John Soderstrom
CS 4320
Project 1

In copying my program output, I realized it did not seem very readable due to the large quantity of information. I added a function to print it to a csv format, though it is not as generalized as the main print functionality. I had to limit the fitness value precision to fit the tables in the report, but they should still present enough information and the original values are still in the original program output. I also added the csv printout to this report.

Begin Source Code

```
import random
from pprint import pprint as pp
from copy import deepcopy
from collections import OrderedDict as od

#Stores a dictionary of runs labeled by run number
#Will print data from all runs at once
class RunCollection:
    #Initialize with the number of runs, issuing a warning if the number is exceeded
    #Sets up a dictionary to store individual run data
    def __init__(self, maxRuns):
        self.maxRuns = maxRuns
        self.currentRun = 0
        self.allRunData = od()

    #Adds an empty run to the dictionary if there is room and returns it
    #Issues a warning if another would exceed the maximum number
    #Increments currentRun so it always points to the most recent
    def addAndUseRun(self, numGens, randomSeed, isMaxFitness = True):
        if self.currentRun < self.maxRuns:
            newRun = SingleRunResults(numGens, randomSeed, isMaxFitness)
            self.currentRun += 1
            self.allRunData['Run ' + str(self.currentRun)] = newRun
            return self.allRunData['Run ' + str(self.currentRun)]
        else:
            print("Warning: more runs than expected, user wanted max of %d runs." %
```

```
self.numRuns)
            return None

    #Allows access to any run added before the current run in case it is needed
    def getRunNum(self, num):
        if num > 0 and num <= self.currentRun:
            return self.allRunData['Run ' + str(num)]
        else:
            print("Warning: run number %d does not exist." % num)
            return None

    #Gets the maximum number of runs allowed
    def getMaxRuns(self):
        return self.maxRuns

    #Gets the current number of runs stored
    def getCurrentRun(self):
        return self.currentRun

    #Calculate averages for data across all runs
    def prepareAverages(self):
        self.averageRunData = {}
        avgOfAvgGen = {}
        avgOfBestGen = {}
        bestRuns = []

        #Initializes values for all generations stored in runs
        #Required so we may increment them in one loop
        for i in self.allRunData['Run 1'].getGenResults():
            avgOfAvgGen[i] = 0
            avgOfBestGen[i] = 0

        #Increments values for all generations from each run, stored in a dictionary
        #and labeled by generation number, for averages across generations
        for key, run in self.allRunData.items():
            tempData = run.getGenResults()
            for genKey, value in tempData.items():
                avgOfAvgGen[genKey] += value['Average Fitness']
                #Checks if we wanted to minimize or maximize fitness before incrementing
                #with the correct value
                if run.getIsMaxFitness():
                    avgOfBestGen[genKey] += value['High Fitness']
                else:
                    avgOfBestGen[genKey] += value['Low Fitness']
```

```
            #Stores all the best of run fitness values. Keeps them all to calculate
            #standard deviation in addition to the average. Not a dictionary because
            #there is only one type of value to track.
            tempBest = run.getBestOfRun()
            bestRuns.append(tempBest['Best Fitness'])

        #Finally, average out the values based on the current number of runs
        #Avoids cases of not using the maximum number of runs and getting data already
        for key, value in avgOfAvgGen.items():
            avgOfAvgGen[key] = value / self.currentRun
        for key, value in avgOfBestGen.items():
            avgOfBestGen[key] = value / self.currentRun

        #Calculated average and standard deviation of the best-of-run fitness values
        avgBest = sum(bestRuns) / len(bestRuns)
        sdBest = 0
        for i in bestRuns:
            sdBest += (i - avgBest) ** 2
        sdBest / len(bestRuns)
        sdBest **= .5

        tempDict = {'Average Fitness' : avgBest, 'Standard Deviation' : sdBest}

        #Places calculated values in one dictionary
        self.averageRunData['Average Avg-of-Generation Fitness'] = avgOfAvgGen
        self.averageRunData['Average Best-of-Generation Fitness'] = avgOfBestGen
        self.averageRunData['Best-of-Runs'] = tempDict

    #Prints all data for all runs and averages across runs
    #Prints a label for each run, then
    #uses the print function of the SingleRunResults class
    #and prints the average run data calculated based on results so far
    def print(self):
        self.prepareAverages()
        for key, value in self.allRunData.items():
            print(key + ":")
            value.print()
        print("\nAverages Across All {} Runs:".format(self.currentRun))
        pp(self.averageRunData)

    #Specialized for this project, needs work to generalize
    def csvPrint(self):
        self.prepareAverages()
```

```
        print("Averages Across All {} Runs".format(self.currentRun))
        print("Average Avg-of-Generation Fitness")
        for key, value in self.averageRunData['Average Avg-of-Generation Fitness'].items():
            print(key + ", " + str(value))
        print()
        print("Average Best-of-Generation Fitness")
        for key, value in self.averageRunData['Average Best-of-Generation Fitness'].items():
            print(key + ", " + str(value))
        print()
        print("Best-of-Runs")
        for key, value in self.averageRunData['Best-of-Runs'].items():
            print(key + ", " + str(value))
        print()

        for key, value in self.allRunData.items():
            print("," + key)
            value.csvPrint()

#Stores data from a single EA run
#Initializes with the number of generations to run, stores the random seed,
#and if we want to min or max the fitness values
class SingleRunResults:
    def __init__(self, numGens, randomSeed, isMaxFitness = True):
        self.numGens = numGens
        self.randomSeed = randomSeed
        self.isMaxFitness = isMaxFitness
        self.bestOfRun = {'Best Fitness' : None}
        self.genResults = od()

    #Returns the dictionary of all data from the run
    def getGenResults(self):
        return self.genResults

    #Stores data by current generation of the best, worst, and average fitness values
    #Also stores the vectors that generated the best and worst fitness values
    def addGenResults(self, currentGen, highFit, hfVector, lowFit, lfVector, avgFit):
        tempDict = {'High Fitness' : highFit, 'High Fitness Vector' : hfVector,
                    'Low Fitness' : lowFit, 'Low Fitness Vector' : lfVector,
                    'Average Fitness' : avgFit}
        self.genResults['Generation ' + str(currentGen)] = tempDict

    #Returns the fitness value and vector of the current best of run, if any exist
    def getBestOfRun(self):
```

```
        return self.bestOfRun

    #Compares either the highest or lowest fitness value, based on desire to min or max,
    #with current best and replaces if it improves it
    def addBestOfRun(self, highFitness, highVector, lowFitness, lowVector):
        if self.isMaxFitness:
            if self.bestOfRun['Best Fitness'] is None or self.bestOfRun['Best Fitness'] < highFitness:
                self.bestOfRun['Best Fitness'] = highFitness
                self.bestOfRun['Best Vector'] = highVector
        else:
            if self.bestOfRun['Best Fitness'] is None or self.bestOfRun['Best Fitness'] > lowFitness:
                self.bestOfRun['Best Fitness'] = lowFitness
                self.bestOfRun['Best Vector'] = lowVector

    #Returns true for maximizing fitness, false for minimizing
    def getIsMaxFitness(self):
        return self.isMaxFitness

    #Prints the random seed, best of run results, and then the overall results
    def print(self):
        print("Random Seed: {}".format(self.randomSeed))
        pp(self.bestOfRun)
        pp(self.genResults)

    #Specialized for printing this particular project format to a csv, needs adjusting
    #for general use
    def csvPrint(self):
        print("Random Seed:,{0},Best Fitness:,{1:.6f},Best
Vector:,\"{2:.6f}".format(self.randomSeed,
                                                    self.bestOfRun['Best Fitness'],
                                                    self.bestOfRun['Best Vector'][0]) +
              chr(10) + "{0:.6f}".format(self.bestOfRun['Best Vector'][1]) +
              chr(10) + "{0:.6f}\"".format(self.bestOfRun['Best Vector'][2]))
        print("Generation,\"Average" + chr(10) + "Fitness\",\"Worst" + chr(10) +
"Fitness\",\"Worst" +
              chr(10) + "Fitness" + chr(10) + "Vector\",\"Best" + chr(10) + "Fitness\",\"Best" +
              chr(10) + "Fitness" + chr(10) + "Vector\"")
        for key, value in self.genResults.items():
            print(key.rsplit(" ")[1] + ",{0:.6f},{1:.6f},\"{2:.6f}".format(value['Average
Fitness'],value['High Fitness'],
                                                value['High Fitness Vector'][0]) +
                  chr(10) + "{0:.6f}".format(value['High Fitness Vector'][1]) +
                  chr(10) + "{0:.6f}\",{1:.6f},\"{2:.6f}".format(value['High Fitness Vector'][2],
value['Low Fitness'],
```

```
                                    value['Low Fitness Vector'][0]) +
                chr(10) + "{0:.6f}".format(value['Low Fitness Vector'][1]) +
                chr(10) + "{0:.6f}\"".format(value['Low Fitness Vector'][2]))
        print()
#Fitness function for the project
#Returns sum of squares of all x values
def _fitness(listVal):
    sumVals = 0
    for x in listVal:
        sumVals += x * x
    return sumVals

#Obtain a list of fitness values for a generation
def _getFitnessValues(listVal):
    fitList = []
    for i in listVal:
        fitList.append(_fitness(i))
    return fitList

#Manipulate fitness values for processing minimum fitness
#Modifies given list, no return needed
def _minModFitness(fitList):
    #Store minimum and maximum fitness values to reverse them
    #without resulting in heavily exaggerated values
    minFit = 0
    maxFit = 0
    for index, i in enumerate(fitList):
        if index is 0:
            minFit = i
            maxFit = i
        else:
            if minFit > i:
                minFit = i
            if maxFit < i:
                maxFit = i
    #Sum of the min and max fitnesses will flip them and
    #keep other values in between
    modFit = minFit + maxFit
    for index, i in enumerate(fitList):
        fitList[index] = modFit - i

#Convert fitness values to percents for random selection
#Modifies given list, no return needed
def _percentFitness(fitList):
```

```
    #Calculate sum to convert all fitness values to percents
    sumFit = 0
    for i in fitList:
        sumFit += i

    #Store previous value to generate stacking percentages
    #A random number between 0 and 1 may be used to select a
    #single index
    prevVal = 0
    for index, i in enumerate(fitList):
        fitList[index] /= sumFit
        fitList[index] += prevVal
        prevVal = fitList[index]
    #Set final value to 2.0 instead of 1.0 to ensure no
    #edge case percent errors
    fitList[-1] = 2.0

#Obtain fitness data about a generation
#Returns a dictionary including indices of the best and worst chromosomes
def _getFitnessData(listVal):
    fitList = _getFitnessValues(listVal)
    fitData = {}
    for index, i in enumerate(fitList):
        #Initialize fitData to first fitness value
        if index is 0:
            fitData = {'High Fit' : i, 'High Fit Index' : index,
                       'Low Fit' : i, 'Low Fit Index' : index,
                       'Average Fit' : i}
        else:
            #Increment value for average fitness of generation
            fitData['Average Fit'] += i
            #Update best fitness and index values
            if fitData['High Fit'] < i:
                fitData['High Fit'] = i
                fitData['High Fit Index'] = index
            #Update worst fitness and index values
            if fitData['Low Fit'] > i:
                fitData['Low Fit'] = i
                fitData['Low Fit Index'] = index
    #Finalize average fitness and return list
    fitData['Average Fit'] /= len(listVal)
    return fitData

#Creates the initial generation for the GA
```

```
def _initialGen(popSize, chromoSize, minVal, maxVal):
    initGen = []
    #Creates a chromosome for the entire population size
    for n in range(popSize):
        xVals = []
        #Selects random values inside given range for each element
        #in the chromosome
        for i in range(chromoSize):
            xVals.append(random.uniform(minVal, maxVal))
        initGen.append(xVals)
    return initGen

#Selection process, proportional, for a new generation
def _selectNewGen(prevGen, isMaxFit = True):
    indices = range(len(prevGen))
    #Get a list of all fitness values and convert to percents
    #Reverse fitness values if attempting to minimize fitness
    fitnessVals = _getFitnessValues(prevGen)
    if not isMaxFit:
        _minModFitness(fitnessVals)
    _percentFitness(fitnessVals)

    #Create a new list using percentages in selection
    tempList = []
    for i in indices:
        copyIndex = 0
        randomChoice = random.random()
        #randomChoice will never be > 1, so this loop will always end before
        #out of range errors
        #Ends when we reach the percent range our randomChoice is in
        while randomChoice > fitnessVals[copyIndex]:
            copyIndex += 1
        #Avoids copying references, prevents manipulating multiple lines at once
        tempList.append(deepcopy(prevGen[copyIndex]))

    #Set our new selection list to the input
    for i in indices:
        prevGen[i] = tempList[i]

#Crossover process for a new generation
def _crossNewGen(prevGenSelect, pointCross, prob):
    crossCount = 0
    #Warn if crossover will not change either point meaningfully
    #Expecting that all chromosomes will have the same length
```

```
    if pointCross is 0 or pointCross >= len(prevGenSelect[0]):
        print("Warning: crossover will not change points.")
        print("    Reconsider your point crossover value.")
        #If the line is too far right, lower it to avoid errors
        pointCross = len(prevGenSelect[0])
    #Working with pairs, not including the last element if length is odd
    halfLength = int(len(prevGenSelect) / 2)
    for i in range(halfLength):
        #Skip crossover of points based on given probability
        if random.random() < prob:
            crossCount += 1
            #Calculate indices of two elements to swap
            firstVal = 2 * i
            secondVal = firstVal + 1
            #Swap all points up to crossover line
            for j in range(pointCross):
                swap = prevGenSelect[firstVal][j]
                prevGenSelect[firstVal][j] = prevGenSelect[secondVal][j]
                prevGenSelect[secondVal][j] = swap
    #print("Number of Crossovers: %d" % crossCount)

#Get the mutated value of an input, making sure to keep it within bounds
def _getMutatedValue(value, alpha, minVal, maxVal):
    count = 0
    bounds = maxVal - minVal
    sign = 1
    if random.random() < .5:
        sign = -1
    tempVal = maxVal + 1
    #Continue looping until mutated value is within bounds or 5 attempts are made
    while count < 5 and (tempVal < minVal or tempVal > maxVal):
        count += 1
        #Set the modifier to a randomized value maxed at the alpha * bounds
        #and set to the appropriate sign
        modifier = random.random() * alpha * sign * bounds
        #print("Modifier: %f" % modifier)
        tempVal = value + modifier
    #Clamp mutated value to min or max if still outside bounds
    if tempVal < minVal:
        tempVal = minVal
    elif tempVal > maxVal:
        tempVal = maxVal
    return tempVal
```

```
#Mutation process for a new generation
def _mutateNewGen(prevGenCross, prob, alpha, minVal = -1.0, maxVal = 5.0):
    mutateCount = 0
    for index, i in enumerate(prevGenCross):
        for innerIndex, j in enumerate(i):
            #Skip mutation based on given probability
            if random.random() < prob:
                mutateCount += 1
                prevGenCross[index][innerIndex] = _getMutatedValue(j, alpha, minVal, maxVal)
    #print("Number of Mutations: %d" % mutateCount)

#Process a new generation using the previous one
#Cycles through selection, crossover, and mutation
#Manipulates list directly so no returns are needed
def _generateNewGen(prevGen, isMaxFitness = True, pointCross = 1, probC = .8, probM = .1,
alpha = .01):
    _selectNewGen(prevGen, isMaxFitness)
    _crossNewGen(prevGen, pointCross, probC)
    _mutateNewGen(prevGen, probM, alpha)

#Run through a number of generations
#Report feedback every 10 generations, and on the final one
#Expects a SingleRunResults class
def singleRun(storeRun, numGens = 50):
    isMaxFitness = storeRun.getIsMaxFitness()
    cell = _initialGen(numGens, 3, -1.0, 5.0)
    fitData = _getFitnessData(cell)
    storeRun.addGenResults(0, fitData['High Fit'], cell[fitData['High Fit Index']],
                           fitData['Low Fit'], cell[fitData['Low Fit Index']],
                           fitData['Average Fit'])
    storeRun.addBestOfRun(fitData['High Fit'], cell[fitData['High Fit Index']],
                          fitData['Low Fit'], cell[fitData['Low Fit Index']])
    for i in range(numGens):
        _generateNewGen(cell, isMaxFitness)
        #print("\nCell # %d" % (i + 1))
        #pprint.pprint(cell)
        if (i + 1) % 10 is 0:
            fitData = _getFitnessData(cell)
            storeRun.addGenResults(i + 1, fitData['High Fit'], cell[fitData['High Fit Index']],
                                   fitData['Low Fit'], cell[fitData['Low Fit Index']],
                                   fitData['Average Fit'])
            storeRun.addBestOfRun(fitData['High Fit'], cell[fitData['High Fit Index']],
                                  fitData['Low Fit'], cell[fitData['Low Fit Index']])
```

```
seeds = [ 54,  30, 101,  67,  34,
          22,  99,  32,  43,  95,
           2, 145, 245, 723,  46,
         123, 823, 711, 911, 194,
           8, 238, 234, 995, 204,
         899, 375, 112, 276, 419]

moreRuns = RunCollection(30)
for i in range(moreRuns.getMaxRuns()):
    random.seed(seeds[i])
    singleRun(moreRuns.addAndUseRun(50, seeds[i], False))
moreRuns.print()
#moreRuns.csvPrint()
```

