DSC 140B - Homework 02

Due: Wednesday, January 21

Instructions:

- Write your solutions to the following problems **by hand**, either on on another piece of paper that you scan or using a tablet. Typed solutions will not be accepted for credit!
- Unless otherwise noted by the problem's instructions, show your work or provide some justification for your answer.
- Homework problems are graded pass/fail on completeness and effort, not correctness.
- Homeworks are due via Gradescope at 11:59 PM.

Problem 1. (1 credit)

Suppose (just like in the last homework) that in a group of 1000 people, 600 currently live in California and 400 currently live in Texas. In any given year, 5% of the people living in California move to Texas, and 3% of the people living in Texas move to California. You may assume that the people do not move to any other states.

We can represent the current number of people living in California and Texas with a *population vector*:

$$\vec{p} = (\#\text{ in California}, \#\text{ in Texas})^T.$$

The initial situation described above is represented by the population vector $(600, 400)^T$.

a) Let $\vec{f}(\vec{p})$ be the linear transformation which takes in a current population vector, $\vec{p} = (c, t)^T$, and returns the population vector after one year has passed. In part (b) of the corresponding problem on the last homework, you should have found the following formula for $\vec{f}$ with respect to the standard basis:

$$\vec{f}(\vec{p}) = (.95c + .03t, \quad .05c + .97t)^T$$

Write the *matrix* A representing $\vec{f}$ with respect to the standard basis.

Solution: For the columns of A we use the vectors $\vec{f}(\hat{e}^{(1)})$ and $\vec{f}(\hat{e}^{(2)})$. So we start by computing them:

$$\begin{aligned}\vec{f}(\hat{e}^{(1)}) &= \vec{f}((1,0)^T)\\ &= (.95, .05)^T\end{aligned}$$

$$\begin{aligned}\vec{f}(\hat{e}^{(2)}) &= \vec{f}((0,1)^T)\\ &= (.03, .97)^T\end{aligned}$$

And so the matrix representing this transformation (in the standard basis) is:

$$A = \begin{pmatrix} .95 & .03 \\ .05 & .97 \end{pmatrix}$$

b) Using a matrix multiplication, find the population vector after one year has passed, given that the initial population vector is $(600, 400)^T$. Your result should not contain decimals.

Solution:

$$\begin{aligned} A\vec{x} &= \begin{pmatrix} .95 & .03 \\ .05 & .97 \end{pmatrix} (600, 400) \\ &= \begin{pmatrix} .95 \times 600 + .03 \times 400 \\ .05 \times 600 + .97 \times 400 \end{pmatrix} \\ &= \begin{pmatrix} 582 \\ 418 \end{pmatrix} \end{aligned}$$

c) In part (f) of the last homework, we saw that two eigenvectors of A are

$$\vec{u}^{(1)} = \begin{pmatrix} 375 \\ 625 \end{pmatrix} \qquad \vec{u}^{(2)} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

Verify that these are eigenvectors of the matrix A by performing the matrix multiplication.

Solution:

$$\begin{aligned} A\vec{u}^{(1)} &= \begin{pmatrix} .95 & .03 \\ .05 & .97 \end{pmatrix} \begin{pmatrix} 375 \\ 625 \end{pmatrix} \\ &= \begin{pmatrix} .95 \times 375 + .03 \times 625 \\ .05 \times 375 + .97 \times 625 \end{pmatrix} \\ &= \begin{pmatrix} 356.25 + 18.75 \\ 18.75 + 606.25 \end{pmatrix} \\ &= \begin{pmatrix} 356.25 + 18.75 \\ 18.75 + 606.25 \end{pmatrix} \\ &= \begin{pmatrix} 375 \\ 625 \end{pmatrix} \end{aligned}$$

So $\vec{u}^{(1)}$ is an eigenvector with eigenvalue 1.

$$\begin{aligned} A\vec{u}^{(2)} &= \begin{pmatrix} .95 & .03 \\ .05 & .97 \end{pmatrix} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \\ &= \begin{pmatrix} .95 \times 1 + .03 \times -1 \\ .05 \times 1 + .97 \times -1 \end{pmatrix} \\ &= \begin{pmatrix} .95 - .03 \\ .05 - .97 \end{pmatrix} \\ &= \begin{pmatrix} .92 \\ -.92 \end{pmatrix} \\ &= 0.92 \begin{pmatrix} 1 \\ -1 \end{pmatrix} \end{aligned}$$

So $\vec{u}^{(2)}$ is an eigenvector with eigenvalue 0.92.

d) Write the matrix $A_{\mathcal{U}}$ of the linear transformation $\vec{f}$ with respect to the basis $\mathcal{U} = \{\vec{u}^{(1)}, \vec{u}^{(2)}\}$.

