Corrections

October 3, 2023 11:50 PM

Problem Sets

Problem Set 1

September 20, 2023 11:01 PM

Means Mistake

Problem 1.1

September 20, 2023 10:03 PM

Problem 1.1

a) show the functions $l_1 : \mathbb{R}^n \to \mathbb{R}$ is a norm

Definition of a norm:

$\|\cdot\|$ is a function that : $V \to \mathbb{R}$ that satisfies:

i) $\|v\| \geq 0$, $\forall v \in V$, $\|v\| = 0$ iff $v = 0$

ii) $\|\alpha v\| = \|\alpha\| \|v\|$ $\forall v \in V$, $\alpha \in \mathbb{R}$

iii) $\|u + v\| \leq \|u\| + \|v\|$

l_1 definition:

$$\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{\frac{1}{p}}, \quad 1 \leq p < \infty$$

Where $p = 1$

$$l_1 = \|x\|_1 = \left(\sum_{i=1}^{n} |x_i|^1 \right)^{\frac{1}{1}} = \sum_{i=1}^{n} |x_i|$$

$x_1 \quad \ldots \quad \Big(\quad i=1 \quad \Big) \quad i=1$

i) if $\vec{V}=0$

then, $\|\vec{V}\|$

$$= \sum_{i=1}^{n} |V_i| = (V_1 + V_2 + \ldots V_n)$$

$$= (0 + 0 + 0 \ldots 0)$$

$$= 0$$

$\therefore \|\vec{V}\| = 0$

Otherway:

if $\|\vec{V}\|$ then $\sum_{i=0}^{n} |V_i| = 0$

$$(V_1 + V_2 \ldots V_n) = 0$$

$$V_i = 0 \quad \forall i$$

$$\therefore \vec{V} = 0$$

$\therefore$ Property satified

ii) $\|\alpha V\|$

$$= \sum_{i=1}^{n} |\alpha V_i|$$

Property of absolute functions

$$= \sum_{i=1}^{n} |\alpha| \, |V_i|$$

$$= \sum_{i=1}^{n} |\alpha| \sum_{i=1}^{n} |V_i|$$

$$= \|\alpha\| \, \|V\|$$

$= ||\alpha|| \, ||V||$

iii) $||u+V||$ Triangle inequality: $|x+y| \le |x| + |y|$

$= \sum_{i=1}^{n} |u_i + v_i|$

$\sum_{i=1}^{n} |u_i + v_i| \le \sum_{i=1}^{n} |u_i| + |v_i| = \sum_{i=1}^{n} |u_i| + \sum_{i=1}^{n} |v_i|$

$||u+V|| \le ||u|| + ||V||$

$\therefore$ Property holds

b) $l_\infty : R^n \to R$

Defn: $l_\infty = \max_{i=1\ldots n} |x_i|$

1) if $\vec{V} = 0$ then,

$||\vec{V}|| = \max_{i=1,\ldots n} |v_i|$

$= \max_{i=1\ldots n} |0, 0, 0, \ldots 0|$

$= 0$

other way:

if $||\vec{V}|| = 0$ then,

$\max_{i=1\ldots n} |v_i| = 0$

$$\max_{i=1..n} |v_i| = 0$$

$$v_i = 0 \quad \forall i$$

$$\therefore \vec{v} = 0$$

$\therefore$ Property holds

2) $\|\alpha v\|$

$$= \max_{i...n} |\alpha v_i|$$

$$= \max_{i...n} |\alpha||v_i|$$

due to scaling

$$= |\alpha| \max_{i...n} |v_i|$$

$$= \|\alpha\| \|v\|$$

Property holds

Property holds

3) $\|u + v\| = \max_{i...n} |u_i + v_i|$

$$\leq \max_{i...n} |u_i| + \max_{i..n} |v_i|$$

the max of the sum will be less than or equal to the sum of ax of u, and the max of v

$$\leq \|u\| + \|v\|$$

Problem 1.2

September 20, 2023 11:01 PM

Exercise 2.6 (norm-inequalities)

1. Show the following inequalities hold for any vector x:

$$\underbrace{\frac{1}{\sqrt{n}}\|x\|_2 \le \overbrace{\|x\|_\infty}^{}}_{①} \overbrace{\le \|x\|_2}^{②} \underbrace{\le \overbrace{\|x\|_1}}_{③} \overbrace{\underbrace{\le \sqrt{n}\,\|x\|_2}}^{④} \underbrace{\le n\|x\|_\infty}_{⑤}$$

① $\frac{1}{\sqrt{n}}\|x\|_2$

$$= \frac{1}{\sqrt{\;}} \sqrt{\sum^{n} x^2} = \sqrt{\frac{\sum_{i=1}^{n} x^2}{}} = \sqrt{\frac{(x_1^2 + \dots x_n^2)}{n}}$$

mean of values in $\vec{x}$

$$= \frac{1}{\sqrt{n}} \sqrt{\sum_{i=1}^{n} x^2} = \sqrt{\frac{\sum_{i=1} x^2}{n}} = \sqrt{\frac{(x_1 + \dots x_n)}{n}}$$ } mean of values in $\vec{x}$

$$\|x\|_\infty = \max_{i \dots n} |x_i|$$ } max # in values of $\vec{x}$

$$\text{mean}(\vec{x}) \le \max(\vec{x})$$

$$\therefore \frac{1}{\sqrt{n}} \|x\|_2 \le \|x\|_\infty$$

max of $\vec{x}$

(2) $$\|x\|_\infty = \max_{i \dots n} |x_i| = x_j$$

$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$$

$$= \sqrt{(x_1^2 + x_2^2 \dots x_n^2)}$$

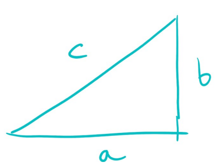
$$x_j \le \sqrt{x_1^2 + x_2^2 \dots x_n^2}$$

$$\therefore \|x\|_\infty \le \|x\|_2$$

(3) $$\|x\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$$

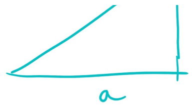
$$= \sqrt{(x_1^2 + x_2^2 + \dots x_n^2)}$$

$$\|x\|_1 = \sum_{i=1}^{n} |x_i| = (|x_1| + |x_2| + \dots |x_n|)$$



$\rightarrow$ $c \le a + b$, $c^2 = a^2 + b^2$

$\therefore \sqrt{a^2 + b^2} \le a + b$



$$\therefore \sqrt{a^2+b^2} \le a+b$$

Generalizing to multiple dimensions:

$$\sqrt{x_1^2 + x_2^2 \cdots + x_n^2} \le x_1 + x_2 + \cdots x_n \le |x_1| + |x_2| + \cdots + |x_n|$$

$$\sqrt{x_1^2 + x_2^2 \cdots x_n^2} \le |x_1| + |x_2| + \cdots |x_n|$$

$$\sqrt{\sum_{i=1}^{n} x_i^2} \le \sum_{i=1}^{n} |x_i|$$

$$\therefore \|x\|_2 \le \|x\|_1$$

4) $\|x\|_1 = \sum_{i=1}^{n} |x_i| = |x_1| + |x_2| + \cdots |x_n|$

Cauchy-Schwartz inequality

$$|\langle u, v \rangle| \le \|u\|_2 \|v\|_2$$

$$|u^T v| \le \|u\|_2 \|v\|_2$$

let $u = \begin{bmatrix} 1 \\ \vdots \\ \vdots \end{bmatrix}$, $V = \begin{bmatrix} |x_1| \\ |x_2| \\ \vdots \end{bmatrix}$

