EECS 16A Designing Information Devices and Systems I
Spring 2020 Homework 12

This homework is due April 24, 2020, at 23:59.
Self-grades are due April 27, 2020, at 23:59.

Submission Format
Your homework submission should consist of **one** file.

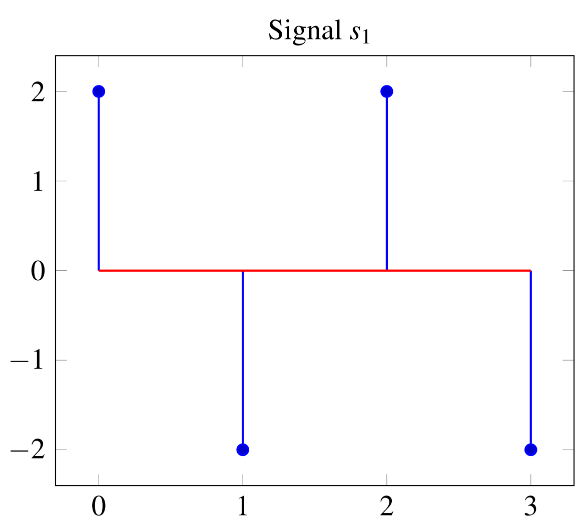
- `hw12.pdf`: A single PDF file that contains all of your answers (any handwritten answers should be scanned) as well as your IPython notebook saved as a PDF.

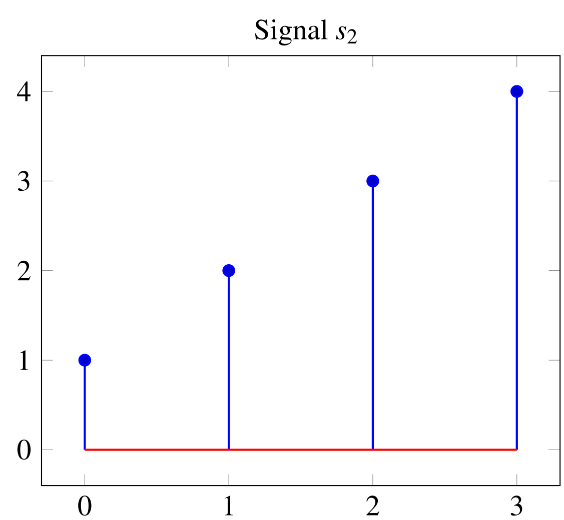
  If you do not attach a PDF "printout" of your IPython notebook, you will not receive credit for problems that involve coding. Make sure that your results and your plots are visible. Assign the IPython printout to the correct problem(s) on Gradescope.

Submit the file to the appropriate assignment on Gradescope.

Unless otherwise stated (as in problem 3), use the standard Euclidean inner product: for $\vec{x}, \vec{y} \in \mathbb{R}^n$, $\langle \vec{x}, \vec{y} \rangle = \vec{x}^T \vec{y}$.

1. Mechanical Linear Correlation





Assume that both of the above signals extend to $\pm\infty$, and are 0 everywhere outside of the region shown in the above graphs. First, we will demonstrate the procedure for linear correlation by computing the linear correlation between signal s_1 with itself (*i.e.* $\text{corr}_{\vec{s}_1}(\vec{s}_1)[k]$). This is referred to as the linear autocorrelation. This can be computed by evaluating the inner product between the signal and the shifted version of the signal (outlined in the below tables). Here, we compute this quantity for shifts between -3 and 3. For all shifts outside this range, the inner product is zero. Finally, we plot the non-zero values of the linear autocorrelation.

$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_1[n+3]$	2		-2		2		-2		0		0		0		0		0		0	
$\langle \vec{s}_1[n], \vec{s}_1[n+3] \rangle$	0	+	0	+	0	+	-4	+	0	+	0	+	0	+	0	+	0	+	0	= -4

$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_1[n+2]$	0		2		-2		2		-2		0		0		0		0		0	
$\langle \vec{s}_1[n], \vec{s}_1[n+2] \rangle$	0	+	0	+	0	+	4	+	4	+	0	+	0	+	0	+	0	+	0	= 8

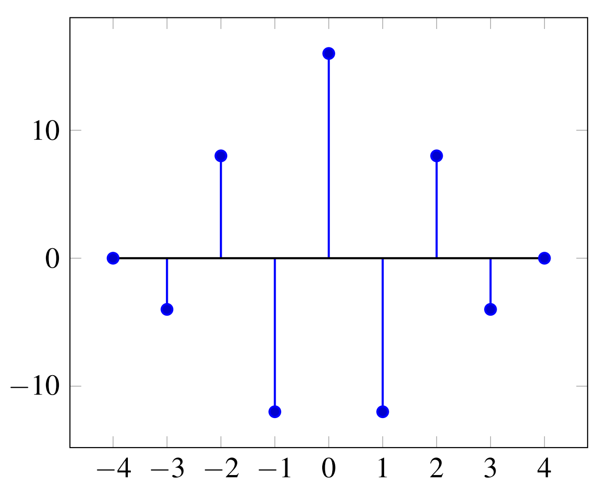
$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_1[n+1]$	0		0		2		-2		2		-2		0		0		0		0	
$\langle \vec{s}_1[n], \vec{s}_1[n+1] \rangle$	0	+	0	+	0	+	-4	+	-4	+	-4	+	0	+	0	+	0	+	0	= -12

$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_1[n+0]$	0		0		0		2		-2		2		-2		0		0		0	
$\langle \vec{s}_1[n], \vec{s}_1[n+0] \rangle$	0	+	0	+	0	+	4	+	4	+	4	+	4	+	0	+	0	+	0	= 16

$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_1[n-1]$	0		0		0		0		2		-2		2		-2		0		0	
$\langle \vec{s}_1[n], \vec{s}_1[n-1] \rangle$	0	+	0	+	0	+	0	+	-4	+	-4	+	-4	+	0	+	0	+	0	= -12

$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_1[n-2]$	0		0		0		0		0		2		-2		2		-2		0	
$\langle \vec{s}_1[n], \vec{s}_1[n-2] \rangle$	0	+	0	+	0	+	0	+	0	+	4	+	4	+	0	+	0	+	0	= 8

$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_1[n-3]$	0		0		0		0		0		0		2		-2		2		-2	
$\langle \vec{s}_1[n], \vec{s}_1[n-3] \rangle$	0	+	0	+	0	+	0	+	0	+	0	+	-4	+	0	+	0	+	0	= -4



(a) Using the procedure demonstrated above, compute $\text{corr}_{\vec{s}_1}(\vec{s}_2)[k]$, the linear cross-correlation of s_2 with s_1. Like the example, use tables like the one given below for $k = -3$ and plot the resulting correlation.

$\vec{s}_1[n]$	0	0	0	2	-2	2	-2	0	0	0
$\vec{s}_2[n+3]$	1	2	3	4	0	0	0	0	0	0
$\langle \vec{s}_1[n], \vec{s}_2[n+3] \rangle$										

Solution:

$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_2[n+3]$	1		2		3		4		0		0		0		0		0		0	
$\langle \vec{s}_1[n], \vec{s}_2[n+3] \rangle$	0	+	0	+	0	+	8	+	0	+	0	+	0	+	0	+	0	+	0	= 8

$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_2[n+2]$	0		1		2		3		4		0		0		0		0		0	
$\langle \vec{s}_1[n], \vec{s}_2[n+2] \rangle$	0	+	0	+	0	+	6	+	-8	+	0	+	0	+	0	+	0	+	0	= -2

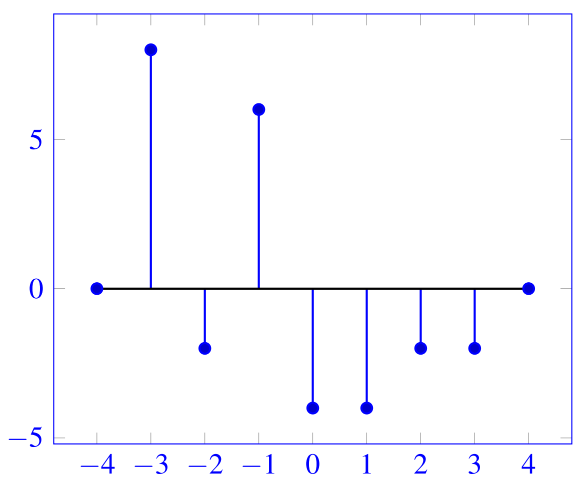
$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_2[n+1]$	0		0		1		2		3		4		0		0		0		0	
$\langle \vec{s}_1[n], \vec{s}_2[n+1] \rangle$	0	+	0	+	0	+	4	+	-6	+	8	+	0	+	0	+	0	+	0	= 6

