Pr. 1.
Show that a weighted sum of squared Euclidean distances between $\boldsymbol{x} \in \mathbb{F}^N$ and some points $\{\boldsymbol{z}_k\}$ is equivalent, within a constant, to the squared distance between $\boldsymbol{x}$ and a weighted average of the points:

$$\sum_k \alpha_k \|\boldsymbol{x} - \boldsymbol{z}_k\|_2^2 = \alpha \left\| \boldsymbol{x} - \sum_k \frac{\alpha_k}{\alpha} \boldsymbol{z}_k \right\|_2^2 + \beta,$$

where each α_k is real, and where you must determine α and β. Hint: This is related to **completing the square**.

Pr. 2.
The **logarithm** of a square matrix $\boldsymbol{A}$ is defined as $\log(\boldsymbol{A}) = -\sum_{k=1}^{\infty} \frac{1}{k}(I - \boldsymbol{A})^k$, provided that $\|\boldsymbol{A} - \boldsymbol{I}\| < 1$, i.e., if $\boldsymbol{A}$ is sufficiently close to $\boldsymbol{I}$. Show that if $\boldsymbol{A}$ is diagonalizable with (nonnegative) eigenvalues $\{\lambda_j\}$, then a matrix version of **entropy** is $H(\boldsymbol{A}) \triangleq -\text{trace}(\boldsymbol{A} \log(\boldsymbol{A})) = -\sum_j \lambda_j \log \lambda_j$.

Pr. 3.
A **flat** or **linear variety** or **affine subspace** is a subset of a vector space $\mathcal{V}$ defined by $\{\boldsymbol{v} \in \mathcal{V} : \boldsymbol{v} = \boldsymbol{\mu} + \text{span}(\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots\})\}$ for some vectors $\boldsymbol{\mu}, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots \in \mathcal{V}$. Show that any such affine subspace is a **convex set**, or provide a counter-example.

Pr. 4.
A real, symmetric matrix $\boldsymbol{A}$ is positive semidefinite iff all of its eigenvalues are non-negative, *i.e.*, iff we can write $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{U}'$ for $\boldsymbol{U}$ orthogonal and $\boldsymbol{\Sigma}$ diagonal with $\sigma_i \geq 0$. We write $\boldsymbol{A} \succeq \boldsymbol{0}$. A useful property is that if $\boldsymbol{A} \succeq \boldsymbol{0}, \boldsymbol{B} \succeq \boldsymbol{0}$, then $\text{trace}(\boldsymbol{A}\boldsymbol{B}) \geq 0$. Let $\boldsymbol{X}, \boldsymbol{K}, \boldsymbol{F}_k \in \mathbb{R}^{N \times N}$ for $k = 1, \ldots, K$. Let $\boldsymbol{c}, \boldsymbol{z} \in \mathbb{R}^K$. Show that $\text{trace}(\boldsymbol{K}\boldsymbol{X}) \geq \boldsymbol{c}^\top \boldsymbol{z}$, assuming the following:

- $\boldsymbol{X} \succeq \boldsymbol{0}$
- $\text{trace}(\boldsymbol{F}_k \boldsymbol{X}) = c_k$ for $k = 1, \ldots, K$
- $\boldsymbol{K} - \sum_{k=1}^{K} z_k \boldsymbol{F}_k \succeq \boldsymbol{0}$.

This result is known as **weak duality** for semidefinite optimization problems.

Pr. 5.
For a nonzero vector $\boldsymbol{v} \in \mathbb{F}^N$ and nonzero scalar $b \in \mathbb{F}$, define the hyperplane (flat) set $\mathcal{C} = \left\{\boldsymbol{x} \in \mathbb{F}^N : \boldsymbol{v}'\boldsymbol{x} = b\right\}$.

(a) Verify that $\mathcal{C}$ is a convex, even though it is not a subspace.

(b) Find an expression for the projection of a point $\boldsymbol{u} \in \mathbb{F}^N$ onto the set $\mathcal{C}$.
Hint: first translate the coordinates so that the plane intersects the origin.

Pr. 6.
Let $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{F}^N$ denote orthogonal vectors. Define $\boldsymbol{X} \triangleq \boldsymbol{x}\boldsymbol{x}'$ and $\boldsymbol{Y} \triangleq \boldsymbol{y}\boldsymbol{y}'$. Determine whether $\text{vec}(\boldsymbol{X})$ and $\text{vec}(\boldsymbol{Y})$ are perpendicular vectors. Hint. Consider the **Frobenius inner product**.

