EE263 homework 2 solutions

3.2 *Color perception.* Human color perception is based on the responses of three different types of color light receptors, called *cones.* The three types of cones have different spectral response characteristics and are called L, M, and, S because they respond mainly to long, medium, and short wavelengths, respectively. In this problem we will divide the visible spectrum into 20 bands, and model the cones' response as follows:

$$L_{\text{cone}} = \sum_{i=1}^{20} l_i p_i, \qquad M_{\text{cone}} = \sum_{i=1}^{20} m_i p_i, \qquad S_{\text{cone}} = \sum_{i=1}^{20} s_i p_i,$$

where p_i is the incident power in the ith wavelength band, and l_i, m_i and s_i are nonnegative constants that describe the spectral response of the different cones. The perceived color is a complex function of the three cone responses, *i.e.*, the vector $(L_{\text{cone}}, M_{\text{cone}}, S_{\text{cone}})$, with different cone response vectors perceived as different colors. (Actual color perception is a bit more complicated than this, but the basic idea is right.)

(a) *Metamers.* When are two light spectra, p and $\tilde{p}$, visually indistinguishable? (Visually identical lights with different spectral power compositions are called *metamers.*)

(b) *Visual color matching.* In a color matching problem, an observer is shown a test light and is asked to change the intensities of three primary lights until the sum of the primary lights looks like the test light. In other words, the observer is asked the find a spectrum of the form

$$p_{\text{match}} = a_1 u + a_2 v + a_3 w,$$

where u, v, w are the spectra of the primary lights, and a_i are the (nonnegative) intensities to be found, that is visually indistinguishable from a given test light spectrum p_{test}. Can this always be done? Discuss briefly. Don't worry about the requirement that a_i are nonnegative.

(c) *Visual matching with phosphors.* A computer monitor has three phosphors, R, G, and B. It is desired to adjust the phosphor intensities to create a color that looks like a reference test light. Find weights that achieve the match or explain why no such weights exist. The data for this problem is in `color_perception.m`. Running it will define and plot the vectors `wavelength`, `B_phosphor`, `G_phosphor`, `R_phosphor`, `L_coefficients`, `M_coefficients`, `S_coefficients`, and `test_light`.

(d) *Effects of illumination.* An object's surface can be characterized by its reflectance (*i.e.*, the fraction of light it reflects) for each band of wavelengths. If the object is illuminated with a light spectrum characterized by I_i, and the reflectance of the object is r_i (which is between 0 and 1), then the reflected light spectrum is given by $I_i r_i$, where $i = 1, \ldots, 20$ denotes the wavelength band. Now consider two objects illuminated (at different times) by two different light sources, say an incandescent bulb and sunlight. Sally argues that if the two objects look identical when illuminated by a tungsten bulb, they will look identical when illuminated by sunlight. Beth disagrees: she says that two objects can appear identical when illuminated by a tungsten bulb, but look different when lit by sunlight. Who is right? If Sally is right, explain why. If Beth is right give an example of two objects that appear identical under one light source and different under another. You can use the vectors `sunlight` and `tungsten` defined in `color_perception.m` as the light sources.

Solution.

(a) Let

$$A = \begin{bmatrix} l_1 & l_2 & l_3 & \cdots & l_{20} \\ m_1 & m_2 & m_3 & \cdots & m_{20} \\ s_1 & s_2 & s_3 & \cdots & s_{20} \end{bmatrix}.$$

Now suppose that $c = Ap$ is the cone response to the spectrum p and $\tilde{c} = A\tilde{p}$ is the cone response to spectrum $\tilde{p}$. If the spectra are indistinguishable, then $c = \tilde{c}$ and $Ap = A\tilde{p}$. Solving the last expression for zero gives $A(p - \tilde{p}) = 0$. In other words, p and $\tilde{p}$ are metamers if $(p - \tilde{p}) \in \mathcal{N}(A)$.

(b) In symbols, the problem asks if it is always possible to find nonnegative a_1, a_2, and a_3 such that

$$\begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} = Ap_{\text{test}} = A \begin{bmatrix} u & v & w \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}.$$

Let $P = \begin{bmatrix} u & v & w \end{bmatrix}$ and let $B = AP$. If B is invertible, then

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = B^{-1} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix}.$$

However, B is not necessarily invertible. For example, if $\mathbf{rank}(A) < 3$ or $\mathbf{rank}(P) < 3$ then B will be singular. Physically, A is full rank if the L, M, and S cone responses are linearly independent, which they are. The matrix P is full rank if and only if the spectra of the primary lights are independent. Even if both A and P are full rank, B could still be singular. Primary lights that generate an invertible B are called *visually independent.* If B is invertible, a_1, a_2, and a_3 exist that satisfy

$$Ap_{\text{test}} = A \begin{bmatrix} u & v & w \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}.$$

but one or more of the a_i may be negative in which case in the experimental setup described, no match would be possible. However, in a more complicated experimental setup that allows the primary lights to be combined either with each other or with p_{test}, a match is always possible if B is invertible. In this case, if $a_i < 0$, the ith light should be mixed with p_{test} instead of the other primary lights. For example, suppose $a_1 < 0$, $a_2, a_3 \geq 0$ and $b_1 = -a_1$, then

$$A(b_1 u + p_{\text{test}}) = A(a_2 v + a_3 w),$$

and each spectrum has a nonnegative weight.

