ML Algorithm Implementation

Nahian Siddique

Doga Ozgulbas

Shahrukh Alam Khan

Motivation

The current challenge in big data is the efficient analysis of large scale data. In recent years, there has been an overwhelming increase in the size of data collected in all industries. While the size of this data has grown exponentially, the processing power of computers has not. We must therefore rely on optimization algorithms to improve our data processing speeds. One such novel class of optimization algorithms is swarm intelligence which rely on the collective behavior of many decentralized systems to process data. In this project we explore two of these swarm intelligence algorithms and find their utility in big data.

Background

Optimization is a field of computer science and mathematics which deals with finding the best solution from a set of possible solutions.

Optimization problems can be divided into two categories:

1. Classical or Deterministic solutions
2. Nature Inspired or Stochastic solutions

Deterministic solutions involve the use of mathematical models to find the exact solution. However, a certain category of problems, called NP-Hard problems, cannot be solved in a reasonable time using deterministic solutions. This led to the exploration of nature inspired solutions.

Background

As the name implies, nature inspired solutions make use of models inspired from the behavior of natural phenomena. These algorithms use stochastic search to find near optimal solutions and by being heuristic, perform significantly faster than deterministic solutions.

Nature inspired algorithms can be further divided into two categories:

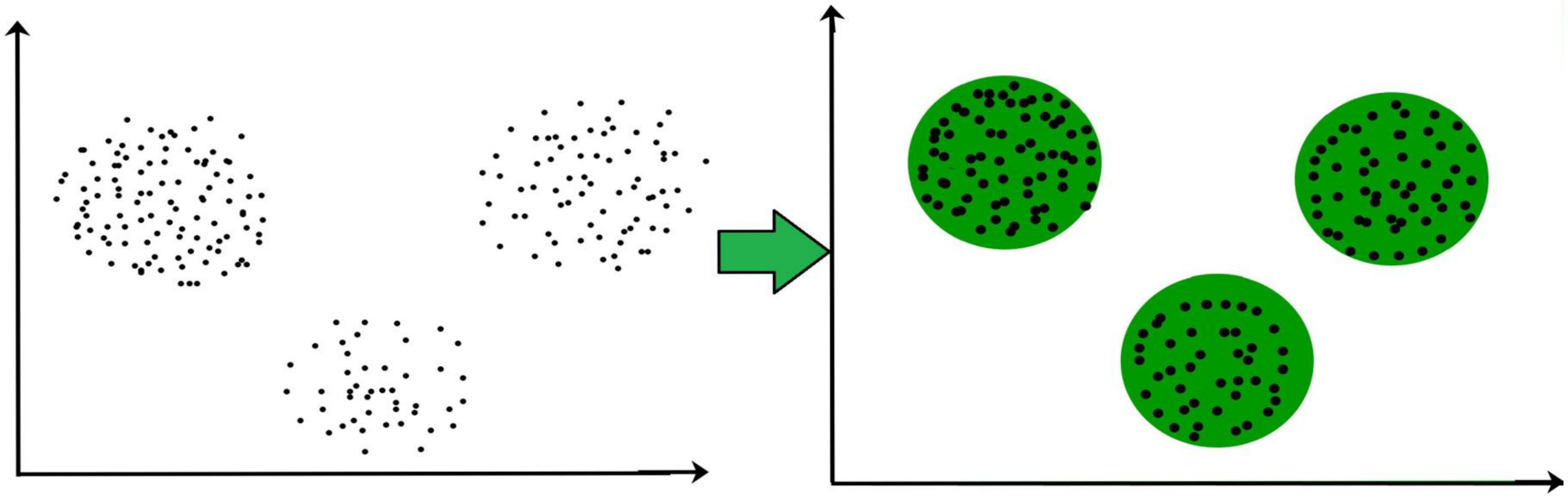
1. Evolutionary algorithms
   - Genetic algorithms
   - Differential evolution
2. Swarm intelligence
   - Particle swarm optimization
   - Chemical reaction optimization
   - Ant colony optimization
   - Gravitational search optimization
   - Binary bat optimization