End Source Code

Begin Program Output

```
Run 1:
Random Seed: 54
{'Best Fitness': 0.010864227269396608,
 'Best Vector': [0.07513053560680051,
                 -0.005081800925254461,
                 0.07206805941738664]}
OrderedDict([('Generation 0',
              {'Average Fitness': 20.713667670377443,
               'High Fitness': 61.05613112218806,
               'High Fitness Vector': [3.4732813542731513,
                                       4.9932051650343965,
                                       4.905135057887841],
               'Low Fitness': 1.3490528879206682,
               'Low Fitness Vector': [-0.9241844849809036,
                                      0.5948274880833557,
                                      0.3756543425300656]}),
             ('Generation 10',
              {'Average Fitness': 0.6454258869416916,
               'High Fitness': 1.4013992261875936,
               'High Fitness Vector': [-0.9241844849809036,
                                       0.560210939520321,
                                       -0.4831624645499015],
               'Low Fitness': 0.19101373984802741,
               'Low Fitness Vector': [0.32786921181365847,
                                      -0.2323123417285178,
                                      0.1718909412195132]}),
             ('Generation 20',
              {'Average Fitness': 0.2040635475545092,
               'High Fitness': 0.27890223083777954,
               'High Fitness Vector': [0.4563741464923516,
                                       -0.1997327158795087,
                                       0.1753046247494703],
               'Low Fitness': 0.1372677065261833,
               'Low Fitness Vector': [0.3051469356143429,
                                      -0.17654290175908696,
                                      0.11395463154185848]}),
             ('Generation 30',
              {'Average Fitness': 0.1388194577211849,
               'High Fitness': 0.2606811880665255,
               'High Fitness Vector': [0.4469755607151039,
                                       -0.17076547455123414,
                                       0.17813811745731395],
```

```
             'Low Fitness': 0.06835725497026088,
             'Low Fitness Vector': [0.1905853829470246,
                                    -0.08594614333341073,
                                    0.15699594651872287]}),
            ('Generation 40',
             {'Average Fitness': 0.0800514569232098,
             'High Fitness': 0.15091279369398442,
             'High Fitness Vector': [0.2988468745950849,
                                     -0.1917635235825536,
                                     0.15757566519592237],
             'Low Fitness': 0.0350781747177334,
             'Low Fitness Vector': [0.12671375984641967,
                                    -0.10125386299025348,
                                    0.09364535766853721]}),
            ('Generation 50',
             {'Average Fitness': 0.03584280565887102,
             'High Fitness': 0.06368981425445264,
             'High Fitness Vector': [0.22533610392237424,
                                     -0.11006760810916405,
                                     0.02825909001834777],
             'Low Fitness': 0.010864227269396608,
             'Low Fitness Vector': [0.07513053560680051,
                                    -0.005081800925254461,
                                    0.07206805941738664]})])
Run 2:
Random Seed: 30
{'Best Fitness': 0.27130749456600023,
 'Best Vector': [0.2923916917460885, 0.06953498272325628, 0.42541683011081854]}
OrderedDict([('Generation 0',
             {'Average Fitness': 20.45680208767602,
             'High Fitness': 52.29699621673738,
             'High Fitness Vector': [1.76919806037575,
                                     4.960910045610012,
                                     4.955431964951767],
             'Low Fitness': 0.5278742883080413,
             'Low Fitness Vector': [0.4560532134406594,
                                    -0.5641293227719519,
                                    -0.04059386662229203]}),
            ('Generation 10',
             {'Average Fitness': 0.9072749419843862,
             'High Fitness': 1.8759916254819573,
             'High Fitness Vector': [0.4560532134406594,
                                     1.0164995301987045,
                                     -0.796703079633949],
```

```
  'Low Fitness': 0.45761606923810316,
  'Low Fitness Vector': [0.3725052006350765,
                         0.23569710025779322,
                         0.5131304138988375]}),
 ('Generation 20',
  {'Average Fitness': 0.5298895811820585,
  'High Fitness': 0.6450720496285065,
  'High Fitness Vector': [0.5391890268427457,
                          0.18212006513796103,
                          0.5667270285022801],
  'Low Fitness': 0.447498121559445,
  'Low Fitness Vector': [0.3994885899846023,
                         0.22380988348864875,
                         0.48766394585242495]}),
 ('Generation 30',
  {'Average Fitness': 0.49430824848697386,
  'High Fitness': 0.6181647474722811,
  'High Fitness Vector': [0.5190870801081099,
                          0.22380988348864875,
                          0.5464636189078971],
  'Low Fitness': 0.3704683119049914,
  'Low Fitness Vector': [0.35309379889306647,
                         0.2044329668762247,
                         0.451663860788572]}),
 ('Generation 40',
  {'Average Fitness': 0.4121672276204283,
  'High Fitness': 0.5524231206213741,
  'High Fitness Vector': [0.3993457224832999,
                          0.34151189761254225,
                          0.5256574344045234],
  'Low Fitness': 0.28630485020216834,
  'Low Fitness Vector': [0.32560894076103164,
                         0.15052795283462966,
                         0.3970201547957772]}),
 ('Generation 50',
  {'Average Fitness': 0.3423229680182353,
  'High Fitness': 0.4701239763603798,
  'High Fitness Vector': [0.40352233927430453,
                          0.2974142789997597,
                          0.4678017151678967],
  'Low Fitness': 0.27130749456600023,
  'Low Fitness Vector': [0.2923916917460885,
                         0.06953498272325628,
                         0.42541683011081854]})])
```

```
Run 3:
Random Seed: 101
{'Best Fitness': 0.028914650268052733,
 'Best Vector': [0.10338645551227404,
                 0.08971581734488458,
                 -0.10088093577480643]}
OrderedDict([('Generation 0',
              {'Average Fitness': 19.467136738025463,
               'High Fitness': 44.82621272056048,
               'High Fitness Vector': [3.7553607017186756,
                                       3.4292998283916507,
                                       4.354696477085457],
               'Low Fitness': 0.40829695429352064,
               'Low Fitness Vector': [0.5001093208545409,
                                      0.2266286056480291,
                                      -0.3268441472475232]}),
             ('Generation 10',
              {'Average Fitness': 1.3331562795796748,
               'High Fitness': 2.8376843019346083,
               'High Fitness Vector': [0.2160402483204827,
                                       1.0913427030906102,
                                       1.2649039558208017],
               'Low Fitness': 0.26401137145175607,
               'Low Fitness Vector': [0.2160402483204827,
                                      0.15042413161005833,
                                      -0.4412601989605907]}),
             ('Generation 20',
              {'Average Fitness': 0.2723863040518082,
               'High Fitness': 0.43668352637530283,
               'High Fitness Vector': [0.431138342209858,
                                       0.19903105523901593,
                                       -0.4595540178284616],
               'Low Fitness': 0.1800141901843861,
               'Low Fitness Vector': [0.18825470406138123,
                                      0.16159184394808634,
                                      -0.3441837191858517]}),
             ('Generation 30',
              {'Average Fitness': 0.21753918082158744,
               'High Fitness': 0.33725092975865834,
               'High Fitness Vector': [0.26614064326777703,
                                       0.16159184394808634,
                                       -0.4902123659488095],
               'Low Fitness': 0.13745833498477011,
               'Low Fitness Vector': [0.12623848511362684,
```

```
                        0.19595188389766804,
                        -0.288314132601823]}),
          ('Generation 40',
           {'Average Fitness': 0.1265257663587583,
            'High Fitness': 0.22316025719390412,
            'High Fitness Vector': [0.22198760443810325,
                        0.22862308078124843,
                        -0.3487309100206448],
            'Low Fitness': 0.06359965877669836,
            'Low Fitness Vector': [0.09972157804639475,
                        0.11187160263092896,
                        -0.2028300031391379]}),
          ('Generation 50',
           {'Average Fitness': 0.06161437059968735,
            'High Fitness': 0.09941228916108716,
            'High Fitness Vector': [0.11867454798812482,
                        0.04682799170675209,
                        -0.28833275917525186],
            'Low Fitness': 0.028914650268052733,
            'Low Fitness Vector': [0.10338645551227404,
                        0.08971581734488458,
                        -0.10088093577480643]})])
Run 4:
Random Seed: 67
{'Best Fitness': 0.001410088429946272,
 'Best Vector': [-0.009745327586925649,
              -0.014354976576191184,
              0.03330242735397497]}
OrderedDict([('Generation 0',
           {'Average Fitness': 23.061175703349072,
            'High Fitness': 58.44790097936681,
            'High Fitness Vector': [4.912639345344759,
                        4.250704303728204,
                        4.0305568553493885],
            'Low Fitness': 0.3838796240715497,
            'Low Fitness Vector': [0.19719239234232022,
                        -0.399847469435308,
                        -0.43025200250555784]}),
          ('Generation 10',
           {'Average Fitness': 0.43587959079979266,
            'High Fitness': 0.9127489537012639,
            'High Fitness Vector': [-0.9237090618484467,
                        -0.1982787589789941,
                        0.14211283016700627],
```

```
  'Low Fitness': 0.056448782920031755,
  'Low Fitness Vector': [-0.10224316554903334,
                         -0.19734167733537733,
                         0.08397249790856848]}),
 ('Generation 20',
  {'Average Fitness': 0.081114860599517 4,
  'High Fitness': 0.16227703694931594,
  'High Fitness Vector': [0.3259350486224677,
                          -0.1436584732328357,
                          0.18816382249836494],
  'Low Fitness': 0.03782030986953616,
  'Low Fitness Vector': [-0.10224316554903334,
                         -0.13404010565105975,
                         0.09695305588325653]}),
 ('Generation 30',
  {'Average Fitness': 0.037256007279131344,
  'High Fitness': 0.0723576584853026,
  'High Fitness Vector': [-0.11212393250986362,
                          -0.16823718861878667,
                          0.1774320450467529],
  'Low Fitness': 0.021059777497790394,
  'Low Fitness Vector': [-0.0785797571594388,
                         -0.09639665308288112,
                         0.07478425326880445]}),
 ('Generation 40',
  {'Average Fitness': 0.015681612288173958,
  'High Fitness': 0.02986386035388138,
  'High Fitness Vector': [-0.09483525157762439,
                          -0.11117749779410965,
                          0.09224803193743927],
  'Low Fitness': 0.005965292290653834,
  'Low Fitness Vector': [-0.06485265305534514,
                         -0.039866357819062434,
                         0.013042208270816182]}),
 ('Generation 50',
  {'Average Fitness': 0.0064851691141305685,
  'High Fitness': 0.01990027570610798,
  'High Fitness Vector': [-0.06564074672492475,
                          -0.03496592516073068,
                          0.11987056416467073],
  'Low Fitness': 0.001410088429946272,
  'Low Fitness Vector': [-0.009745327586925649,
                         -0.014354976576191184,
                         0.03330242735397497]})])
```

Run 5:
Random Seed: 34
{'Best Fitness': 0.6701206137412895,
 'Best Vector': [-0.02322721404060652, 0.6396134072856735, -0.5103682978885077]}
OrderedDict([('Generation 0',
              {'Average Fitness': 24.118486033912497,
               'High Fitness': 45.25719363170004,
               'High Fitness Vector': [-0.0759951113984203,
                                       4.595781140614969,
                                       4.912251427025224],
               'Low Fitness': 1.113320860861844,
               'Low Fitness Vector': [0.40312143371342346,
                                      0.7770499875135599,
                                      -0.5890732445527076]}),
             ('Generation 10',
              {'Average Fitness': 1.541554399858578,
               'High Fitness': 2.0101966934407933,
               'High Fitness Vector': [0.8328429846494634,
                                       0.7325852856764423,
                                       0.8831127083058721],
               'Low Fitness': 0.973235644378196,
               'Low Fitness Vector': [-0.10430038888344455,
                                      0.7770499875135599,
                                      -0.5987907732773898]}),
             ('Generation 20',
              {'Average Fitness': 1.11025916304225,
               'High Fitness': 1.467931945717007,
               'High Fitness Vector': [0.3506075103761527,
                                       0.7517434827210081,
                                       0.8831127083058721],
               'Low Fitness': 0.8550573010000384,
               'Low Fitness Vector': [-0.10430038888344455,
                                      0.6968703894675582,
                                      -0.5987907732773898]}),
             ('Generation 30',
              {'Average Fitness': 1.0437247559951541,
               'High Fitness': 1.303970763999786,
               'High Fitness Vector': [0.43594190404546446,
                                       0.8410008263870533,
                                       -0.6376856830079333],
               'Low Fitness': 0.8215239523048574,
               'Low Fitness Vector': [-0.13263008174670926,
                                      0.7180763639109089,
                                      -0.5369353306620888]}),

```
          ('Generation 40',
           {'Average Fitness': 0.9510851344107624,
            'High Fitness': 1.2845822105861218,
            'High Fitness Vector': [0.5541116941656468,
                                    0.8038009678497781,
                                    -0.5757138569278655],
            'Low Fitness': 0.7663082504373919,
            'Low Fitness Vector': [-0.1474191129004332,
                                   0.6692547866052387,
                                   -0.5446777819913509]}),
          ('Generation 50',
           {'Average Fitness': 0.8926946690654216,
            'High Fitness': 1.196633328802262,
            'High Fitness Vector': [0.6037228736287692,
                                    0.7306087028252971,
                                    -0.5462260923972965],
            'Low Fitness': 0.6701206137412895,
            'Low Fitness Vector': [-0.02322721404060652,
                                   0.6396134072856735,
                                   -0.5103682978885077]})])
Run 6:
Random Seed: 22
{'Best Fitness': 0.17288423332271965,
 'Best Vector': [-0.06478302071777031,
                 0.15361295368087155,
                 0.38090740871088213]}
OrderedDict([('Generation 0',
           {'Average Fitness': 21.501731076087513,
            'High Fitness': 57.42015213104351,
            'High Fitness Vector': [4.627933430472825,
                                    3.6437128476668406,
                                    4.767152292291819],
            'Low Fitness': 0.4543677373556924,
            'Low Fitness Vector': [-0.2914521548241382,
                                   0.2980821222551391,
                                   0.5296889910088101]}),
          ('Generation 10',
           {'Average Fitness': 1.794797596977179,
            'High Fitness': 5.4768870023535845,
            'High Fitness Vector': [2.169444282085106,
                                    0.6440366669548909,
                                    -0.5963348731203455],
            'Low Fitness': 0.3384043679390033,
            'Low Fitness Vector': [-0.17169328534873796,
```

```
                    0.24222502293991394,
                    0.5002527580802711]}),
          ('Generation 20',
           {'Average Fitness': 0.7197856914774121,
            'High Fitness': 0.8571882486399316,
            'High Fitness Vector': [-0.19735244484799594,
                    0.6055719556986714,
                    -0.6719545130615142],
            'Low Fitness': 0.3439204912511842,
            'Low Fitness Vector': [-0.146659659764504,
                    0.24222502293991394,
                    0.5135547426620674]}),
          ('Generation 30',
           {'Average Fitness': 0.33474776262701783,
            'High Fitness': 0.42614592232475057,
            'High Fitness Vector': [-0.2662871348233302,
                    0.27635201610616017,
                    0.5280782587329199],
            'Low Fitness': 0.27143180036368186,
            'Low Fitness Vector': [-0.09641722622865756,
                    0.23548806305786527,
                    0.4546217009858951]}),
          ('Generation 40',
           {'Average Fitness': 0.2904666880834315,
            'High Fitness': 0.39558307278944377,
            'High Fitness Vector': [-0.23278697012816424,
                    0.2653669102918166,
                    0.5205513444898283],
            'Low Fitness': 0.2135225188682341,
            'Low Fitness Vector': [-0.04808515471436144,
                    0.19074883055136593,
                    0.4181210595122139]}),
          ('Generation 50',
           {'Average Fitness': 0.25058114318494934,
            'High Fitness': 0.39179694539568577,
            'High Fitness Vector': [-0.146659659764504,
                    0.23820642203484801,
                    0.5599514176201374],
            'Low Fitness': 0.17288423332271965,
            'Low Fitness Vector': [-0.06478302071777031,
                    0.15361295368087155,
                    0.38090740871088213]})])
Run 7:
Random Seed: 99
```

```
{'Best Fitness': 0.000438271869765925,
 'Best Vector': [0.010501915770376076,
                 -0.0181069842787043,
                 -0.00034460883419701327]}
OrderedDict([('Generation 0',
              {'Average Fitness': 21.18443816186571,
               'High Fitness': 54.50396613305413,
               'High Fitness Vector': [4.897328533722107,
                                       4.663463570320403,
                                       2.961797915817332],
               'Low Fitness': 0.08070924735945936,
               'Low Fitness Vector': [0.06587304517065995,
                                      -0.12693131309149197,
                                      0.2454759276105769]}),
             ('Generation 10',
              {'Average Fitness': 0.3681685238942507,
               'High Fitness': 1.2080072480217863,
               'High Fitness Vector': [0.5624532719383188,
                                       -0.9176648781833496,
                                       0.22258646916738623],
               'Low Fitness': 0.04456873311813303,
               'Low Fitness Vector': [0.06587304517065995,
                                      0.2004526674496725,
                                      -0.006942848866872377]}),
             ('Generation 20',
              {'Average Fitness': 0.06681360166603131,
               'High Fitness': 0.14719632193310864,
               'High Fitness Vector': [-0.1765911422318069,
                                       -0.1648883591886936,
                                       0.2980330844427087],
               'Low Fitness': 0.015303753703584541,
               'Low Fitness Vector': [0.06587304517065995,
                                      0.10424581260713937,
                                      -0.00986438935798447]}),
             ('Generation 30',
              {'Average Fitness': 0.015669301794656528,
               'High Fitness': 0.03993617192905435,
               'High Fitness Vector': [-0.12161145253475834,
                                       0.15729343248030442,
                                       0.02013957894307819],
               'Low Fitness': 0.005332663383886891,
               'Low Fitness Vector': [0.06587304517065995,
                                      0.030744140148048604,
                                      -0.006942848866872377]}),
```

```
          ('Generation 40',
           {'Average Fitness': 0.0035120715770869306,
            'High Fitness': 0.01776233040733359,
            'High Fitness Vector': [0.10763816052610659,
                                    0.021227649222360187,
                                    -0.07566864419547986],
            'Low Fitness': 0.0004503952952273823,
            'Low Fitness Vector': [0.011035066060021206,
                                   0.01674572966132319,
                                   -0.006942848866872377]}),
          ('Generation 50',
           {'Average Fitness': 0.0017620865070838742,
            'High Fitness': 0.005985656222027356,
            'High Fitness Vector': [0.06236763337411658,
                                    0.022251290490409192,
                                    -0.040010181214917805],
            'Low Fitness': 0.000438271869765925,
            'Low Fitness Vector': [0.010501915770376076,
                                   -0.0181069842787043,
                                   -0.00034460883419701327]})])
Run 8:
Random Seed: 32
{'Best Fitness': 0.00950432567771008,
 'Best Vector': [-0.0073447078975525815,
                 -0.09660698968113784,
                 0.010838380338327076]}
OrderedDict([('Generation 0',
              {'Average Fitness': 22.781416552699316,
               'High Fitness': 54.8999794650025,
               'High Fitness Vector': [4.057051727308259,
                                       4.0194651762171105,
                                       4.7206154730210255],
               'Low Fitness': 0.3811253851827624,
               'Low Fitness Vector': [0.21002148596602366,
                                      -0.5109954177950011,
                                      -0.27549962542242756]}),
             ('Generation 10',
              {'Average Fitness': 1.7312489031825529,
               'High Fitness': 3.530996384259385,
               'High Fitness Vector': [1.3077367867200698,
                                       0.3942684473141371,
                                       1.290493422056444],
               'Low Fitness': 0.3762883401142779,
               'Low Fitness Vector': [0.21002148596602366,
```

```
                    -0.5389005054833581,
                    -0.2043662416757776]}),
          ('Generation 20',
           {'Average Fitness': 0.16064089106484236,
            'High Fitness': 0.5937292211128361,
            'High Fitness Vector': [0.29233071100976166,
                    -0.4326683764662308,
                    -0.5666304373393997],
            'Low Fitness': 0.08801371812149204,
            'Low Fitness Vector': [0.20614212686811817,
                    -0.20902624377273016,
                    -0.04274542157990613]}),
          ('Generation 30',
           {'Average Fitness': 0.09153101731792844,
            'High Fitness': 0.14874914777157747,
            'High Fitness Vector': [0.1995689531466257,
                    -0.32972460611282295,
                    0.014250081939452972],
            'Low Fitness': 0.05105517355230095,
            'Low Fitness Vector': [0.16799731949009755,
                    -0.15077914917877405,
                    -0.009885462526763408]}),
          ('Generation 40',
           {'Average Fitness': 0.059465405605530534,
            'High Fitness': 0.1280481290973662,
            'High Fitness Vector': [0.3070045175915274,
                    -0.1763453154598481,
                    -0.05194886900719209],
            'Low Fitness': 0.03560781281077315,
            'Low Fitness Vector': [0.08460033605887436,
                    -0.16154924632262052,
                    -0.04850192740594911]}),
          ('Generation 50',
           {'Average Fitness': 0.024280002717966962,
            'High Fitness': 0.04941820052652549,
            'High Fitness Vector': [0.11992299413291688,
                    -0.18078782880113894,
                    -0.04850192740594911],
            'Low Fitness': 0.00950432567771008,
            'Low Fitness Vector': [-0.0073447078975525815,
                    -0.09660698968113784,
                    0.010838380338327076]})])
Run 9:
Random Seed: 43
```