Solution: The matrix has as its columns $[f(\vec{u}^{(1)})]_{\mathcal{U}}$ and $[f(\vec{u}^{(2)})]_{\mathcal{U}}$.

We know already that $f(\vec{u}^{(1)}) = \vec{u}^{(1)}$, which means

$$[f(\vec{u}^{(1)})]_{\mathcal{U}} = (1, 0)^T.$$

Similarly, $f(\vec{u}^{(2)}) = 0.92\vec{u}^{(2)}$, so

$$[f(\vec{u}^{(2)})]_{\mathcal{U}} = (0, .92)^T.$$

Therefore:

$$A_{\mathcal{U}} = \begin{pmatrix} 1 & 0 \\ 0 & 0.92 \end{pmatrix}$$

e) In part (f) of the last homework, you found that the initial population vector $\vec{x} = (600, 400)^T$ expressed in the new basis has coordinates $[\vec{x}]_{\mathcal{U}} = (1, 225)^T$.

Compute $A_{\mathcal{U}}[\vec{x}]_{\mathcal{U}}$ and then convert the resulting to a coordinate vector in the standard basis.

Hint: your result should be familiar.

Solution:

$$\begin{aligned} A_{\mathcal{U}}[\vec{x}]_{\mathcal{U}} &= \begin{pmatrix} 1 & 0 \\ 0 & .92 \end{pmatrix} \begin{pmatrix} 1 \\ 225 \end{pmatrix} \\ &= \begin{pmatrix} 1 \\ .92 \times 225 \end{pmatrix} \\ &= \begin{pmatrix} 1 \\ 207 \end{pmatrix} \end{aligned}$$

This is telling us that $\vec{f}(\vec{x}) = 1\vec{u}^{(1)} + 207\vec{u}^{(2)}$. Therefore, in the standard basis:

$$\begin{aligned} \vec{f}(\vec{x}) &= \vec{u}^{(1)} + 207\vec{u}^{(2)} \\ &= \begin{pmatrix} 375 \\ 625 \end{pmatrix} + 207 \begin{pmatrix} 1 \\ -1 \end{pmatrix} \\ &= \begin{pmatrix} 375 + 207 \\ 625 - 207 \end{pmatrix} \\ &= \begin{pmatrix} 582 \\ 418 \end{pmatrix} \end{aligned}$$

Problem 2. (1.5 credits)

In this problem, we will prove that for a symmetric matrix A, the unit vector $\vec{x}$ that maximizes $\|A\vec{x}\|^2$ is the eigenvector corresponding to the largest eigenvalue (in absolute value).[1]

Let A be a $d \times d$ symmetric matrix and let $\hat{u}^{(1)}, \hat{u}^{(2)}, \ldots, \hat{u}^{(d)}$ be d of its eigenvectors. Assume that they are

[1] This implies, by the way, that the unit vector that maximizes $\|A\vec{x}\|$ is also the eigenvector corresponding to the largest eigenvalue (in absolute value), since the square root function is monotonically increasing.

all mutually orthogonal and have unit norm (the spectral theorem guarantees this is possible to assume). Let $\lambda_1, \lambda_2, \ldots, \lambda_d$ be the corresponding eigenvalues, and assume that they are in decreasing order by magnitude. That is: $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_d|$.

a) The eigenvectors $\hat{u}^{(1)}, \ldots, \hat{u}^{(d)}$ form an orthonormal basis for $\mathbb{R}^d$. This means that any vector $\vec{x} \in \mathbb{R}^d$ can be written as a linear combination:

$$\vec{x} = a_1\hat{u}^{(1)} + a_2\hat{u}^{(2)} + \cdots + a_d\hat{u}^{(d)}$$

for some coefficients $a_1, \ldots, a_d \in \mathbb{R}$. This is called the *eigendecomposition* of $\vec{x}$ with respect to the eigenvectors of A.