Background

- Swarm intelligence is a class of population based metaheuristics.
- The collective behavior of many disconnected systems with defined rules can result in predictable and controlled outcomes.
- Three stages:
    - Population initialization
    - Iteration
    - Termination
- Two modes:
    - Intensification or Exploitation – local search
    - Diversification or Exploration – global search
- The rules to define the behavior of the population has been modeled after various natural systems.

Clustering

To extract useful information, data mining techniques can be used. Among many techniques of data mining, clustering is most popular technique. Clustering bind together the similar data in same group, whereas, dissimilar data is scattered in different groups.



Algorithms

- K-Means
- Particle Swarm Optimization
- KMPSO
- Chemical Reaction Optimization

K-Means Clustering

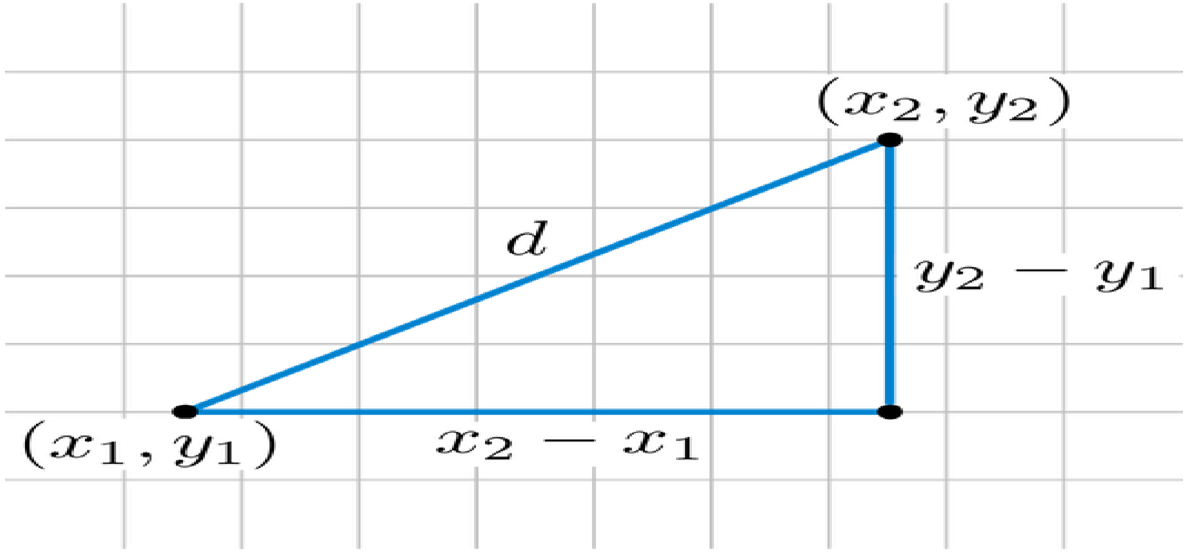
- K-means clustering is one of the simplest and popular unsupervised machine learning algorithms.
- Typically, unsupervised algorithms make inferences from datasets using only input vectors without referring to known, or labelled, outcomes.
- The objective of K-means is simple: group similar data points together and discover underlying patterns. To achieve this objective, K-means looks for a fixed number (*k*) of clusters in a dataset.