Pr. 7.
The definition of the Frobenius norm of a matrix $\boldsymbol{A}$ requires accessing every element a_{ij} of that matrix. Some matrices are represented numerically as *operators* that act on vectors, rather than storing an entire dense matrix, *i.e.*, functions that compute the matrix-vector product $\boldsymbol{A}\boldsymbol{x}$ for any vector $\boldsymbol{x}$. One can *estimate* the Frobenius norm of such an operator by generating K random vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K$ and then computing

$$\hat{F} = \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{A}\boldsymbol{x}\|_2^2 .$$

Show that this average is an **unbiased estimate** of the squared Frobenius norm of $\boldsymbol{A}$, i.e., $\mathbb{E}\left(\hat{F}\right) = \|\boldsymbol{A}\|_{\mathrm{F}}^2$.

Pr. 8.
Unconstrained "Procrustes" method
For given $M \times N$ matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$, the **orthogonal Procrustes problem** involves minimizing $\|\boldsymbol{Y} - \boldsymbol{Q}\boldsymbol{X}\|_{\mathrm{F}}$ over $M \times M$ matrices $\boldsymbol{Q}$, subject to the *constraint* that $\boldsymbol{Q}$ is orthogonal (or unitary). Here we relax that constraint.

(a) Verify that $(\boldsymbol{A} \otimes \boldsymbol{B})^+ = \boldsymbol{A}^+ \otimes \boldsymbol{B}^+$.

(b) Determine analytically the *unconstrained* minimizer $\hat{\boldsymbol{Q}} = \arg\min_{\boldsymbol{Q} \in \mathbb{F}^{M \times M}} \|\boldsymbol{Y} - \boldsymbol{Q}\boldsymbol{X}\|_{\mathrm{F}}$.
Hint. The **vec trick** and the previous part may be useful.

(c) Write a function called `procunc` that returns that minimizer $\hat{\boldsymbol{Q}}$ given $\boldsymbol{X}, \boldsymbol{Y} \in \mathbb{F}^{M \times N}$. Your solution should be efficient for the case where $M \ll N$.

In `Julia`, your file should be named `procunc.jl` and should contain the following function:

```julia
"""
    Qh = procunc(X, Y)

In:
* `X` and `Y` are `M × N` matrices, typically with `M ≪ N`

Out:
* `Qh` `M × M` matrix that minimizes the "unconstrained" Procrustes problem:
  `\\argmin_{Q} ‖ Y - Q X ‖_F`
"""
function procunc(X::Matrix{<:Number}, Y::Matrix{<:Number})
```

Email your solution as an attachment to `eecs551@autograder.eecs.umich.edu`.

(d) Submit a screenshot of your code to gradescope so that the grader can verify efficiency.

Pr. 9.
Consider the matrix $\boldsymbol{A} = \boldsymbol{B} \oplus \boldsymbol{C}$, where $\boldsymbol{B}$ is the $N \times N$ circulant matrix whose first column is $\boldsymbol{b} \in \mathbb{F}^N$, and $\boldsymbol{C}$ is the $M \times M$ circulant matrix whose first column is $\boldsymbol{c} \in \mathbb{F}^M$. This problem shows how to use Fourier transforms to efficiently compute $\boldsymbol{y} = \boldsymbol{A}^{-1}\boldsymbol{x}$.

(a) Show that

$$\boldsymbol{A} = \frac{1}{MN}(\boldsymbol{F}_N \otimes \boldsymbol{F}_M)'(\boldsymbol{\Lambda}_b \oplus \boldsymbol{\Lambda}_c)(\boldsymbol{F}_N \otimes \boldsymbol{F}_M),$$

where $\boldsymbol{\Lambda}_b = \mathrm{Diag}(\boldsymbol{f}_b)$ and $\boldsymbol{\Lambda}_c = \mathrm{Diag}(\boldsymbol{f}_c)$ are the diagonal matrices containing the DFTs of $\boldsymbol{b}$ and $\boldsymbol{c}$, respectively, and $\boldsymbol{F}_N$ is the $N \times N$ DFT matrix.