let $u = \begin{bmatrix} 1 \\ \vdots \\ \vdots \\ 1 \end{bmatrix}$, $V = \begin{bmatrix} |x_1| \\ |x_2| \\ \vdots \\ |x_n| \end{bmatrix}$

$$|u^T v| = \sum_{i=1}^{n} u_i v_i = \sum_{i=1}^{n} |x_i| = \|x\|_1$$

$$\|u\|_2 = \sqrt{\sum_{i=1}^{n} 1^2} = \sqrt{n}$$

$$\|u\|_2 = \sqrt{\sum_{i=1} 1^2} = \sqrt{n}$$

$$\|v\| = \sqrt{\sum_{i}^{n} |x_i^2|} = \|x\|_2$$

$$\therefore \|x\|_1 \leq \sqrt{n}\|x\|_2$$

5 $\sqrt{n}\|x\|_2 \leq n\|x\|_\infty$

$$\sqrt{n}\|x\|_2 = \sqrt{nx_1^2 + nx_2^2 + \ldots + nx_n^2}$$

$$n\|x\|_\infty = \max_{i \ldots n} |x_i| \cdot n$$

let $x_j = \max_{i \ldots n} |x_i| \quad \therefore \quad x_i \leq x_j$

$$n\|x\|_\infty = n \cdot x_j$$

$$\sqrt{nx_1^2 + nx_2^2 \ldots nx_n^2} \leq \sqrt{n}x_1 + \sqrt{n}x_2 \ldots \sqrt{n}x_n = \sqrt{n}\sum_{i=1}^{n} |x_i| \leq n \cdot x_j$$

Missing Part b of the question

Problem 1.3

September 21, 2023 10:54 PM

Problem 1.3:

5.1 *Linear independence of stacked vectors.* Consider the stacked vectors

$$c_1 = \begin{bmatrix} a_1 \\ b_1 \end{bmatrix}, \ \ldots, \ c_k = \begin{bmatrix} a_k \\ b_k \end{bmatrix},$$

where $a_1, \ldots, a_k$ are n-vectors and $b_1, \ldots, b_k$ are m-vectors.

(a) Suppose $a_1, \ldots, a_k$ are linearly independent. (We make no assumptions about the vectors $b_1, \ldots, b_k$.) Can we conclude that the stacked vectors $c_1, \ldots, c_k$ are linearly where $a_1, \ldots, a_k$ are n-vectors and $b_1, \ldots, b_k$ are m-vectors.

(a) Suppose $a_1, \ldots, a_k$ are linearly independent. (We make no assumptions about the vectors $b_1, \ldots, b_k$.) Can we conclude that the stacked vectors $c_1, \ldots, c_k$ are linearly independent?

(b) Now suppose that $a_1, \ldots, a_k$ are linearly dependent. (Again, with no assumptions about $b_1, \ldots, b_k$.) Can we conclude that the stacked vectors $c_1, \ldots, c_k$ are linearly dependent?

a) if $a_1 \ldots a_k$ are linearly independant,

$$a_1 \neq a_2 \neq \ldots \neq a_k \ \& \ a_1 \neq l a_2$$

$$\therefore \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \neq \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \neq \ldots \neq \begin{bmatrix} a_k \\ b_k \end{bmatrix}$$

even if $b_1 = b_2 = \ldots b_k$

i.e. $\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 1 \\ 0 \end{bmatrix} \begin{matrix} \}\, a_1 \\ \\ \}\, b_1 \end{matrix} \quad \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 1 \\ 0 \end{bmatrix} \begin{matrix} \}\, a_2 \\ \\ \}\, b_2 \end{matrix}$ $\rightarrow$ are linear independant since a_i are linear independant

b) In the same way, we cannot guarantee that $c_1 \ldots c_k$ are independent even if $a_1 \ldots a_k$ are dependant because $b_1 \ldots b_k$ could be independant and then $c_1 \ldots c_k$ would be independant

Problem 1.4

September 21, 2023 11:02 PM

Exercise 2.5 (Orthogonality) Let $x, y \in \mathbb{R}^n$ be two unit-norm vectors, that is, such that $\|x\|_2 = \|y\|_2 = 1$. Show that the vectors $x - y$ and $x + y$ are orthogonal. Use this to find an orthogonal basis for the subspace spanned by x and y.

Exercise 2.5 (Orthogonality) Let $x, y \in \mathbb{R}^n$ be two unit-norm vectors, that is, such that $\|x\|_2 = \|y\|_2 = 1$. Show that the vectors $x - y$ and $x + y$ are orthogonal. Use this to find an orthogonal basis for the subspace spanned by x and y.

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

$$x - y = \begin{bmatrix} x_1 - y_1 \\ \vdots \\ x_n - y_n \end{bmatrix} \qquad x + y = \begin{bmatrix} x_1 + y_1 \\ \vdots \\ x_n + y_n \end{bmatrix}$$

Orthogonal:

$$\langle u, v \rangle = 0$$

$$\langle x - y, x + y \rangle = \begin{bmatrix} x_1 - y_1 & \cdots & x_n - y_n \end{bmatrix} \begin{bmatrix} x_1 + y_1 \\ \vdots \\ x_n + y_n \end{bmatrix}$$

$$= (x_1 - y_1)(x_1 + y_1) + (x_2 - y_2)(x_2 - y_2) + \dots + (x_n - y_n)(x_n + y_n)$$

Difference of Squares

$$= (x_1^2 - y_2^2) + (x_2^2 - y_2^2) + \dots + (x_n^2 - y_n^2)$$

$$= x_1^2 + x_2^2 + x_3^2 \dots + x_n^2 - (y_1^2 + y_2^2 + y_3^2 \dots + y_n^2)$$

From Problem statement:

$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2} = 1 \qquad \|y\|_2 = \sqrt{\sum_{i=1}^{n} y_i^2} = 1$$

$$\therefore \sum_{i=1}^{n} x_i^2 = 1 \qquad \therefore \sum_{i=1}^{n} y_i^2 = 1$$

$$= \sum_{i=1}^{n} x_i^2 - \sum_{i=1}^{n} y_i^2$$

$$= 1 - 1$$

$= 1 - 1$

$= 0$

$\therefore \langle x-y, x+y \rangle = 0$ and $\vec{x-y}$ and $\vec{x+y}$ are orthogonal vectors to each other.

Basis definition:

1. Two orthogonal vectors ✓

2. Span entire subspace

$\text{span}\{x, y\} = a_1\vec{x} + a_2\vec{y}$ $\forall$ possible a_1 & a_2 values

$$\text{span}\{x-y, x+y\} = b_1(\vec{x-y}) + b_2(x+y) \quad \forall \text{ possible } b_1 \text{ \& } b_2$$

$$= b_1\vec{x} - b_1\vec{y} + b_2\vec{x} + b_2\vec{y}$$

$$= b_1\vec{x} + b_2\vec{x} - b_1\vec{y} + b_2\vec{y}$$

$$= (b_1 + b_2)\vec{x} + (b_2 - b_1)\vec{y}$$

let $b_1 + b_2 = a_1$, $b_2 - b_1 = a_2$

$$= a_1\vec{x} - a_2\vec{y}$$

$$= \text{span}\{x, y\}$$

$\therefore$ $\vec{x-y}$ & $\vec{x+y}$ span the entire subspace spanned by x and y. Therefore $\vec{x-y}$ & $\vec{x+y}$ are a basis that spans this subspace

Problem 1.5

September 25, 2023 6:45 PM

Exercise 2.4 (Inner product) Let $x, y \in \mathbb{R}^n$. Under which condition on $\alpha \in \mathbb{R}^n$ does the function

$$f(x,y) = \sum_{k=1}^{n} \alpha_k x_k y_k$$

define an inner product on $\mathbb{R}^n$?

