**Data Mining Lab**

**Week 1:**

**(distance measures and proximity matrices for numeric, nominal and binary data)**

1. **Numeric data**:

dataFile = open('./training.csv')

data = []

for line in dataFile:

data.append([int(x) for x in line.split(',')])

#proximmity measures

def euclid(tup1, tup2):

dist = 0

for index in range(len(tup1)):

dist += (tup1[index] - tup2[index])\*\*2

return dist\*\*0.5

def manhattan(tup1, tup2):

dist = 0

for index in range(len(tup1)):

dist += abs(tup1[index] - tup2[index])

return dist

def supremum(tup1, tup2):

dist = max([tup1[index] - tup2[index] for index in range(len(tup1))])

return dist

def sumOfSquares(row):

dist = 0

for val in row:

dist += (val\*\*2)

return dist\*\*0.5

def cosine(tup1, tup2):

numerator = sum([tup1[index]\*tup2[index] for index in range(len(tup1))])

denominator = sumOfSquares(tup1)\*sumOfSquares(tup2)

return numerator/denominator

#function to print proximity matrix

def printProxMatrix(data, function):

for i in range(len(data)):

for j in range(len(data)):

print(str(function(data[i],data[j])), end = ' ')

print("")

printProxMatrix(data, euclid)

printProxMatrix(data, manhattan)

printProxMatrix(data, supremum)

printProxMatrix(data, cosine)