How K-Means algorithm works

To process the learning data, the K-means algorithm in data mining starts with a first group of randomly selected centroids, which are used as the beginning points for every cluster, and then performs iterative (repetitive) calculations to optimize the positions of the centroids

It halts creating and optimizing clusters when either:

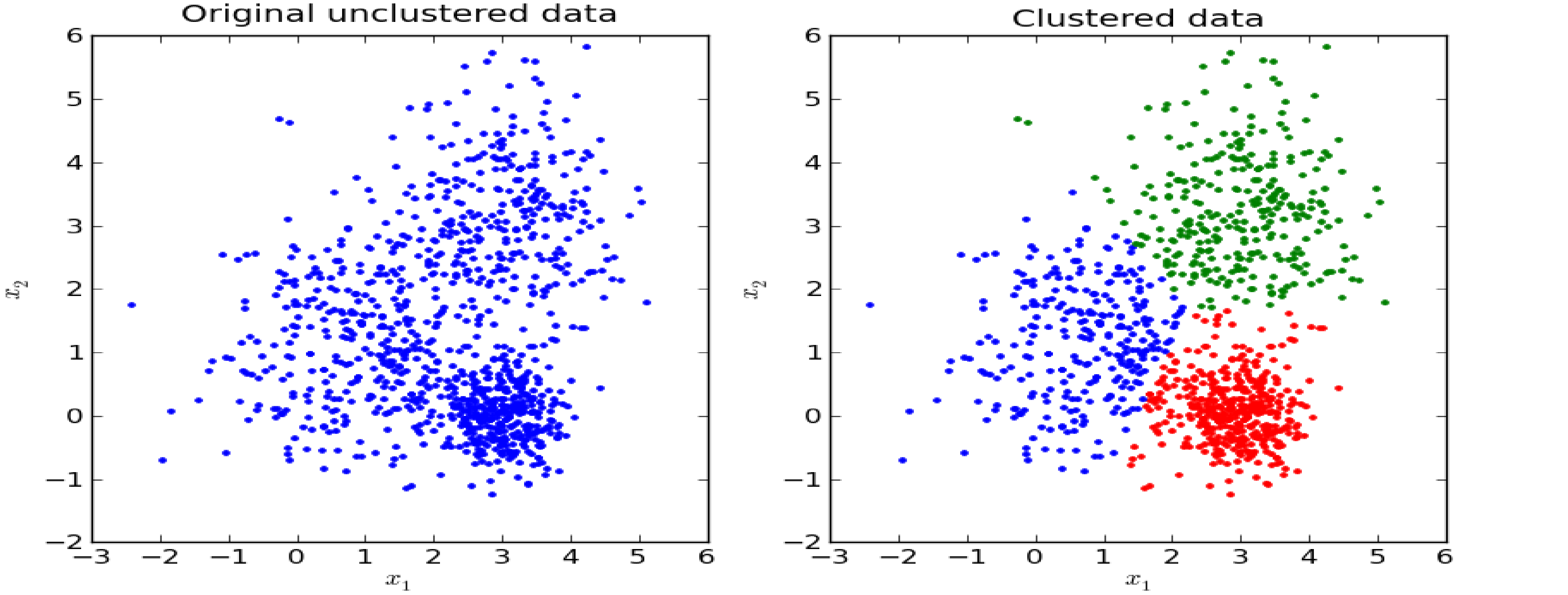
- The centroids have stabilized — there is no change in their values because the clustering has been successful.
- The defined number of iterations has been achieved.