{'Best Fitness': 0.12195684188609847,
 'Best Vector': [-0.2609348527241178,
                 -0.23207824480749534,
                 0.003087524415072443]}
OrderedDict([('Generation 0',
              {'Average Fitness': 21.603413173526175,
               'High Fitness': 55.52866450944496,
               'High Fitness Vector': [3.407544300722418,
                                       4.935097594837858,
                                       4.422908327953985],
               'Low Fitness': 3.3520158640809217,
               'Low Fitness Vector': [-0.9115765905302364,
                                      1.2568147383508577,
                                      0.9702889761005515]}),
             ('Generation 10',
              {'Average Fitness': 0.8271593383098755,
               'High Fitness': 3.586181234411465,
               'High Fitness Vector': [1.1667867174915534,
                                       0.997563658566779,
                                       1.1088988851124408],
               'Low Fitness': 0.292714395101187,
               'Low Fitness Vector': [-0.22033895035660467,
                                      -0.44228216777433405,
                                      0.2203443353612082]}),
             ('Generation 20',
              {'Average Fitness': 0.41465075497703524,
               'High Fitness': 0.6169202784425738,
               'High Fitness Vector': [0.6243318038652413,
                                       -0.44228216777433405,
                                       0.17752904324020732],
               'Low Fitness': 0.33102019029533336,
               'Low Fitness Vector': [-0.3461197359056477,
                                      -0.4289585868183349,
                                      0.16497226890224642]}),
             ('Generation 30',
              {'Average Fitness': 0.39060860486478033,
               'High Fitness': 0.4886427092503629,
               'High Fitness Vector': [-0.47687923063181775,
                                       -0.46848670579678,
                                       0.20432600209970775],
               'Low Fitness': 0.28410194334682703,
               'Low Fitness Vector': [-0.3200922189572873,
                                      -0.3750191857191651,
                                      0.2024932716224454]}),

```
          ('Generation 40',
           {'Average Fitness': 0.32053098259115287,
            'High Fitness': 0.39598120483167415,
            'High Fitness Vector': [-0.4237468037307152,
                                    -0.4051858996336647,
                                    0.22856998468244838],
            'Low Fitness': 0.22940735122690972,
            'Low Fitness Vector': [-0.31498333560156666,
                                   -0.34488201179182504,
                                   0.10606246962353838]}),
          ('Generation 50',
           {'Average Fitness': 0.2349519907756913,
            'High Fitness': 0.3326852897752477,
            'High Fitness Vector': [-0.36100038987113187,
                                    -0.34403677764157814,
                                    0.2898321995882062],
            'Low Fitness': 0.12195684188609847,
            'Low Fitness Vector': [-0.2609348527241178,
                                   -0.23207824480749534,
                                   0.003087524415072443]})])
Run 10:
Random Seed: 95
{'Best Fitness': 0.3144562710453478,
 'Best Vector': [-0.4617378571609745,
                 0.29743417920444476,
                 -0.11308108308075902]}
OrderedDict([('Generation 0',
           {'Average Fitness': 17.84830340424672,
            'High Fitness': 41.845403444642685,
            'High Fitness Vector': [3.5609688698586464,
                                    2.0828996138194977,
                                    4.9826131047163305],
            'Low Fitness': 0.6193318558940665,
            'Low Fitness Vector': [-0.6964187621156739,
                                   0.33208564491173376,
                                   -0.15508671158708842]}),
          ('Generation 10',
           {'Average Fitness': 1.26185743864075,
            'High Fitness': 3.126330928916737,
            'High Fitness Vector': [1.1791040681925888,
                                    0.9688952836762365,
                                    -0.8929089844762284],
            'Low Fitness': 0.545329916236857,
            'Low Fitness Vector': [-0.6410905962260858,
```

```
                    0.33208564491173376,
                    -0.15508671158708842]}),
          ('Generation 20',
           {'Average Fitness': 0.6569997868527699,
            'High Fitness': 0.7870263499546014,
            'High Fitness Vector': [-0.7552977547154944,
                    0.34849560715334793,
                    -0.30838687305258905],
            'Low Fitness': 0.5009937285408592,
            'Low Fitness Vector': [-0.6299171761050301,
                    0.2831010273344605,
                    -0.15508671158708842]}),
          ('Generation 30',
           {'Average Fitness': 0.5323674749337931,
            'High Fitness': 0.6337647886481206,
            'High Fitness Vector': [-0.6792992646437611,
                    0.3459291204185098,
                    -0.22945662193330166],
            'Low Fitness': 0.4280455005649217,
            'Low Fitness Vector': [-0.5645967371091231,
                    0.2831010273344605,
                    -0.17067464173926528]}),
          ('Generation 40',
           {'Average Fitness': 0.47494674046781726,
            'High Fitness': 0.6262391463838576,
            'High Fitness Vector': [-0.6792992646437611,
                    0.33318395849338867,
                    -0.2319053799310807],
            'Low Fitness': 0.37407083741130825,
            'Low Fitness Vector': [-0.5129994844033998,
                    0.2971688386462851,
                    -0.15030983916820742]}),
          ('Generation 50',
           {'Average Fitness': 0.38992486258768233,
            'High Fitness': 0.4752808479597167,
            'High Fitness Vector': [-0.5735125483243634,
                    0.3687032676144499,
                    -0.10208871301294209],
            'Low Fitness': 0.3144562710453478,
            'Low Fitness Vector': [-0.4617378571609745,
                    0.29743417920444476,
                    -0.11308108308075902]})])
Run 11:
Random Seed: 2
```

```
{'Best Fitness': 0.31495856940351175,
 'Best Vector': [-0.041405980379447996,
                 0.2311206774353956,
                 -0.5097326227093302]}
OrderedDict([('Generation 0',
              {'Average Fitness': 22.48416338778013,
               'High Fitness': 51.876564898161064,
               'High Fitness Vector': [4.968917377698457,
                                       4.696372838559461,
                                       2.265062284575925],
               'Low Fitness': 1.755160271264227,
               'Low Fitness Vector': [-0.09147934746069075,
                                      -0.7823860345395159,
                                      1.0652060332207682]}),
             ('Generation 10',
              {'Average Fitness': 1.0908456213627327,
               'High Fitness': 1.8191429733915407,
               'High Fitness Vector': [0.8573787496967642,
                                       0.6094444498959686,
                                       -0.8441694826579283],
               'Low Fitness': 0.6801546777754095,
               'Low Fitness Vector': [-0.05729026787935054,
                                      0.41674044269037247,
                                      -0.7093658480699181]}),
             ('Generation 20',
              {'Average Fitness': 0.7468006091799695,
               'High Fitness': 0.9833959540454063,
               'High Fitness Vector': [-0.023480699877051536,
                                       0.6239360416764319,
                                       -0.7704209412235858],
               'Low Fitness': 0.5102002495140991,
               'Low Fitness Vector': [-0.06491157612119244,
                                      0.3224250755288535,
                                      -0.6340574165403121]}),
             ('Generation 30',
              {'Average Fitness': 0.6481510847056327,
               'High Fitness': 0.8302743050259324,
               'High Fitness Vector': [-0.05640633813534318,
                                       0.47570106605893536,
                                       -0.775113621215939],
               'Low Fitness': 0.4723813448147 1306,
               'Low Fitness Vector': [-1.8687294655022474e-05,
                                      0.3224250755288535,
                                      -0.6069789247871059]}),
```

```
          ('Generation 40',
           {'Average Fitness': 0.5111523468361913,
            'High Fitness': 0.6867017242805815,
            'High Fitness Vector': [-0.019008981489228072,
                                    0.4138053546773367,
                                    -0.7177085141919993],
            'Low Fitness': 0.4237170937710886,
            'Low Fitness Vector': [0.045184491001327216,
                                   0.26933990345905007,
                                   -0.5908734821843922]}),
          ('Generation 50',
           {'Average Fitness': 0.4328198709756,
            'High Fitness': 0.6586352717218374,
            'High Fitness Vector': [-0.034185622258601556,
                                    0.41674044269037247,
                                    -0.6955530306014502],
            'Low Fitness': 0.31495856940351175,
            'Low Fitness Vector': [-0.041405980379447996,
                                   0.2311206774353956,
                                   -0.5097326227093302]})])
Run 12:
Random Seed: 145
{'Best Fitness': 0.6195230731646729,
 'Best Vector': [-0.3222518369546714, -0.13881700981406286, 0.7045613277283215]}
OrderedDict([('Generation 0',
           {'Average Fitness': 23.007498112689763,
            'High Fitness': 61.02525616248627,
            'High Fitness Vector': [4.365134929182448,
                                    4.1561512133272265,
                                    4.969633819958593],
            'Low Fitness': 1.2258817704644946,
            'Low Fitness Vector': [-0.975226768182059,
                                   -0.33758158246606973,
                                   0.4010650773445321]}),
          ('Generation 10',
           {'Average Fitness': 1.8239155685124695,
            'High Fitness': 2.8898867663928764,
            'High Fitness Vector': [0.864159971344415,
                                    1.460556198965173,
                                    0.09944797626549395],
            'Low Fitness': 0.6195230731646729,
            'Low Fitness Vector': [-0.3222518369546714,
                                   -0.13881700981406286,
                                   0.7045613277283215]}),
```

```
        ('Generation 20',
         {'Average Fitness': 1.5476046785956976,
          'High Fitness': 1.738226739991183,
          'High Fitness Vector': [0.2993348916549017,
                                  -0.8946147083960454,
                                  0.921026539330214],
          'Low Fitness': 1.4006120547128509,
          'Low Fitness Vector': [-0.30838567354940316,
                                 -0.857817550120591,
                                 0.7547578298815051]}),
        ('Generation 30',
         {'Average Fitness': 1.5078001202977376,
          'High Fitness': 1.6973240902313451,
          'High Fitness Vector': [-0.2835750889613012,
                                  -0.9042472271875002,
                                  0.8940057109860388],
          'Low Fitness': 1.3220125010968893,
          'Low Fitness Vector': [-0.2835750889613012,
                                 -0.7382826574976337,
                                 0.8345875554162734]}),
        ('Generation 40',
         {'Average Fitness': 1.5436782066384576,
          'High Fitness': 1.8481246290163404,
          'High Fitness Vector': [-0.25965672243497395,
                                  -0.9466689864986655,
                                  0.940489683894649],
          'Low Fitness': 1.2900881210705588,
          'Low Fitness Vector': [-0.2656185752253529,
                                 -0.793919650631348,
                                 0.7676108922541489]}),
        ('Generation 50',
         {'Average Fitness': 1.410813160827661,
          'High Fitness': 1.7534639345862373,
          'High Fitness Vector': [0.31066646420362715,
                                  -0.9196124264494435,
                                  0.9007015419800394],
          'Low Fitness': 1.1930296294969083,
          'Low Fitness Vector': [0.31497076474675795,
                                 -0.7753757989070278,
                                 0.7018656690001578]})])
Run 13:
Random Seed: 245
{'Best Fitness': 1.6496893774748018,
 'Best Vector': [-0.5250586534371328, 0.7734081219839165, 0.8808193144878795]}
```