If $\vec{x}$ is a unit vector (i.e., $\|\vec{x}\| = 1$), show that the coefficients must satisfy $a_1^2 + a_2^2 + \cdots + a_d^2 = 1$.

Hint: Use the fact that the eigenvectors are orthonormal, meaning $\hat{u}^{(i)} \cdot \hat{u}^{(j)} = 0$ if $i \neq j$ and $\hat{u}^{(i)} \cdot \hat{u}^{(i)} = 1$ for all i.

Solution: We start by remembering that $\|\vec{x}\|^2 = \vec{x} \cdot \vec{x}$. Writing $\vec{x}$ in terms of its eigendecomposition:

$$\|\vec{x}\|^2 = \left(a_1\hat{u}^{(1)} + a_2\hat{u}^{(2)} + \cdots + a_d\hat{u}^{(d)}\right) \cdot \left(a_1\hat{u}^{(1)} + a_2\hat{u}^{(2)} + \cdots + a_d\hat{u}^{(d)}\right)$$

Expanding this, we get what looks like a mess:

$$\begin{aligned}
&= a_1^2(\hat{u}^{(1)} \cdot \hat{u}^{(1)}) + a_1a_2(\hat{u}^{(1)} \cdot \hat{u}^{(2)}) + \cdots + a_1a_d(\hat{u}^{(1)} \cdot \hat{u}^{(d)}) \\
&\quad + a_2a_1(\hat{u}^{(2)} \cdot \hat{u}^{(1)}) + a_2^2(\hat{u}^{(2)} \cdot \hat{u}^{(2)}) + \cdots + a_2a_d(\hat{u}^{(2)} \cdot \hat{u}^{(d)}) \\
&\quad + \ldots \\
&\quad + a_da_1(\hat{u}^{(d)} \cdot \hat{u}^{(1)}) + a_da_2(\hat{u}^{(d)} \cdot \hat{u}^{(2)}) + \cdots + a_d^2(\hat{u}^{(d)} \cdot \hat{u}^{(d)})
\end{aligned}$$

However, we're in luck: because the eigenvectors are orthonormal, all of the "cross terms" (those with $i \neq j$) are zero and "vanish". Thus, we are left with only the terms where $i = j$:

$$= a_1^2(\hat{u}^{(1)} \cdot \hat{u}^{(1)}) + a_2^2(\hat{u}^{(2)} \cdot \hat{u}^{(2)}) + \cdots + a_d^2(\hat{u}^{(d)} \cdot \hat{u}^{(d)})$$

Now, because the $\hat{u}^{(i)}$ are unit vectors, we have $\hat{u}^{(i)} \cdot \hat{u}^{(i)} = 1$ for all i. Thus:

$$= a_1^2 + a_2^2 + \cdots + a_d^2$$

We've just shown that $\|\vec{x}\|^2 = a_1^2 + a_2^2 + \cdots + a_d^2$. Since we are told that $\vec{x}$ is a unit vector, we have $\|\vec{x}\|^2 = 1$. Therefore, $a_1^2 + a_2^2 + \cdots + a_d^2 = 1$.

b) Again let $\vec{x} = a_1\hat{u}^{(1)} + a_2\hat{u}^{(2)} + \cdots + a_d\hat{u}^{(d)}$ be the eigendecomposition of $\vec{x}$. Show that:

$$A\vec{x} = a_1\lambda_1\hat{u}^{(1)} + a_2\lambda_2\hat{u}^{(2)} + \cdots + a_d\lambda_d\hat{u}^{(d)}$$

Solution: We have:

$$\begin{aligned}
A\vec{x} &= A\left(a_1\hat{u}^{(1)} + a_2\hat{u}^{(2)} + \cdots + a_d\hat{u}^{(d)}\right) \\
&= a_1A\hat{u}^{(1)} + a_2A\hat{u}^{(2)} + \cdots + a_dA\hat{u}^{(d)}
\end{aligned}$$

Using the definition of eigenvectors and eigenvalues, we know that $A\hat{u}^{(i)} = \lambda_i \hat{u}^{(i)}$ for each i. Therefore:

$$= a_1\lambda_1\hat{u}^{(1)} + a_2\lambda_2\hat{u}^{(2)} + \cdots + a_d\lambda_d\hat{u}^{(d)}$$

c) Again let $\vec{x} = a_1\hat{u}^{(1)} + a_2\hat{u}^{(2)} + \cdots + a_d\hat{u}^{(d)}$ be the eigendecomposition of $\vec{x}$. Show that:

$$\|A\vec{x}\|^2 = \lambda_1^2 a_1^2 + \lambda_2^2 a_2^2 + \cdots + \lambda_d^2 a_d^2$$