$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Clustering example

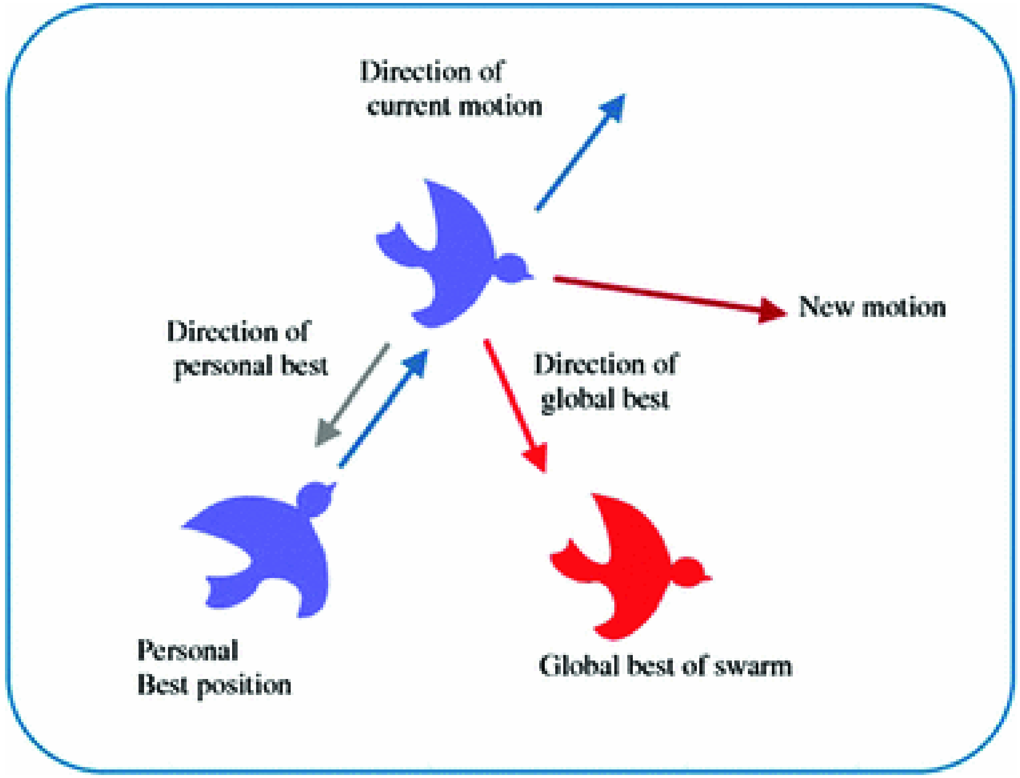
A cluster refers to a collection of data points aggregated together because of certain similarities.



Particle Swarm Optimization

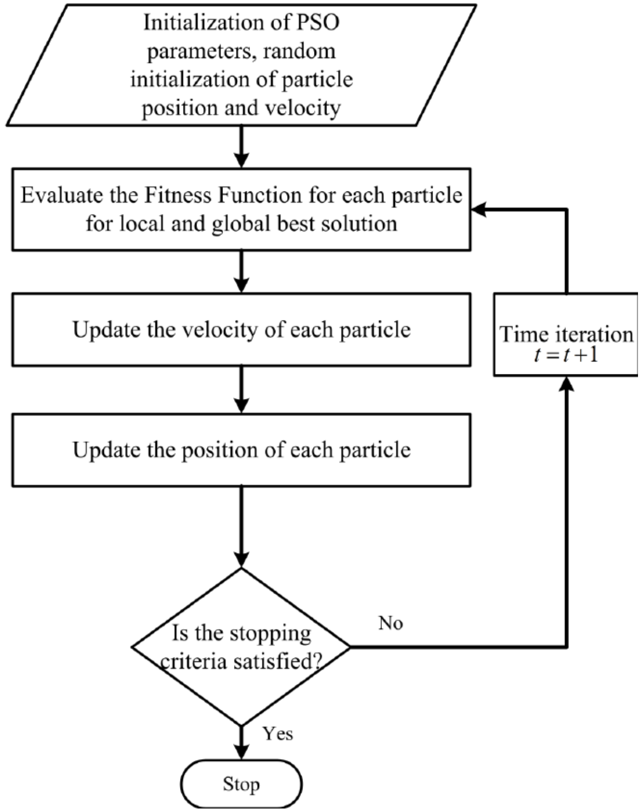
PSO simulates the behaviors of bird flocking.

Suppose the following scenario: a group of birds are randomly searching for food in an area. There is only one piece of food in the area being searched. All the birds do not know where the food is. But they know how far the food is in each iteration. So what's the best strategy to find the food? The effective one is to follow the bird which is nearest to the food.