$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_2[n+0]$	0		0		0		1		2		3		4		0		0		0	
$\langle \vec{s}_1[n], \vec{s}_2[n+0] \rangle$	0	+	0	+	0	+	2	+	-4	+	6	+	-8	+	0	+	0	+	0	= -4

$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_2[n-1]$	0		0		0		0		1		2		3		4		0		0	
$\langle \vec{s}_1[n], \vec{s}_2[n-1] \rangle$	0	+	0	+	0	+	0	+	-2	+	4	+	-6	+	0	+	0	+	0	= -4

$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_2[n-2]$	0		0		0		0		0		1		2		3		4		0	
$\langle \vec{s}_1[n], \vec{s}_2[n-2] \rangle$	0	+	0	+	0	+	0	+	0	+	2	+	-4	+	0	+	0	+	0	= -2

$\vec{s}_1[n]$	0		0		0		2		-2		2		-2		0		0		0	
$\vec{s}_2[n-2]$	0		0		0		0		0		0		1		2		3		4	
$\langle \vec{s}_1[n], \vec{s}_2[n-2] \rangle$	0	+	0	+	0	+	0	+	0	+	0	+	-2	+	0	+	0	+	0	= -2



(b) Will the linear cross-correlation of s_2 with s_1 ($\text{corr}_{\vec{s}_1}(\vec{s}_2)[k]$) be the same as the cross-correlation of s_1 with s_2 ($\text{corr}_{\vec{s}_2}(\vec{s}_1)[k]$)? You can use the iPython notebook to figure this out. How are they related to each other?

Solution: See sol12.ipynb. They do not have the same result, but they are related: one is the reverse of the other. If you were able to observe this, give yourself full points.

You were not explicitly required to show why, but a sketch of why this is the case follows. Let us compare $\text{corr}_{\vec{s}_2}(\vec{s}_1)[k]$ and $\text{corr}_{\vec{s}_1}(\vec{s}_2)[k]$.

By definition:

$$\text{corr}_{\vec{s}_2}(\vec{s}_1)[k] = \sum_{n=-\infty}^{\infty} \vec{s}_2[n]\vec{s}_1[n-k]$$

$$\text{corr}_{\vec{s}_1}(\vec{s}_2)[k] = \sum_{n=-\infty}^{\infty} \vec{s}_1[n]\vec{s}_2[n-k]$$

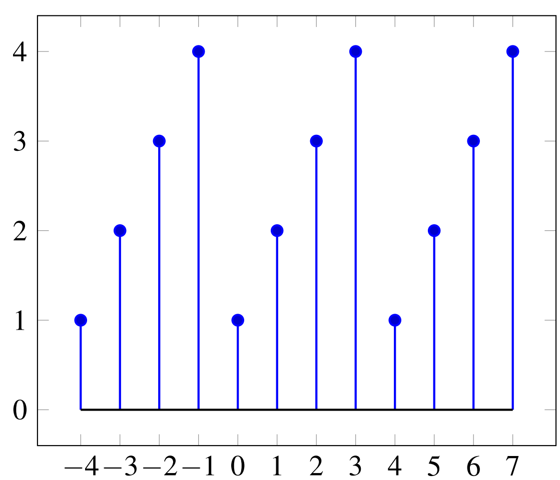
Using a substitution of index, $m = n-k$ we have:

$$\begin{aligned}\text{corr}_{\vec{s}_1}(\vec{s}_2)[k] &= \sum_{m=-\infty}^{\infty} \vec{s}_1[m+k]\vec{s}_2[m] \\ &= \sum_{m=-\infty}^{\infty} \vec{s}_2[m]\vec{s}_1[m-(-k)] \\ &= \text{corr}_{\vec{s}_2}(\vec{s}_1)[-k]\end{aligned}$$

So we can conclude that $\text{corr}_{\vec{s}_1}(\vec{s}_2)[k] = \text{corr}_{\vec{s}_2}(\vec{s}_1)[-k]$.

Now, we will review the procedure to perform linear cross-correlation between one signal that is periodic with a period of 4 and another that is finite length and extended with zeros as in the previous parts. As an example, we will compute the linear correlation $\text{corr}_{\vec{p}_2}(\vec{s}_1)[k]$ between the periodic signal $\vec{p}_2$ (with period 4), formed by repeating $\vec{s}_2$, and the finite length signal $\vec{s}_1$ extended with zeros. The result will be a periodic signal with period 4.

The periodic signal, $\vec{p}_2$, formed by repeating $\vec{s}_2$ is plotted below for indices -4 to 7. It is defined and non-zero for all indices from $-\infty$ to $+\infty$.



We compute one period of the result of the cross-correlation by starting at a shift of $k=-1$ and ending at a shift of $k=2$.

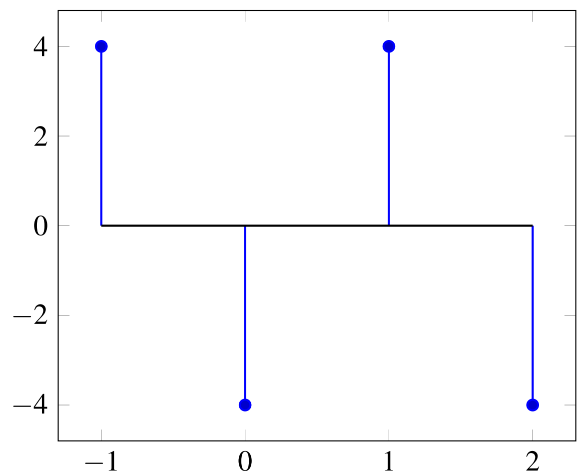
$\vec{p}_2[n]$	2	3	4	1	2	3	4	1	2	3	
$\vec{s}_1[n+1]$	0	0	2	-2	2	-2	0	0	0	0	
$\langle \vec{p}_2[n], \vec{s}_1[n+1] \rangle$	0 +	0 +	8 +	-2 +	4 +	-6 +	0 +	0 +	0 +	0	= 4

$\vec{p}_2[n]$	2	3	4	1	2	3	4	1	2	3	
$\vec{s}_1[n+0]$	0	0	0	2	-2	2	-2	0	0	0	
$\langle \vec{p}_2[n], \vec{s}_1[n+0] \rangle$	0 +	0 +	0 +	2 +	-4 +	6 +	-8 +	0 +	0 +	0	= -4

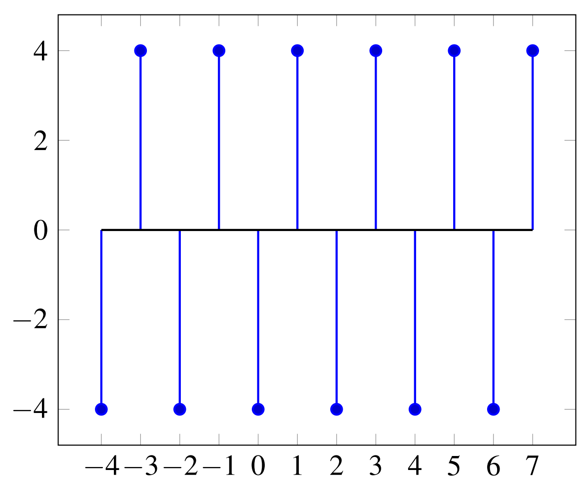
$\vec{p}_2[n]$	2	3	4	1	2	3	4	1	2	3	
$\vec{s}_1[n-1]$	0	0	0	0	2	-2	2	-2	0	0	
$\langle \vec{p}_2[n], \vec{s}_1[n-1] \rangle$	0 +	0 +	0 +	0 +	4 +	-6 +	8 +	-2 +	0 +	0	= 4

$\vec{p}_2[n]$	2	3	4	1	2	3	4	1	2	3	
$\vec{s}_1[n-2]$	0	0	0	0	0	2	-2	2	-2	0	
$\langle \vec{p}_2[n], \vec{s}_1[n-2] \rangle$	0 +	0 +	0 +	0 +	0 +	6 +	-8 +	2 +	-4 +	0	= -4

The computed single period of the resulting linear cross correlation is plotted below.



The resulting linear cross correlation for shifts from $k = -4$ to $k = 7$ is plotted below.



(c) Repeat the procedure described above to compute the correlation $\text{corr}_{\vec{p}_1}(\vec{s}_1)[k]$ between the periodic signal $\vec{p}_1$ (with period 4), formed by repeating s_1, and the finite-length signal s_1 extended with zeros. Like the example, evaluate tables like the one below for $k = -3$ for different shifts and plot a single period of the result.