Solution: Using the result from part (b):

$$\begin{aligned}\|A\vec{x}\|^2 &= (A\vec{x})^T(A\vec{x}) \\ &= \left(a_1\lambda_1\hat{u}^{(1)} + a_2\lambda_2\hat{u}^{(2)} + \cdots + a_d\lambda_d\hat{u}^{(d)}\right)^T \left(a_1\lambda_1\hat{u}^{(1)} + a_2\lambda_2\hat{u}^{(2)} + \cdots + a_d\lambda_d\hat{u}^{(d)}\right)\end{aligned}$$

Remember that, for a vector $\vec{z}$, $\vec{z}^T\vec{z} = \vec{z}\cdot\vec{z}$, so we can rewrite this as a dot product:

$$= \left(a_1\lambda_1\hat{u}^{(1)} + a_2\lambda_2\hat{u}^{(2)} + \cdots + a_d\lambda_d\hat{u}^{(d)}\right) \cdot \left(a_1\lambda_1\hat{u}^{(1)} + a_2\lambda_2\hat{u}^{(2)} + \cdots + a_d\lambda_d\hat{u}^{(d)}\right)$$

Similar to part (a), when we expand this dot product, all cross terms vanish due to orthonormality of the eigenvectors. We are left with:

$$= a_1^2\lambda_1^2(\hat{u}^{(1)}\cdot\hat{u}^{(1)}) + a_2^2\lambda_2^2(\hat{u}^{(2)}\cdot\hat{u}^{(2)}) + \cdots + a_d^2\lambda_d^2(\hat{u}^{(d)}\cdot\hat{u}^{(d)})$$

Using the fact that the eigenvectors are unit vectors so that $\hat{u}^{(i)}\cdot\hat{u}^{(i)} = 1$ for all i, we have:

$$= a_1^2\lambda_1^2 + a_2^2\lambda_2^2 + \cdots + a_d^2\lambda_d^2$$

Therefore, $\|A\vec{x}\|^2 = \lambda_1^2 a_1^2 + \lambda_2^2 a_2^2 + \cdots + \lambda_d^2 a_d^2$.

d) Remember our original goal: we want to find a unit vector $\vec{x}$ that maximizes $\|A\vec{x}\|^2$.

From part (c), we know that we can write $\|A\vec{x}\|^2$ as

$$\lambda_1^2 a_1^2 + \lambda_2^2 a_2^2 + \cdots + \lambda_d^2 a_d^2,$$

and from part (a), we know that if $\vec{x}$ is a unit vector, then

$$a_1^2 + a_2^2 + \cdots + a_d^2 = 1.$$

So, maximizing $\|A\vec{x}\|^2$ over unit vectors $\vec{x}$ is equivalent to maximizing $\lambda_1^2a_1^2+\lambda_2^2a_2^2+\cdots+\lambda_d^2a_d^2$. subject to the constraint $a_1^2 + a_2^2 + \cdots + a_d^2 = 1$.

What choice of $a_1, a_2, \ldots, a_d$ maximizes this quantity? You don't need to *rigorously* prove that your choice is the best one; just explain your reasoning.

Hint: think of the constraint as a "budget". That is, you have a total of 1 unit to distribute among $a_1^2, a_2^2, \ldots, a_d^2$. You'll want to use the fact that $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_d|$ to figure out the best allocation.

Solution: Since $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_d|$, we have $\lambda_1^2 \geq \lambda_2^2 \geq \cdots \geq \lambda_d^2$.

The objective $\sum_{i=1}^d \lambda_i^2 a_i^2$ is a weighted sum of the a_i^2 values, where the weights are λ_i^2. Since we must have $\sum_{i=1}^d a_i^2 = 1$, we are distributing a fixed "budget" of 1 among the terms.

To maximize the weighted sum, we should allocate all of our budget to the term with the largest weight, which is λ_1^2. This means setting $a_1^2 = 1$ (i.e., $a_1 = \pm 1$) and $a_i^2 = 0$ (i.e., $a_i = 0$) for all $i \neq 1$.