Particle Swarm Optimization

- Each particle has a Position and Velocity (Changes of the Position)
- Each particle's movement is influenced by its local best known position, also guided toward the best known positions in the search-space
- The best position is the optimum result


Initialization of PSO parameters, random initialization of particle position and velocity
Evaluate the Fitness Function for each particle for local and global best solution
Update the velocity of each particle
Update the position of each particle
Is the stopping criteria satisfied?
No
Yes
Stop
Time iteration $t = t + 1$

KMPSO

- In the KMPSO algorithm, a swarm of particles are initialized, each with a set of cluster centroids which are randomly sampled points from the dataset.
- Then K-Means is run for a fixed number of iterations, and the centroids of the K-Means clusters are set to the global best particle.
- The PSO clustering algorithm is then executed. The diversity of the swarm ensures that a global search is conducted.
- The resulting performance is significantly better than just simple PSO.

Chemical Reaction Optimization

- Chemical Reaction Optimization (CRO) is a recently established metaheuristics for optimization, inspired by the nature of chemical reactions.
- A chemical reaction is a natural process of transforming unstable molecules to more stable ones.
- In CRO, different solutions are modelled as different molecular structures, and interaction between different molecule, i.e. a chemical reaction, creates new molecules with different structures.
- By finding the molecule with the highest stability, we in turn find the optimal solution.

Chemical Reaction Optimization

- A molecule possesses two kinds of energies: potential energy (PE) and kinetic energy (KE).
- Potential energy is determined by its structure, i.e., stability. Meaning, a molecule with less PE has a more stable structure and is therefor a better solution.
- Kinetic energy is the energy described by the motion of the molecule.
- Every reaction will cause some change in both PE and KE, and eventually lead us to the most stable configuration, where we will find the optimal solution.

Chemical Reaction Optimization

Two types of chemical reactions, each with two possible outcomes:

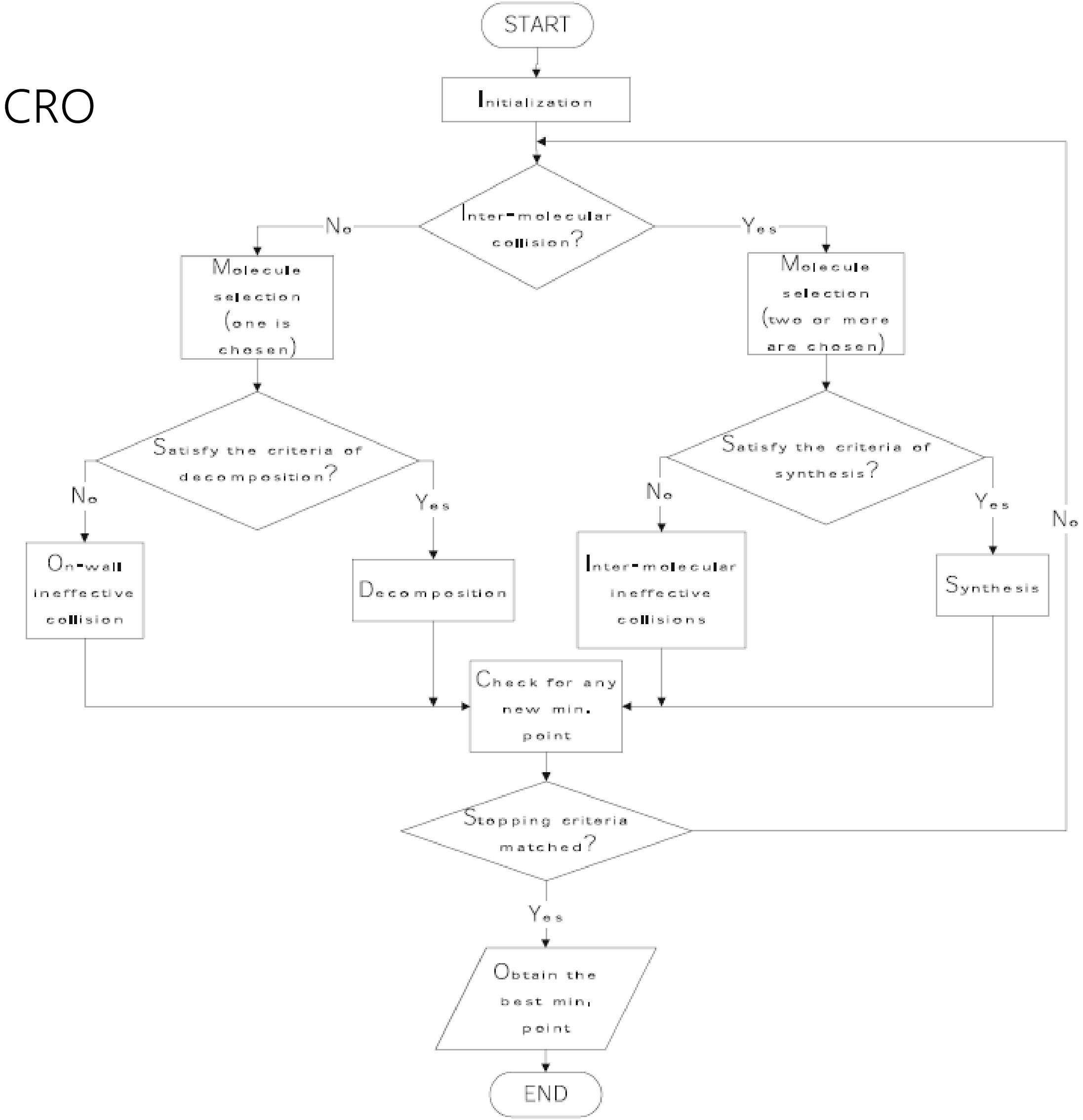
1. Unimolecular reaction
   - On-wall ineffective collision
   - Decomposition
2. Intermolecular reaction
   - Intermolecular ineffective collision
   - Synthesis

Chemical Reaction Optimization

On-wall ineffective collision and intermolecular ineffective collision implement intensification. These two reactions change the structure of the molecules slightly based on the KE and PE. This allows the system to perform local search.

Decomposition and synthesis implement diversification. Either one molecule splits into two or two molecules combine into one. In both cases there is a drastic change in the molecular structure and the system the system can perform global search.