$\vec{p}_1[n]$	-2	2	-2	2	-2	2	-2	2	-2	2
$\vec{s}_1[n+3]$	2	-2	2	-2	0	0	0	0	0	0
$\langle \vec{p}_1[n], \vec{s}_1[n+3] \rangle$										

Solution: We have computed below shifts from $k = -3$ to $k = 3$. However, so long as you have enough values for a single period, give yourself full credit.

$\vec{p}_1[n]$	-2	2	-2	2	-2	2	-2	2	-2	2	
$\vec{s}_1[n+3]$	2	-2	2	-2	0	0	0	0	0	0	
$\langle \vec{p}_1[n], \vec{s}_1[n+3] \rangle$	-4 +	-4 +	-4 +	-4 +	0 +	0 +	0 +	0 +	0 +	0	= -16

$\vec{p}_1[n]$	-2	2	-2	2	-2	2	-2	2	-2	2	
$\vec{s}_1[n+2]$	0	2	-2	2	-2	0	0	0	0	0	
$\langle \vec{p}_1[n], \vec{s}_1[n+2] \rangle$	0 +	4 +	4 +	4 +	4 +	0 +	0 +	0 +	0 +	0	= 16

$\vec{p}_1[n]$	-2	2	-2	2	-2	2	-2	2	-2	2	
$\vec{s}_1[n+1]$	0	0	2	-2	2	-2	0	0	0	0	
$\langle \vec{p}_1[n], \vec{s}_1[n+1] \rangle$	0 +	0 +	-4 +	-4 +	-4 +	-4 +	0 +	0 +	0 +	0	= -16

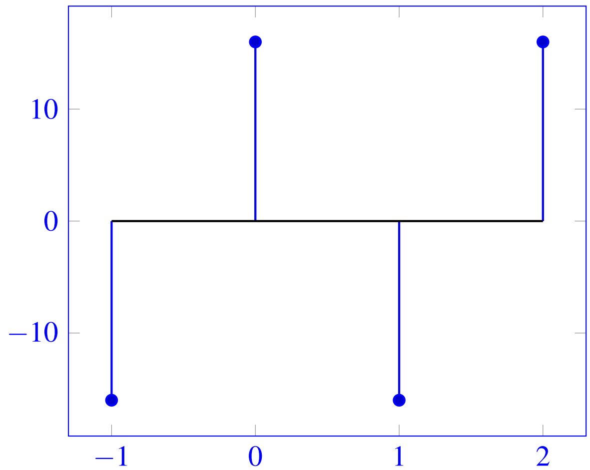
$\vec{p}_1[n]$	-2	2	-2	2	-2	2	-2	2	-2	2	
$\vec{s}_1[n+0]$	0	0	0	2	-2	2	-2	0	0	0	
$\langle \vec{p}_1[n], \vec{s}_1[n+0] \rangle$	0 +	0 +	0 +	4 +	4 +	4 +	4 +	0 +	0 +	0	= 16

$\vec{p}_1[n]$	-2	2	-2	2	-2	2	-2	2	-2	2	
$\vec{s}_1[n-1]$	0	0	0	0	2	-2	2	-2	0	0	
$\langle \vec{p}_1[n], \vec{s}_1[n-1] \rangle$	0 +	0 +	0 +	0 +	-4 +	-4 +	-4 +	-4 +	0 +	0	= -16

$\vec{p}_1[n]$	-2	2	-2	2	-2	2	-2	2	-2	2	
$\vec{s}_1[n-2]$	0	0	0	0	0	2	-2	2	-2	0	
$\langle \vec{p}_1[n], \vec{s}_1[n-2] \rangle$	0 +	0 +	0 +	0 +	0 +	4 +	4 +	4 +	4 +	0	= 16

$\vec{p}_1[n]$	-2	2	-2	2	-2	2	-2	2	-2	2	
$\vec{s}_1[n-3]$	0	0	0	0	0	0	2	-2	2	-2	
$\langle \vec{p}_1[n], \vec{s}_1[n-3] \rangle$	0 +	0 +	0 +	0 +	0 +	0 +	-4 +	-4 +	-4 +	-4	= -16

Like the example, the period was plotted from $k = -1$ to $k = 2$. Give yourself full credit if you plotted four consecutive values sufficient for a single period, i.e. your plot starts from a shift of $k = k_0$ and ends at $k = k_0 + 3$



2. Mechanical Trilateration

Trilateration is the problem of finding one's coordinates given distances from known location coordinates. For each of the following trilateration problems, you are given 3 positions and the corresponding distance from each position to your location. Find your location or possible locations. If a solution does not exist, state that it does not.

(a) $\vec{s}_1 = \begin{bmatrix} 4 \\ 5 \end{bmatrix}$, $d_1 = 5$, $\vec{s}_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$, $d_2 = 2$, $\vec{s}_3 = \begin{bmatrix} -11 \\ 6 \end{bmatrix}$, $d_3 = 13$.

Solution: First the problem will be done in abstract so that we can immediately write the linear system of equations for all three parts. However, if you solved directly using concrete values, give yourself full credit.

$$\begin{aligned} \|\vec{x} - \vec{s}_1\|^2 &= d_1^2 \\ \|\vec{x} - \vec{s}_2\|^2 &= d_2^2 \\ \|\vec{x} - \vec{s}_3\|^2 &= d_3^2 \end{aligned}$$

We can expand each left hand side out in terms of the definition of the norm:

$$\|\vec{x} - \vec{s}_i\|^2 = \langle \vec{x} - \vec{s}_i, \vec{x} - \vec{s}_i \rangle = (\vec{x} - \vec{s}_i)^T (\vec{x} - \vec{s}_i)$$

$$\begin{aligned} \vec{x}^T\vec{x} - 2\vec{x}^T\vec{s}_1 + \vec{s}_1^T\vec{s}_1 &= d_1^2 \\ \vec{x}^T\vec{x} - 2\vec{x}^T\vec{s}_2 + \vec{s}_2^T\vec{s}_2 &= d_2^2 \\ \vec{x}^T\vec{x} - 2\vec{x}^T\vec{s}_3 + \vec{s}_3^T\vec{s}_3 &= d_3^2 \end{aligned}$$

Finally, take one equation and subtract it from the other two to get a system of linear equations in $\vec{x}$:

$$\begin{aligned} 2\vec{x}^T\vec{s}_3 - 2\vec{x}^T\vec{s}_1 &= d_1^2 - d_3^2 + \vec{s}_3^T\vec{s}_3 - \vec{s}_1^T\vec{s}_1 \\ 2\vec{x}^T\vec{s}_3 - 2\vec{x}^T\vec{s}_2 &= d_2^2 - d_3^2 + \vec{s}_3^T\vec{s}_3 - \vec{s}_2^T\vec{s}_2 \end{aligned}$$

We can express as a matrix equation in $\vec{x}$:

$$\begin{bmatrix} 2(\vec{s}_3 - \vec{s}_1)^T \\ 2(\vec{s}_3 - \vec{s}_2)^T \end{bmatrix} \vec{x} = \begin{bmatrix} d_1^2 - d_3^2 + \|\vec{s}_3\|^2 - \|\vec{s}_1\|^2 \\ d_2^2 - d_3^2 + \|\vec{s}_3\|^2 - \|\vec{s}_2\|^2 \end{bmatrix}$$

We have that:

$$2(\vec{s}_3 - \vec{s}_1) = \begin{bmatrix} -30 \\ 2 \end{bmatrix}$$
$$2(\vec{s}_3 - \vec{s}_2) = \begin{bmatrix} -24 \\ 14 \end{bmatrix}$$
$$d_1^2 - d_3^2 + \|\vec{s}_3\|^2 - \|\vec{s}_1\|^2 = 25 - 169 + 157 - 41 = -28$$
$$d_2^2 - d_3^2 + \|\vec{s}_3\|^2 - \|\vec{s}_2\|^2 = 4 - 169 + 157 - 2 = -10$$

Which gives us the system $\begin{bmatrix} -30 & 2 \\ -24 & 14 \end{bmatrix} \vec{x} = \begin{bmatrix} -28 \\ -10 \end{bmatrix}$ with solution $\vec{x} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$.
A solution existing for this system of linear equations does not necessarily guarantee consistency of the system of nonlinear equations, but we can validate:

$$\left\| \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 4 \\ 5 \end{bmatrix} \right\|^2 = \left\| \begin{bmatrix} -3 \\ -4 \end{bmatrix} \right\|^2 = 25 = d_1^2$$
$$\left\| \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 1 \\ -1 \end{bmatrix} \right\|^2 = \left\| \begin{bmatrix} 0 \\ 2 \end{bmatrix} \right\|^2 = 4 = d_2^2$$
$$\left\| \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \begin{bmatrix} -11 \\ 6 \end{bmatrix} \right\|^2 = \left\| \begin{bmatrix} 12 \\ -5 \end{bmatrix} \right\|^2 = 169 = d_3^2$$

(b) $\vec{s}_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, $d_1 = 5\sqrt{2}$, $\vec{s}_2 = \begin{bmatrix} 10 \\ 0 \end{bmatrix}$, $d_2 = 5\sqrt{2}$, $\vec{s}_3 = \begin{bmatrix} 5 \\ 0 \end{bmatrix}$, $d_3 = 5$.