The maximum value is:

$$\|A\vec{x}\|^2 = \lambda_1^2 \cdot 1 + \lambda_2^2 \cdot 0 + \cdots + \lambda_d^2 \cdot 0 = \lambda_1^2$$

The maximizing vector is $\vec{x} = \pm \hat{u}^{(1)}$, the eigenvector corresponding to the largest eigenvalue (in absolute value). Therefore:

$$\max_{\|\vec{x}\|=1} \|A\vec{x}\| = |\lambda_1|$$

This quantity $|\lambda_1|$ is called the *spectral norm* of A, often denoted $\|A\|_2$.

e) The last part effectively proved that the unit vector maximizing $\|A\vec{x}\|^2$ is the eigenvector corresponding to the largest eigenvalue (in absolute value). Explain (in just a sentence or two) why this follows.

Solution: The last part showed that $\|A\vec{x}\|^2$ is maximized when $\vec{x}$ is maximized by setting $a_1 = 1$ and $a_i = 0$ for all $i \neq 1$ in the eigendecomposition $\vec{x} = a_1\hat{u}^{(1)} + a_2\hat{u}^{(2)} + \cdots + a_d\hat{u}^{(d)}$. That is, by taking $\vec{x} = \hat{u}^{(1)}$. But this is simply the eigenvector corresponding to the largest eigenvalue (in absolute value) λ_1.

f) Now consider a related but different problem: maximizing $vecx^T A\vec{x}$ subject to $\|\vec{x}\| = 1$. Using a similar approach as above, show that this is maximized by taking $\vec{x}$ to be the eigenvector corresponding to the largest eigenvalue (*not* in absolute value). What is the maximum value of $\vec{x}^T A\vec{x}$ in this case?

Note: Here we care about the largest eigenvalue itself, not the largest in absolute value. For instance, if $\lambda_1 = 5$ and $\lambda_2 = -10$, the maximum of $\vec{x}^T A\vec{x}$ is 5, not 10.

Solution: We again use the strategy of writing $\vec{x}$ in terms of its eigendecomposition:

$$\vec{x} = a_1\hat{u}^{(1)} + a_2\hat{u}^{(2)} + \cdots + a_d\hat{u}^{(d)}$$

with the constraint that $a_1^2 + a_2^2 + \cdots + a_d^2 = 1$.

We'll start by computing $\vec{x}^T A\vec{x}$. We already know from part (b) that:

$$A\vec{x} = a_1\lambda_1\hat{u}^{(1)} + a_2\lambda_2\hat{u}^{(2)} + \cdots + a_d\lambda_d\hat{u}^{(d)}$$

Therefore:

$$\begin{aligned}
\vec{x}^T A\vec{x} &= \vec{x}^T \left(a_1\lambda_1\hat{u}^{(1)} + a_2\lambda_2\hat{u}^{(2)} + \cdots + a_d\lambda_d\hat{u}^{(d)}\right) \\
&= \left(a_1\hat{u}^{(1)} + a_2\hat{u}^{(2)} + \cdots + a_d\hat{u}^{(d)}\right)^T \left(a_1\lambda_1\hat{u}^{(1)} + a_2\lambda_2\hat{u}^{(2)} + \cdots + a_d\lambda_d\hat{u}^{(d)}\right)
\end{aligned}$$

Rewriting as a dot product:

$$= \left(a_1\hat{u}^{(1)} + a_2\hat{u}^{(2)} + \cdots + a_d\hat{u}^{(d)}\right) \cdot \left(a_1\lambda_1\hat{u}^{(1)} + a_2\lambda_2\hat{u}^{(2)} + \cdots + a_d\lambda_d\hat{u}^{(d)}\right)$$

Expanding and using orthonormality to eliminate cross terms:

$$= a_1^2\lambda_1(\hat{u}^{(1)} \cdot \hat{u}^{(1)}) + a_2^2\lambda_2(\hat{u}^{(2)} \cdot \hat{u}^{(2)}) + \cdots + a_d^2\lambda_d(\hat{u}^{(d)} \cdot \hat{u}^{(d)})$$

Using the fact that $\hat{u}^{(i)} \cdot \hat{u}^{(i)} = 1$ for all i:

$$= a_1^2 \lambda_1 + a_2^2 \lambda_2 + \cdots + a_d^2 \lambda_d$$

Remember that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$, and that we're subject to the constraint $a_1^2 + a_2^2 + \cdots + a_d^2 = 1$. That is, we have a budget of 1 to distribute among the a_i^2 values. Clearly, we want to allocate all of our budget to the term with the largest weight, which is λ_1. Thus, we set $a_1^2 = 1$ (i.e., $a_1 = \pm 1$) and $a_i^2 = 0$ (i.e., $a_i = 0$) for all $i \neq 1$.