30

```
OrderedDict([('Generation 0',
              {'Average Fitness': 22.36126547385806,
               'High Fitness': 55.01027748346911,
               'High Fitness Vector': [4.574889329034127,
                                       3.7575448152208697,
                                       4.467831920760399],
               'Low Fitness': 2.23047790754123,
               'Low Fitness Vector': [1.304180290462618,
                                      -0.6775991294921451,
                                      -0.26542625571250245]}),
             ('Generation 10',
              {'Average Fitness': 3.557217043129361,
               'High Fitness': 7.008294511771549,
               'High Fitness Vector': [1.9030139205678192,
                                       1.6964274489156486,
                                       0.7134188394359884],
               'Low Fitness': 1.8941270193872266,
               'Low Fitness Vector': [-0.6371215737966105,
                                      0.8011197302052104,
                                      0.9200055964319389]}),
             ('Generation 20',
              {'Average Fitness': 2.084087441641293,
               'High Fitness': 2.8485369036106105,
               'High Fitness Vector': [-0.7196120418821773,
                                       1.1432857991009358,
                                       1.011727727386627],
               'Low Fitness': 1.7534410536747242,
               'Low Fitness Vector': [-0.6090580758159597,
                                      0.7363024178656437,
                                      0.9167050034789229]}),
             ('Generation 30',
              {'Average Fitness': 1.9826354047332593,
               'High Fitness': 2.3807703477965747,
               'High Fitness Vector': [-0.644568614487979,
                                       0.9347544286458819,
                                       1.0447659102117244],
               'Low Fitness': 1.7544567675767233,
               'Low Fitness Vector': [-0.5399373380894288,
                                      0.789035091178037,
                                      0.9167050034789229]}),
             ('Generation 40',
              {'Average Fitness': 1.9790145788785083,
               'High Fitness': 2.5221213845915837,
               'High Fitness Vector': [-0.7098414044871387,
```

```
                                0.9420646664727852,
                                1.0633723380128102],
                 'Low Fitness': 1.7339262642809299,
                 'Low Fitness Vector': [-0.5772785866619743,
                                0.7485503551927494,
                                0.9167050034789229]}),
                ('Generation 50',
                 {'Average Fitness': 1.8375272386610972,
                 'High Fitness': 2.415009211392654,
                 'High Fitness Vector': [-0.6293670765346514,
                                0.8941761378611774,
                                1.1042442342374033],
                 'Low Fitness': 1.6496893774748018,
                 'Low Fitness Vector': [-0.5250586534371328,
                                0.7734081219839165,
                                0.8808193144878795]})])
Run 14:
Random Seed: 723
{'Best Fitness': 7.217006388830133e-06,
 'Best Vector': [-0.0024743206700619837,
                 -0.0010462134296090192,
                 1.345623423856801e-05]}
OrderedDict([('Generation 0',
                {'Average Fitness': 23.15054567330937,
                 'High Fitness': 51.535937338967834,
                 'High Fitness Vector': [3.9797961669960547,
                                4.252715414647794,
                                4.196614279410057],
                 'Low Fitness': 0.3802546246652318,
                 'Low Fitness Vector': [-0.08013739461712222,
                                0.5835889945353352,
                                -0.1823636699193342 8]}),
                ('Generation 10',
                 {'Average Fitness': 1.1910263422714058,
                 'High Fitness': 3.086813186483658,
                 'High Fitness Vector': [1.7348027641439336,
                                0.2263896585060916,
                                0.16130802374252906],
                 'Low Fitness': 0.13722140500192748,
                 'Low Fitness Vector': [0.17985445177228554,
                                0.2808086584418254,
                                0.16130802374252906]}),
                ('Generation 20',
                 {'Average Fitness': 0.12819873739559964,
```

```
             'High Fitness': 0.23476850723362894,
             'High Fitness Vector': [-0.40503838148924054,
                                     0.24933885958422158,
                                     0.09242591549692777],
             'Low Fitness': 0.06925896856531233,
             'Low Fitness Vector': [0.13786867465811958,
                                    0.14774289441046268,
                                    0.1685919163677333]}),
            ('Generation 30',
             {'Average Fitness': 0.07968582884024777,
             'High Fitness': 0.12701133848240534,
             'High Fitness Vector': [0.23602452701173704,
                                     0.21279916026049267,
                                     0.16130802374252906],
             'Low Fitness': 0.0460328566006686,
             'Low Fitness Vector': [0.11725991649736134,
                                    0.1307709464957247,
                                    0.12321496717652752]}),
            ('Generation 40',
             {'Average Fitness': 0.03411326488225837,
             'High Fitness': 0.08846675975024076,
             'High Fitness Vector': [0.23670000030463934,
                                     0.1307709464957247,
                                     0.12385002688185742],
             'Low Fitness': 0.01263756237652846,
             'Low Fitness Vector': [0.018308541069889696,
                                    0.089226746485178,
                                    0.06588586655793788]}),
            ('Generation 50',
             {'Average Fitness': 0.006980047555987461,
             'High Fitness': 0.025254825743440974,
             'High Fitness Vector': [-0.10301423607805887,
                                     0.05833746244044524,
                                     0.10601713722179339],
             'Low Fitness': 7.217006388830133e-06,
             'Low Fitness Vector': [-0.0024743206700619837,
                                    -0.0010462134296090192,
                                    1.345623423856801e-05]})])
Run 15:
Random Seed: 46
{'Best Fitness': 0.06170315504185826,
 'Best Vector': [0.02523105922781369, -0.12127845241220506, 0.2153092791140255]}
OrderedDict([('Generation 0',
             {'Average Fitness': 19.96149079095754,
```

```
  'High Fitness': 50.44859375630187,
  'High Fitness Vector': [3.3493472859667754,
                          4.385717027367495,
                          4.471683426870432],
  'Low Fitness': 0.3849927564293952,
  'Low Fitness Vector': [0.22191017101937938,
                         -0.31092460826246704,
                         0.4889524725414285]}),
('Generation 10',
 {'Average Fitness': 1.2012418867983579,
  'High Fitness': 5.03361624948524,
  'High Fitness Vector': [1.1935061486840572,
                          1.7936510815008555,
                          -0.6260791646185012],
  'Low Fitness': 0.1994042901494109,
  'Low Fitness Vector': [-0.03673361839765614,
                         -0.2073139994739255,
                         0.3937966950736782]}),
('Generation 20',
 {'Average Fitness': 0.27758262219958174,
  'High Fitness': 0.33940807008299956,
  'High Fitness Vector': [0.11337982656857334,
                          -0.3364817328354211,
                          0.46187999358962506],
  'Low Fitness': 0.15757157453095377,
  'Low Fitness Vector': [0.04430086660782892,
                         -0.20732152158375017,
                         0.3355991573841444]}),
('Generation 30',
 {'Average Fitness': 0.21557187556861218,
  'High Fitness': 0.288707280170878,
  'High Fitness Vector': [0.14055019840448998,
                          -0.3245424889361199,
                          0.404505988552067],
  'Low Fitness': 0.1463829914146863,
  'Low Fitness Vector': [-0.029613963376472666,
                         -0.18132625334152663,
                         0.3355991573841444]}),
('Generation 40',
 {'Average Fitness': 0.16007453845537536,
  'High Fitness': 0.22765297792951034,
  'High Fitness Vector': [0.12263094268390365,
                          -0.2691969303507166,
                          0.3743629822988898],
```

```
             'Low Fitness': 0.09962107563159717,
             'Low Fitness Vector': [0.0693653327400002,
                                    -0.16967328339517068,
                                    0.25694455267113236]}),
            ('Generation 50',
             {'Average Fitness': 0.11113579314942502,
             'High Fitness': 0.1589753201832707,
             'High Fitness Vector': [0.098399360043364,
                                     -0.1914839201849059,
                                     0.3355991573841444],
             'Low Fitness': 0.06170315504185826,
             'Low Fitness Vector': [0.02523105922781369,
                                    -0.12127845241220506,
                                    0.2153092791140255]})])
Run 16:
Random Seed: 123
{'Best Fitness': 0.18385756484476382,
 'Best Vector': [-0.00789683410747943,
                 -0.41452406706712575,
                 -0.10938465467318449]}
OrderedDict([('Generation 0',
             {'Average Fitness': 18.392935635520885,
             'High Fitness': 55.4248481593254,
             'High Fitness Vector': [2.832805035458608,
                                     4.950758598511494,
                                     4.7843550338232905],
             'Low Fitness': 1.5521103347638454,
             'Low Fitness Vector': [0.15906182025412208,
                                    0.538174083779255,
                                    1.112285182698057]}),
            ('Generation 10',
             {'Average Fitness': 0.938838573339636,
             'High Fitness': 2.4094252233940674,
             'High Fitness Vector': [-0.9127906667486552,
                                     0.582288669339326,
                                     1.112285182698057],
             'Low Fitness': 0.36260110653674016,
             'Low Fitness Vector': [-0.03819330946429606,
                                    -0.5866445672392572,
                                    -0.1303477248653361]}),
            ('Generation 20',
             {'Average Fitness': 0.43165850366796393,
             'High Fitness': 0.6034592704093108,
             'High Fitness Vector': [-0.09349207030581926,
```

```
                          -0.7489308118848228,
                          -0.18390525334743102],
             'Low Fitness': 0.30192033237486876,
             'Low Fitness Vector': [-0.022406062610367688,
                          -0.5312032330979619,
                          -0.13871346682799537]}),
            ('Generation 30',
             {'Average Fitness': 0.3932463181512152,
             'High Fitness': 0.5021878128003245,
             'High Fitness Vector': [0.039035745114975276,
                          -0.654881991383378,
                          -0.2679432790076744],
             'Low Fitness': 0.23129222581883419,
             'Low Fitness Vector': [-0.022406062610367688,
                          -0.4746793200157916,
                          -0.07395767253287644]}),
            ('Generation 40',
             {'Average Fitness': 0.34859340371264547,
             'High Fitness': 0.4601003237508646,
             'High Fitness Vector': [0.09234912860291464,
                          -0.6613024349351114,
                          -0.119377769061244],
             'Low Fitness': 0.23048560002445237,
             'Low Fitness Vector': [-0.02111196813216527,
                          -0.4738883280896024,
                          -0.07395767253287644]}),
            ('Generation 50',
             {'Average Fitness': 0.26325857410323766,
             'High Fitness': 0.35184571247945634,
             'High Fitness Vector': [-0.09269960635829427,
                          -0.5662147071957819,
                          -0.15051046746212887],
             'Low Fitness': 0.18385756484476382,
             'Low Fitness Vector': [-0.00789683410747943,
                          -0.41452406706712575,
                          -0.10938465467318449]})])
Run 17:
Random Seed: 823
{'Best Fitness': 0.014567538723469012,
 'Best Vector': [0.04634045095177876,
                 0.01566603169119951,
                 -0.11033891779469804]}
OrderedDict([('Generation 0',
              {'Average Fitness': 23.127805685785155,
```

```
  'High Fitness': 63.3782884423761,
  'High Fitness Vector': [4.667999932739873,
                          4.1986890855844505,
                          4.894800816469517],
  'Low Fitness': 0.21314125672018952,
  'Low Fitness Vector': [0.05335909226295077,
                         -0.19837209380702725,
                         -0.41345202429263717]}),
('Generation 10',
 {'Average Fitness': 0.5438450342983274,
  'High Fitness': 1.7612247829671297,
  'High Fitness Vector': [-0.479618906700443,
                          0.25362323486130717,
                          1.2111423293903518],
  'Low Fitness': 0.11754929370988822,
  'Low Fitness Vector': [0.0032822708640613332,
                         0.02065418556544274,
                         -0.3422161963240364]}),
('Generation 20',
 {'Average Fitness': 0.12458830381256272,
  'High Fitness': 0.20410616495326614,
  'High Fitness Vector': [0.007264483140068702,
                          0.06341075480786934,
                          -0.4472498948157172],
  'Low Fitness': 0.0895529444988955,
  'Low Fitness Vector': [-0.05739032243318733,
                         0.05650968962154227,
                         -0.28821164162605795]}),
('Generation 30',
 {'Average Fitness': 0.10149928113637402,
  'High Fitness': 0.13835661195701696,
  'High Fitness Vector': [0.03381050019578098,
                          -0.011572272301708642,
                          -0.3702425482670828],
  'Low Fitness': 0.05139819365376067,
  'Low Fitness Vector': [0.007897245302051093,
                         -0.011572272301708642,
                         -0.22627838978606643]}),
('Generation 40',
 {'Average Fitness': 0.060736130183001835,
  'High Fitness': 0.09274129343864984,
  'High Fitness Vector': [-0.020109377270261016,
                          0.01879808973532078,
                          -0.3032878800854973],
```

```
             'Low Fitness': 0.03879146507222249,
             'Low Fitness Vector': [-0.03794945844483288,
                                    0.04886804157771942,
                                    -0.18698453997142298]}),
            ('Generation 50',
             {'Average Fitness': 0.0430329149391968,
             'High Fitness': 0.09946525135520141,
             'High Fitness Vector': [-0.03793544739890146,
                                     0.11008382355131119,
                                     -0.29310016202344613],
             'Low Fitness': 0.014567538723469012,
             'Low Fitness Vector': [0.04634045095177876,
                                    0.01566603169119951,
                                    -0.11033891779469804]})])
Run 18:
Random Seed: 711
{'Best Fitness': 0.00010345296556149348,
 'Best Vector': [-0.0031388741950012606,
                 0.0037144422837771113,
                 0.00893327223753633]}
OrderedDict([('Generation 0',
             {'Average Fitness': 19.762473588576,
             'High Fitness': 41.999358569226175,
             'High Fitness Vector': [4.52581032243808,
                                     1.8694820156438077,
                                     4.2451662497154885],
             'Low Fitness': 1.6788182697927896,
             'Low Fitness Vector': [-0.4190915678049192,
                                    -0.7366119293770366,
                                    -0.9800935634351671]}),
            ('Generation 10',
             {'Average Fitness': 1.5294211900883818,
             'High Fitness': 2.710319304999038,
             'High Fitness Vector': [-0.9770242997765888,
                                     -0.024621125861490928,
                                     1.324815693901011],
             'Low Fitness': 0.024481678641840143,
             'Low Fitness Vector': [0.01415982219510474,
                                    0.14265533995746726,
                                    0.06269475304092528]}),
            ('Generation 20',
             {'Average Fitness': 0.034727603439968856,
             'High Fitness': 0.07744488489228352,
             'High Fitness Vector': [0.1492878936393759,
```

```
                        0.1593414682688551,
                        0.17253494195358618],
          'Low Fitness': 0.01302672879509646,
          'Low Fitness Vector': [0.055226699466361495,
                        0.07161933122932815,
                        0.06962335711104736]}),
         ('Generation 30',
          {'Average Fitness': 0.010603714972458912,
          'High Fitness': 0.02277386927403729,
          'High Fitness Vector': [-0.05384088991436982,
                        0.07161933122932815,
                        0.12143187078164358],
          'Low Fitness': 0.004929401904342626,
          'Low Fitness Vector': [0.00038547677475369094,
                        -0.009046626807065819,
                        0.06962335711104736]}),
         ('Generation 40',
          {'Average Fitness': 0.004212804787025738,
          'High Fitness': 0.01509756318666644,
          'High Fitness Vector': [-0.008660406030854446,
                        0.039396860235343556,
                        0.11606225897182937],
          'Low Fitness': 0.0002913521564114508,
          'Low Fitness Vector': [0.014062422339767858,
                        0.0037144422837771113,
                        0.00893327223753633]}),
         ('Generation 50',
          {'Average Fitness': 0.001135440618263926,
          'High Fitness': 0.006347706663689593,
          'High Fitness Vector': [-0.07908290731466658,
                        0.0037144422837771113,
                        0.00893327223753633],
          'Low Fitness': 0.00010345296556149348,
          'Low Fitness Vector': [-0.0031388741950012606,
                        0.0037144422837771113,
                        0.00893327223753633]})])
Run 19:
Random Seed: 911
{'Best Fitness': 0.12987766026429204,
 'Best Vector': [0.14568924133160815, -0.2441076686536167, 0.2215033889786081]}
OrderedDict([('Generation 0',
          {'Average Fitness': 18.681766215386745,
          'High Fitness': 44.41229018462699,
          'High Fitness Vector': [4.385939899071708,
```

```
                    2.3081195820546396,
                    4.45515492225509],
  'Low Fitness': 1.041653617389662,
  'Low Fitness Vector': [0.786436455626857,
                    -0.32035343042301556,
                    0.5661669349820198]}),
 ('Generation 10',
  {'Average Fitness': 0.6116596107746931,
  'High Fitness': 1.9451368928189372,
  'High Fitness Vector': [0.3330044240836696,
                    0.2864260419317919,
                    1.3237088308472253],
  'Low Fitness': 0.4209767742725255,
  'Low Fitness Vector': [0.23763050961678225,
                    -0.3116168121047852,
                    0.517110701480297]}),
 ('Generation 20',
  {'Average Fitness': 0.4444824841044492,
  'High Fitness': 0.5436295707793883,
  'High Fitness Vector': [0.3199959188179373,
                    -0.28327031861370094,
                    0.6008245245591635],
  'Low Fitness': 0.37287900834079923,
  'Low Fitness Vector': [0.28279714349649004,
                    -0.18530361848875243,
                    0.5084951847815274]}),
 ('Generation 30',
  {'Average Fitness': 0.39162920397918355,
  'High Fitness': 0.556737453081738,
  'High Fitness Vector': [0.34688958605454284,
                    -0.344868656522496,
                    0.5634453637372613],
  'Low Fitness': 0.2558301128084775,
  'Low Fitness Vector': [0.14568924133160815,
                    -0.23061382518728185,
                    0.4259366401252512]}),
 ('Generation 40',
  {'Average Fitness': 0.29552496610993634,
  'High Fitness': 0.39460769375026067,
  'High Fitness Vector': [0.2272903130886428,
                    -0.2884115844549398,
                    0.5096720173587309],
  'Low Fitness': 0.1960957343188388,
  'Low Fitness Vector': [0.14568924133160815,
```

```
                        -0.2441076686536167,
                        0.33953177374666327]}),
          ('Generation 50',
           {'Average Fitness': 0.220445615373803,
            'High Fitness': 0.34129451841951053,
            'High Fitness Vector': [0.1840643682994842,
                        -0.3174305166490108,
                        0.45459068824819154],
            'Low Fitness': 0.12987766026429204,
            'Low Fitness Vector': [0.14568924133160815,
                        -0.2441076686536167,
                        0.2215033889786081]})])
Run 20:
Random Seed: 194
{'Best Fitness': 0.2924038104219147,
 'Best Vector': [0.4836527562451746, -0.20184883274459892, 0.133194859202611]}
OrderedDict([('Generation 0',
           {'Average Fitness': 17.428005948488714,
            'High Fitness': 46.90577111083829,
            'High Fitness Vector': [3.800332220665399,
                        3.964579414294872,
                        4.092109015063017],
            'Low Fitness': 0.3871973774871995,
            'Low Fitness Vector': [0.32412766273201443,
                        -0.42255130690744425,
                        0.32185249536067273]}),
          ('Generation 10',
           {'Average Fitness': 1.01908461983506,
            'High Fitness': 4.383462813790947,
            'High Fitness Vector': [0.9669740385523642,
                        -0.08275181044255864,
                        1.8551485548131064],
            'Low Fitness': 0.3474919875497008,
            'Low Fitness Vector': [0.3202727240799759,
                        -0.4108594206551834,
                        0.2758838636437069]}),
          ('Generation 20',
           {'Average Fitness': 0.6009911941610399,
            'High Fitness': 0.7230880523033802,
            'High Fitness Vector': [-0.6293288328686606,
                        -0.49384114199764345,
                        0.2883646972496964],
            'Low Fitness': 0.4592964812102028,
            'Low Fitness Vector': [-0.48889021174930025,
```

```
                        -0.40222523163742663,
                        0.24186298828085628]}),
          ('Generation 30',
           {'Average Fitness': 0.5434731777600206,
            'High Fitness': 0.6448981832992275,
            'High Fitness Vector': [-0.6656709350004427,
                        -0.35161064072308673,
                        0.279553835468527],
            'Low Fitness': 0.43020934932307103,
            'Low Fitness Vector': [-0.5644837917465713,
                        -0.24031389788771668,
                        0.23198411294848764]}),
          ('Generation 40',
           {'Average Fitness': 0.49464794851687854,
            'High Fitness': 0.5874403230964163,
            'High Fitness Vector': [-0.6027492482971709,
                        -0.373646992810581,
                        0.2907259732759033],
            'Low Fitness': 0.37623639023515415,
            'Low Fitness Vector': [0.5395721630944414,
                        -0.2595330432348379,
                        0.133194859202611]}),
          ('Generation 50',
           {'Average Fitness': 0.41798902043778,
            'High Fitness': 0.5947201054277591,
            'High Fitness Vector': [-0.6148991251872036,
                        -0.36786731549415663,
                        0.2851189391515294],
            'Low Fitness': 0.2924038104219147,
            'Low Fitness Vector': [0.4836527562451746,
                        -0.20184883274459892,
                        0.133194859202611]})])
Run 21:
Random Seed: 8
{'Best Fitness': 0.23406589257875982,
 'Best Vector': [0.09186093116571864,
              0.023355866673953438,
              -0.47442804027168906]}
OrderedDict([('Generation 0',
           {'Average Fitness': 18.57398843484723,
            'High Fitness': 42.668748938480675,
            'High Fitness Vector': [4.6999983926373625,
                        4.06095779663169,
                        2.0217284243100453],
```

```
  'Low Fitness': 0.5531414086668922,
  'Low Fitness Vector': [-0.4293296768948538,
                         -0.4884647579545005,
                         -0.36085955362260247]}),
 ('Generation 10',
  {'Average Fitness': 1.5043902463758354,
  'High Fitness': 4.646436404408044,
  'High Fitness Vector': [1.0674990593170226,
                          1.1380923445624633,
                          1.4871543221917594],
  'Low Fitness': 0.36070640735919235,
  'Low Fitness Vector': [0.14805419043800278,
                         -0.10208354529685819,
                         -0.5730316865868451]}),
 ('Generation 20',
  {'Average Fitness': 0.5189696274115334,
  'High Fitness': 1.2252226460888669,
  'High Fitness Vector': [0.8019635518027579,
                          -0.06217395408454276,
                          -0.7604022008794197],
  'Low Fitness': 0.3622746998321909,
  'Low Fitness Vector': [0.12900926702937182,
                         -0.14717212587372153,
                         -0.5691850966237122]}),
 ('Generation 30',
  {'Average Fitness': 0.38868052608303477,
  'High Fitness': 0.4728772739470364,
  'High Fitness Vector': [0.32458756571784064,
                          -0.1978759012508561,
                          -0.5730316865868451],
  'Low Fitness': 0.2982312850302924,
  'Low Fitness Vector': [0.09186093116571864,
                         -0.1328003886030305,
                         -0.5216866024181027]}),
 ('Generation 40',
  {'Average Fitness': 0.36825464401994196,
  'High Fitness': 0.5147339892259267,
  'High Fitness Vector': [0.23041141467062987,
                          -0.10653112173460211,
                          -0.6710407508619526],
  'Low Fitness': 0.2626932739239201,
  'Low Fitness Vector': [0.09186093116571864,
                         -0.15780249820466233,
                         -0.4789083574230614]}),
```

```
          ('Generation 50',
           {'Average Fitness': 0.29839899995449387,
            'High Fitness': 0.394211922883687,
            'High Fitness Vector': [0.09186093116571864,
                                    -0.08135342134247824,
                                    -0.615755725141819],
            'Low Fitness': 0.23406589257875982,
            'Low Fitness Vector': [0.09186093116571864,
                                   0.023355866673953438,
                                   -0.47442804027168906]})])
Run 22:
Random Seed: 238
{'Best Fitness': 0.009427347725947513,
 'Best Vector': [0.05097084155075844,
                 0.0029607733086107063,
                 0.08258665060995032]}
OrderedDict([('Generation 0',
              {'Average Fitness': 22.719937971230483,
               'High Fitness': 58.15475983779056,
               'High Fitness Vector': [3.5824820337019094,
                                       4.798850821708359,
                                       4.721399486061585],
               'Low Fitness': 0.9794286246466966,
               'Low Fitness Vector': [0.9638381343110378,
                                      0.21739425763332965,
                                      -0.05643059668803651 6]}),
             ('Generation 10',
              {'Average Fitness': 0.6360112274631051,
               'High Fitness': 0.9934749770332937,
               'High Fitness Vector': [-0.8178893870416994,
                                       0.3975098939321515,