Solution: Using what was shown in part (a) we have that:

$$2(\vec{s}_3 - \vec{s}_1) = \begin{bmatrix} 10 \\ 0 \end{bmatrix}$$
$$2(\vec{s}_3 - \vec{s}_2) = \begin{bmatrix} -10 \\ 0 \end{bmatrix}$$
$$d_1^2 - d_3^2 + \|\vec{s}_3\|^2 - \|\vec{s}_1\|^2 = 50 - 25 + 25 - 0 = 50$$
$$d_2^2 - d_3^2 + \|\vec{s}_3\|^2 - \|\vec{s}_2\|^2 = 50 - 25 + 25 - 100 = -50$$

Which gives us the system $\begin{bmatrix} 10 & 0 \\ -10 & 0 \end{bmatrix} \vec{x} = \begin{bmatrix} 50 \\ -50 \end{bmatrix}$ with solution $\vec{x} = \begin{bmatrix} 5 \\ \alpha \end{bmatrix}$. However, not all values of α are valid, so we check with the third distance equation:

$$\left\| \begin{bmatrix} 5 \\ \alpha \end{bmatrix} - \begin{bmatrix} 5 \\ 0 \end{bmatrix} \right\|^2 = 5^2 \implies \alpha^2 = 25 \implies \alpha = \pm 5$$

The system of nonlinear equations is consistent with this solution. We do not have enough information to uniquely determine our location, but we know we are at either $\vec{x} = \begin{bmatrix} 5 \\ 5 \end{bmatrix}$ or $\vec{x} = \begin{bmatrix} 5 \\ -5 \end{bmatrix}$.

(c) $\vec{s}_1 = \begin{bmatrix} 3 \\ 4 \end{bmatrix}$, $d_1 = 5$, $\vec{s}_2 = \begin{bmatrix} 0 \\ -2 \end{bmatrix}$, $d_2 = 2$, $\vec{s}_3 = \begin{bmatrix} -12 \\ 5 \end{bmatrix}$, $d_3 = 12$.

Solution: Using again what was shown in part (a) we have that:

$$2(\vec{s}_3 - \vec{s}_1) = \begin{bmatrix} -30 \\ 2 \end{bmatrix}$$
$$2(\vec{s}_3 - \vec{s}_2) = \begin{bmatrix} -24 \\ 14 \end{bmatrix}$$
$$d_1^2 - d_3^2 + \|\vec{s}_3\|^2 - \|\vec{s}_1\|^2 = 25 - 144 + 169 - 25 = 25$$
$$d_2^2 - d_3^2 + \|\vec{s}_3\|^2 - \|\vec{s}_2\|^2 = 4 - 144 + 169 - 4 = 25$$

Which gives us the system $\begin{bmatrix} -30 & 2 \\ -24 & 14 \end{bmatrix} \vec{x} = \begin{bmatrix} 25 \\ 25 \end{bmatrix}$. While a solution, $\vec{x} = \begin{bmatrix} -\frac{75}{93} \\ \frac{75}{186} \end{bmatrix}$, for this system of linear equations exists, it will yield inconsistent distances when substituted back into the nonlinear equations. Therefore there is no solution.

3. Inner Products

For each of the following functions, show whether it defines an inner product on the given vector space. If not, give a counterexample.

(a) For $\mathbb{R}^2$:

$$\langle \vec{p}, \vec{q} \rangle = \vec{p}^T \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix} \vec{q}$$

Solution:

Yes, the function defines an inner product on $\mathbb{R}^2$. To show this, we will show that all of the three axioms apply. Let $\vec{p} = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}$ and $\vec{q} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$.

i. Symmetry:

$$\begin{aligned}
\langle \vec{p}, \vec{q} \rangle &= \begin{bmatrix} p_1 & p_2 \end{bmatrix} \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \\
&= 3p_1q_1 + p_1q_2 + p_2q_1 + 2p_2q_2 \\
&= \begin{bmatrix} q_1 & q_2 \end{bmatrix} \begin{bmatrix} 3p_1 + p_2 \\ p_1 + 2p_2 \end{bmatrix} \\
&= \begin{bmatrix} q_1 & q_2 \end{bmatrix} \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} \\
&= \langle \vec{q}, \vec{p} \rangle
\end{aligned}$$

ii. Linearity: Let $\vec{p}_1, \vec{p}_2 \in \mathbb{R}^2$.

$$\begin{aligned}
\langle \alpha \vec{p}_1 + \beta \vec{p}_2, q(x) \rangle &= (\alpha \vec{p}_1 + \beta \vec{p}_2)^T \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix} \vec{q} \\
&= \alpha \vec{p}_1^{\,T} \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix} \vec{q} + \beta \vec{p}_2^{\,T} \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix} \vec{q} \\
&= \alpha \langle \vec{p}_1, \vec{q} \rangle + \beta \langle \vec{p}_2, \vec{q} \rangle
\end{aligned}$$

iii. Positive-definiteness:

$$\langle \vec{p}, \vec{p} \rangle = 3p_1p_1 + p_1p_2 + p_2p_1 + 2p_2p_2 = 3p_1^2 + 2p_2^2 + 2p_1p_2 = 2p_1^2 + p_2^2 + (p_1 + p_2)^2$$

Since all of the components are non-negative, $\langle \vec{p}, \vec{p} \rangle \geq 0$.
Furthermore, the inner product will be 0 if and only if $p_1 = p_2 = 0 \implies \vec{p} = \vec{0}$.

(b) For $\mathbb{R}^2$:

$$\langle \vec{p}, \vec{q} \rangle = \vec{p}^T \begin{bmatrix} 3 & 1 \\ 1 & -2 \end{bmatrix} \vec{q}$$

Solution:
No, the function does not define an inner product on $\mathbb{R}^2$. To show this, we will give a counterexample for the positive-definiteness axiom. Let $\vec{p} = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}$.

$$\begin{aligned} \langle \vec{p}, \vec{p} \rangle &= \begin{bmatrix} p_1 & p_2 \end{bmatrix} \begin{bmatrix} 3 & 1 \\ 1 & -2 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} \\ &= 3p_1p_1 + p_1p_2 + p_2p_1 - 2p_2p_2 \\ &= 3p_1^2 - 2p_2^2 + 2p_1p_2 \end{aligned}$$

Let $p_1 = 0$ and $p_2 = 1$. Then $3p_1^2 - 2p_2^2 + 2p_1p_2 = -2 < 0$.

(c) For $\mathbb{R}^2$:

$$\langle \vec{p}, \vec{q} \rangle = \vec{p}^T \begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix} \vec{q}$$

Solution:
No, the function does not define an inner product on $\mathbb{R}^2$. To show this, we will give a counterexample for the symmetry axiom. Let $\vec{p} = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}$ and $\vec{q} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$.

$$\begin{aligned} \langle \vec{p}, \vec{q} \rangle &= \begin{bmatrix} p_1 & p_2 \end{bmatrix} \begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \\ &= 3p_1q_1 + p_1q_2 + 2p_2q_2 \end{aligned}$$

$$\begin{aligned} \langle \vec{q}, \vec{p} \rangle &= \begin{bmatrix} q_1 & q_2 \end{bmatrix} \begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} \\ &= 3p_1q_1 + p_2q_1 + 2p_2q_2 \end{aligned}$$

Since there exists p_1, q_1, p_2, q_2 such that $p_1q_2 \neq p_2q_1$, we have $\langle \vec{p}, \vec{q} \rangle \neq \langle \vec{q}, \vec{p} \rangle$.