Schematic diagram of CRO

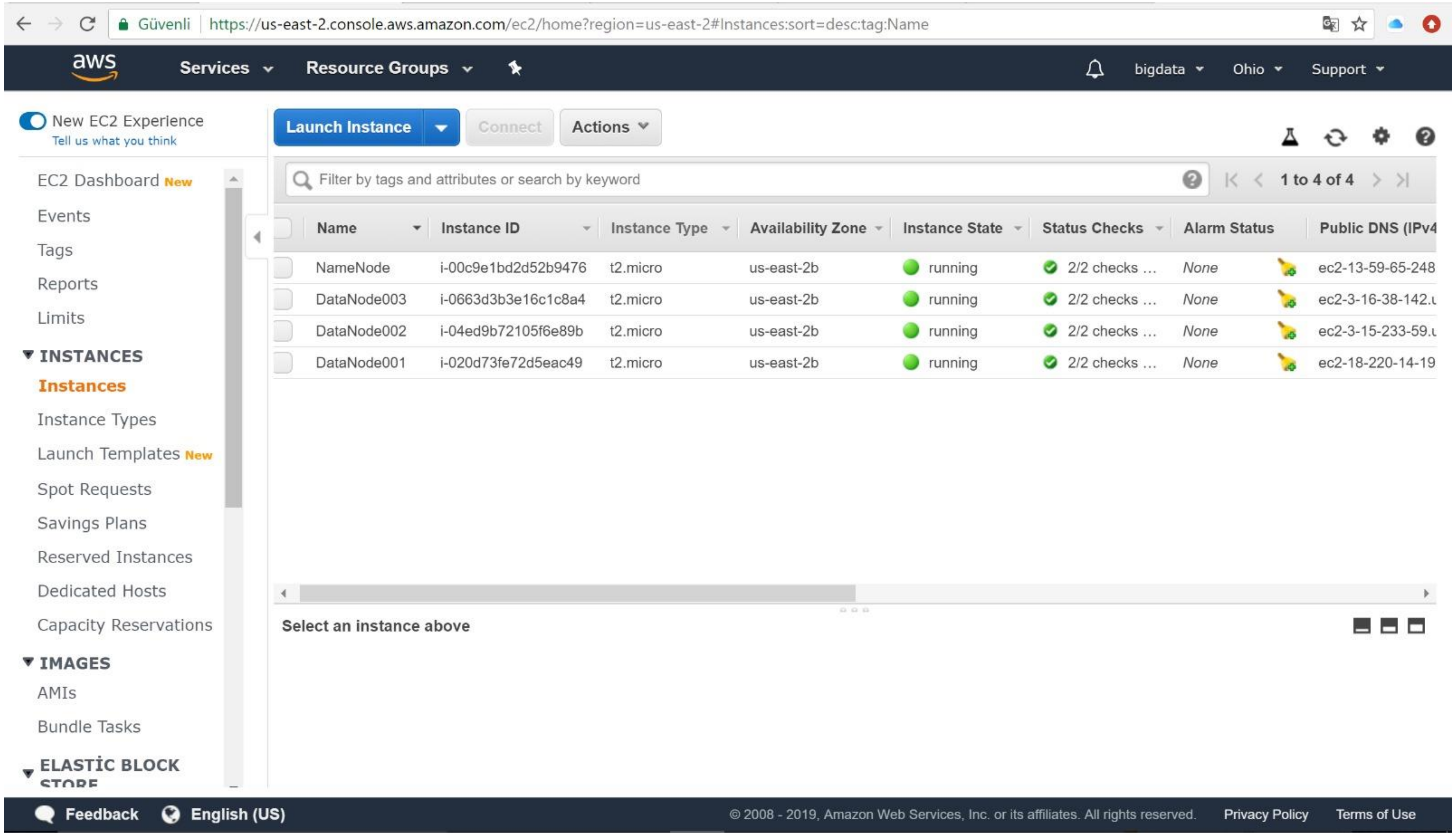

START
Initialization
Inter-molecular collision?
No
Yes
Molecule selection (one is chosen)
Molecule selection (two or more are chosen)
Satisfy the criteria of decomposition?
No
Yes
Satisfy the criteria of synthesis?
No
Yes
No
On-wall ineffective collision
Decomposition
Inter-molecular ineffective collisions
Synthesis
Check for any new min. point
Stopping criteria matched?
Yes
Obtain the best min. point
END