                                       -0.4080659405339981 4],
               'Low Fitness': 0.14650009780488693,
               'Low Fitness Vector': [-0.00614646604697992,
                                      0.02055803813053969,
                                      0.382151391242062]}),
             ('Generation 20',
              {'Average Fitness': 0.17445057750338244,
               'High Fitness': 0.4601126029845065,
               'High Fitness Vector': [-0.5015824050377 3,
                                       -0.020404243347995177,
                                       0.4561922410502713 4],
               'Low Fitness': 0.1005444282287662,
               'Low Fitness Vector': [0.07146104248638743,
```

```
                        0.02055803813053969,
                        0.30824521846047864]}),
          ('Generation 30',
           {'Average Fitness': 0.11240564950579418,
            'High Fitness': 0.1701337880429504,
            'High Fitness Vector': [-0.07986082836461297,
                        0.02055803813053969,
                        0.4041452748753739],
            'Low Fitness': 0.07258045542181811,
            'Low Fitness Vector': [0.03896800790340735,
                        0.04355335922512609,
                        0.2629924992886018]}),
          ('Generation 40',
           {'Average Fitness': 0.08900016852512005,
            'High Fitness': 0.14334511873096475,
            'High Fitness Vector': [-0.00614646604697992,
                        -0.024994921210500995,
                        0.37773349546972784],
            'Low Fitness': 0.0449874486131735,
            'Low Fitness Vector': [0.05097084155075844,
                        0.0029607733086107063,
                        0.20586562546038614]}),
          ('Generation 50',
           {'Average Fitness': 0.02855299400863166,
            'High Fitness': 0.07115528033532864,
            'High Fitness Vector': [0.08564241731552465,
                        -0.01117710198629376,
                        0.25238012814575983],
            'Low Fitness': 0.009427347725947513,
            'Low Fitness Vector': [0.05097084155075844,
                        0.0029607733086107063,
                        0.08258665060995032]})])
Run 23:
Random Seed: 234
{'Best Fitness': 0.10477879752518902,
 'Best Vector': [0.06791058631878089, 0.10944856126753419, 0.2969645807625806]}
OrderedDict([('Generation 0',
           {'Average Fitness': 21.958436273124658,
            'High Fitness': 64.9239808311477,
            'High Fitness Vector': [4.431795781361366,
                        4.70746658276933,
                        4.8086302993228065],
            'Low Fitness': 0.3668328520065032,
            'Low Fitness Vector': [-0.09941637889740496,
```

```
                    0.21518278248837897,
                    0.557355905803455]}),
 ('Generation 10',
  {'Average Fitness': 1.0115714899824664,
   'High Fitness': 3.5833116591132406,
   'High Fitness Vector': [1.0667688773526534,
                           -0.9777933835885773,
                           1.2203425422541827],
   'Low Fitness': 0.30459924579475034,
   'Low Fitness Vector': [0.11220240400973573,
                          0.19487232062496562,
                          0.5040184966679566]}),
 ('Generation 20',
  {'Average Fitness': 0.3720235727534073,
   'High Fitness': 0.7119075166465814,
   'High Fitness Vector': [0.1667372018025366,
                           0.718697595678557,
                           0.40936534800530316],
   'Low Fitness': 0.2087086031452976,
   'Low Fitness Vector': [0.1072117796241772,
                          0.19487232062496562,
                          0.3990476363911395]}),
 ('Generation 30',
  {'Average Fitness': 0.2870179508324638,
   'High Fitness': 0.40582136512278744,
   'High Fitness Vector': [0.11408780124614933,
                           0.21176512354411448,
                           0.5898820824367803],
   'Low Fitness': 0.18561984001453655,
   'Low Fitness Vector': [0.11052533429353315,
                          0.19487232062496562,
                          0.3680064797637127]}),
 ('Generation 40',
  {'Average Fitness': 0.19673325765475722,
   'High Fitness': 0.2944697056265244,
   'High Fitness Vector': [0.26192751634413136,
                           0.2000195421424836,
                           0.43111003765794],
   'Low Fitness': 0.10477879752518902,
   'Low Fitness Vector': [0.06791058631878089,
                          0.10944856126753419,
                          0.2969645807625806]}),
 ('Generation 50',
  {'Average Fitness': 0.19378995022211903,
```

```
            'High Fitness': 0.28270197361624505,
            'High Fitness Vector': [0.26192751634413136,
                                    0.20162479002397884,
                                    0.41646535731778017],
            'Low Fitness': 0.13188458725555302,
            'Low Fitness Vector': [0.12440315807381075,
                                   0.1337940808861586,
                                   0.3138591809022695]})])
Run 24:
Random Seed: 995
{'Best Fitness': 0.36500918741430893,
 'Best Vector': [-0.07837915176536242,
                 0.5968383287771136,
                 -0.051477230747129926]}
OrderedDict([('Generation 0',
              {'Average Fitness': 22.21088838016609,
               'High Fitness': 45.06745917581607,
               'High Fitness Vector': [1.4371076167000174,
                                       4.5349509451334775,
                                       4.736707801740766],
               'Low Fitness': 2.91582688831919,
               'Low Fitness Vector': [0.5258928691269564,
                                      0.30107340614334377,
                                      1.5964392824764282]}),
             ('Generation 10',
              {'Average Fitness': 1.4507711118926077,
               'High Fitness': 4.6755175372312445,
               'High Fitness Vector': [-0.283725664562653,
                                       1.8960530806123876,
                                       -1.0],
               'Low Fitness': 0.5936744060963886,
               'Low Fitness Vector': [-0.14682271142473408,
                                      0.7540523023993762,
                                      -0.059351687023012634]}),
             ('Generation 20',
              {'Average Fitness': 0.6332032671658783,
               'High Fitness': 0.8574815816464829,
               'High Fitness Vector': [-0.164085481118267,
                                       0.9084446220947584,
                                       -0.07270423041186325],
               'Low Fitness': 0.5373778525458052,
               'Low Fitness Vector': [-0.11573784073509569,
                                      0.717750913106051,
                                      -0.09389478954240692]}),
```

```
          ('Generation 30',
           {'Average Fitness': 0.622348102049065,
            'High Fitness': 0.7527782892830007,
            'High Fitness Vector': [-0.2633132207758093,
                                    0.8267010445555026,
                                    -0.0031336844945561923],
            'Low Fitness': 0.47862309537248776,
            'Low Fitness Vector': [-0.1813641315142059,
                                   0.6609946411805263,
                                   -0.09389478954240692]}),
          ('Generation 40',
           {'Average Fitness': 0.597055933845015,
            'High Fitness': 0.7346596121297014,
            'High Fitness Vector': [-0.1813641315142059,
                                    0.8327537859883258,
                                    -0.09103733218800358],
            'Low Fitness': 0.44231269481327046,
            'Low Fitness Vector': [-0.032573112304308954,
                                   0.6558535233953602,
                                   -0.10539375227200651]}),
          ('Generation 50',
           {'Average Fitness': 0.5162207664892527,
            'High Fitness': 0.6774062598676911,
            'High Fitness Vector': [-0.21827288467055161,
                                    0.7866019596676214,
                                    -0.10497887755327229],
            'Low Fitness': 0.36500918741430893,
            'Low Fitness Vector': [-0.07837915176536242,
                                   0.5968383287771136,
                                   -0.051477230747129926]})])
Run 25:
Random Seed: 204
{'Best Fitness': 0.060768035680048536,
 'Best Vector': [-0.059260118610160006,
                 -0.17205848920193847,
                 -0.16628935719373242]}
OrderedDict([('Generation 0',
           {'Average Fitness': 24.419741935281785,
            'High Fitness': 57.00654293263412,
            'High Fitness Vector': [4.904155620535963,
                                    4.169564590986889,
                                    3.9459513052986326],
            'Low Fitness': 0.6279787978501206,
            'Low Fitness Vector': [-0.09224020210221529,
```

```
                    0.1889680129708482,
                    0.7640429523528847]}),
 ('Generation 10',
  {'Average Fitness': 0.83244158142177,
   'High Fitness': 4.3628430291279106,
   'High Fitness Vector': [2.0204889002778357,
                    -0.3331667328661658,
                    -0.4116643791892338],
   'Low Fitness': 0.22214556617168357,
   'Low Fitness Vector': [-0.051793525183026,
                    -0.22359659171718502,
                    -0.4116643791892338]}),
 ('Generation 20',
  {'Average Fitness': 0.24667501984036563,
   'High Fitness': 0.383091023366281,
   'High Fitness Vector': [-0.051793525183026,
                    0.16359993957221586,
                    -0.594679337027413],
   'Low Fitness': 0.17454169496212463,
   'Low Fitness Vector': [0.0035347591541436615,
                    -0.2486696052040215,
                    -0.33569722651151424]}),
 ('Generation 30',
  {'Average Fitness': 0.1758836179200873,
   'High Fitness': 0.2677120069757417,
   'High Fitness Vector': [0.06350852186537646,
                    -0.19479907015439918,
                    -0.4751126149590202],
   'Low Fitness': 0.11263839649653071,
   'Low Fitness Vector': [0.0019121605564796362,
                    -0.2486696052040215,
                    -0.22538448834428032]}),
 ('Generation 40',
  {'Average Fitness': 0.13234280099516432,
   'High Fitness': 0.1996153944695679,
   'High Fitness Vector': [-0.07869097875103188,
                    -0.2271255145441725,
                    -0.3766126989040829],
   'Low Fitness': 0.08075796015046159,
   'Low Fitness Vector': [0.0019121605564796362,
                    -0.17368328149774168,
                    -0.2249186998020486]}),
 ('Generation 50',
  {'Average Fitness': 0.09716644552323701,
```

```
            'High Fitness': 0.17636006058312229,
            'High Fitness Vector': [-0.012152938444894776,
                                    -0.19279952725968003,
                                    -0.37288162861519575],
            'Low Fitness': 0.060768035680048536,
            'Low Fitness Vector': [-0.059260118610160006,
                                   -0.17205848920193847,
                                   -0.16628935719373242]})])
Run 26:
Random Seed: 899
{'Best Fitness': 0.009212349993631665,
 'Best Vector': [-0.007441170138427357,
                 -0.03630712432760062,
                 0.08853683811647439]}
OrderedDict([('Generation 0',
              {'Average Fitness': 18.994845261177762,
               'High Fitness': 48.966916766321994,
               'High Fitness Vector': [4.575740533682047,
                                       4.840277007793554,
                                       2.1450486760365615],
               'Low Fitness': 1.5150242912659784,
               'Low Fitness Vector': [0.2772863224005888,
                                      1.103472914474521,
                                      0.46955735932539877]}),
             ('Generation 10',
              {'Average Fitness': 0.3421810618977366,
               'High Fitness': 1.2618714131705702,
               'High Fitness Vector': [0.29302454133013717,
                                       -0.9328487244565997,
                                       -0.5529930276491086],
               'Low Fitness': 0.11837160558689266,
               'Low Fitness Vector': [0.04587711780016446,
                                      -0.1887912584987026,
                                      0.28394498827011994]}),
             ('Generation 20',
              {'Average Fitness': 0.13669124863331045,
               'High Fitness': 0.23795804471873877,
               'High Fitness Vector': [0.34017367086738476,
                                       -0.17787656840175056,
                                       0.3009980810253851],
               'Low Fitness': 0.06899093060006305,
               'Low Fitness Vector': [-0.04244892080940401,
                                      -0.16461392204870848,
                                      0.20022806094531737]}),
```

```
          ('Generation 30',
           {'Average Fitness': 0.10367949838117724,
            'High Fitness': 0.18758460822155182,
            'High Fitness Vector': [0.006320935577654799,
                                    -0.2086852681344696,
                                    0.3794668797913982],
            'Low Fitness': 0.06267328857644218,
            'Low Fitness Vector': [0.007993098903303384,
                                   -0.1887912584987026,
                                   0.16421711135213723]}),
          ('Generation 40',
           {'Average Fitness': 0.06740095074467392,
            'High Fitness': 0.110884660203688,
            'High Fitness Vector': [-0.005020150528070595,
                                    -0.14726473464120238,
                                    0.2986177426467151],
            'Low Fitness': 0.02514650014300794,
            'Low Fitness Vector': [-0.040222966388296345,
                                   -0.10048660123302201,
                                   0.11589243327573005]}),
          ('Generation 50',
           {'Average Fitness': 0.026352122936202736,
            'High Fitness': 0.08528361839414192,
            'High Fitness Vector': [0.03732778938103564,
                                    -0.22048360826826505,
                                    0.18782234430193123],
            'Low Fitness': 0.009212349993631665,
            'Low Fitness Vector': [-0.007441170138427357,
                                   -0.03630712432760062,
                                   0.08853683811647439]})])
Run 27:
Random Seed: 375
{'Best Fitness': 0.23203680678640498,
 'Best Vector': [-0.19745738831536577,
                 -0.022834099118795456,
                 0.43877783729754727]}
OrderedDict([('Generation 0',
              {'Average Fitness': 19.40060887218309,
               'High Fitness': 56.05321067737911,
               'High Fitness Vector': [4.489222355123953,
                                       3.9670715435888333,
                                       4.490260203115424],
               'Low Fitness': 2.8494192633186306,
               'Low Fitness Vector': [0.8981472636314201,
```

```
                    1.2680685902526871,
                    -0.659357874423735]}),
 ('Generation 10',
  {'Average Fitness': 1.0711562540497306,
   'High Fitness': 3.310904240986249,
   'High Fitness Vector': [1.5603803821000188,
                           0.8653319591276141,
                           -0.3568163458363606],
   'Low Fitness': 0.554031084884447,
   'Low Fitness Vector': [-0.4194515355137027,
                          0.059366235678028234,
                          0.6120189084506007]}),
 ('Generation 20',
  {'Average Fitness': 0.6971145117771791,
   'High Fitness': 0.9990316823243781,
   'High Fitness Vector': [-0.43260533433172565,
                           0.7938158053994022,
                           -0.42631065448826444],
   'Low Fitness': 0.5292245872081823,
   'Low Fitness Vector': [-0.3265467777096195,
                          0.059366235678028234,
                          0.6473541837642416]}),
 ('Generation 30',
  {'Average Fitness': 0.5950864668299443,
   'High Fitness': 0.7714270697174825,
   'High Fitness Vector': [-0.4663917086352779,
                           0.1400638802646952,
                           0.7309500347348961],
   'Low Fitness': 0.4011334423105428,
   'Low Fitness Vector': [-0.3908287202689882,
                          0.017738614917518797,
                          0.49806796249533636]}),
 ('Generation 40',
  {'Average Fitness': 0.5176555604685974,
   'High Fitness': 0.6600274511561524,
   'High Fitness Vector': [-0.49976198273182754,
                           0.14070217775678945,
                           0.6248746345840913],
   'Low Fitness': 0.3644730139656036,
   'Low Fitness Vector': [-0.3407149251825576,
                          0.017738614917518797,
                          0.49806796249533636]}),
 ('Generation 50',
  {'Average Fitness': 0.3725754505729613,
```

```
            'High Fitness': 0.5258989399675422,
            'High Fitness Vector': [-0.3921222813814217,
                                    -0.0019469571260818039,
                                    0.6100289056837558],
            'Low Fitness': 0.23203680678640498,
            'Low Fitness Vector': [-0.19745738831536577,
                                   -0.022834099118795456,
                                   0.43877783729754727]})])
Run 28:
Random Seed: 112
{'Best Fitness': 1.1565880585008177,
 'Best Vector': [0.7442696950355625, 0.6096310557529626, -0.48062527546313744]}
OrderedDict([('Generation 0',
              {'Average Fitness': 20.497858039464386,
               'High Fitness': 46.403523753156286,
               'High Fitness Vector': [4.093417060690058,
                                       4.527893809103013,
                                       3.0241756188909505],
               'Low Fitness': 3.095194561790023,
               'Low Fitness Vector': [-0.02453921604939202,
                                      0.3255383309553057,
                                      1.7287617486931328]}),
             ('Generation 10',
              {'Average Fitness': 2.201514792130483,
               'High Fitness': 5.1496980346842935,
               'High Fitness Vector': [2.029205282693773,
                                       0.8810144226702015,
                                       -0.5058038576553863],
               'Low Fitness': 1.4663819300426437,
               'Low Fitness Vector': [0.8572065474215358,
                                      0.6081391248738927,
                                      -0.6014529656572574]}),
             ('Generation 20',
              {'Average Fitness': 1.5688276079705883,
               'High Fitness': 2.158666681869295,
               'High Fitness Vector': [0.8915889351961932,
                                       1.1092458603621447,
                                       0.36511569916186687],
               'Low Fitness': 1.3669390820430802,
               'Low Fitness Vector': [0.7885327232421261,
                                      0.6665042287238065,
                                      -0.5485684456043025]}),
             ('Generation 30',
              {'Average Fitness': 1.458323063288145,
```

```
            'High Fitness': 1.7831657915392398,
            'High Fitness Vector': [0.8330022620402948,
                                    0.7732331150333717,
                                    -0.7009875696407037],
            'Low Fitness': 1.2143616605449394,
            'Low Fitness Vector': [0.7725972626233146,
                                   0.590708501421707,
                                   -0.5181878005897299]}),
           ('Generation 40',
            {'Average Fitness': 1.4839660095217981,
            'High Fitness': 1.7338938590805189,
            'High Fitness Vector': [0.9621179424246158,
                                    0.6649315222417821,
                                    -0.6050528858491373],
            'Low Fitness': 1.3091059356513954,
            'Low Fitness Vector': [0.8018754786206457,
                                   0.5990394194813177,
                                   -0.5543044527565981]}),
           ('Generation 50',
            {'Average Fitness': 1.3951785536461767,
            'High Fitness': 1.7802724693036716,
            'High Fitness Vector': [0.8974142035739245,
                                    0.7919959519664591,
                                    -0.5896292297674808],
            'Low Fitness': 1.1565880585008177,
            'Low Fitness Vector': [0.7442696950355625,
                                   0.6096310557529626,
                                   -0.48062527546313744]})])
Run 29:
Random Seed: 276
{'Best Fitness': 0.809425463330915,
 'Best Vector': [0.08292753182537524, 0.6220589694265316, 0.6446635761017275]}
OrderedDict([('Generation 0',
            {'Average Fitness': 22.200516292171496,
            'High Fitness': 58.618826169426754,
            'High Fitness Vector': [4.855510261262794,
                                    3.3668737626080105,
                                    4.8689842204365625],
            'Low Fitness': 0.8891579108115842,
            'Low Fitness Vector': [-0.23781559129778884,
                                   0.4257286051781124,
                                   -0.8070667940637666]}),
           ('Generation 10',
            {'Average Fitness': 2.765428268715314,
```

```
  'High Fitness': 4.739702107894251,
  'High Fitness Vector': [1.0259992465217063,
                          -0.13832019924184147,
                          1.9151749728191518],
  'Low Fitness': 1.4019638564256391,
  'Low Fitness Vector': [0.15709516955323438,
                         0.9167400124259599,
                         0.7327159843663373]}),
 ('Generation 20',
  {'Average Fitness': 1.502447426375874,
  'High Fitness': 3.0886181593685995,
  'High Fitness Vector': [0.6579155199607958,
                          -0.8943026498373988,
                          1.3623465412504714],
  'Low Fitness': 1.1573881990635435,
  'Low Fitness Vector': [0.16952643996858927,
                         0.8028635618049953,
                         0.6957435492628848]}),
 ('Generation 30',
  {'Average Fitness': 1.2662376718791248,
  'High Fitness': 1.870956522727446,
  'High Fitness Vector': [-0.5718045869608602,
                          1.0046869104153935,
                          0.7311636267607527],
  'Low Fitness': 0.8756592916163359,
  'Low Fitness Vector': [0.23554617666491484,
                         0.6610727931288479,
                         0.61899923461964]}),
 ('Generation 40',
  {'Average Fitness': 1.0391948285861359,
  'High Fitness': 1.2790007205005962,
  'High Fitness Vector': [0.14292211693549822,
                          0.9167400124259599,
                          0.6466542651282235],
  'Low Fitness': 0.910477635337112,
  'Low Fitness Vector': [0.15606611128164413,
                         0.7172021794957122,
                         0.6097065179027839]}),
 ('Generation 50',
  {'Average Fitness': 0.9558577009186968,
  'High Fitness': 1.1828409738577457,
  'High Fitness Vector': [0.2725953839255647,
                          0.8320956099274113,
                          0.6450966024245897],
```

```
            'Low Fitness': 0.809425463330915,
            'Low Fitness Vector': [0.08292753182537524,
                                   0.6220589694265316,
                                   0.6446635761017275]})])
Run 30:
Random Seed: 419
{'Best Fitness': 0.011147354083049966,
 'Best Vector': [0.03363133589894888,
                 0.0866186333619381,
                 -0.050134815081061004]}
OrderedDict([('Generation 0',
              {'Average Fitness': 21.624658204360717,
               'High Fitness': 49.45389369102641,
               'High Fitness Vector': [4.498017673434243,
                                       3.7570535933399682,
                                       3.886679688020462],
               'Low Fitness': 0.6415853258815475,
               'Low Fitness Vector': [0.22882433009924052,
                                      0.7140119710226176,
                                      -0.2818007400142484]}),
             ('Generation 10',
              {'Average Fitness': 0.46326035423839684,
               'High Fitness': 1.6864984040794422,
               'High Fitness Vector': [0.22882433009924052,
                                       -0.6709491005655841,
                                       1.088101619557801],
               'Low Fitness': 0.11170944060145016,
               'Low Fitness Vector': [-0.2223347609089899,
                                      0.156867394229385,
                                      -0.19408584523523387]}),
             ('Generation 20',
              {'Average Fitness': 0.09046946231858749,
               'High Fitness': 0.14313172011764524,
               'High Fitness Vector': [0.19177733026158486,
                                       0.22342206791367047,
                                       -0.2375621082676349],
               'Low Fitness': 0.0585870317224723,
               'Low Fitness Vector': [0.1197508445163461,
                                      0.156867394229385,
                                      -0.14014059935570747]}),
             ('Generation 30',
              {'Average Fitness': 0.05703230546702956,
               'High Fitness': 0.07987837577094022,
               'High Fitness Vector': [0.09712730133181643,
```

```
                                0.1934889568316278,
                                -0.18167742482528537],
                  'Low Fitness': 0.032988305560003185,
                  'Low Fitness Vector': [0.1280463788845052,
                                0.11392558200893971,
                                -0.06011149788928302]}),
                 ('Generation 40',
                  {'Average Fitness': 0.0451621779882551,
                  'High Fitness': 0.06297806998015842,
                  'High Fitness Vector': [0.1129317230528716,
                                0.15592263378337395,
                                -0.16097399846721766],
                  'Low Fitness': 0.022968428895956707,
                  'Low Fitness Vector': [0.06842443484323273,
                                0.09543595946558497,
                                -0.09580450539105209]}),
                 ('Generation 50',
                  {'Average Fitness': 0.024695657665248918,
                  'High Fitness': 0.05378243528839357,
                  'High Fitness Vector': [0.07941189073396639,
                                0.1827564915801667,
                                -0.11864338027786718],
                  'Low Fitness': 0.011147354083049966,
                  'Low Fitness Vector': [0.03363133589894888,
                                0.0866186333619381,
                                -0.0501348150810610041})])

Averages Across All 30 Runs:
{'Average Avg-of-Generation Fitness': {'Generation 0': 21.12320002593753,
                                       'Generation 10': 1.2210781592915534,
                                       'Generation 20': 0.5526066227472156,
                                       'Generation 30': 0.4745854224740938,
                                       'Generation 40': 0.42343158690920296,
                                       'Generation 50': 0.36314621289362653},
 'Average Best-of-Generation Fitness': {'Generation 0': 1.1317750938801319,
                                        'Generation 10': 0.45410821024342957,
                                        'Generation 20': 0.4209748672207123,
                                        'Generation 30': 0.3636089839610129,
                                        'Generation 40': 0.332663582999799,
                                        'Generation 50': 0.2820540025689745},
 'Best-of-Runs': {'Average Fitness': 0.26203359103355445,
                  'Standard Deviation': 2.049849182629643}}
```