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad \vec{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

$$f(x,y) = \sum_{k=1}^{n} \alpha_k x_k y_k$$

Definition of inner product:

$$\langle x, y \rangle = x^T y = [x_1 \ldots x_n] \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

$$= x_1 y_1 + x_2 y_2 + \ldots x_n y_n$$

$$= \sum_{i=1}^{n} x_i y_i$$

$$f(x,y) = \sum_{k=1}^{n} a_k x_k y_k$$

$$f(x,y) = \sum_{k=1} a_k x_k y_k$$

If $a_k = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$

If $a_k = \begin{bmatrix} \vdots \\ 1 \end{bmatrix}$

$$f(x,y) = \sum_{k=1}^{n} x_k y_k = \langle x, y \rangle$$

~~$\therefore$ the condition for α is if $\alpha = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$, then $f(x,y)$ defines the inner product operation~~

Technically any $a_k > 0$ would do

Problem 1.6

September 25, 2023 6:50 PM

3.24 *Distance versus angle nearest neighbor.* Suppose $z_1, \ldots, z_m$ is a collection of n-vectors, and x is another n-vector. The vector z_j is the (distance) nearest neighbor of x (among the given vectors) if

$$\|x - z_j\| \le \|x - z_i\|, \quad i = 1, \ldots, m,$$

i.e., x has smallest distance to z_j. We say that z_j is the *angle nearest neighbor* of x if

$$\angle(x, z_j) \le \angle(x, z_i), \quad i = 1, \ldots, m,$$

i.e., x has smallest angle to z_j.

(a) Give a simple specific numerical example where the (distance) nearest neighbor is not the same as the angle nearest neighbor.

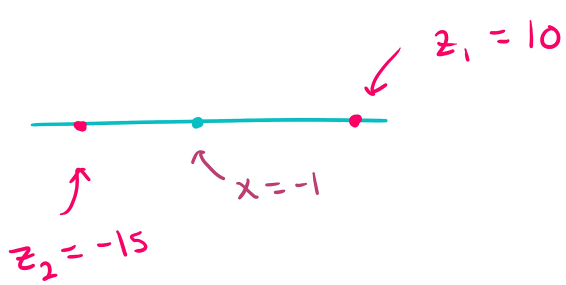
(b) Now suppose that the vectors $z_1, \ldots, z_m$ are normalized, which means that $\|z_i\| = 1$, $i = 1, \ldots, m$. Show that in this case the distance nearest neighbor and the angle nearest neighbor are always the same. *Hint.* You can use the fact that arccos is a decreasing function, *i.e.*, for any u and v with $-1 \le u < v \le 1$, we have $\arccos(u) > \arccos(v)$.

Definition of angle between Vectors:

$$a^T b = \|a\| \, \|b\| \cos\theta$$

$$\therefore \ \theta = \arccos\left(\frac{a^T b}{\|a\| \, \|b\|}\right)$$

a) 1D example:



Distance nearest neighbor:

Distance nearest neighbor:

$$\|x - z_1\| = |-1 - 10| = 11$$

$$\|x - z_2\| = |-1 - (-15)| = 14$$

$\therefore$ z_1 is the distance nearest neighbor

Angle nearest neighbor

Angle nearest neighbor

$$\theta_{z_1} = \arccos\left(\frac{x^T z_1}{\|x\| \|z_1\|}\right)$$

$$= \arccos\left(\frac{(-1)(10)}{|-1||10|}\right)$$

$$= \arccos(-1)$$

$$= \pi$$

$$\theta_{z_2} = \arccos\left(\frac{x^T z_2}{\|x\| \|z_2\|}\right)$$

$$= \arccos\left(\frac{(-1)(-15)}{|-1| |-15|}\right)$$

$$= \arccos(1)$$

$$= 0$$

$\therefore$ z_2 is the angle nearest neighbor.

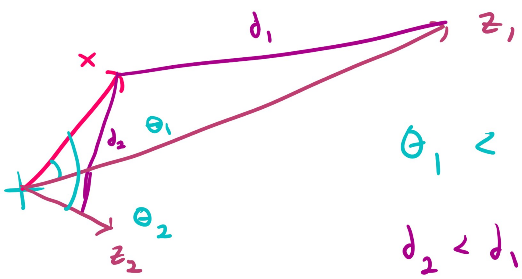
In this example the distance nearest neighbor was different than the angle nearest neighbor

b) 2D example:

For non-normalized vectors, the distance and angle nearest neighbors will be different when the magnitude of the angle nearest

When the magnitude of the angle nearest vector is less than another z vector

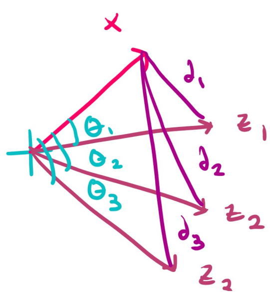
i.e.



$\theta_1 < \theta_2 \rightarrow z_1$ is angle nearest neighbor

$d_2 < d_1 \rightarrow z_2$ is distance nearest neighbor

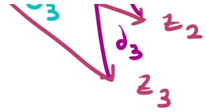
But if all the z_i vectors are normalized



$d_1 < d_2$ & d_3 &

$\theta_1 < \theta_2 < \theta_3$

So z_1 is both the angle



So z_1 is both the angle and distance nearest neighbor

Hint Property:

If $-1 \leq u < v \leq 1$, then,

$$\arccos u \geq \arccos(v)$$

If $\|z_i\| = 1$ for $\forall\, i \ldots m$ then

$$\theta_i = \arccos\left(\frac{x^T z_i}{\quad}\right)$$

$$\theta_i = \arccos\left(\frac{x^T z_i}{\|x\|\|z_i\|}\right)$$

$$= \arccos\left(\frac{x^T z_i}{\|x\|}\right)$$

$\theta_1 = \arccos\left(\frac{x^T z_1}{\|x\|}\right)$ and $z_2 = \arccos\left(\frac{x^T z_2}{\|x\|}\right)$

the only difference being the value of $x^T z_i$

If $\theta_1 < \theta_2 \rightarrow$ so z_1 is the angle nearest neighbor

$$\arccos\left(\frac{x^T z_1}{\|x\|}\right) < \arccos\left(\frac{x^T z_2}{\|x\|}\right)$$

$$\arccos\left(\frac{x^T z_1}{\|x\|}\right) < \arccos\left(\frac{x^T z_2}{\|x\|}\right)$$

which based on the property above means

$$-1 \leq \frac{x^T z_2}{\|x\|} < \frac{x^T z_1}{\|x\|} \leq 1$$

which means $z_2 < z_1$

$$z_2 < z_1$$

$$-z_2 > -z_1$$

inequality laws with multiplying or dividing by negative #s

$$\| -z_2 + x \| > \| -z_1 + x \|$$

$$\| x - z_2 \| > \| x - z_1 \|$$

$$\| x - z_2 \| > \| x - z_1 \|$$

$\therefore$ z_1 is also the distance nearest neighbor.