(d) For $\mathbb{P}_2$ (the vector space containing all polynomials of up to degree 2):

$$\langle p(x), q(x) \rangle = p(-1)q(-1) + p(0)q(0) + p(1)q(1)$$

Solution:
Yes, the function defines an inner product on $\mathbb{P}_2$. To show this, we will show that all of the three axioms apply:

i. Symmetry:

$$\begin{aligned}\langle p(x), q(x)\rangle &= p(-1)q(-1) + p(0)q(0) + p(1)q(1) \\ &= q(-1)p(-1) + q(0)p(0) + q(1)p(1) \\ &= \langle q(x), p(x)\rangle\end{aligned}$$

ii. Linearity:

$$\begin{aligned}&\langle \alpha p_1(x) + \beta p_2(x), q(x)\rangle \\ &= (\alpha p_1(-1) + \beta p_2(-1))q(-1) + (\alpha p_1(0) + \beta p_2(0))q(0) + (\alpha p_1(1) + \beta p_2(1))q(1) \\ &= \alpha p_1(-1)q(-1) + \beta p_2(-1)q(-1) + \alpha p_1(0)q(0) + \beta p_2(0)q(0) + \alpha p_1(1)q(1) + \beta p_2(1)q(1) \\ &= \alpha p_1(-1)q(-1) + \alpha p_1(0)q(0) + \alpha p_1(1)q(1) + \beta p_2(-1)q(-1) + \beta p_2(0)q(0) + \beta p_2(1)q(1) \\ &= \alpha(p_1(-1)q(-1) + p_1(0)q(0) + p_1(1)q(1)) + \beta(p_2(-1)q(-1) + p_2(0)q(0) + p_2(1)q(1)) \\ &= \alpha\langle p_1(x), q(x)\rangle + \beta\langle p_2(x), q(x)\rangle\end{aligned}$$

iii. Positive-definiteness:

$$\langle p(x), p(x)\rangle = p(-1)p(-1) + p(0)p(0) + p(1)q(1) = (p(-1))^2 + (p(0))^2 + (p(1))^2$$

Since all of the components are non-negative, $\langle p(x), p(x)\rangle \geq 0$.
We also know that each second-order polynomial can only have two zeroes. Therefore, $\langle p(x), p(x)\rangle = 0$ if and only if $p(-1) = p(0) = p(1) = 0$, which is only true if $p(x) = 0$.

(e) For $\mathbb{P}_3$ (the vector space containing all polynomials of up to degree 3):

$$\langle p(x), q(x)\rangle = p(-1)q(-1) + p(0)q(0) + p(1)q(1)$$

Solution:
No, the function does not define an inner product on $\mathbb{P}_3$. To show this, we will give a counterexample for the positive-definiteness axiom.
Let $p(x) = x^3 - x = (x+1)x(x-1)$. Then $\langle p(x), p(x)\rangle = 0$, but $p(x) \neq 0$.

(f) For $\mathbb{P}_1$ (the vector space containing all polynomials of up to degree 1):

$$\langle p(x), q(x)\rangle = \int_0^1 p(x)q(x)dx$$

In other words, if $p(x) = ax + b$ and $q(x) = cx + d$, then $\langle p(x), q(x)\rangle = \frac{ac}{3} + \frac{ad}{2} + \frac{bc}{2} + bd$.
Solution:
Yes, the function defines an inner product on $\mathbb{P}_1$. To show this, we will show that all of the three axioms apply:

i. Symmetry:

$$\begin{aligned}\langle p(x), q(x)\rangle &= \int_0^1 p(x)q(x)dx \\ &= \int_0^1 q(x)p(x)dx \\ &= \langle q(x), p(x)\rangle\end{aligned}$$

ii. Linearity:

$$\begin{aligned}
\langle \alpha p_1(x) + \beta p_2(x), q(x) \rangle &= \int_0^1 (\alpha p_1(x) + \beta p_2(x)) q(x) dx \\
&= \int_0^1 \alpha p_1(x) q(x) + \beta p_2(x) q(x) dx \\
&= \alpha \int_0^1 p_1(x) q(x) dx + \beta \int_0^1 p_2(x) q(x) dx \quad = \alpha \langle p_1(x), q(x) \rangle + \beta \langle p_2(x), q(x) \rangle
\end{aligned}$$

iii. Positive-definiteness:

$$\langle p(x), p(x) \rangle = \int_0^1 p(x) p(x) dx = \int_0^1 (p(x))^2 dx$$

Geometrically, the integral represents the area under the curve. Since we know that $(p(x))^2$ is always non-negative, $\langle p(x), p(x) \rangle \geq 0$.
Furthermore, the integral will evaluate to 0 if and only if $(p(x))^2 = 0 \implies p(x) = 0$.

4. GPS Receivers

The Global Positioning System (GPS) is a space-based satellite navigation system that provides location and time information in all weather conditions, anywhere on or near the Earth where there is an unobstructed line of sight to four or more GPS satellites. In this problem, we will understand how a receiver (e.g. your cellphone) can disambiguate signals from the different GPS satellites that are simultaneously received.

GPS satellites employ "spread-spectrum" technology (very similar to Code-division multiple access, CDMA, which is commonly used in cellphone transmissions) and a special coding scheme where each transmitter is assigned a code that serves as its "signature".

Each GPS satellite uses a unique 1023 element long sequence as its "signature." These codes used by the satellites are called "Gold codes," and they have some special properties:

- The auto-correlation of a Gold code (correlation with itself) is very **high**.
- The cross-correlation between different Gold codes is very low, i.e. different Gold codes are almost orthogonal to each other.

Gold codes are generated using a linear feedback shift register (LFSR). Understanding how this works is out of scope for the class, but you can read more about LFSR and CDMA if you are interested.

The important thing to know is that the Gold codes are 1023 element vectors where each element is either $+1$ or -1, and that any Gold code is "almost orthogonal" to any other Gold code.

A receiver listening for signature transmissions from a satellite has copies of all of the different GPS satellites' Gold codes. The receiver can determine how long it took for a particular GPS satellite's signal to reach it by taking the correlation of the received signal with a satellite's Gold code. The shift value (delay) that corresponds to maximizing the correlation determines the "propagation delay" between when the GPS satellite transmitted it's signal and when the receiver received it. This time delay can then be converted into a distance (in the case of GPS, electromagnetic waves are used for transmissions, distance is equal to the speed of light multiplied by the time delay).

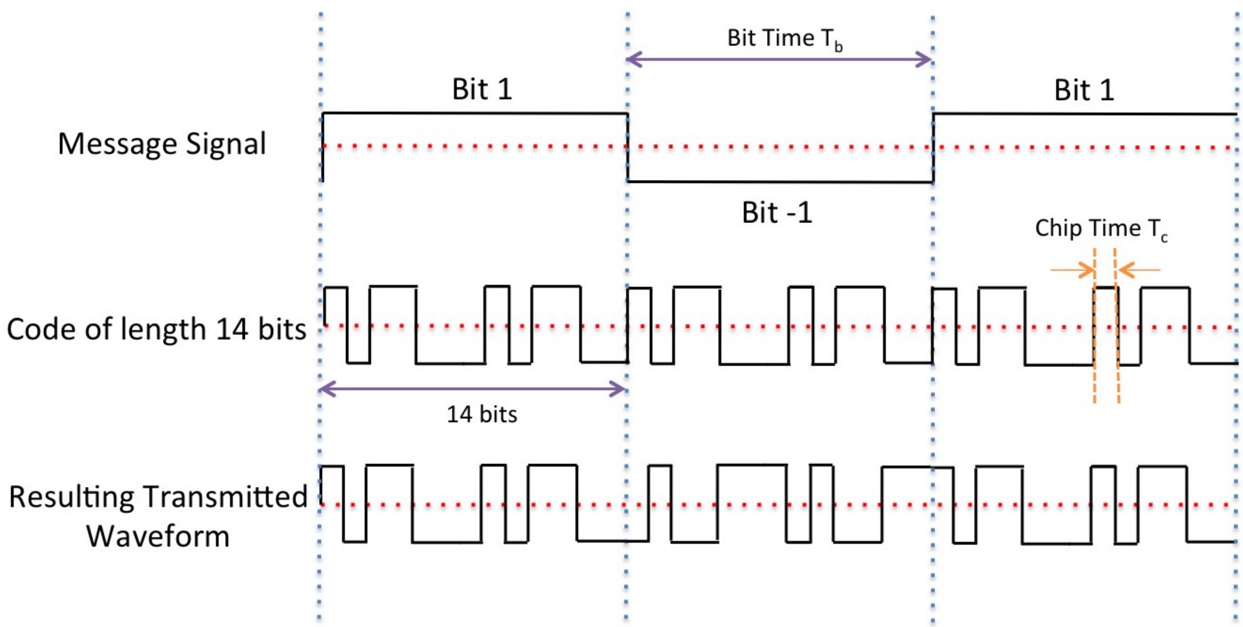
The GPS satellite is constantly transmitting its signature. In addition to identifying itself through its signature, it can also "modulate" the signature to communicate more information. Modulating a signature means multiplying the entire signature block by $+1$ or -1, as shown in the figure.