End Program Output

Begin Tabular Data Format

Averages Across All 30 Runs
Average Avg-of-Generation Fitness

Generation 0	21.12320002593753
Generation 10	1.2210781592915534
Generation 20	0.5526066227472156
Generation 30	0.4745854224740938
Generation 40	0.42343158690920296
Generation 50	0.36314621289362653

Average Best-of-Generation Fitness

Generation 0	1.1317750938801319
Generation 10	0.45410821024342957
Generation 20	0.4209748672207123
Generation 30	0.3636089839610129
Generation 40	0.332663582999799
Generation 50	0.2820540025689745

Best-of-Runs

Average Fitness	0.26203359103355445
Standard Deviation	2.049849182629643

Run 1

Random Seed:	54	Best Fitness:	0.010864	Best Vector:	0.075131 -0.005082 0.072068
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	20.713668	61.056131	3.473281 4.993205 4.905135	1.349053	-0.924184 0.594827 0.375654
10	0.645426	1.401399	-0.924184 0.560211 -0.483162	0.191014	0.327869 -0.232312 0.171891
20	0.204064	0.278902	0.456374 -0.199733 0.175305	0.137268	0.305147 -0.176543 0.113955
30	0.138819	0.260681	0.446976 -0.170765 0.178138	0.068357	0.190585 -0.085946 0.156996
40	0.080051	0.150913	0.298847 -0.191764 0.157576	0.035078	0.126714 -0.101254 0.093645
50	0.035843	0.06369	0.225336 -0.110068 0.028259	0.010864	0.075131 -0.005082 0.072068

	Run 2				
Random Seed:	30	Best Fitness:	0.271307	Best Vector:	0.292392 0.069535 0.425417
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	20.456802	52.296996	1.769198 4.960910 4.955432	0.527874	0.456053 -0.564129 -0.040594
10	0.907275	1.875992	0.456053 1.016500 -0.796703	0.457616	0.372505 0.235697 0.513130
20	0.52989	0.645072	0.539189 0.182120 0.566727	0.447498	0.399489 0.223810 0.487664
30	0.494308	0.618165	0.519087 0.223810 0.546464	0.370468	0.353094 0.204433 0.451664
40	0.412167	0.552423	0.399346 0.341512 0.525657	0.286305	0.325609 0.150528 0.397020
50	0.342323	0.470124	0.403522 0.297414 0.467802	0.271307	0.292392 0.069535 0.425417

Run 3

Random Seed:	101	Best Fitness:	0.028915	Best Vector:	0.103386 0.089716 -0.100881
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	19.467137	44.826213	3.755361 3.429300 4.354696	0.408297	0.500109 0.226629 -0.326844
10	1.333156	2.837684	0.216040 1.091343 1.264904	0.264011	0.216040 0.150424 -0.441260
20	0.272386	0.436684	0.431138 0.199031 -0.459554	0.180014	0.188255 0.161592 -0.344184
30	0.217539	0.337251	0.266141 0.161592 -0.490212	0.137458	0.126238 0.195952 -0.288314
40	0.126526	0.22316	0.221988 0.228623 -0.348731	0.0636	0.099722 0.111872 -0.202830
50	0.061614	0.099412	0.118675 0.046828 -0.288333	0.028915	0.103386 0.089716 -0.100881

Run 4

Random Seed:	67	Best Fitness:	0.00141	Best Vector:	-0.009745 -0.014355 0.033302
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	23.061176	58.447901	4.912639 4.250704 4.030557	0.38388	0.197192 -0.399847 -0.430252
10	0.43588	0.912749	-0.923709 -0.198279 0.142113	0.056449	-0.102243 -0.197342 0.083972
20	0.081115	0.162277	0.325935 -0.143658 0.188164	0.03782	-0.102243 -0.134040 0.096953
30	0.037256	0.072358	-0.112124 -0.168237 0.177432	0.02106	-0.078580 -0.096397 0.074784
40	0.015682	0.029864	-0.094835 -0.111177 0.092248	0.005965	-0.064853 -0.039866 0.013042
50	0.006485	0.0199	-0.065641 -0.034966 0.119871	0.00141	-0.009745 -0.014355 0.033302

Run 5

Random Seed:	34	Best Fitness:	0.670121	Best Vector:	-0.023227 0.639613 -0.510368
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	24.118486	45.257194	-0.075995 4.595781 4.912251	1.113321	0.403121 0.777050 -0.589073
10	1.541554	2.010197	0.832843 0.732585 0.883113	0.973236	-0.104300 0.777050 -0.598791
20	1.110259	1.467932	0.350608 0.751743 0.883113	0.855057	-0.104300 0.696870 -0.598791
30	1.043725	1.303971	0.435942 0.841001 -0.637686	0.821524	-0.132630 0.718076 -0.536935
40	0.951085	1.284582	0.554112 0.803801 -0.575714	0.766308	-0.147419 0.669255 -0.544678
50	0.892695	1.196633	0.603723 0.730609 -0.546226	0.670121	-0.023227 0.639613 -0.510368

Run 6

Random Seed:	22	Best Fitness:	0.172884	Best Vector:	-0.064783 0.153613 0.380907
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	21.501731	57.420152	4.627933 3.643713 4.767152	0.454368	-0.291452 0.298082 0.529689
10	1.794798	5.476887	2.169444 0.644037 -0.596335	0.338404	-0.171693 0.242225 0.500253
20	0.719786	0.857188	-0.197352 0.605572 -0.671955	0.34392	-0.146660 0.242225 0.513555
30	0.334748	0.426146	-0.266287 0.276352 0.528078	0.271432	-0.096417 0.235488 0.454622
40	0.290467	0.395583	-0.232787 0.265367 0.520551	0.213523	-0.048085 0.190749 0.418121
50	0.250581	0.391797	-0.146660 0.238206 0.559951	0.172884	-0.064783 0.153613 0.380907

Run 7

Random Seed:	99	Best Fitness:	0.000438	Best Vector:	0.010502 -0.018107 -0.000345
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	21.184438	54.503966	4.897329 4.663464 2.961798	0.080709	0.065873 -0.126931 0.245476
10	0.368169	1.208007	0.562453 -0.917665 0.222586	0.044569	0.065873 0.200453 -0.006943
20	0.066814	0.147196	-0.176591 -0.164888 0.298033	0.015304	0.065873 0.104246 -0.009864
30	0.015669	0.039936	-0.121611 0.157293 0.020140	0.005333	0.065873 0.030744 -0.006943
40	0.003512	0.017762	0.107638 0.021228 -0.075669	0.00045	0.011035 0.016746 -0.006943
50	0.001762	0.005986	0.062368 0.022251 -0.040010	0.000438	0.010502 -0.018107 -0.000345

Run 8

Random Seed:	32	Best Fitness:	0.009504	Best Vector:	-0.007345 -0.096607 0.010838
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	22.781417	54.899979	4.057052 4.019465 4.720615	0.381125	0.210021 -0.510995 -0.275500
10	1.731249	3.530996	1.307737 0.394268 1.290493	0.376288	0.210021 -0.538901 -0.204366
20	0.160641	0.593729	0.292331 -0.432668 -0.566630	0.088014	0.206142 -0.209026 -0.042745
30	0.091531	0.148749	0.199569 -0.329725 0.014250	0.051055	0.167997 -0.150779 -0.009885
40	0.059465	0.128048	0.307005 -0.176345 -0.051949	0.035608	0.084600 -0.161549 -0.048502
50	0.02428	0.049418	0.119923 -0.180788 -0.048502	0.009504	-0.007345 -0.096607 0.010838

Run 9

Random Seed:	43	Best Fitness:	0.121957	Best Vector:	-0.260935 -0.232078 0.003088
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	21.603413	55.528665	3.407544 4.935098 4.422908	3.352016	-0.911577 1.256815 0.970289
10	0.827159	3.586181	1.166787 0.997564 1.108899	0.292714	-0.220339 -0.442282 0.220344
20	0.414651	0.61692	0.624332 -0.442282 0.177529	0.33102	-0.346120 -0.428959 0.164972
30	0.390609	0.488643	-0.476879 -0.468487 0.204326	0.284102	-0.320092 -0.375019 0.202493
40	0.320531	0.395981	-0.423747 -0.405186 0.228570	0.229407	-0.314983 -0.344882 0.106062
50	0.234952	0.332685	-0.361000 -0.344037 0.289832	0.121957	-0.260935 -0.232078 0.003088

Run 10

Random Seed:	95	Best Fitness:	0.314456	Best Vector:	-0.461738 0.297434 -0.113081
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	17.848303	41.845403	3.560969 2.082900 4.982613	0.619332	-0.696419 0.332086 -0.155087
10	1.261857	3.126331	1.179104 0.968895 -0.892909	0.54533	-0.641091 0.332086 -0.155087
20	0.657	0.787026	-0.755298 0.348496 -0.308387	0.500994	-0.629917 0.283101 -0.155087
30	0.532367	0.633765	-0.679299 0.345929 -0.229457	0.428046	-0.564597 0.283101 -0.170675
40	0.474947	0.626239	-0.679299 0.333184 -0.231905	0.374071	-0.512999 0.297169 -0.150310
50	0.389925	0.475281	-0.573513 0.368703 -0.102089	0.314456	-0.461738 0.297434 -0.113081

Run 11

Random Seed:	2	Best Fitness:	0.314959	Best Vector:	-0.041406 0.231121 -0.509733
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	22.484163	51.876565	4.968917 4.696373 2.265062	1.75516	-0.091479 -0.782386 1.065206
10	1.090846	1.819143	0.857379 0.609444 -0.844169	0.680155	-0.057290 0.416740 -0.709366
20	0.746801	0.983396	-0.023481 0.623936 -0.770421	0.5102	-0.064912 0.322425 -0.634057
30	0.648151	0.830274	-0.056406 0.475701 -0.775114	0.472381	-0.000019 0.322425 -0.606979
40	0.511152	0.686702	-0.019009 0.413805 -0.717709	0.423717	0.045184 0.269340 -0.590873
50	0.43282	0.658635	-0.034186 0.416740 -0.695553	0.314959	-0.041406 0.231121 -0.509733

Run 12

Random Seed:	145	Best Fitness:	0.619523	Best Vector:	-0.322252 -0.138817 0.704561
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	23.007498	61.025256	4.365135 4.156151 4.969634	1.225882	-0.975227 -0.337582 0.401065
10	1.823916	2.889887	0.864160 1.460556 0.099448	0.619523	-0.322252 -0.138817 0.704561
20	1.547605	1.738227	0.299335 -0.894615 0.921027	1.400612	-0.308386 -0.857818 0.754758
30	1.5078	1.697324	-0.283575 -0.904247 0.894006	1.322013	-0.283575 -0.738283 0.834588
40	1.543678	1.848125	-0.259657 -0.946669 0.940490	1.290088	-0.265619 -0.793920 0.767611
50	1.410813	1.753464	0.310666 -0.919612 0.900702	1.19303	0.314971 -0.775376 0.701866

Run 13

Random Seed:	245	Best Fitness:	1.649689	Best Vector:	-0.525059 0.773408 0.880819
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	22.361265	55.010277	4.574889 3.757545 4.467832	2.230478	1.304180 -0.677599 -0.265426
10	3.557217	7.008295	1.903014 1.696427 0.713419	1.894127	-0.637122 0.801120 0.920006
20	2.084087	2.848537	-0.719612 1.143286 1.011728	1.753441	-0.609058 0.736302 0.916705
30	1.982635	2.38077	-0.644569 0.934754 1.044766	1.754457	-0.539937 0.789035 0.916705
40	1.979015	2.522121	-0.709841 0.942065 1.063372	1.733926	-0.577279 0.748550 0.916705
50	1.837527	2.415009	-0.629367 0.894176 1.104244	1.649689	-0.525059 0.773408 0.880819

Run 14

Random Seed:	723	Best Fitness:	0.000007	Best Vector:	-0.002474 -0.001046 0.000013
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	23.150546	51.535937	3.979796 4.252715 4.196614	0.380255	-0.080137 0.583589 -0.182364
10	1.191026	3.086813	1.734803 0.226390 0.161308	0.137221	0.179854 0.280809 0.161308
20	0.128199	0.234769	-0.405038 0.249339 0.092426	0.069259	0.137869 0.147743 0.168592
30	0.079686	0.127011	0.236025 0.212799 0.161308	0.046033	0.117260 0.130771 0.123215
40	0.034113	0.088467	0.236700 0.130771 0.123850	0.012638	0.018309 0.089227 0.065886
50	0.00698	0.025255	-0.103014 0.058337 0.106017	0.000007	-0.002474 -0.001046 0.000013

Run 15

Random Seed:	46	Best Fitness:	0.061703	Best Vector:	0.025231 -0.121278 0.215309
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	19.961491	50.448594	3.349347 4.385717 4.471683	0.384993	0.221910 -0.310925 0.488952
10	1.201242	5.033616	1.193506 1.793651 -0.626079	0.199404	-0.036734 -0.207314 0.393797
20	0.277583	0.339408	0.113380 -0.336482 0.461880	0.157572	0.044301 -0.207322 0.335599
30	0.215572	0.288707	0.140550 -0.324542 0.404506	0.146383	-0.029614 -0.181326 0.335599
40	0.160075	0.227653	0.122631 -0.269197 0.374363	0.099621	0.069365 -0.169673 0.256945
50	0.111136	0.158975	0.098399 -0.191484 0.335599	0.061703	0.025231 -0.121278 0.215309

Run 16

Random Seed:	123	Best Fitness:	0.183858	Best Vector:	-0.007897 -0.414524 -0.109385
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	18.392936	55.424848	2.832805 4.950759 4.784355	1.55211	0.159062 0.538174 1.112285
10	0.938839	2.409425	-0.912791 0.582289 1.112285	0.362601	-0.038193 -0.586645 -0.130348
20	0.431659	0.603459	-0.093492 -0.748931 -0.183905	0.30192	-0.022406 -0.531203 -0.138713
30	0.393246	0.502188	0.039036 -0.654882 -0.267943	0.231292	-0.022406 -0.474679 -0.073958
40	0.348593	0.4601	0.092349 -0.661302 -0.119378	0.230486	-0.021112 -0.473888 -0.073958
50	0.263259	0.351846	-0.092700 -0.566215 -0.150510	0.183858	-0.007897 -0.414524 -0.109385

Run 17

Random Seed:	823	Best Fitness:	0.014568	Best Vector:	0.046340 0.015666 -0.110339
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	23.127806	63.378288	4.668000 4.198689 4.894801	0.213141	0.053359 -0.198372 -0.413452
10	0.543845	1.761225	-0.479619 0.253623 1.211142	0.117549	0.003282 0.020654 -0.342216
20	0.124588	0.204106	0.007264 0.063411 -0.447250	0.089553	-0.057390 0.056510 -0.288212
30	0.101499	0.138357	0.033811 -0.011572 -0.370243	0.051398	0.007897 -0.011572 -0.226278
40	0.060736	0.092741	-0.020109 0.018798 -0.303288	0.038791	-0.037949 0.048868 -0.186985
50	0.043033	0.099465	-0.037935 0.110084 -0.293100	0.014568	0.046340 0.015666 -0.110339

Run 18

Random Seed:	711	Best Fitness:	0.000103	Best Vector:	-0.003139 0.003714 0.008933
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	19.762474	41.999359	4.525810 1.869482 4.245166	1.678818	-0.419092 -0.736612 -0.980094
10	1.529421	2.710319	-0.977024 -0.024621 1.324816	0.024482	0.014160 0.142655 0.062695
20	0.034728	0.077445	0.149288 0.159341 0.172535	0.013027	0.055227 0.071619 0.069623
30	0.010604	0.022774	-0.053841 0.071619 0.121432	0.004929	0.000385 -0.009047 0.069623
40	0.004213	0.015098	-0.008660 0.039397 0.116062	0.000291	0.014062 0.003714 0.008933
50	0.001135	0.006348	-0.079083 0.003714 0.008933	0.000103	-0.003139 0.003714 0.008933