Hint. Use the fact that any $N \times N$ circulant matrix $\boldsymbol{B}$ can be written as $\boldsymbol{B} = (1/N)\boldsymbol{F}_N'\boldsymbol{\Lambda}_b\boldsymbol{F}_N$.

Define $\boldsymbol{Q}_b = (1/\sqrt{N})\boldsymbol{F}_N'$ and $\boldsymbol{Q}_c = (1/\sqrt{M})\boldsymbol{F}_M'$ and apply your result from a previous HW.

(b) Show that

$$\boldsymbol{A}^{-1} = \frac{1}{MN}(\boldsymbol{F}_N \otimes \boldsymbol{F}_M)'(\boldsymbol{\Lambda}_b \oplus \boldsymbol{\Lambda}_c)^{-1}(\boldsymbol{F}_N \otimes \boldsymbol{F}_M).$$

(c) Show that the $M \times N$ matrix

$$\boldsymbol{P} = \boldsymbol{1}_M \boldsymbol{f}_b' + \boldsymbol{f}_c \boldsymbol{1}_N',$$

containing all pairwise sums of the entries of $\boldsymbol{f}_b$ and $\boldsymbol{f}_c$ (*i.e.*, eigenvalues of $\boldsymbol{B}$ and $\boldsymbol{C}$), satisfies the relationship $\boldsymbol{\Lambda}_b \oplus \boldsymbol{\Lambda}_c = \mathrm{Diag}(\mathrm{vec}(\boldsymbol{P}))$. In other words, the columns of $\boldsymbol{P}$ stacked into a vector form the diagonal of the matrix $\boldsymbol{\Lambda}_b \oplus \boldsymbol{\Lambda}_c$.

(d) Suppose $\boldsymbol{y} = \boldsymbol{A}^{-1}\boldsymbol{x}$, and let $\boldsymbol{X}$ and $\boldsymbol{Y}$ be the $M \times N$ matrices such that $\boldsymbol{x} = \mathrm{vec}(\boldsymbol{X})$ and $\boldsymbol{y} = \mathrm{vec}(\boldsymbol{Y})$. Use the above results and your answer to a previous HW problem. to argue that

$$\boldsymbol{Y} = \frac{1}{MN}\boldsymbol{F}_M'\left(\frac{\boldsymbol{F}_M \boldsymbol{X} \boldsymbol{F}_N^\top}{\boldsymbol{P}}\right)\overline{\boldsymbol{F}_N}, \tag{1}$$

where the fraction denotes elementwise division, and $\overline{\boldsymbol{F}_N}$ denotes the (elementwise) complex conjugate of $\boldsymbol{F}_N$.

(e) Explain why the `Julia` code `Y = ifft(fft(X) ./ P)` implements (1).

Demo 08/kron-sum-inv illustrates that the FFT-based approach works.

Pr. 10.
(Closed-form photometric stereo via Fourier transforms)
Recall that the photometric stereo problem of reconstructing a surface from its gradients can be formulated as

$$\min_{\boldsymbol{x} \in \mathbb{R}^{mn}} \|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2, \tag{2}$$

where $\boldsymbol{x} = \text{vec}(\boldsymbol{X})$ is the vectorized $m \times n$ surface that we would like to estimate, and

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{I}_n \otimes \boldsymbol{D}_m \\ \boldsymbol{D}_n \otimes \boldsymbol{I}_m \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} \boldsymbol{b}_x \\ \boldsymbol{b}_y \end{bmatrix}.$$

In the above, $\boldsymbol{D}_n \in \mathbb{R}^{n \times n}$ is the circulant first differences matrix

$$\boldsymbol{D}_n = \begin{bmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \\ 1 & & & & -1 \end{bmatrix}$$

and $\boldsymbol{b}_x = \text{vec}(\boldsymbol{B}_x)$ and $\boldsymbol{b}_y = \text{vec}(\boldsymbol{B}_y)$, where $\boldsymbol{B}_x$ and $\boldsymbol{B}_y$ denote $m \times n$ matrices containing the x and y gradients, respectively, of the underlying surface.
In previous homework problems, you constructed the (sparse!) $\boldsymbol{A}$ matrix explicitly and employed various iterative solvers to compute the solution to (2). This problem derives a *closed-form* solution using Fourier transforms!
Recall that the solution to (2) is given by $\hat{\boldsymbol{x}} = \boldsymbol{A}^+\boldsymbol{b}$. If $\boldsymbol{A}^T\boldsymbol{A}$ were invertible, we could compute this solution as $\hat{\boldsymbol{x}} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}$. Unfortunately, it is not.