(c) Weights can be found as described above. The R, G, and B phosphors should be weighted by 0.4226, 0.0987, and 0.5286 respectively. The following Matlab code illustrates the steps.

```
close all; clear all;
color_perception;
A = [L_coefficients; M_coefficients; S_coefficients]; B =
A*[R_phosphor’ G_phosphor’ B_phosphor’];
weights = inv(B)*A*test_light;
```

(d) Beth is right. Let r and $\tilde{r}$ be the reflectances of two objects and let p and $\tilde{p}$ be two spectra. Let A be defined as before. Then, the objects will look identical under p if

$$A\underbrace{\begin{bmatrix} r_1 & 0 & \cdots & 0 \\ 0 & r_2 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & r_{20} \end{bmatrix}}_{R} p = A\underbrace{\begin{bmatrix} \tilde{r}_1 & 0 & \cdots & 0 \\ 0 & \tilde{r}_2 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \tilde{r}_{20} \end{bmatrix}}_{\tilde{R}} p.$$

This is equivalent to saying $(R - \tilde{R})p \in \mathcal{N}(A)$. The objects will look different under $\tilde{p}$ if, additionally, $AR\tilde{p} \neq A\tilde{R}\tilde{p}$ which means that $(R - \tilde{R})\tilde{p} \notin \mathcal{N}(A)$. The following matlab code shows how to find reflectances r_1 and r_2 for two objects such that the objects will have the same color under tungsten light and will have different colors under sunlight.

```
close all; clear all;
color_perception;
A = [L_coefficients; M_coefficients; S_coefficients]; N = null(A);
n = N(:,1);
n= n*10;
for i = 1:20
n(i) = n(i)/tungsten(i);
end
r1 = [0; .2; .3; .7; .7; .8; .8; .2; .9; .8; .2; .8; .9; .2; .8;
.3; .8; .7; .2; .4];
r2 = r1-n;
for i = 1:20
t1(i) = r1(i)*tungsten(i);
t2(i) = r2(i)*tungsten(i);
end color1_tungsten = A*t1'; color2_tungsten = A*t2';
for i = 1:20
s1(i) = r1(i)*sunlight(i);
s2(i) = r2(i)*sunlight(i);
end color1_sun = A*s1'; color2_sun = A*s2';
253.5187
```

3.3 *Halfspace.* Suppose $a, b \in \mathbf{R}^n$ are two given points. Show that the set of points in $\mathbf{R}^n$ that are closer to a than b is a halfspace, *i.e.*:

$$\{x \mid \|x-a\| \leq \|x-b\| \ \} = \{ \ x \mid c^T x \leq d\}$$

for appropriate $c \in \mathbf{R}^n$ and $d \in \mathbf{R}$. Give c and d explicitly, and draw a picture showing a, b, c, and the halfspace.

Solution.
It is easy to see geometrically what is going on: the hyperplane that goes right between a and b splits $\mathbf{R}^n$ into two parts; the points closer to a (than b) and the points closer to b (than a). More precisely, the hyperplane is normal to the line through a and b, and intersects that line at the midpoint between a and b. Now that we have the idea, let's try to derive it algebraically. Let x belong to the set of points in $\mathbf{R}^n$ that are closer to a than b. Therefore $\|x-a\| < \|x-b\|$ or $\|x-a\|^2 < \|x-b\|^2$ so