In the figure below, the signature is of length 14, i.e. it is made of 14 ± 1 symbols. The figure shows 3 blocks of length 14 being transmitted. The message signal (made of $+1$ and -1 as well) multiplies the entire block of the signature, to give the resulting transformed waveform at the bottom of the figure. The message being transmitted in the figure is $\begin{bmatrix} 1 & -1 & 1 \end{bmatrix}$. So to send these three symbols of message, we need to send 14×3 symbols of the gold code.

The waveform that is actually transmitted is the multiplication of the message signal with the signature signal. Now, when a receiver receives a signal, in addition to finding the time delay between transmission and reception, the receiver will be able to decode the message by noting a very high correlation if the message bit is equal to 1, and a very negative correlation if the message bit is equal to -1.

For the problem you will now do, $T_b = 1023 T_c$. (In reality, $T_b = 20 \times 1023 \times T_c$.)

You will use the ideas of linear correlation to figure out which of the satellites are transmitting.



For the purpose of this question we only consider 24 GPS satellites. Download the IPython notebook and the corresponding data files for the following questions:

Note: this code is calculation-heavy, and can take up to a few mins to run for each code block. Be patient!

(a) Auto-correlate (i.e. cross-correlate with itself) the Gold code of satellite 10 and plot it. Python has functions for this. What do you observe?

Solution: The autocorrelation peaks at 1023 when the signals are perfectly aligned (offset 0). The correlation of a Gold code with a shifted version of itself is not significant.

(b) Cross-correlate the Gold code of satellite 10 with satellite 13 and plot it. What do you observe?

Solution: We see that the cross-correlation of a Gold code of any satellite with any other satellite is very low. This indicates that when given some unknown data, we can differentiate between different satellites.

(c) Consider a random signal, i.e. a signal that is not generated due to a specific code but is a random ± 1 sequence. A helper function in the notebook will generate this for you. Cross-correlate it with the

Gold code of satellite 10. What do you observe? What does this mean about our ability to identify satellites in the presence of random ± 1 noise?

Solution: We see that the cross-correlation of the Gold code of any satellite with integer noise is very low. This indicates that we can still figure out the presence of a satellite even if it is buried in noise.

(d) The signal actually received by a receiver will be the satellites' transmissions plus additive noise, and this need not be just noise that takes values ± 1. Use the helper function in the notebook to generate a random noise sequence of length 1023, and compute the cross-correlation of this sequence with the Gold Code of satellite 10. What does this mean about our ability to identify satellites in the presence of real-valued noise?

For the next subparts of this problem, the received signals are corrupted by real-valued noise. Use the observation from this subpart for solving the rest of the question.

Solution: We see that the Gold code of any satellite with Gaussian noise is very low. This indicates that we can still figure out the presence of a satellite even if it is buried in Gaussian noise.

(e) The receiver may receive signals from multiple satellites simultaneously, in which case the signals will all be added together. In addition, noise might be added to the signal. What are the satellites present in `data1.npy`?

Solution: The satellites that are present are satellites 4, 7, 13, and 19.

(f) Let's assume that you can hear only one satellite, Satellite X, at the location you are in (though this never happens in reality). Let's also assume that this satellite is transmitting an unknown sequence of $+1$ and -1 of length 5 (after encoding it with the 1023 bit Gold code corresponding to Satellite X). Find out from `data2.npy` which satellite it is and what sequence of ± 1's it is transmitting.

Solution: Satellite 3 is transmitting 1, -1, -1, -1, 1.

(g) Signals from different transmitters arrive at the receiver with different delays. We use these delays to figure out the distance between the satellite and receiver.

The signals from different satellites are superimposed on each other with different offsets at the start.

What satellites are you able to see in `data3.npy`? Assume that all satellites begin transmission at time 0. What are the delays of all the satellites that are present? You are told that all the satellites have the same message signal given by $\begin{bmatrix}1 & 1 & -1 & -1 & -1\end{bmatrix}$?

Solution: The satellites present in this data are 5 and 20.

The correlation array index where satellite 5's first peak is located is 1022. The array index where satellite 20's first peak is located is 1528.

Since the Gold code length is 1023 and it is shifted back along the received signal at most by 1022, these would correspond to delays of $1022 - 1022 = 0$ and $1528 - 1022 = 506$. Alternatively, you can take the shift array for satellite 5 and access the value with index 1022 which would give 0, and for satellite 20 if you accessed the value with index 1528 you would get a delay of 506.

Give yourself full credit if you identified the correct correlation array index (1022 and 1528) where the messages began or the actual delay (0 and 506).

5. Classification: Targeted promotions

This problem describes a situation that online businesses face regularly. They can only observe the current purchases of a customer but would like to understand the general interests of the customer.

The retail store EehEeh Sixteen would like to create an algorithm that can predict a customer's interests based only on the purchases made by the customer. Based on the interests, the store decides to give the customer a coupon (promotion) that is targeted to the right customer's interests. This can be thought of as a classification problem: given the customer's purchase history, we hope to assign the customer to one of finitely many groups, depending on which promotion we deem to be most suitable for them.

Customer interests are described by a vector: $\vec{s_A} = \begin{bmatrix} \text{party-interest score} \\ \text{family-interest score} \\ \text{student-interest score} \\ \text{office-interest score} \end{bmatrix}$

The store would like to infer this vector for each customer.

(a) Assume we have the interests of a customer c in a vector $\vec{x_c} = \begin{bmatrix} c_{\text{party}} \\ c_{\text{family}} \\ c_{\text{student}} \\ c_{\text{office}} \end{bmatrix}$ and a set of promotions $A_1, A_2, \ldots, A_N$, with their attached vectors of scores $\vec{s_{A_1}}, \vec{s_{A_2}}, \ldots, \vec{s_{A_N}}$ (that are also customer interest vectors).

We would like to select the promotion vector that is closest to the customer interest vector, because this is the promotion that the customer would be the most interested in. To perform this selection, we would like to come up with some measure of similarity. Specifically, we want a function that outputs **a higher value if the two vectors are closer to each other**.

The larger the value of the similarity function between $\vec{x_c}$ and $\vec{s_{A_i}}$ the better suited the promotion is for the customer. You have two choices for the similarity measure:

Distance: $sim_1(\vec{x_c}, \vec{s_A}) = \|\vec{x_c} - \vec{s_A}\|$ is a norm and measures the distance between $\vec{x_c}$ and $\vec{s_A}$. Projection: $sim_2(\vec{x_c}, \vec{s_A}) = \left\langle \vec{x_c}, \frac{\vec{s_A}}{\|\vec{s_A}\|} \right\rangle$ is a normalized inner product. Which one is a better similarity measure? Why?

Solution:

Distance: $sim_1(\vec{x_c}, \vec{s_A}) = \|\vec{x_c} - \vec{s_A}\|$ is not good because the farther these vectors are from each other, the higher the score. Which is the opposite of what we wanted. Projection: $sim_2(\vec{x_c}, \vec{s_A}) = \left\langle \vec{x_c}, \frac{\vec{s_A}}{\|\vec{s_A}\|} \right\rangle$ is a better option because the closer the two vectors, the larger the projection will be. Lock on this choice for the rest of the problem.

(b) Unfortunately, the store does not get to observe each customer's interest vector. It only gets to observe the money the customer spends in four categories: food, movies, art, and books. The store needs to use this information to infer the vector $\vec{s_A}$ for each customer.

The EehEeh Sixteen research division conducted some studies that calculated the distribution of spending for people who are purely interested in only one category. For example, a person who is only interested in Party-spending will have the vector $x_c = [1,\ 0,\ 0,\ 0]^T$, and the spending of this person is given in the first line of Table: 1. Similarly, the remaining rows tell you how a person who is just interested in Family, Students, or Offices will spend.

	Spending Category			
Interest Category	**Food**	**Movies**	**Art**	**Books**
Party	40%	33%	22%	5%
Family	70%	10%	10%	10%
Student	20%	10%	15%	55%
Office	5%	2%	20%	73%

Table 1: The distribution of spending of people in each category.

We want to use this data to infer the interest vectors of customers given their spending, assuming that the spending of each customer is a linear combination of the spending of the "pure customers" (those with interest vectors $[1,\ 0,\ 0,\ 0]^T$, $[0,\ 1,\ 0,\ 0]^T$ etc). Suppose a customer spends $T_{\text{food}}\%$ on food, $T_{\text{movies}}\%$ on movies, $T_{\text{art}}\%$ on art, and $T_{\text{books}}\%$ on books.