This implies that the unit vector $\vec{x}$ that maximizes $\vec{x}^T A \vec{x}$ is $\vec{x} = \pm \hat{u}^{(1)}$, the eigenvector corresponding to the largest eigenvalue λ_1. The maximum value of $\vec{x}^T A \vec{x}$ is then λ_1.

Problem 3. (1.5 credits)

Let $g(\vec{x}) = g(x_1, x_2) = 4x_1^2 + 3x_2^2 + 10x_1x_2$, where we've defined $\vec{x} = (x_1, x_2)^T$. In this problem, we will consider maximizing g subject to the constraint $x_1^2 + x_2^2 = 1$.

You saw how to solve optimization problems like this in your multivariate calculus class using the method of *Lagrange multipliers*. Informally-speaking, the idea behind Lagrange multipliers is that the gradient vector of g and the gradient of the constraint $x_1^2 + x_2^2 - 1$ should be parallel at a constrained optimum. Since two vectors $\vec{a}$ and $\vec{b}$ are parallel if and only if $\vec{a} = \lambda \vec{b}$ for some λ, and since the gradient of the constraint is simply $(2x_1, 2x_2)^T = 2\vec{x}$, this means that a local optimum should satisfy $\nabla g(\vec{x}) = 2\lambda \vec{x}$. This looks similar to the eigenvector equation $A\vec{x} = \lambda \vec{x}$; in this problem we'll make the connection clearer.

a) The Lagrange multiplier approach says that we should define the *Lagrangian*:

$$\mathcal{L}(x_1, x_2, \lambda) = g(\vec{x}) - \lambda(x_1^2 + x_2^2 - 1)$$

We then solve the system of three equations in three unknowns:

$$\frac{\partial \mathcal{L}}{\partial x_1} = 0$$

$$\frac{\partial \mathcal{L}}{\partial x_2} = 0$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = 0$$

Write out and solve this system for x_1, x_2, and λ.

Hint: Try to get a formula for x_1^2 in terms of λ only, and same for x_2^2. When you get to this point, you will be able to substitute your formulas for x_1^2 and x_2^2 into $\partial \mathcal{L}/\partial \lambda = 0$ to get a function of the form

$$\frac{a_1}{(b_1\lambda + c_1)^2 + d_1} + \frac{a_2}{(b_2\lambda + c_2)^2 + d_2} - 1 = 0,$$

where the a, b, c, d's are all constants. We want to solve this for λ, which is not easy to do analytically. Instead, solve it numerically using `scipy.optimize.fsolve`, or similar. Once you've solved for λ, you can plug it back in to your equations for x_1^2 and x_2^2. Some of the possible combinations of x_1 and x_2 you get may not actually solve the original system of equations; be sure to check which ones do by plugging them back into the original equations and making sure they equal zero.

Note: when you use code (like `fsolve`) to solve part of the problem, you do *not* need to include your code in your writeup. Just describe what you did.

Solution: The partial derivatives are:

$$\frac{\partial \mathcal{L}}{\partial x_1} = 8x_1 + 10x_2 - 2\lambda x_1$$
$$\frac{\partial \mathcal{L}}{\partial x_2} = 6x_2 + 10x_1 - 2\lambda x_2$$
$$\frac{\partial \mathcal{L}}{\partial \lambda} = x_1^2 + x_2^2 - 1$$

Setting the last equations to zero and solving for (alternately) x_1 and x_2 yields:

$$x_1 = \sqrt{1 - x_2^2}$$
$$x_2 = \sqrt{1 - x_1^2}$$

We can plug these into the first two equations to reduce the number of variables in each. For instance, the first equation becomes:

$$8x_1 + 10x_2 - 2\lambda x_1 = 0$$
$$\implies (8 - 2\lambda)x_1 + 10\sqrt{1 - x_1^2} = 0$$
$$\implies 10\sqrt{1 - x_1^2} = (2\lambda - 8)x_1$$