$$(x-a)^T(x-a) < (x-b)^T(x-b).$$

Expanding the inner products gives

$$x^T x - x^T a - a^T x + a^T a < x^T x - x^T b - b^T x + b^T b$$

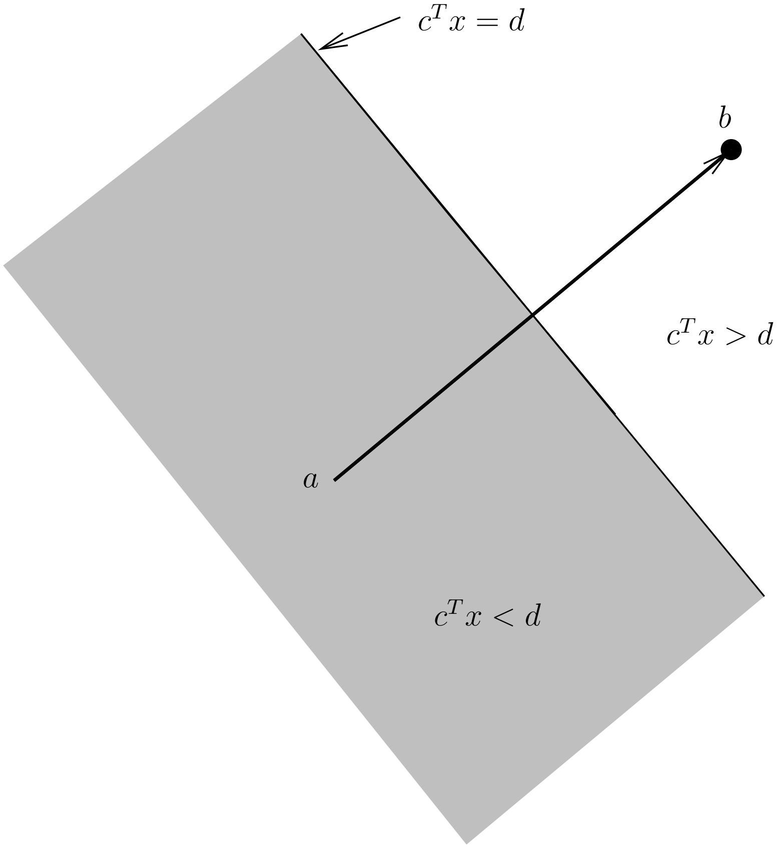
or

$$-2a^T x + a^T a < -2b^T x + b^T b$$

and finally

$$(b-a)^T x < \frac{1}{2}(b^T b - a^T a). \qquad (1)$$

Thus (1) is in the form $c^T x < d$ with $c = b - a$ and $d = \frac{1}{2}(b^T b - a^T a)$ and therefore we have shown that the set of points in $\mathbf{R}^n$ that are closer to a than b is a halfspace. Note that the hyperplane $c^T x = d$ is perpendicular to $c = b - a$.



3.10 *Proof of Cauchy-Schwarz inequality.* You will prove the Cauchy-Schwarz inequality.

(a) Suppose $a \geq 0$, $c \geq 0$, and for all $\lambda \in \mathbf{R}$, $a + 2b\lambda + c\lambda^2 \geq 0$. Show that $|b| \leq \sqrt{ac}$.

(b) Given v, $w \in \mathbf{R}^n$ explain why $(v + \lambda w)^T(v + \lambda w) \geq 0$ for all $\lambda \in \mathbf{R}$.

(c) Apply (a) to the quadratic resulting when the expression in (b) is expanded, to get the Cauchy-Schwarz inequality:

$$|v^Tw| \leq \sqrt{v^Tv}\sqrt{w^Tw}.$$

(d) When does equality hold?

Solution.

(a) If the equation $a + 2b\lambda + c\lambda^2 = 0$ has no real roots (with odd degree) for λ then it never changes sign for $\lambda \in \mathbf{R}$. Since a and c are positive, the value of $a + 2b\lambda + c\lambda^2$ is non-negative at zero and infinity respectively, so the necessary and sufficient condition for $a + 2b\lambda + c\lambda^2$ to be non-negative is the condition for which $a + 2b\lambda + c\lambda^2 = 0$ has no (simple) real roots for λ. Therefore we should have

$$4b^2 - 4ac \leq 0$$

and since $a, c \geq 0$ this gives $|b| \leq \sqrt{ac}$.

(b) Clearly $(v+\lambda w)^T(v+\lambda w) = \|v+\lambda w\|^2$, and the norm of any vector (here $v+\lambda w$) is non-negative. Therefore $(v+\lambda w)^T(v+\lambda w) \geq 0$ and equality holds when $v+\lambda w = 0$ or $v = -\lambda w$ (*i.e.*, v is a scalar multiple of w.)

(c) From the previous part we know that $(v+\lambda w)^T(v+\lambda w) \geq 0$ and since

$$(v+\lambda w)^T(v+\lambda w) = v^Tv + 2(v^Tw)\lambda + (w^Tw)\lambda^2,$$

applying the result of problem (3.10a) with $a = v^Tv \geq 0$, $b = v^Tw$ and $c = w^Tw \geq 0$ gives

$$|v^Tw| \leq \sqrt{v^Tv}\sqrt{w^Tw}.$$

(d) According to part (b), equality holds if and only if v is a scalar multiple of w. If v is a positive scalar multiple of w, then $v^Tw > 0$ so $|v^Tw| = v^Tw$ and we have $v^Tw = \sqrt{v^Tv}\sqrt{w^Tw}$. If v is a negative scalar multiple of w, then $v^Tw < 0$ and $|v^Tw| = -v^Tw$, so $v^Tw = -\sqrt{v^Tv}\sqrt{w^Tw}$.