Problem 1.7

September 26, 2023 12:49 AM

Question a)

Closest Euclidean Pair : Article 6 & Article 8

Closest Angle Pair : Article 5 & Article 8

Question b)

Question b)

Closest Euclidean Pair : Article 5 & Article 8

Closest Angle Pair : Article 5 & Article 8

By normalizing the angle & euclidean distances match similar to the problem set question. Normalizing the features allows for more accurate comparison between the articles because it means the articles with more words are not weighted more than articles with fewer words.

367_Home...

Question a)

```
disp("Question A)")
load 'wordVecV.mat' % goes into a variable V

% disp(V)

% Euclidiean Distance
disp("Euclidean Distance")
euclidean_distances = zeros(10,10);
max_coord = zeros(1,2);
max = 0;
for i = 1:10
    for j = 1:10
        euclidean_distances(i,j) = euclidean_distance(V(:, i), V(:, j));

        if euclidean_distances(i,j) > max
            max = euclidean_distances(i,j);
            max_coord(1) = i;
            max_coord(2) = j;
        end
    end
end

disp(euclidean_distances)
disp(max)
disp(max_coord)

disp("Closest euclidean distance: Article: " + max_coord(1) + " and Article: 
" + max_coord(2))

% Angle Distance
disp("Angle Distance")
angle_distances = zeros(10,10);
max_coord = zeros(1,2);
max = 0;
for i = 1:10
    for j = 1:10
        angle_distances(i,j) = angle_distance(V(:, i), V(:, j));

        if angle_distances(i,j) > max
            max = angle_distances(i,j);
            max_coord(1) = i;
            max_coord(2) = j;
        end
    end
end

disp(angle_distances)
disp(max)
disp(max_coord)
```

→ looking for farthest distance

```
disp("Closest angle distance: Article: " + max_coord(1) + " and Article: " +
max_coord(2))

% Question b)
disp("Question b")

% Euclidiean Distance
disp("Euclidean Distance")
euclidean_distances = zeros(10,10);
max_coord = zeros(1,2);
max = 0;
for i = 1:10
    for j = 1:10
        euclidean_distances(i,j) = euclidean_distance(V(:, i)/sum(V(:, i)),
V(:, j)/sum(V(:, j)));

        if euclidean_distances(i,j) > max
            max = euclidean_distances(i,j);
            max_coord(1) = i;
            max_coord(2) = j;
        end
    end
end

disp(euclidean_distances)
disp(max)
disp(max_coord)

disp("Closest euclidean distance: Article: " + max_coord(1) + " and Article:
" + max_coord(2))

% Angle Distance
disp("Angle Distance")
angle_distances = zeros(10,10);
max_coord = zeros(1,2);
max = 0;
for i = 1:10
    for j = 1:10
        angle_distances(i,j) = angle_distance(V(:, i)/sum(V(:, i)), V(:, j)/
sum(V(:, j)));

        if angle_distances(i,j) > max
            max = angle_distances(i,j);
            max_coord(1) = i;
            max_coord(2) = j;
        end
    end
end

disp(angle_distances)
disp(max)
disp(max_coord)

disp("Closest angle distance: Article: " + max_coord(1) + " and Article: " +
```

```
max_coord(2))

% Question C
disp("Question C")

% Get f_doc
f_doc = zeros(1,1651);

for i = 1:length(f_doc)
    num_articles = 0;
    for j = 1:10
        if V(i, j) ~= 0
            num_articles = num_articles +1;
        end
    end
    f_doc(i) = num_articles;
end

% get w(t,d)
w_t_d = zeros(1651,10);
for i = 1:10
    for j = 1:1651
        w_t_d(j,i) = (V(j,i) / sum(V(:,i))) * sqrt(log(10/f_doc(j)));
    end
end

% Euclidiean Distance
disp("Euclidean Distance")
euclidean_distances = zeros(10,10);
max_coord = zeros(1,2);
max = 0;
for i = 1:10
    for j = 1:10
        euclidean_distances(i,j) = euclidean_distance(w_t_d(:, i), w_t_d(:,
j));

        if euclidean_distances(i,j) > max
            max = euclidean_distances(i,j);
            max_coord(1) = i;
            max_coord(2) = j;
        end
    end
end

disp(euclidean_distances)
disp(max)
disp(max_coord)

disp("TF-IDF: Closest euclidean distance: Article: " + max_coord(1) + " and
Article: " + max_coord(2))

% Angle Distance
disp("Angle Distance")
angle_distances = zeros(10,10);
```

```
max_coord = zeros(1,2);
max = 0;
for i = 1:10
    for j = 1:10
        angle_distances(i,j) = angle_distance(w_t_d(:, i), w_t_d(:, j));

        if angle_distances(i,j) > max
            max = angle_distances(i,j);
            max_coord(1) = i;
            max_coord(2) = j;
        end
    end
end

disp(angle_distances)
disp(max)
disp(max_coord)

disp("TF_IDF: Closest angle distance: Article: " + max_coord(1) + " and
Article: " + max_coord(2))
```

Why use TF-IDF?

TF-IDF scales the data by how much each word shows up in all the documents. This means that similar to normlization, words from larger documents aren't disproportionaly weighted. It also makes sure that rare words that are only in a few documents aren't unfairly biased against jut because they are a rare word.

