Math 108C Homework 2

Rad Mallari, 8360828

April 12th, 2022

Exercise 3) Show that $||u+v||_2 \leq ||u||_2 + ||v||_2$, using Cauchy-Schwarz inequality.

Proof:

The Cauchy-Schwarz inequality states that the dot product satisfies:

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

Using this we get

$$\begin{aligned}(||u+v||_2)^2 &= (u+v)^T(u+v) \\ &= u^T y + u^T v + v^T u + v^T v \\ &= (||u||_2)^2 + 2u^T v + (||v||_2)^2) \\ &\leq ||u||_2)^2 + 2||u||_2||v||_2 + (||v||_2)^2 = (||u||_2 + ||v||_2)^2\end{aligned} \tag{1}$$

Exercise 4)

(a) Consider the vectors in $\mathbb{R}^3$ given by $[1,2,0]^T$, $[0,1,1]^T$, and $[0,0,3]^T$.
Which of these vectors is closest to $u = [1,-2,1]^T$ using Euclidean distance?
Explain your answer.

(b) What is the angle between the vectors $[1,0,3]^T$ and $[0,1,1]^T$?

Proof:

(a) The Euclidean distance of some vectors $u, v \in \mathbb{R}^n$ is defined as

$$||u-v||_2 = \sqrt{(u_1-v_1)^2 + (u_2-v_2)^2 + \cdots + (u_n-v_n)^2}$$

Therefore, by definition, the closest vector to u is the vector $[0,0,3]^T$ since the distance is
$\sqrt{(1-0)^2 + (-2-1)^2 + (1-3)^2} = \sqrt{9} = 3$ and the magnitude between the two vectors is the least.

(b) The angle between the vectors $[1,0,3]^T$ and $[0,1,1]^T$ can be found using the cosine angle given as:

$$\cos(\theta(u,v)) = \frac{u^T v}{||u||_2||v||_2} \implies \theta(u,v) = \cos^{-1}\left(\frac{u^T v}{||u||_2||v||_2}\right)$$

Therefore, the angle θ is

$$\theta = \cos^{-1}\left(\frac{\sqrt{6}}{6}\right) = 65.9^\circ$$

Exercise 7) Construct a random set consisting of points within a circle centered at $(3,4)$ in the xy-plane with radius 2, call the points in this set Class 1.
Constuct another random set consisting of points inside a circle centered at $(0,0)$ with radius $\frac{3}{2}$, call the points in this set Class 2.
Write a Python program that implements kNN algorithm to classify points in Class 1 or 2.

In [6]:
```
from math import sin, cos, pi, sqrt
from random import random


def generate_random_point_circle(r, center):
    r, theta = 2 * sqrt(random()), random() * 2 * pi
    return center[0] + (r * cos(theta)), center[1] + (r * sin(theta))

def find_Eucl_dist():
    # find euclidean distance between points
    pass

r_1, center_1 = 2, (3,4)
r_2, center_2 = 3/2, (0,0)

class_1 = [generate_random_point_circle(r_1, center_1) for i in range(1000)]
class_2 = [generate_random_point_circle(r_2, center_2) for i in range(1000)]
```

In [ ]:
```

```