3.11 *Vector spaces over the Boolean field.* In this course the *scalar field*, *i.e.*, the components of vectors, will usually be the real numbers, and sometimes the complex numbers. It is also possible to consider vector spaces over other fields, for example $\mathbf{Z}_2$, which consists of the two numbers 0 and 1, with Boolean addition and multiplication (*i.e.*, $1+1=0$). Unlike $\mathbf{R}$ or $\mathbf{C}$, the field $\mathbf{Z}_2$ is finite, indeed, has only two elements. A vector in $\mathbf{Z}_2^n$ is called a *Boolean vector*. Much of the linear algebra for $\mathbf{R}^n$ and $\mathbf{C}^n$ carries over to $\mathbf{Z}_2^n$. For example, we define a function $f : \mathbf{Z}_2^n \to \mathbf{Z}_2^m$ to be linear (over $\mathbf{Z}_2$) if $f(x+y) = f(x) + f(y)$ and $f(\alpha x) = \alpha f(x)$ for every $x,\ y \in \mathbf{Z}_2^n$ and $\alpha \in \mathbf{Z}_2$. It is easy to show that every linear function can be expressed as matrix multiplication, *i.e.*, $f(x) = Ax$, where $A \in \mathbf{Z}_2^{m\times n}$ is a Boolean matrix, and all the operations in the matrix multiplication are Boolean, *i.e.*, in $\mathbf{Z}_2$. Concepts like nullspace, range, independence and rank are all defined in the obvious way for vector spaces over $\mathbf{Z}_2$. Although we won't consider them in this course, there are many important applications of vector spaces and linear dynamical systems over $\mathbf{Z}_2$. In this problem you will explore one simple example: block codes. *Linear block codes.* Suppose $x \in \mathbf{Z}_2^n$ is a Boolean vector we wish to transmit over an unreliable channel. In a linear block code, the vector $y = Gx$ is formed, where $G \in \mathbf{Z}_2^{m\times n}$ is the *coding matrix*, and $m > n$. Note that the vector y is 'redundant'; roughly speaking we have *coded* an n-bit vector as a (larger) m-bit vector. This is called an (n,m) code. The coded vector y is transmitted over the channel; the received signal $\hat{y}$ is given by

$$\hat{y} = y + v,$$

where v is a noise vector (which usually is zero). This means that when $v_i = 0$, the ith bit is transmitted correctly; when $v_i = 1$, the ith bit is changed during transmission. In a *linear decoder*, the received signal is multiplied by another matrix: $\hat{x} = H\hat{y}$, where $H \in \mathbf{Z}_2^{n\times m}$. One reasonable requirement is that if the transmission is perfect, *i.e.*, $v = 0$, then the decoding is perfect, *i.e.*, $\hat{x} = x$. This holds if and only if H is a left inverse of G, *i.e.*, $HG = I_n$, which we assume to be the case.

(a) What is the practical significance of $\mathcal{R}(G)$?

(b) What is the practical significance of $\mathcal{N}(H)$?

(c) A one-bit *error correcting code* has the property that for any noise v with one component equal to one, we still have $\hat{x} = x$. Consider $n = 3$. Either design a one-bit error correcting linear block code with the smallest possible m, or explain why it cannot be done. (By design we mean, give G and H explicitly and verify that they have the required properties.)

Remark: linear decoders are never used in practice; there are far better nonlinear ones.

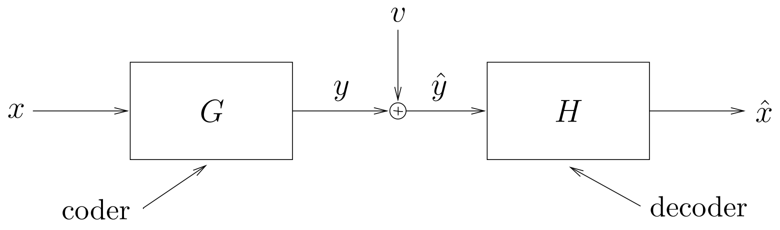
Solution.

(a) $\mathcal{R}(G)$ is the set of all valid block codes that are transmitted over the channel. If $\hat{y} \notin \mathcal{R}(G)$ then $v \neq 0$ but the converse is not necessarily true.

(b) Note that

$$\hat{x} = H\hat{y} = H(Gx + v) = x + Hv \qquad (HG = I)$$

and therefore $\hat{x} = x$ if and only if $Hv = 0$ or $v \in \mathcal{N}(H)$. So $\mathcal{N}(H)$ is the set of all channel noise vectors for which x can be reconstructed perfectly from the noisy channel output $\hat{y}$. In other words, any noise $v \in \mathcal{N}(H)$ is "ok" in the sense that the linear decoder can still detect x correctly from the noisy observation $\hat{y}$.



(c) The problem wants us to find $G \in \mathbf{Z}_2^{m\times 3}$ and $H \in \mathbf{Z}_2^{3\times m}$ such that

- $HG = I_3$ so that x can be recovered from noiseless channel outputs (*i.e.*, $v = 0$.)
- $Hv = 0$ for any v with *one* component equal to one so that x can be recovered from the channel output if an error has occured in only *one* of the bits.