Run 19

Random Seed:	911	Best Fitness:	0.129878	Best Vector:	0.145689 -0.244108 0.221503
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	18.681766	44.41229	4.385940 2.308120 4.455155	1.041654	0.786436 -0.320353 0.566167
10	0.61166	1.945137	0.333004 0.286426 1.323709	0.420977	0.237631 -0.311617 0.517111
20	0.444482	0.54363	0.319996 -0.283270 0.600825	0.372879	0.282797 -0.185304 0.508495
30	0.391629	0.556737	0.346890 -0.344869 0.563445	0.25583	0.145689 -0.230614 0.425937
40	0.295525	0.394608	0.227290 -0.288412 0.509672	0.196096	0.145689 -0.244108 0.339532
50	0.220446	0.341295	0.184064 -0.317431 0.454591	0.129878	0.145689 -0.244108 0.221503

Run 20

Random Seed:	194	Best Fitness:	0.292404	Best Vector:	0.483653 -0.201849 0.133195
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	17.428006	46.905771	3.800332 3.964579 4.092109	0.387197	0.324128 -0.422551 0.321852
10	1.019085	4.383463	0.966974 -0.082752 1.855149	0.347492	0.320273 -0.410859 0.275884
20	0.600991	0.723088	-0.629329 -0.493841 0.288365	0.459296	-0.488890 -0.402225 0.241863
30	0.543473	0.644898	-0.665671 -0.351611 0.279554	0.430209	-0.564484 -0.240314 0.231984
40	0.494648	0.58744	-0.602749 -0.373647 0.290726	0.376236	0.539572 -0.259533 0.133195
50	0.417989	0.59472	-0.614899 -0.367867 0.285119	0.292404	0.483653 -0.201849 0.133195

Run 21

Random Seed:	8	Best Fitness:	0.234066	Best Vector:	0.091861 0.023356 -0.474428
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	18.573988	42.668749	4.699998 4.060958 2.021728	0.553141	-0.429330 -0.488465 -0.360860
10	1.50439	4.646436	1.067499 1.138092 1.487154	0.360706	0.148054 -0.102084 -0.573032
20	0.51897	1.225223	0.801964 -0.062174 -0.760402	0.362275	0.129009 -0.147172 -0.569185
30	0.388681	0.472877	0.324588 -0.197876 -0.573032	0.298231	0.091861 -0.132800 -0.521687
40	0.368255	0.514734	0.230411 -0.106531 -0.671041	0.262693	0.091861 -0.157802 -0.478908
50	0.298399	0.394212	0.091861 -0.081353 -0.615756	0.234066	0.091861 0.023356 -0.474428

Run 22

Random Seed:	238	Best Fitness:	0.009427	Best Vector:	0.050971 0.002961 0.082587
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	22.719938	58.15476	3.582482 4.798851 4.721399	0.979429	0.963838 0.217394 -0.056431
10	0.636011	0.993475	-0.817889 0.397510 -0.408066	0.1465	-0.006146 0.020558 0.382151
20	0.174451	0.460113	-0.501582 -0.020404 0.456192	0.100544	0.071461 0.020558 0.308245
30	0.112406	0.170134	-0.079861 0.020558 0.404145	0.07258	0.038968 0.043553 0.262992
40	0.089	0.143345	-0.006146 -0.024995 0.377733	0.044987	0.050971 0.002961 0.205866
50	0.028553	0.071155	0.085642 -0.011177 0.252380	0.009427	0.050971 0.002961 0.082587

Run 23

Random Seed:	234	Best Fitness:	0.104779	Best Vector:	0.067911 0.109449 0.296965
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	21.958436	64.923981	4.431796 4.707467 4.808630	0.366833	-0.099416 0.215183 0.557356
10	1.011571	3.583312	1.066769 -0.977793 1.220343	0.304599	0.112202 0.194872 0.504018
20	0.372024	0.711908	0.166737 0.718698 0.409365	0.208709	0.107212 0.194872 0.399048
30	0.287018	0.405821	0.114088 0.211765 0.589882	0.18562	0.110525 0.194872 0.368006
40	0.196733	0.29447	0.261928 0.200020 0.431110	0.104779	0.067911 0.109449 0.296965
50	0.19379	0.282702	0.261928 0.201625 0.416465	0.131885	0.124403 0.133794 0.313859

Run 24

Random Seed:	995	Best Fitness:	0.365009	Best Vector:	-0.078379 0.596838 -0.051477
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	22.210888	45.067459	1.437108 4.534951 4.736708	2.915827	0.525893 0.301073 1.596439
10	1.450771	4.675518	-0.283726 1.896053 -1.000000	0.593674	-0.146823 0.754052 -0.059352
20	0.633203	0.857482	-0.164085 0.908445 -0.072704	0.537378	-0.115738 0.717751 -0.093895
30	0.622348	0.752778	-0.263313 0.826701 -0.003134	0.478623	-0.181364 0.660995 -0.093895
40	0.597056	0.73466	-0.181364 0.832754 -0.091037	0.442313	-0.032573 0.655854 -0.105394
50	0.516221	0.677406	-0.218273 0.786602 -0.104979	0.365009	-0.078379 0.596838 -0.051477

Run 25

Random Seed:	204	Best Fitness:	0.060768	Best Vector:	-0.059260 -0.172058 -0.166289
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	24.419742	57.006543	4.904156 4.169565 3.945951	0.627979	-0.092240 0.188968 0.764043
10	0.832442	4.362843	2.020489 -0.333167 -0.411664	0.222146	-0.051794 -0.223597 -0.411664
20	0.246675	0.383091	-0.051794 0.163600 -0.594679	0.174542	0.003535 -0.248670 -0.335697
30	0.175884	0.267712	0.063509 -0.194799 -0.475113	0.112638	0.001912 -0.248670 -0.225384
40	0.132343	0.199615	-0.078691 -0.227126 -0.376613	0.080758	0.001912 -0.173683 -0.224919
50	0.097166	0.17636	-0.012153 -0.192800 -0.372882	0.060768	-0.059260 -0.172058 -0.166289

Run 26

Random Seed:	899	Best Fitness:	0.009212	Best Vector:	-0.007441 -0.036307 0.088537
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	18.994845	48.966917	4.575741 4.840277 2.145049	1.515024	0.277286 1.103473 0.469557
10	0.342181	1.261871	0.293025 -0.932849 -0.552993	0.118372	0.045877 -0.188791 0.283945
20	0.136691	0.237958	0.340174 -0.177877 0.300998	0.068991	-0.042449 -0.164614 0.200228
30	0.103679	0.187585	0.006321 -0.208685 0.379467	0.062673	0.007993 -0.188791 0.164217
40	0.067401	0.110885	-0.005020 -0.147265 0.298618	0.025147	-0.040223 -0.100487 0.115892
50	0.026352	0.085284	0.037328 -0.220484 0.187822	0.009212	-0.007441 -0.036307 0.088537

Run 27

Random Seed:	375	Best Fitness:	0.232037	Best Vector:	-0.197457 -0.022834 0.438778
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	19.400609	56.053211	4.489222 3.967072 4.490260	2.849419	0.898147 1.268069 -0.659358
10	1.071156	3.310904	1.560380 0.865332 -0.356816	0.554031	-0.419452 0.059366 0.612019
20	0.697115	0.999032	-0.432605 0.793816 -0.426311	0.529225	-0.326547 0.059366 0.647354
30	0.595086	0.771427	-0.466392 0.140064 0.730950	0.401133	-0.390829 0.017739 0.498068
40	0.517656	0.660027	-0.499762 0.140702 0.624875	0.364473	-0.340715 0.017739 0.498068
50	0.372575	0.525899	-0.392122 -0.001947 0.610029	0.232037	-0.197457 -0.022834 0.438778

Run 28

Random Seed:	112	Best Fitness:	1.156588	Best Vector:	0.744270 0.609631 -0.480625
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	20.497858	46.403524	4.093417 4.527894 3.024176	3.095195	-0.024539 0.325538 1.728762
10	2.201515	5.149698	2.029205 0.881014 -0.505804	1.466382	0.857207 0.608139 -0.601453
20	1.568828	2.158667	0.891589 1.109246 0.365116	1.366939	0.788533 0.666504 -0.548568
30	1.458323	1.783166	0.833002 0.773233 -0.700988	1.214362	0.772597 0.590709 -0.518188
40	1.483966	1.733894	0.962118 0.664932 -0.605053	1.309106	0.801875 0.599039 -0.554304
50	1.395179	1.780272	0.897414 0.791996 -0.589629	1.156588	0.744270 0.609631 -0.480625

Run 29

Random Seed:	276	Best Fitness:	0.809425	Best Vector:	0.082928 0.622059 0.644664
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	22.200516	58.618826	4.855510 3.366874 4.868984	0.889158	-0.237816 0.425729 -0.807067
10	2.765428	4.739702	1.025999 -0.138320 1.915175	1.401964	0.157095 0.916740 0.732716
20	1.502447	3.088618	0.657916 -0.894303 1.362347	1.157388	0.169526 0.802864 0.695744
30	1.266238	1.870957	-0.571805 1.004687 0.731164	0.875659	0.235546 0.661073 0.618999
40	1.039195	1.279001	0.142922 0.916740 0.646654	0.910478	0.156066 0.717202 0.609707
50	0.955858	1.182841	0.272595 0.832096 0.645097	0.809425	0.082928 0.622059 0.644664

Run 30

Random Seed:	419	Best Fitness:	0.011147	Best Vector:	0.033631 0.086619 -0.050135
Generation	Average Fitness	Worst Fitness	Worst Fitness Vector	Best Fitness	Best Fitness Vector
0	21.624658	49.453894	4.498018 3.757054 3.886680	0.641585	0.228824 0.714012 -0.281801
10	0.46326	1.686498	0.228824 -0.670949 1.088102	0.111709	-0.222335 0.156867 -0.194086
20	0.090469	0.143132	0.191777 0.223422 -0.237562	0.058587	0.119751 0.156867 -0.140141
30	0.057032	0.079878	0.097127 0.193489 -0.181677	0.032988	0.128046 0.113926 -0.060111
40	0.045162	0.062978	0.112932 0.155923 -0.160974	0.022968	0.068424 0.095436 -0.095805
50	0.024696	0.053782	0.079412 0.182756 -0.118643	0.011147	0.033631 0.086619 -0.050135

End Tabular Data Format

Begin CSV Printed Format

Averages Across All 30 Runs
Average Avg-of-Generation Fitness
Generation 0, 21.12320002593753
Generation 10, 1.2210781592915534
Generation 20, 0.5526066227472156
Generation 30, 0.4745854224740938
Generation 40, 0.42343158690920296
Generation 50, 0.36314621289362653

Average Best-of-Generation Fitness
Generation 0, 1.1317750938801319
Generation 10, 0.45410821024342957
Generation 20, 0.4209748672207123
Generation 30, 0.3636089839610129
Generation 40, 0.332663582999799
Generation 50, 0.2820540025689745

Best-of-Runs
Average Fitness, 0.26203359103355445
Standard Deviation, 2.049849182629643

,Run 1
Random Seed:,54,Best Fitness:,0.010864,Best Vector:,"0.075131
-0.005082
0.072068"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,20.713668,61.056131,"3.473281
4.993205
4.905135",1.349053,"-0.924184
0.594827
0.375654"
10,0.645426,1.401399,"-0.924184
0.560211
-0.483162",0.191014,"0.327869
-0.232312

0.171891"
20,0.204064,0.278902,"0.456374
-0.199733
0.175305",0.137268,"0.305147
-0.176543
0.113955"
30,0.138819,0.260681,"0.446976
-0.170765
0.178138",0.068357,"0.190585
-0.085946
0.156996"
40,0.080051,0.150913,"0.298847
-0.191764
0.157576",0.035078,"0.126714
-0.101254
0.093645"
50,0.035843,0.063690,"0.225336
-0.110068
0.028259",0.010864,"0.075131
-0.005082
0.072068"

,Run 2
Random Seed:,30,Best Fitness:,0.271307,Best Vector:,"0.292392
0.069535
0.425417"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,20.456802,52.296996,"1.769198
4.960910
4.955432",0.527874,"0.456053
-0.564129
-0.040594"
10,0.907275,1.875992,"0.456053
1.016500
-0.796703",0.457616,"0.372505
0.235697
0.513130"

20,0.529890,0.645072,"0.539189
0.182120
0.566727",0.447498,"0.399489
0.223810
0.487664"
30,0.494308,0.618165,"0.519087
0.223810
0.546464",0.370468,"0.353094
0.204433
0.451664"
40,0.412167,0.552423,"0.399346
0.341512
0.525657",0.286305,"0.325609
0.150528
0.397020"
50,0.342323,0.470124,"0.403522
0.297414
0.467802",0.271307,"0.292392
0.069535
0.425417"

,Run 3
Random Seed:,101,Best Fitness:,0.028915,Best Vector:,"0.103386
0.089716
-0.100881"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,19.467137,44.826213,"3.755361
3.429300
4.354696",0.408297,"0.500109
0.226629
-0.326844"
10,1.333156,2.837684,"0.216040
1.091343
1.264904",0.264011,"0.216040
0.150424
-0.441260"
20,0.272386,0.436684,"0.431138

0.199031
-0.459554",0.180014,"0.188255
0.161592
-0.344184"
30,0.217539,0.337251,"0.266141
0.161592
-0.490212",0.137458,"0.126238
0.195952
-0.288314"
40,0.126526,0.223160,"0.221988
0.228623
-0.348731",0.063600,"0.099722
0.111872
-0.202830"
50,0.061614,0.099412,"0.118675
0.046828
-0.288333",0.028915,"0.103386
0.089716
-0.100881"

,Run 4
Random Seed:,67,Best Fitness:,0.001410,Best Vector:,"-0.009745
-0.014355
0.033302"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,23.061176,58.447901,"4.912639
4.250704
4.030557",0.383880,"0.197192
-0.399847
-0.430252"
10,0.435880,0.912749,"-0.923709
-0.198279
0.142113",0.056449,"-0.102243
-0.197342
0.083972"
20,0.081115,0.162277,"0.325935
-0.143658

```
0.188164",0.037820,"-0.102243
-0.134040
0.096953"
30,0.037256,0.072358,"-0.112124
-0.168237
0.177432",0.021060,"-0.078580
-0.096397
0.074784"
40,0.015682,0.029864,"-0.094835
-0.111177
0.092248",0.005965,"-0.064853
-0.039866
0.013042"
50,0.006485,0.019900,"-0.065641
-0.034966
0.119871",0.001410,"-0.009745
-0.014355
0.033302"

,Run 5
Random Seed:,34,Best Fitness:,0.670121,Best Vector:,"-0.023227
0.639613
-0.510368"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,24.118486,45.257194,"-0.075995
4.595781
4.912251",1.113321,"0.403121
0.777050
-0.589073"
10,1.541554,2.010197,"0.832843
0.732585
0.883113",0.973236,"-0.104300
0.777050
-0.598791"
20,1.110259,1.467932,"0.350608
0.751743
0.883113",0.855057,"-0.104300
```

0.696870
-0.598791"
30,1.043725,1.303971,"0.435942
0.841001
-0.637686",0.821524,"-0.132630
0.718076
-0.536935"
40,0.951085,1.284582,"0.554112
0.803801
-0.575714",0.766308,"-0.147419
0.669255
-0.544678"
50,0.892695,1.196633,"0.603723
0.730609
-0.546226",0.670121,"-0.023227
0.639613
-0.510368"

,Run 6
Random Seed:,22,Best Fitness:,0.172884,Best Vector:,"-0.064783
0.153613
0.380907"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,21.501731,57.420152,"4.627933
3.643713
4.767152",0.454368,"-0.291452
0.298082
0.529689"
10,1.794798,5.476887,"2.169444
0.644037
-0.596335",0.338404,"-0.171693
0.242225
0.500253"
20,0.719786,0.857188,"-0.197352
0.605572
-0.671955",0.343920,"-0.146660
0.242225

0.513555"
30,0.334748,0.426146,"-0.266287
0.276352
0.528078",0.271432,"-0.096417
0.235488
0.454622"
40,0.290467,0.395583,"-0.232787
0.265367
0.520551",0.213523,"-0.048085
0.190749
0.418121"
50,0.250581,0.391797,"-0.146660
0.238206
0.559951",0.172884,"-0.064783
0.153613
0.380907"

,Run 7
Random Seed:,99,Best Fitness:,0.000438,Best Vector:,"0.010502
-0.018107
-0.000345"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,21.184438,54.503966,"4.897329
4.663464
2.961798",0.080709,"0.065873
-0.126931
0.245476"
10,0.368169,1.208007,"0.562453
-0.917665
0.222586",0.044569,"0.065873
0.200453
-0.006943"
20,0.066814,0.147196,"-0.176591
-0.164888
0.298033",0.015304,"0.065873
0.104246
-0.009864"

30,0.015669,0.039936,"-0.121611
0.157293
0.020140",0.005333,"0.065873
0.030744
-0.006943"
40,0.003512,0.017762,"0.107638
0.021228
-0.075669",0.000450,"0.011035
0.016746
-0.006943"
50,0.001762,0.005986,"0.062368
0.022251
-0.040010",0.000438,"0.010502
-0.018107
-0.000345"

,Run 8
Random Seed:,32,Best Fitness:,0.009504,Best Vector:,"-0.007345
-0.096607
0.010838"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,22.781417,54.899979,"4.057052
4.019465
4.720615",0.381125,"0.210021
-0.510995
-0.275500"
10,1.731249,3.530996,"1.307737
0.394268
1.290493",0.376288,"0.210021
-0.538901
-0.204366"
20,0.160641,0.593729,"0.292331
-0.432668
-0.566630",0.088014,"0.206142
-0.209026
-0.042745"
30,0.091531,0.148749,"0.199569

-0.329725
0.014250",0.051055,"0.167997
-0.150779
-0.009885"
40,0.059465,0.128048,"0.307005
-0.176345
-0.051949",0.035608,"0.084600
-0.161549
-0.048502"
50,0.024280,0.049418,"0.119923
-0.180788
-0.048502",0.009504,"-0.007345
-0.096607
0.010838"

,Run 9
Random Seed:,43,Best Fitness:,0.121957,Best Vector:,"-0.260935
-0.232078
0.003088"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,21.603413,55.528665,"3.407544
4.935098
4.422908",3.352016,"-0.911577
1.256815
0.970289"
10,0.827159,3.586181,"1.166787
0.997564
1.108899",0.292714,"-0.220339
-0.442282
0.220344"
20,0.414651,0.616920,"0.624332
-0.442282
0.177529",0.331020,"-0.346120
-0.428959
0.164972"
30,0.390609,0.488643,"-0.476879
-0.468487

0.204326",0.284102,"-0.320092
-0.375019
0.202493"
40,0.320531,0.395981,"-0.423747
-0.405186
0.228570",0.229407,"-0.314983
-0.344882
0.106062"
50,0.234952,0.332685,"-0.361000
-0.344037
0.289832",0.121957,"-0.260935
-0.232078
0.003088"