↳ Could say more about common words not showing difference

```
function euc_dist = euclidean_distance(x,y)
    euc_dist = sqrt(sum((x-y).^2));
end

function ang = angle_distance(x,y)
    ang = acos((sum(x.*y))/(norm(x)*norm(y)));
end

function norm = norm(x)
    norm = sqrt(sum(x.^2));
end
```

```
Question A)
Euclidean Distance
  Columns 1 through 7

         0   32.2955   35.8050   44.3959   32.2025   55.2540   29.0000
   32.2955         0   34.7707   42.1426   35.3553   53.3292   33.1361
   35.8050   34.7707         0   46.6154   33.6601   56.4535   37.2961
   44.3959   42.1426   46.6154         0   50.2792   51.3809   46.3681
   32.2025   35.3553   33.6601   50.2792         0   59.7662   29.0517
   55.2540   53.3292   56.4535   51.3809   59.7662         0   57.6541
   29.0000   33.1361   37.2961   46.3681   29.0517   57.6541         0
   31.0805   34.9428   37.9737   50.1498   30.9031   62.5700   24.7184
   48.1560   45.9347   45.2217   51.7301   48.5592   50.5569   50.5371
   40.3485   39.7366   40.1497   48.0729   41.9881   51.9326   42.2493
```

```
  Columns 8 through 10

   31.0805   48.1560   40.3485
   34.9428   45.9347   39.7366
   37.9737   45.2217   40.1497
   50.1498   51.7301   48.0729
   30.9031   48.5592   41.9881
   62.5700   50.5569   51.9326
   24.7184   50.5371   42.2493
         0   53.3573   43.2435
   53.3573         0   34.3948
   43.2435   34.3948         0

   62.5700

     6     8

Closest euclidean distance: Article: 6 and Article: 8
Angle Distance
  Columns 1 through 4

   0.0000 + 0.0000i   0.7779 + 0.0000i   0.8375 + 0.0000i   0.8407 + 0.0000i
   0.7779 + 0.0000i   0.0000 + 0.0000i   0.7762 + 0.0000i   0.7839 + 0.0000i
   0.8375 + 0.0000i   0.7762 + 0.0000i   0.0000 + 0.0000i   0.8760 + 0.0000i
   0.8407 + 0.0000i   0.7839 + 0.0000i   0.8760 + 0.0000i   0.0000 + 0.0000i
   0.8835 + 0.0000i   0.8853 + 0.0000i   0.7958 + 0.0000i   1.0011 + 0.0000i
   0.7751 + 0.0000i   0.7583 + 0.0000i   0.8337 + 0.0000i   0.7399 + 0.0000i
   0.8143 + 0.0000i   0.8331 + 0.0000i   0.9138 + 0.0000i   0.8792 + 0.0000i
   0.9034 + 0.0000i   0.8957 + 0.0000i   0.9411 + 0.0000i   0.9905 + 0.0000i
   0.7736 + 0.0000i   0.7396 + 0.0000i   0.7287 + 0.0000i   0.8278 + 0.0000i
   0.7483 + 0.0000i   0.7381 + 0.0000i   0.7443 + 0.0000i   0.8345 + 0.0000i

  Columns 5 through 8

   0.8835 + 0.0000i   0.7751 + 0.0000i   0.8143 + 0.0000i   0.9034 + 0.0000i
   0.8853 + 0.0000i   0.7583 + 0.0000i   0.8331 + 0.0000i   0.8957 + 0.0000i
   0.7958 + 0.0000i   0.8337 + 0.0000i   0.9138 + 0.0000i   0.9411 + 0.0000i
   1.0011 + 0.0000i   0.7399 + 0.0000i   0.8792 + 0.0000i   0.9905 + 0.0000i
   0.0000 + 0.0000i   0.8721 + 0.0000i   0.8925 + 0.0000i   1.0032 + 0.0000i
   0.8721 + 0.0000i   0.0000 + 0.0000i   0.7648 + 0.0000i   0.8864 + 0.0000i
   0.8925 + 0.0000i   0.7648 + 0.0000i   0.0000 + 0.0000i   0.8625 + 0.0000i
   1.0032 + 0.0000i   0.8864 + 0.0000i   0.8625 + 0.0000i   0.0000 + 0.0000i
   0.7587 + 0.0000i   0.7098 + 0.0000i   0.7819 + 0.0000i   0.8305 + 0.0000i
   0.7788 + 0.0000i   0.7498 + 0.0000i   0.7602 + 0.0000i   0.7447 + 0.0000i

  Columns 9 through 10

   0.7736 + 0.0000i   0.7483 + 0.0000i
   0.7396 + 0.0000i   0.7381 + 0.0000i
   0.7287 + 0.0000i   0.7443 + 0.0000i
   0.8278 + 0.0000i   0.8345 + 0.0000i
   0.7587 + 0.0000i   0.7788 + 0.0000i
   0.7098 + 0.0000i   0.7498 + 0.0000i
   0.7819 + 0.0000i   0.7602 + 0.0000i
```