The second property is equivalent to $He_i = 0$ where e_i is the ith unit vector in $\mathbf{Z}_2^m$ for $i = 1, \dots, m$. However, He_i is the ith column of H and therefore $He_i = 0$ for $i = 1, \dots, m$ implies that *all* columns of H are zero, or simply $H = 0$. But if $H = 0$ then we can never have $HG = I$ for any G. Therefore, such a design is not possible. (Note that if we remove the constraint for the decoder to be linear, a simple one-bit error correcting block code can be designed. For example, an obvious one is to repeat each bit three times (a linear operation), *i.e.*,

$$G = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix},$$

and then for the decoder we can take a majority vote (a nonlinear operation) on the repeated bits. In this case, 101 would be coded as $111'000'111$ and if the received bits are $110'010'111$, then the output of the decoder (found by a majority vote) is 101. This example demonstrates that nonlinear decoders are far better than linear ones.)

3.16 *Identifying a point on the unit sphere from spherical distances.* In this problem we consider the *unit sphere* in $\mathbf{R}^n$, which is defined as the set of vectors with norm one: $S^n = \{x \in \mathbf{R}^n \mid \|x\| = 1\}$. We define the *spherical distance* between two vectors on the unit sphere as the distance between them, measured along the sphere, *i.e.*, as the angle between the vectors, measured in radians: If $x,\ y \in S^n$, the spherical distance between them is

$$\mathrm{sphdist}(x, y) = \angle(x, y),$$

where we take the angle as lying between 0 and π. (Thus, the maximum distance between two points in S^n is π, which occurs only when the two points x, y are *antipodal*, which means $x = -y$.) Now suppose $p_1, \ldots, p_k \in S^n$ are the (known) positions of some beacons on the unit sphere, and let $x \in S^n$ be an unknown point on the unit sphere. We have exact measurements of the (spherical) distances between each beacon and the unknown point x, *i.e.*, we are given the numbers

$$\rho_i = \mathrm{sphdist}(x, p_i), \quad i = 1, \ldots, k.$$

We would like to determine, without any ambiguity, the exact position of x, based on this information. Find the conditions on $p_1, \ldots, p_k$ under which we can unambiguously determine x, for any $x \in S^n$, given the distances ρ_i. You can give your solution algebraically, using any of the concepts used in class (*e.g.*, nullspace, range, rank), or you can give a geometric condition (involving the vectors p_i). You must justify your answer.

Solution.
From

$$\rho_i = \arccos\left(\frac{p_i^T x}{\|p_i\|\|x\|}\right) = \arccos(p_i^T x),$$

we find that

$$p_i^T x = \cos \rho_i, \quad i = 1, \ldots, k.$$

Rewriting these equations in matrix form yields $Ax = b$, where $A \in \mathbf{R}^{k \times n}$ is the matrix with rows $p_1^T, \ldots, p_k^T$, and b is the vector with entries $\cos \rho_1, \ldots, \cos \rho_k$. Now if the matrix A has rank n, we can unambiguously find x given ρ. So a condition under which we can always recover x from the spherical distances is that A has rank n, which means nothing more than the vectors $p_1, \ldots, p_k$ span $\mathbf{R}^n$. In fact, that's exactly the condition. If it doesn't hold, *i.e.*, if the vectors $p_1, \ldots, p_k$ don't span $\mathbf{R}^n$, then there is a nonzero vector q such that

$$p_i^T q = 0, \quad i = 1, \ldots, k.$$

Now choose $x = q/\|q\|$, so x belongs to S^n. Then x has a spherical distance of $\pi/2$ to all the vectors $p_1, \ldots, p_k$, but the same is true for the point $-x$, which also belongs to S^n. It follows that we cannot unambiguously determine x (or $-x$) from the distances; x and $-x$ are indistinguishable. We can also give a very nice geometrical condition. The condition that $p_1, \ldots, p_k$ span $\mathbf{R}^n$, is the same as the condition that there exists no nonzero x with $p_i^T x = 0$. This can be interpreted geometrically as stating that the points $p_1, \ldots, p_k$ do not lie in a common plane (with normal vector x) passing through zero. We can restate this as saying that the points $p_1, \ldots, p_k$ do not lie on a common great circle.

3.17 *Some true/false questions.* Determine if the following statements are true or false. No justification or discussion is needed for your answers. What we mean by "true" is that the statement is true for all values of the matrices and vectors given. You can't assume anything about the dimensions of the matrices (unless it's explicitly stated), but you can assume that the dimensions are such that all expressions make sense. For example, the statement "$A + B = B + A$" is true, because no matter what the dimensions of A and B (which must, however, be the same), and no matter what values A and B have, the statement holds. As another example, the statement $A^2 = A$ is false, because there are (square) matrices for which this doesn't hold. (There are also matrices for which it does hold, *e.g.*, an identity matrix. But that doesn't make the statement true.)

a. If all coefficients (*i.e.*, entries) of the matrix A are positive, then A is full rank.

Solution.