,Run 10
Random Seed:,95,Best Fitness:,0.314456,Best Vector:,"-0.461738
0.297434
-0.113081"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,17.848303,41.845403,"3.560969
2.082900
4.982613",0.619332,"-0.696419
0.332086
-0.155087"
10,1.261857,3.126331,"1.179104
0.968895
-0.892909",0.545330,"-0.641091
0.332086
-0.155087"
20,0.657000,0.787026,"-0.755298
0.348496
-0.308387",0.500994,"-0.629917
0.283101
-0.155087"
30,0.532367,0.633765,"-0.679299
0.345929
-0.229457",0.428046,"-0.564597

0.283101
-0.170675"
40,0.474947,0.626239,"-0.679299
0.333184
-0.231905",0.374071,"-0.512999
0.297169
-0.150310"
50,0.389925,0.475281,"-0.573513
0.368703
-0.102089",0.314456,"-0.461738
0.297434
-0.113081"

,Run 11
Random Seed:,2,Best Fitness:,0.314959,Best Vector:,"-0.041406
0.231121
-0.509733"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,22.484163,51.876565,"4.968917
4.696373
2.265062",1.755160,"-0.091479
-0.782386
1.065206"
10,1.090846,1.819143,"0.857379
0.609444
-0.844169",0.680155,"-0.057290
0.416740
-0.709366"
20,0.746801,0.983396,"-0.023481
0.623936
-0.770421",0.510200,"-0.064912
0.322425
-0.634057"
30,0.648151,0.830274,"-0.056406
0.475701
-0.775114",0.472381,"-0.000019
0.322425

-0.606979"
40,0.511152,0.686702,"-0.019009
0.413805
-0.717709",0.423717,"0.045184
0.269340
-0.590873"
50,0.432820,0.658635,"-0.034186
0.416740
-0.695553",0.314959,"-0.041406
0.231121
-0.509733"

,Run 12
Random Seed:,145,Best Fitness:,0.619523,Best Vector:,"-0.322252
-0.138817
0.704561"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,23.007498,61.025256,"4.365135
4.156151
4.969634",1.225882,"-0.975227
-0.337582
0.401065"
10,1.823916,2.889887,"0.864160
1.460556
0.099448",0.619523,"-0.322252
-0.138817
0.704561"
20,1.547605,1.738227,"0.299335
-0.894615
0.921027",1.400612,"-0.308386
-0.857818
0.754758"
30,1.507800,1.697324,"-0.283575
-0.904247
0.894006",1.322013,"-0.283575
-0.738283
0.834588"

40,1.543678,1.848125,"-0.259657
-0.946669
0.940490",1.290088,"-0.265619
-0.793920
0.767611"
50,1.410813,1.753464,"0.310666
-0.919612
0.900702",1.193030,"0.314971
-0.775376
0.701866"

,Run 13
Random Seed:,245,Best Fitness:,1.649689,Best Vector:,"-0.525059
0.773408
0.880819"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,22.361265,55.010277,"4.574889
3.757545
4.467832",2.230478,"1.304180
-0.677599
-0.265426"
10,3.557217,7.008295,"1.903014
1.696427
0.713419",1.894127,"-0.637122
0.801120
0.920006"
20,2.084087,2.848537,"-0.719612
1.143286
1.011728",1.753441,"-0.609058
0.736302
0.916705"
30,1.982635,2.380770,"-0.644569
0.934754
1.044766",1.754457,"-0.539937
0.789035
0.916705"
40,1.979015,2.522121,"-0.709841

0.942065
1.063372",1.733926,"-0.577279
0.748550
0.916705"
50,1.837527,2.415009,"-0.629367
0.894176
1.104244",1.649689,"-0.525059
0.773408
0.880819"

,Run 14
Random Seed:,723,Best Fitness:,0.000007,Best Vector:,"-0.002474
-0.001046
0.000013"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,23.150546,51.535937,"3.979796
4.252715
4.196614",0.380255,"-0.080137
0.583589
-0.182364"
10,1.191026,3.086813,"1.734803
0.226390
0.161308",0.137221,"0.179854
0.280809
0.161308"
20,0.128199,0.234769,"-0.405038
0.249339
0.092426",0.069259,"0.137869
0.147743
0.168592"
30,0.079686,0.127011,"0.236025
0.212799
0.161308",0.046033,"0.117260
0.130771
0.123215"
40,0.034113,0.088467,"0.236700
0.130771

0.123850",0.012638,"0.018309
0.089227
0.065886"
50,0.006980,0.025255,"-0.103014
0.058337
0.106017",0.000007,"-0.002474
-0.001046
0.000013"

,Run 15
Random Seed:,46,Best Fitness:,0.061703,Best Vector:,"0.025231
-0.121278
0.215309"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,19.961491,50.448594,"3.349347
4.385717
4.471683",0.384993,"0.221910
-0.310925
0.488952"
10,1.201242,5.033616,"1.193506
1.793651
-0.626079",0.199404,"-0.036734
-0.207314
0.393797"
20,0.277583,0.339408,"0.113380
-0.336482
0.461880",0.157572,"0.044301
-0.207322
0.335599"
30,0.215572,0.288707,"0.140550
-0.324542
0.404506",0.146383,"-0.029614
-0.181326
0.335599"
40,0.160075,0.227653,"0.122631
-0.269197
0.374363",0.099621,"0.069365

88

-0.169673
0.256945"
50,0.111136,0.158975,"0.098399
-0.191484
0.335599",0.061703,"0.025231
-0.121278
0.215309"

,Run 16
Random Seed:,123,Best Fitness:,0.183858,Best Vector:,"-0.007897
-0.414524
-0.109385"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,18.392936,55.424848,"2.832805
4.950759
4.784355",1.552110,"0.159062
0.538174
1.112285"
10,0.938839,2.409425,"-0.912791
0.582289
1.112285",0.362601,"-0.038193
-0.586645
-0.130348"
20,0.431659,0.603459,"-0.093492
-0.748931
-0.183905",0.301920,"-0.022406
-0.531203
-0.138713"
30,0.393246,0.502188,"0.039036
-0.654882
-0.267943",0.231292,"-0.022406
-0.474679
-0.073958"
40,0.348593,0.460100,"0.092349
-0.661302
-0.119378",0.230486,"-0.021112
-0.473888

-0.073958"
50,0.263259,0.351846,"-0.092700
-0.566215
-0.150510",0.183858,"-0.007897
-0.414524
-0.109385"

,Run 17
Random Seed:,823,Best Fitness:,0.014568,Best Vector:,"0.046340
0.015666
-0.110339"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,23.127806,63.378288,"4.668000
4.198689
4.894801",0.213141,"0.053359
-0.198372
-0.413452"
10,0.543845,1.761225,"-0.479619
0.253623
1.211142",0.117549,"0.003282
0.020654
-0.342216"
20,0.124588,0.204106,"0.007264
0.063411
-0.447250",0.089553,"-0.057390
0.056510
-0.288212"
30,0.101499,0.138357,"0.033811
-0.011572
-0.370243",0.051398,"0.007897
-0.011572
-0.226278"
40,0.060736,0.092741,"-0.020109
0.018798
-0.303288",0.038791,"-0.037949
0.048868
-0.186985"

90

50,0.043033,0.099465,"-0.037935
0.110084
-0.293100",0.014568,"0.046340
0.015666
-0.110339"

,Run 18
Random Seed:,711,Best Fitness:,0.000103,Best Vector:,"-0.003139
0.003714
0.008933"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,19.762474,41.999359,"4.525810
1.869482
4.245166",1.678818,"-0.419092
-0.736612
-0.980094"
10,1.529421,2.710319,"-0.977024
-0.024621
1.324816",0.024482,"0.014160
0.142655
0.062695"
20,0.034728,0.077445,"0.149288
0.159341
0.172535",0.013027,"0.055227
0.071619
0.069623"
30,0.010604,0.022774,"-0.053841
0.071619
0.121432",0.004929,"0.000385
-0.009047
0.069623"
40,0.004213,0.015098,"-0.008660
0.039397
0.116062",0.000291,"0.014062
0.003714
0.008933"
50,0.001135,0.006348,"-0.079083

0.003714
0.008933",0.000103,"-0.003139
0.003714
0.008933"

,Run 19
Random Seed:,911,Best Fitness:,0.129878,Best Vector:,"0.145689
-0.244108
0.221503"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,18.681766,44.412290,"4.385940
2.308120
4.455155",1.041654,"0.786436
-0.320353
0.566167"
10,0.611660,1.945137,"0.333004
0.286426
1.323709",0.420977,"0.237631
-0.311617
0.517111"
20,0.444482,0.543630,"0.319996
-0.283270
0.600825",0.372879,"0.282797
-0.185304
0.508495"
30,0.391629,0.556737,"0.346890
-0.344869
0.563445",0.255830,"0.145689
-0.230614
0.425937"
40,0.295525,0.394608,"0.227290
-0.288412
0.509672",0.196096,"0.145689
-0.244108
0.339532"
50,0.220446,0.341295,"0.184064
-0.317431

0.454591",0.129878,"0.145689
-0.244108
0.221503"

,Run 20
Random Seed:,194,Best Fitness:,0.292404,Best Vector:,"0.483653
-0.201849
0.133195"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,17.428006,46.905771,"3.800332
3.964579
4.092109",0.387197,"0.324128
-0.422551
0.321852"
10,1.019085,4.383463,"0.966974
-0.082752
1.855149",0.347492,"0.320273
-0.410859
0.275884"
20,0.600991,0.723088,"-0.629329
-0.493841
0.288365",0.459296,"-0.488890
-0.402225
0.241863"
30,0.543473,0.644898,"-0.665671
-0.351611
0.279554",0.430209,"-0.564484
-0.240314
0.231984"
40,0.494648,0.587440,"-0.602749
-0.373647
0.290726",0.376236,"0.539572
-0.259533
0.133195"
50,0.417989,0.594720,"-0.614899
-0.367867
0.285119",0.292404,"0.483653

93

-0.201849
0.133195"

,Run 21
Random Seed:,8,Best Fitness:,0.234066,Best Vector:,"0.091861
0.023356
-0.474428"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,18.573988,42.668749,"4.699998
4.060958
2.021728",0.553141,"-0.429330
-0.488465
-0.360860"
10,1.504390,4.646436,"1.067499
1.138092
1.487154",0.360706,"0.148054
-0.102084
-0.573032"
20,0.518970,1.225223,"0.801964
-0.062174
-0.760402",0.362275,"0.129009
-0.147172
-0.569185"
30,0.388681,0.472877,"0.324588
-0.197876
-0.573032",0.298231,"0.091861
-0.132800
-0.521687"
40,0.368255,0.514734,"0.230411
-0.106531
-0.671041",0.262693,"0.091861
-0.157802
-0.478908"
50,0.298399,0.394212,"0.091861
-0.081353
-0.615756",0.234066,"0.091861
0.023356

94

-0.474428"

,Run 22
Random Seed:,238,Best Fitness:,0.009427,Best Vector:,"0.050971
0.002961
0.082587"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,22.719938,58.154760,"3.582482
4.798851
4.721399",0.979429,"0.963838
0.217394
-0.056431"
10,0.636011,0.993475,"-0.817889
0.397510
-0.408066",0.146500,"-0.006146
0.020558
0.382151"
20,0.174451,0.460113,"-0.501582
-0.020404
0.456192",0.100544,"0.071461
0.020558
0.308245"
30,0.112406,0.170134,"-0.079861
0.020558
0.404145",0.072580,"0.038968
0.043553
0.262992"
40,0.089000,0.143345,"-0.006146
-0.024995
0.377733",0.044987,"0.050971
0.002961
0.205866"
50,0.028553,0.071155,"0.085642
-0.011177
0.252380",0.009427,"0.050971
0.002961
0.082587"

,Run 23
Random Seed:,234,Best Fitness:,0.104779,Best Vector:,"0.067911
0.109449
0.296965"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,21.958436,64.923981,"4.431796
4.707467
4.808630",0.366833,"-0.099416
0.215183
0.557356"
10,1.011571,3.583312,"1.066769
-0.977793
1.220343",0.304599,"0.112202
0.194872
0.504018"
20,0.372024,0.711908,"0.166737
0.718698
0.409365",0.208709,"0.107212
0.194872
0.399048"
30,0.287018,0.405821,"0.114088
0.211765
0.589882",0.185620,"0.110525
0.194872
0.368006"
40,0.196733,0.294470,"0.261928
0.200020
0.431110",0.104779,"0.067911
0.109449
0.296965"
50,0.193790,0.282702,"0.261928
0.201625
0.416465",0.131885,"0.124403
0.133794
0.313859"

,Run 24
Random Seed:,995,Best Fitness:,0.365009,Best Vector:,"-0.078379
0.596838
-0.051477"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,22.210888,45.067459,"1.437108
4.534951
4.736708",2.915827,"0.525893
0.301073
1.596439"
10,1.450771,4.675518,"-0.283726
1.896053
-1.000000",0.593674,"-0.146823
0.754052
-0.059352"
20,0.633203,0.857482,"-0.164085
0.908445
-0.072704",0.537378,"-0.115738
0.717751
-0.093895"
30,0.622348,0.752778,"-0.263313
0.826701
-0.003134",0.478623,"-0.181364
0.660995
-0.093895"
40,0.597056,0.734660,"-0.181364
0.832754
-0.091037",0.442313,"-0.032573
0.655854
-0.105394"
50,0.516221,0.677406,"-0.218273
0.786602
-0.104979",0.365009,"-0.078379
0.596838
-0.051477"

,Run 25

Random Seed:,204,Best Fitness:,0.060768,Best Vector:,"-0.059260
-0.172058
-0.166289"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,24.419742,57.006543,"4.904156
4.169565
3.945951",0.627979,"-0.092240
0.188968
0.764043"
10,0.832442,4.362843,"2.020489
-0.333167
-0.411664",0.222146,"-0.051794
-0.223597
-0.411664"
20,0.246675,0.383091,"-0.051794
0.163600
-0.594679",0.174542,"0.003535
-0.248670
-0.335697"
30,0.175884,0.267712,"0.063509
-0.194799
-0.475113",0.112638,"0.001912
-0.248670
-0.225384"
40,0.132343,0.199615,"-0.078691
-0.227126
-0.376613",0.080758,"0.001912
-0.173683
-0.224919"
50,0.097166,0.176360,"-0.012153
-0.192800
-0.372882",0.060768,"-0.059260
-0.172058
-0.166289"

,Run 26
Random Seed:,899,Best Fitness:,0.009212,Best Vector:,"-0.007441

98

-0.036307
0.088537"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,18.994845,48.966917,"4.575741
4.840277
2.145049",1.515024,"0.277286
1.103473
0.469557"
10,0.342181,1.261871,"0.293025
-0.932849
-0.552993",0.118372,"0.045877
-0.188791
0.283945"
20,0.136691,0.237958,"0.340174
-0.177877
0.300998",0.068991,"-0.042449
-0.164614
0.200228"
30,0.103679,0.187585,"0.006321
-0.208685
0.379467",0.062673,"0.007993
-0.188791
0.164217"
40,0.067401,0.110885,"-0.005020
-0.147265
0.298618",0.025147,"-0.040223
-0.100487
0.115892"
50,0.026352,0.085284,"0.037328
-0.220484
0.187822",0.009212,"-0.007441
-0.036307
0.088537"

,Run 27
Random Seed:,375,Best Fitness:,0.232037,Best Vector:,"-0.197457
-0.022834

0.438778"
Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,19.400609,56.053211,"4.489222
3.967072
4.490260",2.849419,"0.898147
1.268069
-0.659358"
10,1.071156,3.310904,"1.560380
0.865332
-0.356816",0.554031,"-0.419452
0.059366
0.612019"
20,0.697115,0.999032,"-0.432605
0.793816
-0.426311",0.529225,"-0.326547
0.059366
0.647354"
30,0.595086,0.771427,"-0.466392
0.140064
0.730950",0.401133,"-0.390829
0.017739
0.498068"
40,0.517656,0.660027,"-0.499762
0.140702
0.624875",0.364473,"-0.340715
0.017739
0.498068"
50,0.372575,0.525899,"-0.392122
-0.001947
0.610029",0.232037,"-0.197457
-0.022834
0.438778"

,Run 28
Random Seed:,112,Best Fitness:,1.156588,Best Vector:,"0.744270
0.609631
-0.480625"

100

Generation,"Average
Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,20.497858,46.403524,"4.093417
4.527894
3.024176",3.095195,"-0.024539
0.325538
1.728762"
10,2.201515,5.149698,"2.029205
0.881014
-0.505804",1.466382,"0.857207
0.608139
-0.601453"
20,1.568828,2.158667,"0.891589
1.109246
0.365116",1.366939,"0.788533
0.666504
-0.548568"
30,1.458323,1.783166,"0.833002
0.773233
-0.700988",1.214362,"0.772597
0.590709
-0.518188"
40,1.483966,1.733894,"0.962118
0.664932
-0.605053",1.309106,"0.801875
0.599039
-0.554304"
50,1.395179,1.780272,"0.897414
0.791996
-0.589629",1.156588,"0.744270
0.609631
-0.480625"

,Run 29
Random Seed:,276,Best Fitness:,0.809425,Best Vector:,"0.082928
0.622059
0.644664"
Generation,"Average

Fitness","Worst
Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,22.200516,58.618826,"4.855510
3.366874
4.868984",0.889158,"-0.237816
0.425729
-0.807067"
10,2.765428,4.739702,"1.025999
-0.138320
1.915175",1.401964,"0.157095
0.916740
0.732716"
20,1.502447,3.088618,"0.657916
-0.894303
1.362347",1.157388,"0.169526
0.802864
0.695744"
30,1.266238,1.870957,"-0.571805
1.004687
0.731164",0.875659,"0.235546
0.661073
0.618999"
40,1.039195,1.279001,"0.142922
0.916740
0.646654",0.910478,"0.156066
0.717202
0.609707"
50,0.955858,1.182841,"0.272595
0.832096
0.645097",0.809425,"0.082928
0.622059
0.644664"

,Run 30
Random Seed:,419,Best Fitness:,0.011147,Best Vector:,"0.033631
0.086619
-0.050135"
Generation,"Average
Fitness","Worst

Fitness","Worst
Fitness
Vector","Best
Fitness","Best
Fitness
Vector"
0,21.624658,49.453894,"4.498018
3.757054
3.886680",0.641585,"0.228824
0.714012
-0.281801"
10,0.463260,1.686498,"0.228824
-0.670949
1.088102",0.111709,"-0.222335
0.156867
-0.194086"
20,0.090469,0.143132,"0.191777
0.223422
-0.237562",0.058587,"0.119751
0.156867
-0.140141"
30,0.057032,0.079878,"0.097127
0.193489
-0.181677",0.032988,"0.128046
0.113926
-0.060111"
40,0.045162,0.062978,"0.112932
0.155923
-0.160974",0.022968,"0.068424
0.095436
-0.095805"
50,0.024696,0.053782,"0.079412
0.182756
-0.118643",0.011147,"0.033631
0.086619
-0.050135"

End CSV Printed Format