Use the information in Table 1 to devise a system of linear equations so you can solve for the customer's preferences, x_c.

Solution:

For a given customer, $T_{\text{food}}\%, T_{\text{movies}}\%, T_{\text{art}}\%$ and $T_{\text{books}}\%$ represent the customer's percent spending on food, movies, art and books, respectively.

The system of linear equations that describes the spending above, assuming the spending are observed:

$$\begin{aligned}
0.4c_{\text{party}} + 0.7c_{\text{family}} + 0.2c_{\text{student}} + 0.05c_{\text{office}} &= T_{\text{food}}\% \\
0.33c_{\text{party}} + 0.1c_{\text{family}} + 0.1c_{\text{student}} + 0.02c_{\text{office}} &= T_{\text{movies}}\% \\
0.22c_{\text{party}} + 0.1c_{\text{family}} + 0.15c_{\text{student}} + 0.2c_{\text{office}} &= T_{\text{art}}\% \\
0.05c_{\text{party}} + 0.1c_{\text{family}} + 0.55c_{\text{student}} + 0.73c_{\text{office}} &= T_{\text{books}}\%
\end{aligned}$$

(c) We will combine the results from the previous parts to complete the partially filled out algorithm below. The algorithm takes the raw spending of a customer, $M_{\text{food}}, M_{\text{movies}}, M_{\text{art}}, M_{\text{books}}$, and the promotion scores, $\vec{s_{A_1}}, \vec{s_{A_2}}, \ldots, \vec{s_{A_N}}$, as inputs. The algorithm's output should be the best promotion for that customer.

For this part, use the second similarity metric from part (a). In lines 2 to 5, we first normalize the spending subtotals to get spending percentages.

Algorithm 1 The EehEeh Sixteen promotions algorithm

1: **procedure** PROMOTION($M_{\text{food}}, M_{\text{movies}}, M_{\text{art}}, M_{\text{books}}, \vec{s_{A_1}}, \vec{s_{A_2}}, \ldots, \vec{s_{A_N}}$)

2: $\quad T_{\text{food}}\% = \frac{M_{\text{food}}}{M_{\text{food}}+M_{\text{movies}}+M_{\text{art}}+M_{\text{books}}}$

3: $\quad T_{\text{movies}}\% = \frac{M_{\text{movies}}}{M_{\text{food}}+M_{\text{movies}}+M_{\text{art}}+M_{\text{books}}}$

4: $\quad T_{\text{art}}\% = \frac{M_{\text{art}}}{M_{\text{food}}+M_{\text{movies}}+M_{\text{art}}+M_{\text{books}}}$

5: $\quad T_{\text{books}}\% = \frac{M_{\text{books}}}{M_{\text{food}}+M_{\text{movies}}+M_{\text{art}}+M_{\text{books}}}$

6: Set up and solve the system from part b

7: Assign $\vec{x_c} = \begin{bmatrix} c_{\text{party}} \\ c_{\text{family}} \\ c_{\text{student}} \\ c_{\text{office}} \end{bmatrix}$

8: *Pick promotion A using similarity metric from part a.*

9: Print promotion A

10: **end procedure**

How should we pick the promotion A using the similarity metric from part a? Complete the specification of Algorithm 2 by writing down what step 8 should be.

Solution:

Use algorithm 2 for reference. You may want to simplify things by using simpler notation.

(d) Run the algorithm to figure out what promotion we should give to Jane Doe who spent \$6 on food, \$4 on movies, \$1 on art and \$5 on books. Use the values in Table 1 and assume there are 4 promotions, A_1,

Algorithm 2 The EehEeh Sixteen promotions algorithm

procedure $\text{PROMOTION}(M_{\text{food}}, M_{\text{movies}}, M_{\text{art}}, M_{\text{books}}, \vec{s_{A_1}}, \vec{s_{A_2}}, \ldots, \vec{s_{A_N}})$

$T_{\text{food}}\% = \frac{M_{\text{food}}}{M_{\text{food}}+M_{\text{movies}}+M_{\text{art}}+M_{\text{books}}}$

$T_{\text{movies}}\% = \frac{M_{\text{movies}}}{M_{\text{food}}+M_{\text{movies}}+M_{\text{art}}+M_{\text{books}}}$

$T_{\text{art}}\% = \frac{M_{\text{art}}}{M_{\text{food}}+M_{\text{movies}}+M_{\text{art}}+M_{\text{books}}}$

$T_{\text{books}}\% = \frac{M_{\text{books}}}{M_{\text{food}}+M_{\text{movies}}+M_{\text{art}}+M_{\text{books}}}$

Solve the system

$$\begin{aligned}
0.4c_{\text{party}} + 0.7c_{\text{family}} + 0.2c_{\text{student}} + 0.05c_{\text{office}} &= T_{\text{food}}\% \\
0.33c_{\text{party}} + 0.1c_{\text{family}} + 0.1c_{\text{student}} + 0.02c_{\text{office}} &= T_{\text{movies}}\% \\
0.22c_{\text{party}} + 0.1c_{\text{family}} + 0.15c_{\text{student}} + 0.2c_{\text{office}} &= T_{\text{art}}\% \\
0.05c_{\text{party}} + 0.1c_{\text{family}} + 0.55c_{\text{student}} + 0.73c_{\text{office}} &= T_{\text{books}}\%
\end{aligned}$$

Assign $\vec{x_c} = \begin{bmatrix} c_{\text{party}} \\ c_{\text{family}} \\ c_{\text{student}} \\ c_{\text{office}} \end{bmatrix}$

Pick promotion A such that $\left\langle \vec{x_c}, \frac{\vec{s_A}}{\|\vec{s_A}\|} \right\rangle$ *is highest.*

Print promotion A

end procedure

A_2, A_3, and A_4, with associated score vectors $\vec{s_{A_1}} = \begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \end{bmatrix}$, $\vec{s_{A_2}} = \begin{bmatrix} \frac{2}{3} \\ -\frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{3} \end{bmatrix}$, $\vec{s_{A_3}} = \begin{bmatrix} -\frac{1}{2} \\ -\frac{1}{2} \\ \frac{5}{2} \\ -\frac{1}{2} \end{bmatrix}$ and $\vec{s_{A_4}} = \begin{bmatrix} 0 \\ \frac{1}{2} \\ 0 \\ \frac{1}{2} \end{bmatrix}$.

You may use IPython to run your algorithm. *Hint: Note that the preference vectors do not all have the same magnitude!*

Solution:

First, we normalize to get: $T_{\text{food}}\% = 37.5\%, T_{\text{movies}}\% = 25\%, T_{\text{art}}\% = 6.25\%, T_{\text{books}}\% = 31.25\%$ which yields the following system of linear equations:

$$\begin{aligned}
0.4c_{\text{party}} + 0.7c_{\text{family}} + 0.2c_{\text{student}} + 0.05c_{\text{office}} &= 0.375 \\
0.33c_{\text{party}} + 0.1c_{\text{family}} + 0.1c_{\text{student}} + 0.02c_{\text{office}} &= 0.25 \\
0.22c_{\text{party}} + 0.1c_{\text{family}} + 0.15c_{\text{student}} + 0.2c_{\text{office}} &= 0.0625 \\
0.05c_{\text{party}} + 0.1c_{\text{family}} + 0.55c_{\text{student}} + 0.73c_{\text{office}} &= 0.3125
\end{aligned}$$

The solution to this system is $\vec{x_c} = \begin{bmatrix} -0.02295082 \\ -0.22311475 \\ 3.18704918 \\ -1.94098361 \end{bmatrix}$

Running sim_2 we get $\left\langle \vec{x_c}, \frac{\vec{s_{A_1}}}{\|\vec{s_{A_1}}\|} \right\rangle = -2.68704918032$, $\left\langle \vec{x_c}, \frac{\vec{s_{A_2}}}{\|\vec{s_{A_2}}\|} \right\rangle = 1.01497422276$, $\left\langle \vec{x_c}, \frac{\vec{s_{A_3}}}{\|\vec{s_{A_3}}\|} \right\rangle = 3.42479185518$ and $\left\langle \vec{x_c}, \frac{\vec{s_{A_4}}}{\|\vec{s_{A_4}}\|} \right\rangle = -1.53024862597$ and therefore, the promotion A_3 will be printed.

(e) Will there ever be a customer for which the system devised in part (b) will yield no solutions or infinite solutions?

Solution:

No. Let us work with what we know about invertibility of a matrix. Specifically, we know that a matrix is invertible when the columns of that matrix are linearly independent. In this problem, the vectors on the columns of our system matrix represent the spending percentages for a person "purely" interested in a single topic. This implies each "pure" person's spending vectors will not be collinear with anyone elses' spending vector (otherwise those people would be interested in the same topic). In our course's terminology, this implies that the columns are linearly independent, that the matrix is invertible, and their will be single unique solution for each customer.