False. The matrix $\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$ has all entries positive and is singular, hence not full rank.

b. If A and B are onto, then $A + B$ must be onto.

Solution.

False. The 1×1 matrix $A = 1$ is full rank, and so is the matrix $B = -1$. But $A + B = 0$ (the 1×1 zero), which is not onto.

c. If A and B are onto, then so is the matrix $\begin{bmatrix} A & C \\ 0 & B \end{bmatrix}$.

Solution.
True. To show this matrix is onto, we need to show that we can solve the equations

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} A & C \\ 0 & B \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

for any y_1 and y_2. (These are all vectors.) The bottom block row is $y_2 = Bx_2$. Using the fact that B is onto, we can find at least one x_2 such that $y_2 = Bx_2$. The top block row is

$$y_1 = Ax_1 + Cx_2,$$

which we can rewrite as

$$Ax_1 = y_1 - Cx_2.$$

Using the fact that A is onto, we can find at least one x_1 that satisfies this equation. Now we're done.

d. If A and B are onto, then so is the matrix $\begin{bmatrix} A \\ B \end{bmatrix}$.

Solution.
False. Let A and B both be the 1×1 matrix 1. These are each onto, but $\begin{bmatrix} A \\ B \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ is not.

e. If the matrix $\begin{bmatrix} A \\ B \end{bmatrix}$ is onto, then so are the matrices A and B.

Solution.
True. To say that $\begin{bmatrix} A \\ B \end{bmatrix}$ is onto means that for any vector y, we can find at least one x that satisfies

$$y = \begin{bmatrix} A \\ B \end{bmatrix} x.$$

Let's use this to show that A and B are both onto. First let's consider the equation $z = Au$. We can solve this by finding an x that satisfies

$$\begin{bmatrix} z \\ 0 \end{bmatrix} = \begin{bmatrix} A \\ B \end{bmatrix} x.$$

In a similar way can solve the equation $w = Bv$ for any vector w.

f. If A is full rank and skinny, then so is the matrix $\begin{bmatrix} A \\ B \end{bmatrix}$.

Solution.
True. Since the matrix A is skinny and full rank, its has zero nullspace: whenever

we have $Ax = 0$, we can conclude $x = 0$. The matrix $\begin{bmatrix} A \\ B \end{bmatrix}$ is also skinny, so to show it is full rank we must show that it, too, has zero nullspace. To do this suppose that

$$\begin{bmatrix} A \\ B \end{bmatrix} x = 0.$$

This means that $Ax = 0$ and $Bx = 0$. From the first, we conclude that $x = 0$. This shows that $\begin{bmatrix} A \\ B \end{bmatrix}$ is full rank.

Solutions to additional exercises

1. *Temperatures in a multi-core processor.* We are concerned with the temperature of a processor at two critical locations. These temperatures, denoted $T = (T_1, T_2)$ (in degrees C), are affine functions of the power dissipated by three processor cores, denoted $P = (P_1, P_2, P_3)$ (in W). We make 4 measurements. In the first, all cores are idling, and dissipate 10W. In the next three measurements, one of the processors is set to full power, 100W, and the other two are idling. In each experiment we measure and note the temperatures at the two critical locations.

P_1	P_2	P_3	T_1	T_2
10W	10W	10W	27°	29°
100W	10W	10W	45°	37°
10W	100W	10W	41°	49°
10W	10W	100W	35°	55°

Suppose we operate all cores at the same power, p. How large can we make p, without T_1 or T_2 exceeding 70°?

You must fully explain your reasoning and method, in addition to providing the numerical solution.

Solution. The temperature vector T is an affine function of the power vector P, *i.e.*, we have $T = AP + b$ for some matrix $A \in \mathbf{R}^{2\times 3}$ and some vector $b \in \mathbf{R}^2$. Once we find A and b, we can predict the temperature T for *any* value of P.

The first approach is to (somewhat laboriously) write equations describing the measurements in terms of the elements of A. Let a_{ij} denote the (i, j) entry of A. We can write out the relations $T = AP + b$ for the 4 experiments listed above as the set of 8 equations

$$\begin{aligned} 10a_{11} + 10a_{12} + 10a_{13} + b_1 &= 27, \\ 10a_{21} + 10a_{22} + 10a_{23} + b_2 &= 29, \\ 100a_{11} + 10a_{12} + 10a_{13} + b_1 &= 45, \end{aligned}$$

$$
\begin{array}{rcl}
100a_{21} + 10a_{22} + 10a_{23} + b_2 & = & 37, \\
10a_{11} + 100a_{12} + 10a_{13} + b_1 & = & 41, \\
10a_{21} + 100a_{22} + 10a_{23} + b_2 & = & 49, \\
10a_{11} + 10a_{12} + 100a_{13} + b_1 & = & 35, \\
10a_{21} + 10a_{22} + 100a_{23} + b_2 & = & 55.
\end{array}
$$

