

# COSC 343: Test 1

Micah Sherry

February 22, 2024

## 1 vector P-Norm

Listing 1: vector p-norm

```
import numpy as np
import matplotlib

def p_norm(vec, p):
    sum = 0
    for element in vec:
        sum += np.abs(element) ** p
    return sum ** (1/p)

vec = [3,4]
p = 2
pnorm = p_norm(vec,p)
print("the", p, "norm of", vec, "is", pnorm)
```

## 2 Matrix 1 norm

Listing 2: matrix 1-norm

```
import numpy as np
import matplotlib

def one_norm(matrix):
    max = 0
    for j in range(len(matrix[0])):
        sum = 0
        for i in range(len(matrix)):
            sum += np.abs(matrix[i][j])
        if sum > max:
            max = sum
    return max

def inf_norm(matrix):
    max = 0
    for i in range(len(matrix)):
        sum = 0
        for j in range(len(matrix[0])):
            sum += np.abs(matrix[i][j])
        if sum > max:
            max = sum
```

```
return max
```

```
matrix = [[1,2,0],
          [1,0,0],
          [1,0,0],
          [1,0,0]]
#matrix = np.zeros(9).reshape(3,3)

print(inf_norm(matrix))
```

### 3 Matrix $\infty$ norm

Listing 3: matrix *infty* norm

```
import numpy as np
import matplotlib

def one_norm(matrix):
    max = 0
    for j in range(len(matrix[0])):
        sum = 0
        for i in range(len(matrix)):
            sum += np.abs(matrix[i][j])
        if sum > max:
            max = sum
    return max

def inf_norm(matrix):
    max = 0
    for i in range(len(matrix)):
        sum = 0
        for j in range(len(matrix[0])):
            sum += np.abs(matrix[i][j])
        if sum > max:
            max = sum
    return max
```

```
matrix = [[1,2,0],
          [1,0,0],
          [1,0,0],
          [1,0,0]]
#matrix = np.zeros(9).reshape(3,3)

print(inf_norm(matrix))
```