Squaring both sides:

$$\implies 100(1 - x_1^2) = (2\lambda - 8)^2 x_1^2$$
$$\implies 100 - 100x_1^2 = (2\lambda - 8)^2 x_1^2$$
$$\implies 100 = \left[(2\lambda - 8)^2 + 100\right] x_1^2$$
$$\implies x_1^2 = 100/\left[(2\lambda - 8)^2 + 100\right]$$

Similarly for x_2, we have:

$$6x_2 + 10x_1 - 2\lambda x_2 = 0$$
$$\implies (6 - 2\lambda)x_2 + 10\sqrt{1 - x_2^2} = 0$$
$$\implies 10\sqrt{1 - x_2^2} = (2\lambda - 6)x_2$$

Squaring both sides:

$$\implies 100(1 - x_2^2) = (2\lambda - 6)^2 x_2^2$$
$$\implies 100 - 100x_2^2 = (2\lambda - 6)^2 x_2^2$$
$$\implies 100 = \left[(2\lambda - 6)^2 + 100\right] x_2^2$$
$$\implies x_2^2 = 100/\left[(2\lambda - 6)^2 + 100\right]$$

Plugging these back into the formula for $\partial\mathcal{L}/\partial\lambda = 0$, we get:

$$\begin{aligned}
&\partial\mathcal{L}/\partial\lambda = 0 \\
&\implies x_1^2 + x_2^2 - 1 = 0 \\
&\implies \frac{100}{(2\lambda - 8)^2 + 100} + \frac{100}{(2\lambda - 6)^2 + 100} - 1 = 0
\end{aligned}$$

Solving this by hand is not possible. Instead, we can solve it numerically using a solver, such as `scipy.optimize.fsolve`. We also plot the function so that we can get a sense for where its roots are:

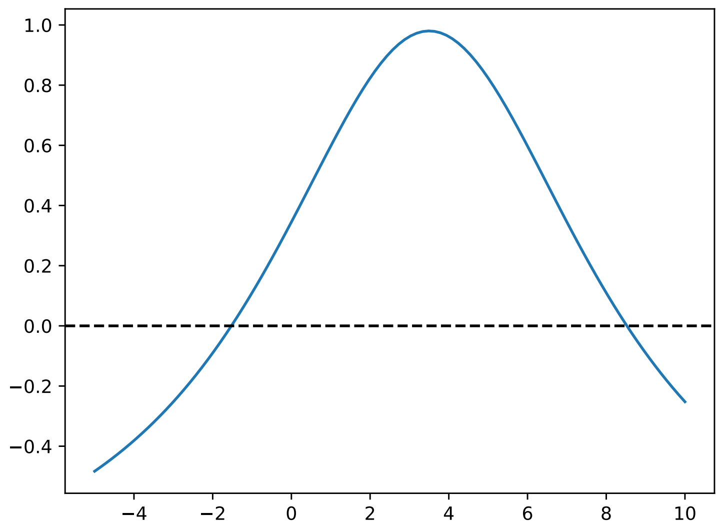
```
import numpy as np
import matplotlib.pyplot as plt
import scipy.optimize

def f(x):
    return (
            100 / ((2*x - 8)**2 + 100)
            +
            100 / ((2*x - 6)**2 + 100)
            -
            1
        )

x = np.linspace(-5, 10, 100)
plt.plot(x, f(x))
plt.axhline(0, color='black', linestyle='--')
plt.show()

print(scipy.optimize.fsolve(f, -2))
print(scipy.optimize.fsolve(f, 9))
```

Now, there are multiple roots of the function, as the plot shows:



We guess that they are around -2 and 8. Using these as the starting location in `scipy.optimize.fsolve`, we get two approximated roots: -1.525 and 8.525.