Incorrect Articles

```
   0.8305 + 0.0000i   0.7447 + 0.0000i
   0.0000 + 0.0000i   0.5327 + 0.0000i
   0.5327 + 0.0000i   0.0000 + 0.0000i

    1.0032

     5     8

Closest angle distance: Article: 5 and Article: 8
Question b
Euclidean Distance
  Columns 1 through 7

         0    0.0852    0.0977    0.0857    0.1081    0.0860    0.0988
    0.0852         0    0.0930    0.0841    0.1101    0.0866    0.1025
    0.0977    0.0930         0    0.1002    0.1031    0.1000    0.1157
    0.0857    0.0841    0.1002         0    0.1188    0.0809    0.1045
    0.1081    0.1101    0.1031    0.1188         0    0.1092    0.1171
    0.0860    0.0866    0.1000    0.0809    0.1092         0    0.0952
    0.0988    0.1025    0.1157    0.1045    0.1171    0.0952         0
    0.1060    0.1075    0.1172    0.1131    0.1289    0.1071    0.1106
    0.0895    0.0877    0.0903    0.0938    0.0979    0.0849    0.0992
    0.0826    0.0840    0.0896    0.0896    0.0983    0.0861    0.0945

  Columns 8 through 10

    0.1060    0.0895    0.0826
    0.1075    0.0877    0.0840
    0.1172    0.0903    0.0896
    0.1131    0.0938    0.0896
    0.1289    0.0979    0.0983
    0.1071    0.0849    0.0861
    0.1106    0.0992    0.0945
         0    0.1033    0.0909
    0.1033         0    0.0643
    0.0909    0.0643         0

    0.1289

     5     8

Closest euclidean distance: Article: 5 and Article: 8
Angle Distance
  Columns 1 through 4

   0.0000 + 0.0000i   0.7779 + 0.0000i   0.8375 + 0.0000i   0.8407 + 0.0000i
   0.7779 + 0.0000i   0.0000 + 0.0000i   0.7762 + 0.0000i   0.7839 + 0.0000i
   0.8375 + 0.0000i   0.7762 + 0.0000i   0.0000 + 0.0000i   0.8760 + 0.0000i
   0.8407 + 0.0000i   0.7839 + 0.0000i   0.8760 + 0.0000i   0.0000 + 0.0000i
   0.8835 + 0.0000i   0.8853 + 0.0000i   0.7958 + 0.0000i   1.0011 + 0.0000i
   0.7751 + 0.0000i   0.7583 + 0.0000i   0.8337 + 0.0000i   0.7399 + 0.0000i
   0.8143 + 0.0000i   0.8331 + 0.0000i   0.9138 + 0.0000i   0.8792 + 0.0000i
   0.9034 + 0.0000i   0.8957 + 0.0000i   0.9411 + 0.0000i   0.9905 + 0.0000i
   0.7736 + 0.0000i   0.7396 + 0.0000i   0.7287 + 0.0000i   0.8278 + 0.0000i
```

```
  0.7483 + 0.0000i   0.7381 + 0.0000i   0.7443 + 0.0000i   0.8345 + 0.0000i

 Columns 5 through 8

  0.8835 + 0.0000i   0.7751 + 0.0000i   0.8143 + 0.0000i   0.9034 + 0.0000i
  0.8853 + 0.0000i   0.7583 + 0.0000i   0.8331 + 0.0000i   0.8957 + 0.0000i
  0.7958 + 0.0000i   0.8337 + 0.0000i   0.9138 + 0.0000i   0.9411 + 0.0000i
  1.0011 + 0.0000i   0.7399 + 0.0000i   0.8792 + 0.0000i   0.9905 + 0.0000i
  0.0000 + 0.0000i   0.8721 + 0.0000i   0.8925 + 0.0000i   1.0032 + 0.0000i
  0.8721 + 0.0000i   0.0000 + 0.0000i   0.7648 + 0.0000i   0.8864 + 0.0000i
  0.8925 + 0.0000i   0.7648 + 0.0000i   0.0000 + 0.0000i   0.8625 + 0.0000i
  1.0032 + 0.0000i   0.8864 + 0.0000i   0.8625 + 0.0000i   0.0000 + 0.0000i
  0.7587 + 0.0000i   0.7098 + 0.0000i   0.7819 + 0.0000i   0.8305 + 0.0000i
  0.7788 + 0.0000i   0.7498 + 0.0000i   0.7602 + 0.0000i   0.7447 + 0.0000i

 Columns 9 through 10

  0.7736 + 0.0000i   0.7483 + 0.0000i
  0.7396 + 0.0000i   0.7381 + 0.0000i
  0.7287 + 0.0000i   0.7443 + 0.0000i
  0.8278 + 0.0000i   0.8345 + 0.0000i
  0.7587 + 0.0000i   0.7788 + 0.0000i
  0.7098 + 0.0000i   0.7498 + 0.0000i
  0.7819 + 0.0000i   0.7602 + 0.0000i
  0.8305 + 0.0000i   0.7447 + 0.0000i
  0.0000 + 0.0000i   0.5327 + 0.0000i
  0.5327 + 0.0000i   0.0000 + 0.0000i

   1.0032

     5     8

Closest angle distance: Article: 5 and Article: 8
Question C
Euclidean Distance
 Columns 1 through 7

        0    0.1075    0.1169    0.1043    0.1176    0.0978    0.1131
   0.1075         0    0.1201    0.1039    0.1251    0.1050    0.1207
   0.1169    0.1201         0    0.1148    0.1312    0.1167    0.1265
   0.1043    0.1039    0.1148         0    0.1217    0.0983    0.1166
   0.1176    0.1251    0.1312    0.1217         0    0.1167    0.1246
   0.0978    0.1050    0.1167    0.0983    0.1167         0    0.1100
   0.1131    0.1207    0.1265    0.1166    0.1246    0.1100         0
   0.1210    0.1254    0.1327    0.1246    0.1344    0.1201    0.1303
   0.0990    0.1071    0.1143    0.1027    0.1122    0.0972    0.1097
   0.0968    0.1051    0.1130    0.1016    0.1118    0.0960    0.1073

 Columns 8 through 10

   0.1210    0.0990    0.0968
   0.1254    0.1071    0.1051
   0.1327    0.1143    0.1130
   0.1246    0.1027    0.1016
```

```
  0.1344    0.1122    0.1118
  0.1201    0.0972    0.0960
  0.1303    0.1097    0.1073
       0    0.1136    0.0965
  0.1136         0    0.0821
  0.0965    0.0821         0

  0.1344

   5     8

TF-IDF: Closest euclidean distance: Article: 5 and Article: 8
Angle Distance
  Columns 1 through 4

   0.0000 + 0.0000i   1.4965 + 0.0000i   1.5440 + 0.0000i   1.5177 + 0.0000i
   1.4965 + 0.0000i   0.0000 + 0.0000i   1.4857 + 0.0000i   1.3853 + 0.0000i
   1.5440 + 0.0000i   1.4857 + 0.0000i   0.0000 + 0.0000i   1.4603 + 0.0000i
   1.5177 + 0.0000i   1.3853 + 0.0000i   1.4603 + 0.0000i   0.0000 + 0.0000i
   1.5269 + 0.0000i   1.5389 + 0.0000i   1.5422 + 0.0000i   1.5470 + 0.0000i
   1.4762 + 0.0000i   1.4659 + 0.0000i   1.5532 + 0.0000i   1.4237 + 0.0000i
   1.5354 + 0.0000i   1.5431 + 0.0000i   1.5378 + 0.0000i   1.5424 + 0.0000i
   1.5464 + 0.0000i   1.5104 + 0.0000i   1.5346 + 0.0000i   1.5575 + 0.0000i
   1.5344 + 0.0000i   1.5342 + 0.0000i   1.5417 + 0.0000i   1.5382 + 0.0000i
   1.5121 + 0.0000i   1.5149 + 0.0000i   1.5327 + 0.0000i   1.5362 + 0.0000i

  Columns 5 through 8

   1.5269 + 0.0000i   1.4762 + 0.0000i   1.5354 + 0.0000i   1.5464 + 0.0000i
   1.5389 + 0.0000i   1.4659 + 0.0000i   1.5431 + 0.0000i   1.5104 + 0.0000i
   1.5422 + 0.0000i   1.5532 + 0.0000i   1.5378 + 0.0000i   1.5346 + 0.0000i
   1.5470 + 0.0000i   1.4237 + 0.0000i   1.5424 + 0.0000i   1.5575 + 0.0000i
   0.0000 + 0.0000i   1.5250 + 0.0000i   1.4864 + 0.0000i   1.5389 + 0.0000i
   1.5250 + 0.0000i   0.0000 + 0.0000i   1.4940 + 0.0000i   1.5437 + 0.0000i
   1.4864 + 0.0000i   1.4940 + 0.0000i   0.0000 + 0.0000i   1.5409 + 0.0000i
   1.5389 + 0.0000i   1.5437 + 0.0000i   1.5409 + 0.0000i   0.0000 + 0.0000i
   1.4754 + 0.0000i   1.5170 + 0.0000i   1.5194 + 0.0000i   1.4607 + 0.0000i
   1.4825 + 0.0000i   1.5139 + 0.0000i   1.4907 + 0.0000i   1.1948 + 0.0000i

  Columns 9 through 10

   1.5344 + 0.0000i   1.5121 + 0.0000i
   1.5342 + 0.0000i   1.5149 + 0.0000i
   1.5417 + 0.0000i   1.5327 + 0.0000i
   1.5382 + 0.0000i   1.5362 + 0.0000i
   1.4754 + 0.0000i   1.4825 + 0.0000i
   1.5170 + 0.0000i   1.5139 + 0.0000i
   1.5194 + 0.0000i   1.4907 + 0.0000i
   1.4607 + 0.0000i   1.1948 + 0.0000i
   0.0000 + 0.0000i   1.2867 + 0.0000i
   1.2867 + 0.0000i   0.0000 + 0.0000i

  1.5575
```

```
     4      8
```

```
TF_IDF: Closest angle distance: Article: 4 and Article: 8
```

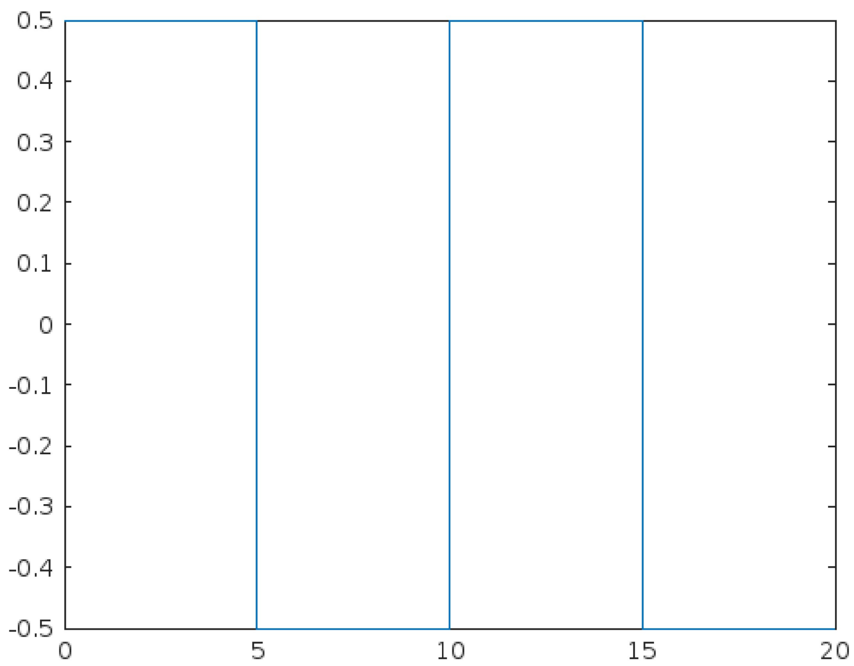
Published with MATLAB® R2023b

Homework...