Next, we define a vector of unknowns, $x = (a_{11}, a_{12}, a_{13}, a_{21}, a_{22}, a_{23}, b_1, b_2) \in \mathbf{R}^8$. We rewrite the 8 equations above as $Cx = d$, where

$$
C = \begin{bmatrix}
10 & 10 & 10 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 10 & 10 & 10 & 0 & 1 \\
100 & 10 & 10 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 100 & 10 & 10 & 0 & 1 \\
10 & 100 & 10 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 10 & 100 & 10 & 0 & 1 \\
10 & 10 & 100 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 10 & 10 & 100 & 0 & 1
\end{bmatrix}, \qquad
d = \begin{bmatrix} 27 \\ 29 \\ 45 \\ 37 \\ 41 \\ 49 \\ 35 \\ 55 \end{bmatrix}.
$$

We solve for x as $x = C^{-1}d$. (It turns out that C is invertible.) Putting the entries of x into the appropriate places in A and b, we have

$$
A = \begin{bmatrix} 0.200 & 0.156 & 0.089 \\ 0.089 & 0.222 & 0.289 \end{bmatrix}, \qquad b = \begin{bmatrix} 22.6 \\ 23.0 \end{bmatrix}.
$$

At this point we can predict T for any P (assuming we trust the affine model).

Substituting $P = (p, p, p)$ into $T = AP + b$, we get

$$
T_1 = 0.444p + 22.6, \qquad T_2 = 0.600p + 23.0.
$$

Both of these temperatures are increasing in p (it would be quite surprising if this were not the case). The value of p for which $T_1 = 70$ is $p = (70 - 22.6)/0.444 = 106.8$W. The value of p for which $T_2 = 70$ is $p = (70 - 23)/0.6 = 78.3$W. Thus, the maximum value of p for which both temperatures do not exceed 70° is $p = 78.3$W.

Alternative solution. Another way of solving this problem is to directly exploit the fact that T is an affine function of P. This means that if we form any linear combination of the power vectors used in the experiment, *with the coefficients summing to one*, the temperature vector will also be the same linear combination of the temperatures.

By averaging the last three experiments we find if the powers are $P = (40, 40, 40)$, then the temperature vector is $T = (40.33, 47.00)$. (Note that this is really a prediction, based on the observed experimental data and the affineness assumption; it's not a new experiment!)

Now we form a new power vector of the form

$$P = (1-\theta)(10,10,10) + \theta(40,40,40) = (10+30\theta, 10+30\theta, 10+30\theta),$$

where $\theta \in \mathbf{R}$. The coefficients $1-\theta$ and θ sum to one, so since T is affine, we find that the corresponding temperature vector is

$$T = (1-\theta)(27,29) + \theta(40.33, 47.00) = (27 + 13.33\theta, 29 + 18\theta),$$

just as above. The first coefficient hits 70 at $\theta = 3.22$; the second coefficient hits 70 at $\theta = 2.23$. Thus, θ can be as large as $\theta = 2.27$. This corresponds to the powers $P = (78.3, 78.3, 78.3)$.

2. *Relative deviation between vectors.* Suppose a and b are nonzero vectors of the same size. The relative deviation of b from a is defined as the distance between a and b, divided by the norm of a,

$$\eta_{ab} = \frac{\|a-b\|}{\|a\|}.$$

This is often expressed as a percentage. The relative deviation is not a symmetric function of a and b; in general, $\eta_{ab} \neq \eta_{ba}$.

Suppose $\eta_{ab} = 0.1$ (*i.e.*, 10%). How big and how small can be η_{ba} be? How big and how small can $\angle(a,b)$ be? Explain your reasoning. For bounding $\angle(a,b)$, you can just draw some pictures; you don't have to give a formal argument.

Solution. We'll work out a more general case. We have

$$\|a-b\| = \eta_{ab}\|a\|.$$

We need to get upper and lower bounds on $\|b\|$. We can use the triangle inequality to get an upper bound:

$$\begin{array}{rcl} \|b\| & = & \|a + (-(a-b))\| \\ & \leq & \|a\| + \|a-b\| \\ & = & (1+\eta_{ab})\|a\|. \end{array}$$

This inequality is tight, if a and $a-b$ are anti-aligned, which is the same as a and b being aligned. Now we can say that

$$\eta_{ba} = \frac{\|a-b\|}{\|b\|} \geq \frac{\eta_{ab}\|a\|}{(1+\eta_{ab})\|a\|} = \frac{\eta_{ab}}{(1+\eta_{ab})}.$$

This is a general bound, and it is tight when a and b are aligned. For $\eta_{ab} = 0.1$, we find that $\eta_{ba} \geq 0.1/1.1 = 0.0909$.