We plug these back into our formulas for x_1^2 and x_2^2:

$$x_1^2 = 100/\left[(2\lambda - 8)^2 + 100\right] =$$
$$x_2^2 = 100/\left[(2\lambda - 6)^2 + 100\right] =$$

With $\lambda = -1.525$, we get:

$$x_1 = \pm 0.671$$
$$x_2 = \pm 0.741$$

With $\lambda = 8.525$, we get:

$$x_1 = \pm 0.741$$
$$x_2 = \pm 0.671$$

This seems to yield 8 possible combinations of x_1, x_2, λ, but not all combinations actually solve the original system of equations. For example, $x_1 = .67$, $x_2 = .74$, and $\lambda = -1.525$ does not satisfy $\partial L / \partial x_1 = 0$. Filtering out the four non-solutions, we have:

x_1	x_2	λ	$g(x_1, x_2)$
0.67	-0.74	-1.525	-1.52
-0.67	0.74	-1.525	-1.52
0.74	0.67	8.525	8.52
-0.74	-0.67	8.525	8.52

b) The equation $g(\vec{x}) = 4x_1^2 + 3x_2^2 + 10x_1x_2$ can be written in matrix-vector form as $g(\vec{x}) = \vec{x}^T A \vec{x}$ for an appropriately-defined matrix, A. Find this matrix A, and show that the matrix form is equivalent to the original form.

You can assume that A is symmetric.

Solution: Define

$$A = \begin{pmatrix} 4 & 5 \\ 5 & 3 \end{pmatrix}$$

We have:

$$\begin{aligned} A\vec{x} &= \begin{pmatrix} 4 & 5 \\ 5 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\ &= \begin{pmatrix} 4x_1 + 5x_2 \\ 5x_1 + 3x_2 \end{pmatrix} \end{aligned}$$

So:

$$\begin{aligned} \vec{x}^T A\vec{x} &= \begin{pmatrix} x_1 & x_2 \end{pmatrix} \begin{pmatrix} 4x_1 + 5x_2 \\ 5x_1 + 3x_2 \end{pmatrix} \\ &= 4x_1^2 + 5x_1x_2 + 5x_1x_2 + 3x_2^2 \\ &= 4x_1^2 + 10x_1x_2 + 3x_2^2 \end{aligned}$$

c) Using whatever method you choose (e.g., numpy), compute the eigenvectors and eigenvalues of A. Show that the eigenvectors are the same as your solution to part (a).

Solution:

```
>>> import numpy as np
>>> A = np.array([[4, 5], [5, 3]])
>>> evals, evecs = np.linalg.eigh(A)
>>> print(evals)
[-1.52493781  8.52493781]
>>> print(evecs)
[[ 0.67100532 -0.74145253]
 [-0.74145253 -0.67100532]]
```

The columns of the `evecs` array are the two eigenvectors of A. We have two solutions: $\hat{u}^{(1)} = (0.671, -0.741)^T$ and $\hat{u}^{(2)} = (-0.741, -0.671)^T$. These are the same as the solutions to the first part. Note that in the first part, we also have $(-0.671, 0.741)^T$ and $(0.741, 0.671)^T$, but these are just $-\hat{u}^{(1)}$ and $-\hat{u}^{(2)}$, and are therefore not really "different" solutions.

d) We saw in lecture that a matrix can be interpreted as the representation of a linear transformation $\vec{f}(\vec{x})$. It turns out that A represents the *gradient* of $\vec{g}$.

Show that A represents the linear transformation $\vec{f}(\vec{x}) = \frac{1}{2}\nabla g(\vec{x})$, $\nabla g(\vec{x}) = (\partial g/\partial x_1,$ where $\partial g/\partial x_2)^T$ is the *gradient* of g.

Solution: We have:

$$\frac{\partial g}{\partial x_1} = 8x_1 + 6x_2 \qquad \frac{\partial g}{\partial x_2} = 6x_2 + 6x_1$$

So the gradient vector is:

$$\nabla g(\vec{x}) = \begin{pmatrix} 8x_1 + 10x_2 \\ 10x_2 + 6x_1 \end{pmatrix}$$

and

$$\vec{f}(\vec{x}) = \frac{1}{2}\nabla g(\vec{x}) = \begin{pmatrix} 4x_1 + 5x_2 \\ 5x_2 + 3x_1 \end{pmatrix}$$

Whereas:

$$A\vec{x} = \begin{pmatrix} 4 & 5 \\ 5 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 4x_1 + 5x_2 \\ 5x_1 + 3x_2 \end{pmatrix} = \frac{1}{2}\nabla g(\vec{x})$$

Therefore, for a function g of the form $ax_1^2 + bx_2^2 + cx_1x_2$, the gradient is a linear transformation that can be computed by a matrix multiplication, and the method of Lagrange multipliers is equivalent to finding an eigenvector of the matrix representing the gradient.