Table of Contents

Question A)

```
[ time_pos , sq_wave , B_unnorm ] = generate_data;

% Plot
plot(time_pos, sq_wave)
```

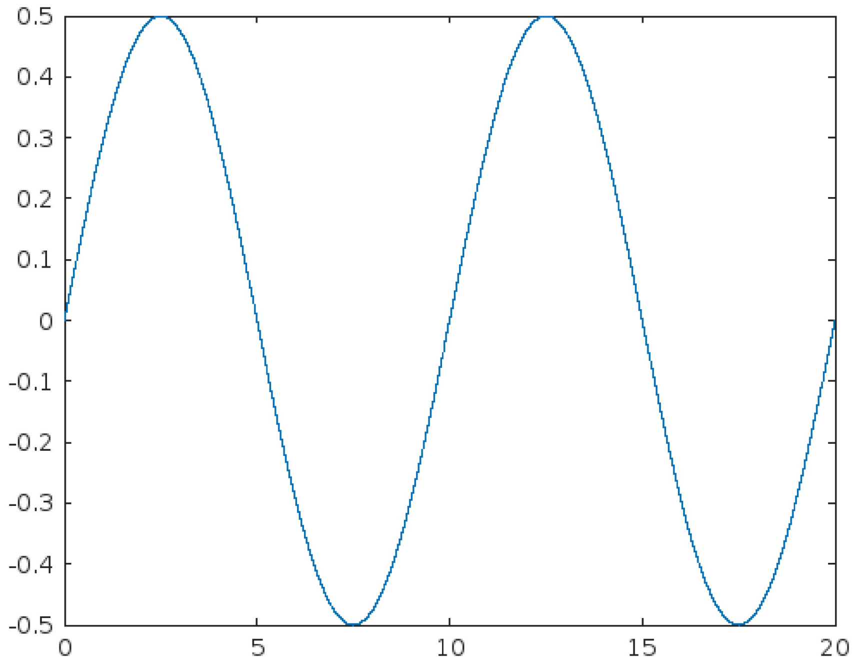


Question B)

To test the orthoganality of the basis vectors, I would multiply the two functions in a dot product. However since they are functions I would be taking the integral of the two functions multilpied. But since the functions are actually

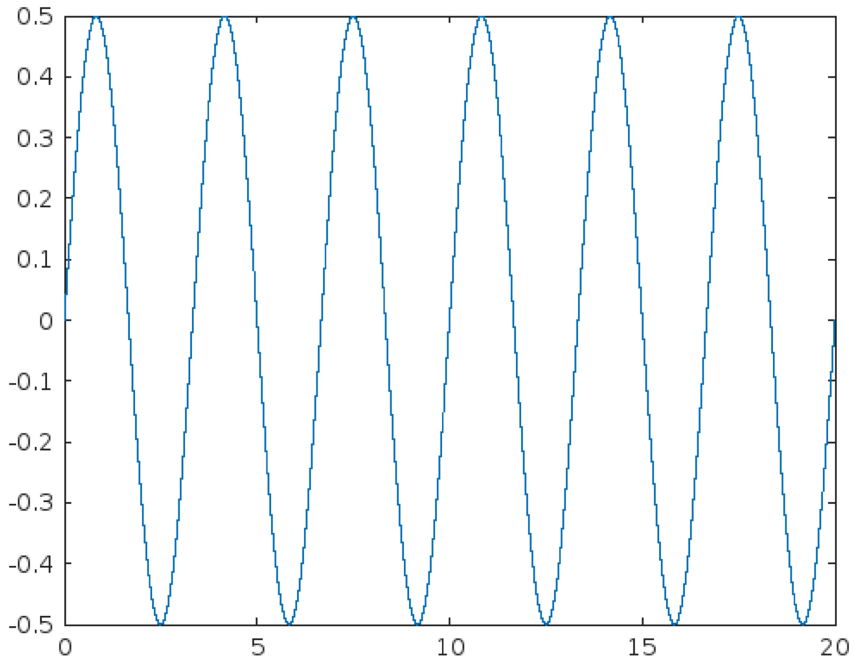
```
disp('B1')
plot(time_pos, B_unnorm(1,:))
```

B1



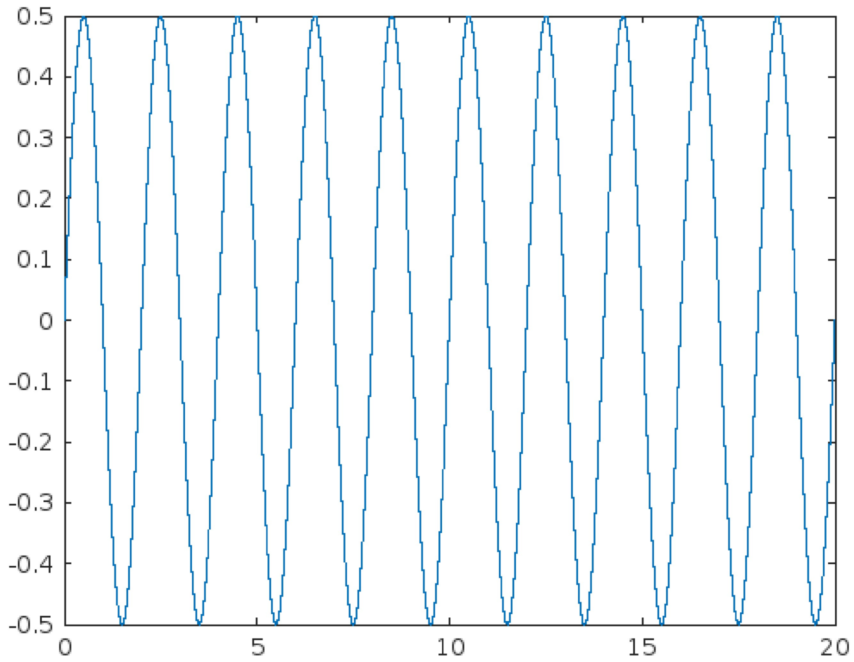

```
disp('B2')
plot(time_pos, B_unnorm(2,:))
```

B2



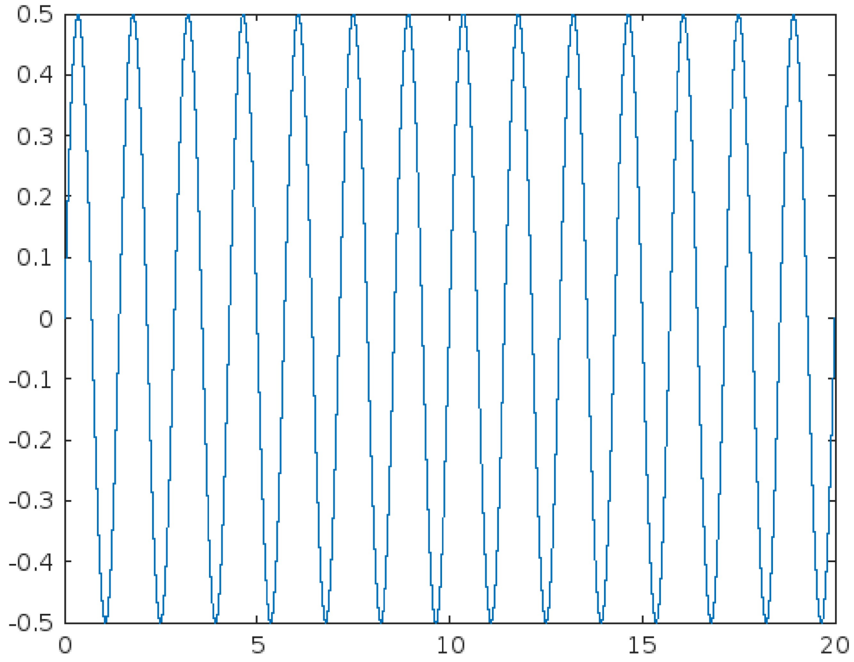

```
disp('B3')
plot(time_pos, B_unnorm(3,:))
```