Now let's get a lower bound on $\|b\|$, again using the triangle inequality:

$$\begin{aligned} \|a\| &= \|b + (a-b)\| \\ &\leq \|b\| + \|a-b\| \\ &= \|b\| + \eta_{ab}\|a\|. \end{aligned}$$

This inequality is tight if a and $a-b$ are aligned, which is the same as a and b being anti-aligned. Subtracting, we get

$$(1-\eta_{ab})\|a\| \leq \|b\|.$$

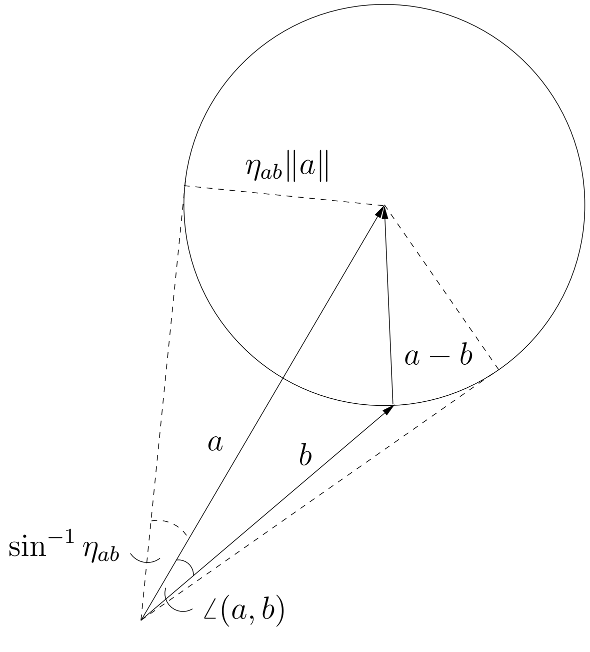
Assuming that $\eta_{ab} < 1$ (which is the case for $\eta_{ab} = 0.1$), we then have

$$\eta_{ba} = \frac{\|a-b\|}{\|b\|} \leq \frac{\eta_{ab}\|a\|}{(1-\eta_{ab})\|a\|} = \frac{\eta_{ab}}{(1-\eta_{ab})}.$$

This is a general bound, tight when a and b are anti-aligned. For $\eta_{ab} = 0.1$, we find that $\eta_{ba} \geq 0.1/0.9 = 0.1111$.

In summary, when $\eta_{ab} = 0.1$, η_{ba} can range between 0.0909 and 0.1111. The lower limit occurs when a and b are aligned; the upper limit occurs when a and b are aligned.

Now let's look at the angle. We first give a geomtric argument. Let's look at the plane spanned by a and b. Then vector b must be in a ball of radius $\eta_{ab}\|a\|$, centered in a, as shown below. Assuming that $\eta_{ab} < 1$ (which is the case here), the ball does not include the origin.



Now, we look on how small and how large the angle between a and b can be, as b varies over the ball. When a and b are aligned, $\angle(a,b) = 0$. Now let's see how large the angle can be. The largest angle is obtained when b and $a-b$ are orthogonal; in this case $(0,a,b)$ are the vertices of a right triangle. In this case we have $\angle(a,b) = \arcsin \eta_{ab}$. For η_{ab}, we find that $\angle(a,b) = 0.1002$. Therefore $\angle(a,b)$ can take values in the interval $[0, 0.1002]$.

3. *Single sensor failure detection and identification.* We have $y = Ax$, where $A \in \mathbf{R}^{m\times n}$ is known, and $x \in \mathbf{R}^n$ is to be found. Unfortunately, up to one sensor may have failed (but you don't know which one has failed, or even whether any has failed). You are given $\tilde{y}$ and not y, where $\tilde{y}$ is the same as y in all entries except, possibly, one (say, the kth entry). If all sensors are operating correctly, we have $y = \tilde{y}$. If the kth sensor fails, we have $\tilde{y}_i = y_i$ for all $i \neq k$.

   The file `one_bad_sensor.m`, available on the course web site, defines A and $\tilde{y}$ (as `A` and `ytilde`). Determine which sensor has failed (or if no sensors have failed). You must explain your method, and submit your code.

   For this exercise, you can use the Matlab code `rank([F g])==rank(F)` to check if $g \in \mathcal{R}(F)$. (We will see later a much better way to check if $g \in \mathcal{R}(F)$.)

   Solution. Let $y^{(i)}$ be the measurement vector y with the ith entry removed. Likewise, let $A^{(i)}$ be the measurement matrix with the ith row of A removed. This corresponds to the system without the ith sensor.

   If the ith sensor is faulty, we will almost surely have $y \notin \mathcal{R}(A)$ (unless the sensor failure happens to give the same response y_i as that predicted by A, which is highly unlikely). However, once we remove its faulty measurement, we will certainly have $y^{(i)} \in \mathcal{R}(A^{(i)})$.

   To test if a vector z is in $\mathcal{R}(C)$, we can use Matlab and compare `rank([C z]) == rank(C)`. If they are equal, $z \in \mathcal{R}(C)$. Otherwise `rank([C z]) == rank(C) + 1`. To find a faulty sensor, we remove one row of A at a time, and use the above test.

   The 11th sensor is faulty.