6. Audio File Matching

Lots of different quantities we interact with every day can be expressed as vectors. For example, an audio clip can be thought of as a vector. The series of numbers in the clip determines the sounds we hear.

Many audio processing application rely on representing audio files as vectors. In this problem we explore how we could possibly use the idea of inner products to build an application like *Shazam*.

Audio files can naturally be represented as vectors or signals. Every component of the vector/signal determines the sound we hear at a given time. We will use inner products to determine if a particular audio clip was part of a longer song. The ideas here are similar to the themes of the Acoustic Positioning System in the lab.

For this problem the magnitude of each vector determines the volume and the angle of each vector captures the tune.

Let us consider a very simplified model for an audio signal. At each time, the audio signal can be either -1 or $+1$. A vector of length N makes up the audio file.

(a) Say we want to compare two audio files of the same length N to decide how similar they are. First, consider two vectors that are exactly identical $\vec{x}_1 = \begin{bmatrix}1 & 1 & \cdots & 1\end{bmatrix}^T$ and $\vec{x}_2 = \begin{bmatrix}1 & 1 & \cdots & 1\end{bmatrix}^T$. What is the inner product of these two vectors? What if $\vec{x}_1 = \begin{bmatrix}1 & 1 & \cdots & 1\end{bmatrix}^T$ and $\vec{x}_2 = \begin{bmatrix}1 & -1 & 1 & -1 & \cdots & 1 & -1\end{bmatrix}^T$ (where the length of the vector is an even number)? Can you come up with an idea to compare the similarity two general vectors of length N ?

Solution:

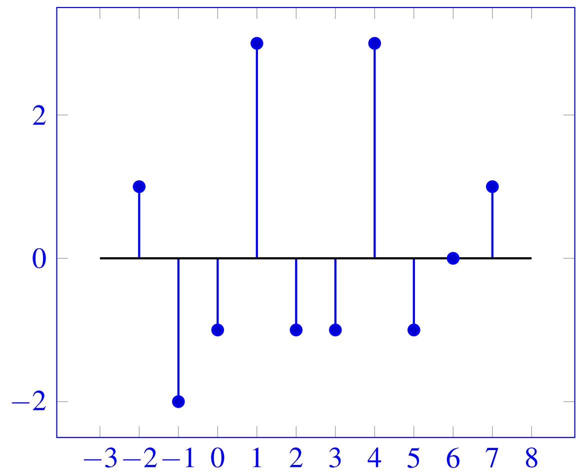
The inner product of $\vec{x}_1 = \begin{bmatrix}1 & 1 & \cdots & 1\end{bmatrix}^T$ and $\vec{x}_2 = \begin{bmatrix}1 & 1 & \cdots & 1\end{bmatrix}^T$ is $\vec{x}_1 \cdot \vec{x}_2 = N$. The inner product of $\vec{x}_1 = \begin{bmatrix}1 & 1 & \cdots & 1\end{bmatrix}^T$ and $\vec{x}_2 = \begin{bmatrix}1 & -1 & 1 & -1 & \cdots & 1 & -1\end{bmatrix}^T$ is $\vec{x}_1 \cdot \vec{x}_2 = 0$ when the vector length is even. To compare two vectors of length N composed of 1 and -1, we take the inner product of the two vectors, a large inner product means the vectors have a similar direction.

In many circumstances, an inner product with a very large negative value would mean the vectors are very different, but it turns out that humans are unable to perceive the sign of sound, so two sounds vectors $\vec{x}$ and $-\vec{x}$ sound exactly the same. As a result, for this problem we are interested in is the **absolute value** of the dot product, but in many other problems, we will interpret a large negative dot product as very different vectors. Don't take off points in parts (a), (b), or (c) if you didn't mention the absolute value.

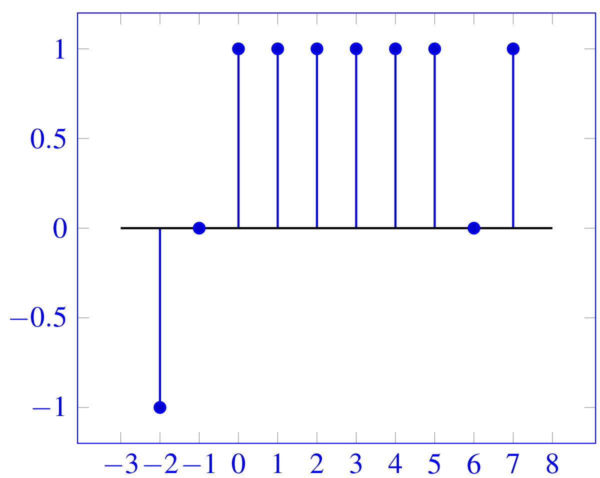
(b) Next suppose we want to find a short audio clip in a longer one. We might want to do this for an application like *Shazam*, which is able to identify a song from a signature tune. Consider the vector of length 8, $\vec{x} = \begin{bmatrix}-1 & 1 & 1 & -1 & 1 & 1 & -1 & 1\end{bmatrix}^T$. Let us label the elements of $\vec{x}$ so that $\vec{x} =$

$\begin{bmatrix} x[0] & x[1] & x[2] & x[3] & x[4] & x[5] & x[6] & x[7] \end{bmatrix}^T$. We want to find the short segment $\vec{y} = \begin{bmatrix} y[0] & x[1] & y[2] \end{bmatrix}^T = \begin{bmatrix} 1 & 1 & -1 \end{bmatrix}^T$ in the longer vector. To do this, perform the linear cross correlation between these two finite length sequences and identify at what shift(s) the linear cross correlation is maximized. Applying the same technique to identify what shift(s) gives the best match for $\vec{y} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T$?
(If you wish, you may use iPython to do this part of the question, but you do not have to)

Solution:



The above plot is $\text{corr}_{\vec{x}}(\vec{y})[k]$ where $\vec{y} = \begin{bmatrix} 1 & 1 & -1 \end{bmatrix}^T$. At shifts 1 and 4 the cross correlation is maximum with value 3.



The above plot is $\text{corr}_{\vec{x}}(\vec{y})[k]$ where $\vec{y} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T$. At shifts 0 through 5 the cross correlation is maximum with value 1. There is not a really good match like before.

(c) Now suppose our audio vector is represented using integers and not just by 1 and -1. We want to find the short audio clip $\vec{y} = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}^T$ in the song given by $\vec{x} = \begin{bmatrix} 1 & 2 & 3 & 1 & 2 & 2 & 3 & 10 \end{bmatrix}^T$. Where do you expect to see the peak in the correlation of the two signals? Is the peak where you want it to be, i.e. does it pull out the clip of the song that you intended? Why?

Solution:

Applying the technique in part (b), we get the best match to be $\begin{bmatrix} 2 & 3 & 10 \end{bmatrix}^T$ as this has the largest dot product with $\vec{y} = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}^T$. This is not where we expect to see the peak, as we observe the short audio clip $\vec{y}$ appears at the beginning of the song.

This happens because the volume at the end of the song is louder than the beginning of the song. Despite the angle not matching as well, the louder volume causes the linear cross correlation to be larger.

(d) **(Optional:)** Let us think about how to get around the issue in the previous part. Cross-correlation compares segments of $\vec{x}$ of length 3 (which is the length of $\vec{y}$) with $\vec{y}$. Instead of directly taking the cross correlation, we want to normalize each inner product computed at each shift by the magnitudes of both segments, i.e. we want to consider $\frac{<\vec{x}_k,\vec{y}>}{\|\vec{x}_k\|\|\vec{y}\|}$, where $\vec{x}_k$ is the length 3 segment starting from the k-th index of $\vec{x}$. This is referred to as normalized cross correlation. Using this procedure, now which segment matches the short audio clip best?

Solution: Using the normalized cross correlation procedure, the best match for the short audio clip is at the 0^{th} shift and it perfectly matches the clip.

(e) **(Optional:)** Now we will implement the strategy above in the IPython notebook, where you can use a "normalized" correlation calculation. Run the appropriate cells in the IPython notebook.

Solution:

See `sol12.ipynb`.

7. Homework Process and Study Group

Who else did you work with on this homework? List names and student ID's. (In case of homework party, you can also just describe the group.) How did you work on this homework?

Solution:

I worked on this homework with...

I first worked by myself for 2 hours, but got stuck on problem 5, so I went to office hours on...

Then I went to homework party for a few hours, where I finished the homework.