B3



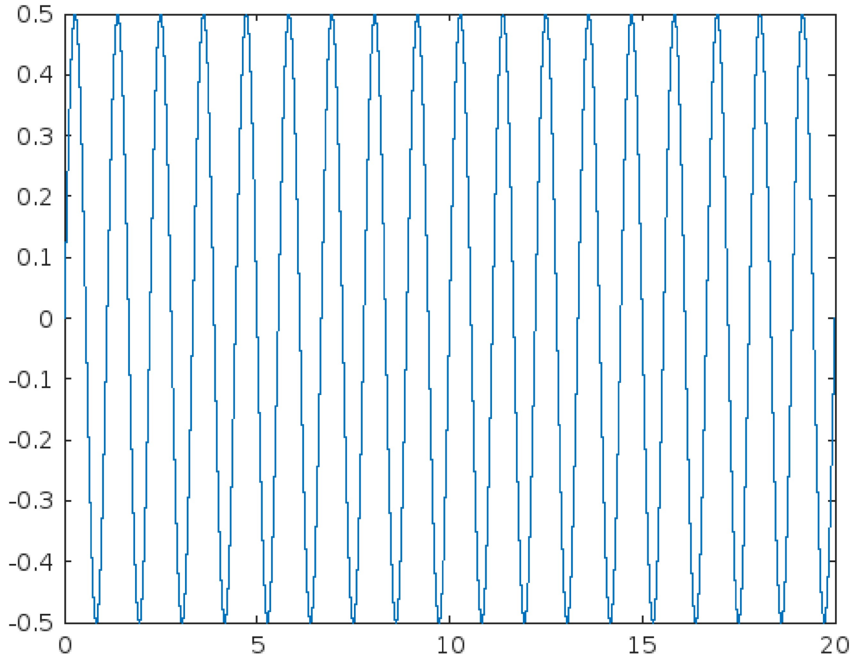

```
disp('B4')
plot(time_pos, B_unnorm(4,:))
```

B4



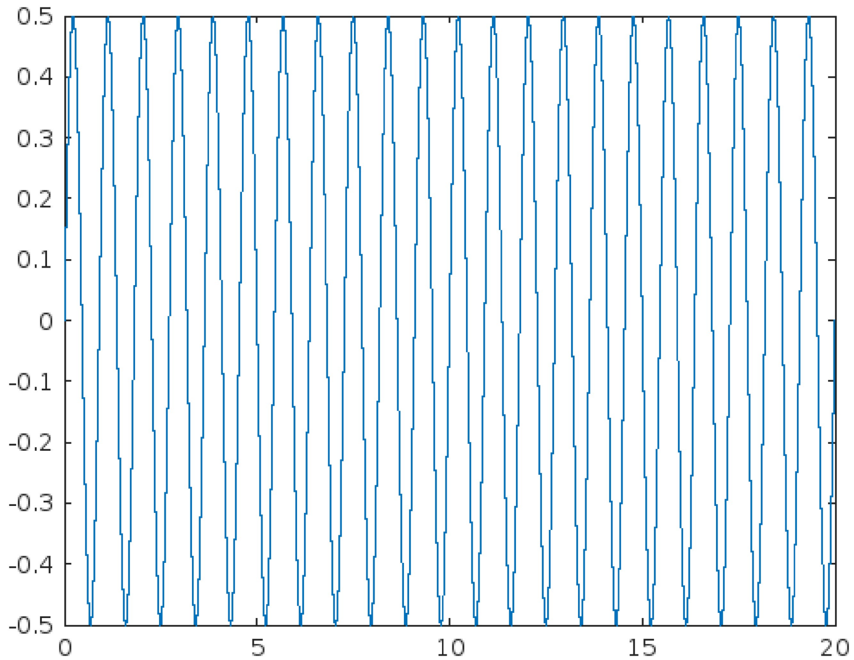

```
disp('B5')
plot(time_pos, B_unnorm(5,:))
```

B5




```
disp('B6')
plot(time_pos, B_unnorm(6,:))
```

B6



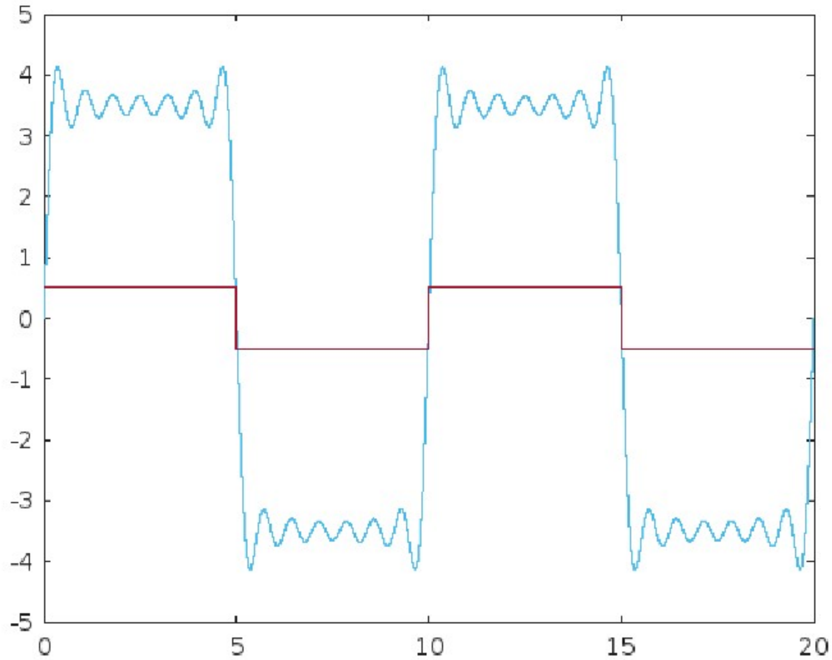
Question C)

Orthonormal Basis

```
B_norm = zeros(30,200001);
for i = 1:30
    B_norm(i,:) = B_unnorm(i,:)./(sqrt(sum(B_unnorm(i,:).^2)));
end

% l2 Projection
proj = zeros(30,200001);
for i = 1:30
    proj(i,:) = sum(sq_wave.*(B_norm(i,:))) .* B_norm(i,:);
end

plot(time_pos, proj(1,:));
plot(time_pos, proj(2,:));
plot(time_pos, proj(3,:));
plot(time_pos, proj(4,:));
plot(time_pos, proj(5,:));
plot(time_pos, proj(6,:));
plot(time_pos, proj(30,:));

for i = 1:7
    projection = projection + proj(i,:);
```

```
end

disp("With " + 7 + " Basis Vectors")
plot(time_pos, projection, time_pos,sq_wave)

function [ time_pos , sq_wave , B_unnorm ] = generate_data
    n_comps = 30; period = 10; fundFreq = 1/ period ;
    time_pos = 0:0.0001:2* period ; harmonics = 2*(1: n_comps ) -1;
    sq_wave = floor (0.9* sin (2* pi * fundFreq * time_pos ) ) +.5; % %
generate the signal
    B_unnorm = sin (2* pi * fundFreq *( harmonics .' * time_pos ) ) /2; % %
generate the basis
end
```

With 7 Basis Vectors



Published with MATLAB® R2023b

Not enough graphs shown

→ more approximations should be shown