(a) What is the rank of $\boldsymbol{A}^T\boldsymbol{A}$? You can check this numerically, at least for small values of m and n.

What is the eigenvector corresponding to the smallest eigenvalue? Does this eigenvector make sense intuitively?

(b) To facilitate a Fourier transform-based solution, we will instead consider the regularized problem

$$\min_{\boldsymbol{x} \in \mathbb{R}^{mn}} \|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu\|\boldsymbol{x}\|_2^2, \tag{3}$$

for some (small) value of $\mu > 0$. The solution to (3) (for any $\boldsymbol{A}$) is

$$\hat{\boldsymbol{x}} = (\mu\boldsymbol{I}_{mn} + \boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}.$$

For practical values of m and n, the $mn \times mn$ matrix $\mu\boldsymbol{I}_{mn} + \boldsymbol{A}^T\boldsymbol{A}$ is too large to invert directly. However, in this application it has special structure: we can write it as

$$\boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{C}_n \oplus \boldsymbol{C}_m, \tag{4}$$

where $\boldsymbol{C}_n$ and $\boldsymbol{C}_m$ are $n \times n$ and $m \times m$ circulant matrices, respectively.

What are $\boldsymbol{C}_n$ and $\boldsymbol{C}_m$?

Equation (4) is exactly the form considered in the previous problem, so we can write it as

$$\boldsymbol{A}^T\boldsymbol{A} = \frac{1}{mn}(\boldsymbol{F}_n \otimes \boldsymbol{F}_m)'(\boldsymbol{\Lambda}_n \oplus \boldsymbol{\Lambda}_m)(\boldsymbol{F}_n \otimes \boldsymbol{F}_m),$$

where $\boldsymbol{\Lambda}_n$ and $\boldsymbol{\Lambda}_m$ are the diagonal matrices containing the DFT of the first columns of $\boldsymbol{C}_n$ and $\boldsymbol{C}_m$, respectively, and $\boldsymbol{F}_n$ denotes the $n \times n$ DFT matrix.

(c) Show that

$$(\mu\boldsymbol{I}_{mn} + \boldsymbol{A}^T\boldsymbol{A})^{-1} = \frac{1}{mn}(\boldsymbol{F}_n \otimes \boldsymbol{F}_m)'(\mu\boldsymbol{I}_{mn} + \boldsymbol{\Lambda}_n \oplus \boldsymbol{\Lambda}_m)^{-1}(\boldsymbol{F}_n \otimes \boldsymbol{F}_m).$$

(d) Show that $\boldsymbol{\Lambda}_n$ and $\boldsymbol{\Lambda}_m$ are diagonal matrices of the form

$$\begin{aligned} (\boldsymbol{\Lambda}_n)_{jj} &= 2 - 2\cos(2\pi(j-1)/n), \\ (\boldsymbol{\Lambda}_m)_{kk} &= 2 - 2\cos(2\pi(k-1)/m), \end{aligned}$$

for $j = 1, \ldots, n$ and $k = 1, \ldots, m$, respectively.

Hint: The first columns of $\boldsymbol{C}_n$ and $\boldsymbol{C}_m$ have only three nonzero entries, so it is straightforward to compute the required DFTs in closed-form.

Hint: Use $\exp(i\theta) = \cos(\theta) + i\sin(\theta)$ to simplify your answer.

(e) Let $\boldsymbol{Z}$ be the $m \times n$ matrix such that $\text{vec}(\boldsymbol{Z}) = \boldsymbol{A}^T \boldsymbol{b}$. Show that

$$\boldsymbol{Z} = \boldsymbol{D}_m^T \boldsymbol{B}_x + \boldsymbol{B}_y \boldsymbol{D}_n.$$

In parts (a)-(e), we have shown that

$$\hat{\boldsymbol{x}} = (\mu \boldsymbol{I}_{mn} + \boldsymbol{A}^T \boldsymbol{A})^{-1} \boldsymbol{A}^T \boldsymbol{b}$$
$$\frac{1}{mn} (\boldsymbol{F}_n \otimes \boldsymbol{F}_m)' (\mu \boldsymbol{I}_{mn} + \boldsymbol{\Lambda}_n \oplus \boldsymbol{\Lambda}_m)^{-1} (\boldsymbol{F}_n \otimes \boldsymbol{F}_m) \text{vec}(\boldsymbol{D}_m^T \boldsymbol{B}_x + \boldsymbol{B}_y \boldsymbol{D}_n).$$

