13,\ 21,\ 34,\ 55,\dots \tag{5}$$

Notice that we can write the operation of computing the next Fibonacci numbers from the previous two using matrix multiplication

$$\begin{bmatrix} F_N \\ F_{N-1} \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}}_{A} \begin{bmatrix} F_{N-1} \\ F_{N-2} \end{bmatrix} \tag{6}$$

Do you see why? Notice also that we could use powers of A to compute Fibonacci numbers starting from the original two, 0 and 1.

$$\begin{bmatrix} F_N \\ F_{N-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^{N-2} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{7}$$

Diagonalize A and use Equation (3) to show that

$$F_N = \frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^{N-1} - \frac{1}{\sqrt{5}} \left(\frac{1-\sqrt{5}}{2} \right)^{N-1} \tag{8}$$

is an analytical expression for the Nth Fibonacci number.

Note that A has eigenvalues and eigenvectors

$$\left\{ \lambda_1 = \frac{1+\sqrt{5}}{2},\ \lambda_2 = \frac{1-\sqrt{5}}{2} \right\} \qquad \left\{ \vec{p}_1 = \begin{bmatrix} \frac{1+\sqrt{5}}{2} \\ 1 \end{bmatrix},\ \vec{p}_2 = \begin{bmatrix} \frac{1-\sqrt{5}}{2} \\ 1 \end{bmatrix} \right\} \tag{9}$$

Feel free to use the 2×2 inverse formula

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad-bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} \tag{10}$$

(d) **(Bonus In-Scope)** Generalize what you found to a procedure that will give you, in principle, expressions for many linear recurrence relations that are recursively defined as $S_{n+k} = \sum_{i=0}^{k-1} \alpha_i S_{n+i}$ for some coefficients $\vec{\alpha}$ and initial conditions $[S_{k-1}, S_{k-2}, \ldots, S_0]^T = \vec{s}_0$.

Do this by setting up the appropriate matrix A and then invoking a computation of its eigenvalues and eigenvectors. And then using the results. (Feel free to assume diagonalizability of the resulting matrix, although there are some important cases when that does not hold.)

5. Image Compression

In this question, we explore how eigenvalues and eigenvectors can be used for image compression. We have seen that a grayscale image can be represented as a data grid. Say a symmetric, square image is represented by a symmetric matrix A, such that $A^T = A$. We've been transforming the images to vectors in the past to make it easier to process them as data, but here we will understand them as 2D data. Let $\lambda_1 \cdots \lambda_n$ be the eigenvalues of A with corresponding eigenvectors $\vec{v}_1 \cdots \vec{v}_n$. Then, the matrix can be represented as

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

However, the matrix A can also be *approximated* with the k largest eigenvalues and corresponding eigenvectors. That is,

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

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

$$A = U \Lambda V$$

.

(b) Use the IPython notebook `prob12.ipynb` and the image file `pattern.npy`. Use the `numpy.linalg` command `eig` to find the U and Λ matrices for the image. 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?

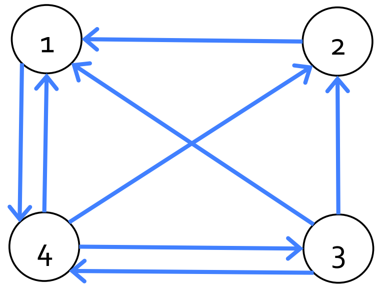
6. Counting the paths of a Random Surfer

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", A, with elements a_{ij}. $a_{ji} = 1$ 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.

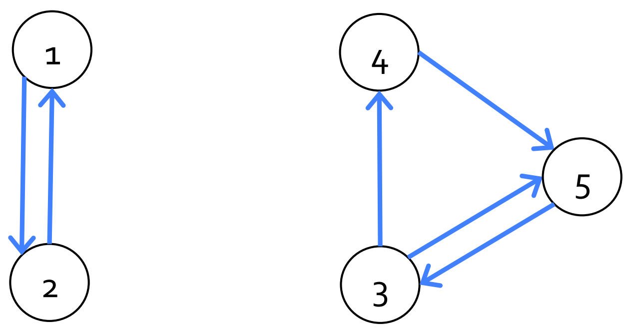
This path counting aspect actually is an implicit part of the how the "importance scores" for each webpage are described. Recall that the "importance score" of a website is the steady-state frequency of the fraction of people on that website.

Consider the following graphs.

Graph A



Graph B



Graph C

(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 3-hop?

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

(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?

(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 the importance scores and eigenvalues and eigenvectors you found?

7. 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), 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) 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.

$$s_i = \sum_j^N q_{ij} r_j, \tag{11}$$

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

$$\vec{s} = Q\vec{r}, \tag{12}$$

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

To find our rating vector we need to find an eigenvector of Q with all nonnegative entries (ratings can't be negative) and a positive eigenvalue. If the matrix 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 when it is unique.

(c) Given $\vec{v}$ is an eigenvector of Q with eigenvalue λ and a real nonzero number c, express $Q^n c\vec{v}$ 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 Q, their eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_m$, and real nonzero numbers $c_1, c_2, \ldots, c_m$, express $Q^n(\sum_{i=1}^m c_i \vec{v}_i)$ 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

$$\lim_{n\to\infty} \frac{1}{\lambda_1^n} Q^n (\sum_{i=1}^m c_i \vec{v}_i) = c_1 \vec{v}_1 \tag{13}$$

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

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

$$\lim_{n\to\infty} \frac{Q^n(\sum_{i=1}^m c_i\vec{v}_i)}{||Q^n(\sum_{i=1}^m c_i\vec{v}_i)||} = \frac{c_1\vec{v}_1}{||c_1\vec{v}_1||} \tag{14}$$

Hints:

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 A in $\mathbb{R}^{N\times N}$, i.e. A has N rows and N columns.

Let's tie it all together. Given the eigenvectors of 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]

$$\lim_{n\to\infty} \frac{Q^n\vec{b}}{||Q^n\vec{b}||} = \frac{c_1\vec{v}_D}{||c_1\vec{v}_D||}. \tag{15}$$

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.

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

In this problem we 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 described 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:

$$R[n+1] = aR[n] + bJ[n], \quad n = 0, 1, 2, \ldots \tag{16}$$

[2] If we select a vector at random c_1 will be nonzero almost certainly.

and

$$J[n+1] = c\,R[n] + d\,J[n], \quad n = 0, 1, 2, \ldots, \tag{17}$$

which we can rewrite as

$$\vec{s}[n+1] = \mathbf{A}\,\vec{s}[n], \tag{18}$$

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 if $0 < d < 1$ and $d > 1$.

The parameters b and c capture the linear fashion in which the other person's feelings influences $R[n]$ 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. And the fate of their interactions depends on the romantic style each of them exhibits, the initial state, and the values of 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. Determine, also, 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: $\{\vec{s}_* | \mathbf{A}\vec{s}_* = \vec{s}_*\}$. 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 $\vec{s}[0] = [3\ 5]^\mathsf{T}$. 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 $\vec{s}[0] = [1\ 0]^\mathsf{T}$. 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.