Using the previous results, we rearrange the above computations into $m \times n$ matrix operations:

$$\hat{\boldsymbol{X}} = \frac{1}{mn} \boldsymbol{F}_m' \left[\frac{\boldsymbol{F}_m \left(\boldsymbol{D}_m^T \boldsymbol{B}_x + \boldsymbol{B}_y \boldsymbol{D}_n \right) \boldsymbol{F}_n^T}{\mu \boldsymbol{1}_m \boldsymbol{1}_n^T + \boldsymbol{1}_m \boldsymbol{f}_n^T + \boldsymbol{f}_m \boldsymbol{1}_n^T} \right] \overline{\boldsymbol{F}_n}, \tag{5}$$

where $\boldsymbol{x} = \text{vec}(\hat{\boldsymbol{X}})$, $\boldsymbol{f}_n \in \mathbb{R}^n$ is the vector with entries $[\boldsymbol{f}_n]_k = 2 - 2\cos(2\pi(k-1)/n)$, and the fraction denotes elementwise division.

We can express this computation using 2D FFT operations in `Julia` as

```
# We know Xhat should be real-valued, so we discard the imaginary part
Xhat = real(ifft(fft(Dm' * Bx + By * Dn) ./ P))
```

where $\boldsymbol{P} \triangleq \mu \boldsymbol{1}_m \boldsymbol{1}_n^T + \boldsymbol{1}_m \boldsymbol{f}_n^T + \boldsymbol{f}_m \boldsymbol{1}_n^T$.

(f) Write a function called `fftsurf` that computes the solution to (3) in closed-form using (5). Your solution should use only 2D FFTs and iFFTs and matrix algebra: no iterations required! Make sure to cast the surface into a real-valued matrix before returning it.

In `Julia`, your file should be named `fftsurf.jl` and should contain the following function:

```
"""
    X = fftsurf(Bx, By, mu)

Compute solution to the regularized photometric stereo problem
`\\min_x 0.5 \\| b - A x \\|^2 + \\mu \\| x \\|^2`
non-iteratively using 2D FFT operations.
Here, `x = X[:]` and A is the 2D circulant first-differences matrix

In:
- `Bx` : `m × n` matrix of x gradients
- `By` : `m × n` matrix of y gradients
- `mu > 0` : regularization parameter

Out:
`X` : `m × n` matrix containing the estimated surface
"""
function fftsurf(Bx, By, mu)
```

Email your solution as an attachment to `eecs551@autograder.eecs.umich.edu`.

Once your code is working, download the `PhotometricStereo_demo_complete` notebook from Canvas. This is an updated photometric stereo notebook with code to use your `fftsurf` implementation to reconstruct the cat. Verify that it produces the same surface for small values of μ as `lsqr` and your previous `lsngd` solver.

(g) Turn in plots of the reconstructed surfaces using `fftsurf` with $\mu = 10^{-9}$ and $\mu = 10^{-2}$.

What happens to the reconstructed surface as μ increases?

(h) Run the last section of the updated notebook to compare the runtimes of the `lsqr`, `lsngd`, and `fftsurf` algorithms. Turn in a copy of the plots that are generated. How much faster is the FFT-based solver?

Pr. 11.

(a) Show that $f(\boldsymbol{X}) = \operatorname{trace}(\boldsymbol{X})$ is a Lipschitz continuous function from $\mathbb{F}^{N \times N}$ into $\mathbb{R}$. Use any matrix norm that you like for $\mathbb{F}^{N \times N}$.

(b) Show that $g(\boldsymbol{X}) = \operatorname{Diag}(\boldsymbol{X})$ is a Lipschitz continuous function from $\mathbb{F}^{N \times N}$ into $\mathbb{F}^{N}$. Use any matrix norms that you like for $\mathbb{F}^{N \times N}$ and $\mathbb{F}^{N}$.

(c) Optional. Can you think of a way to generalize these two examples? Hint. Both examples are linear mappings of the elements of $\boldsymbol{X}$.