AWS Setup/Clustering

- Ubuntu Service

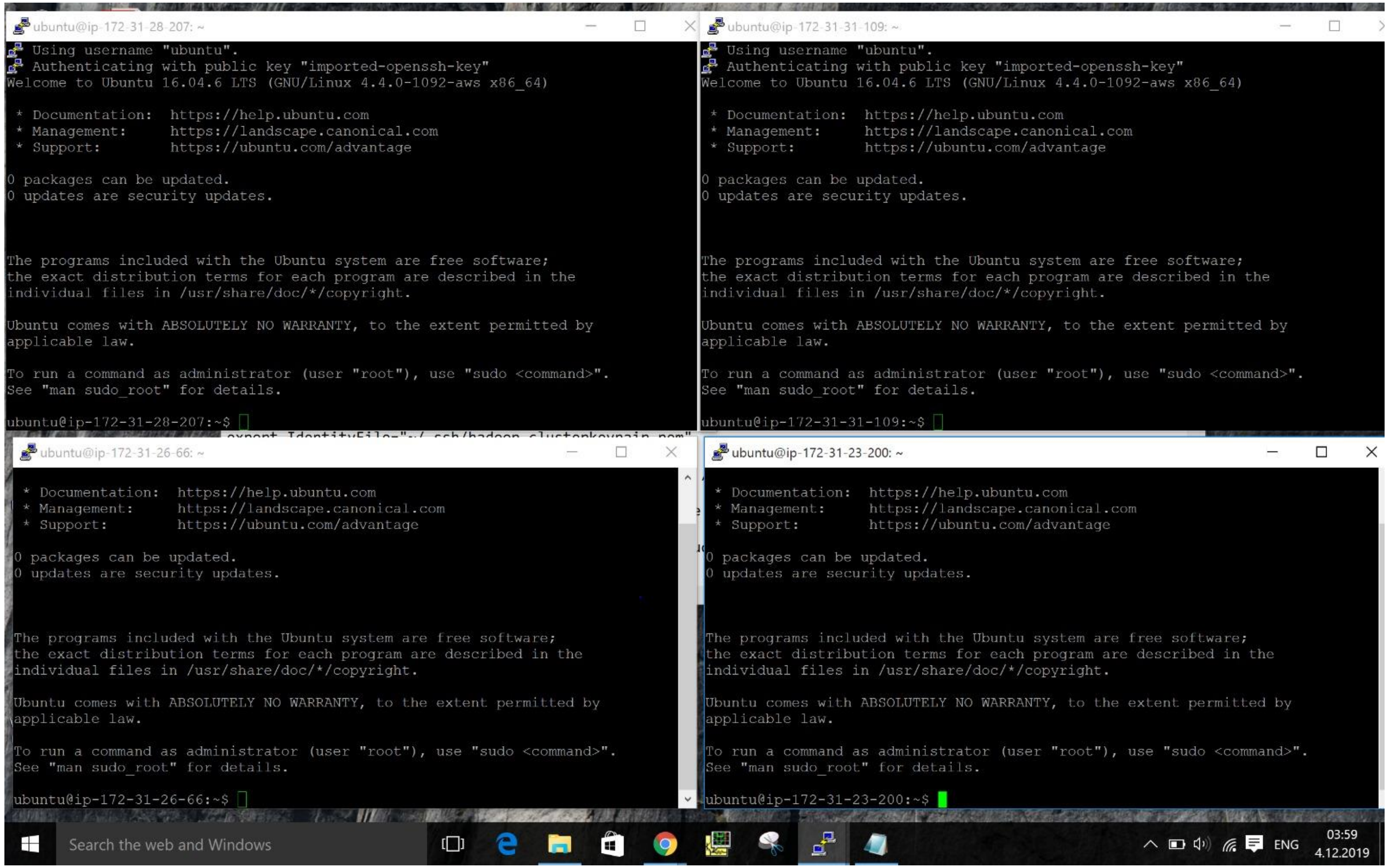
- PuTTY

- WinSCP



Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4
NameNode	i-00c9e1bd2d52b9476	t2.micro	us-east-2b	● running	✓ 2/2 checks ...	None	ec2-13-59-65-248
DataNode003	i-0663d3b3e16c1c8a4	t2.micro	us-east-2b	● running	✓ 2/2 checks ...	None	ec2-3-16-38-142.u
DataNode002	i-04ed9b72105f6e89b	t2.micro	us-east-2b	● running	✓ 2/2 checks ...	None	ec2-3-15-233-59.u
DataNode001	i-020d73fe72d5eac49	t2.micro	us-east-2b	● running	✓ 2/2 checks ...	None	ec2-18-220-14-19

PuTTY



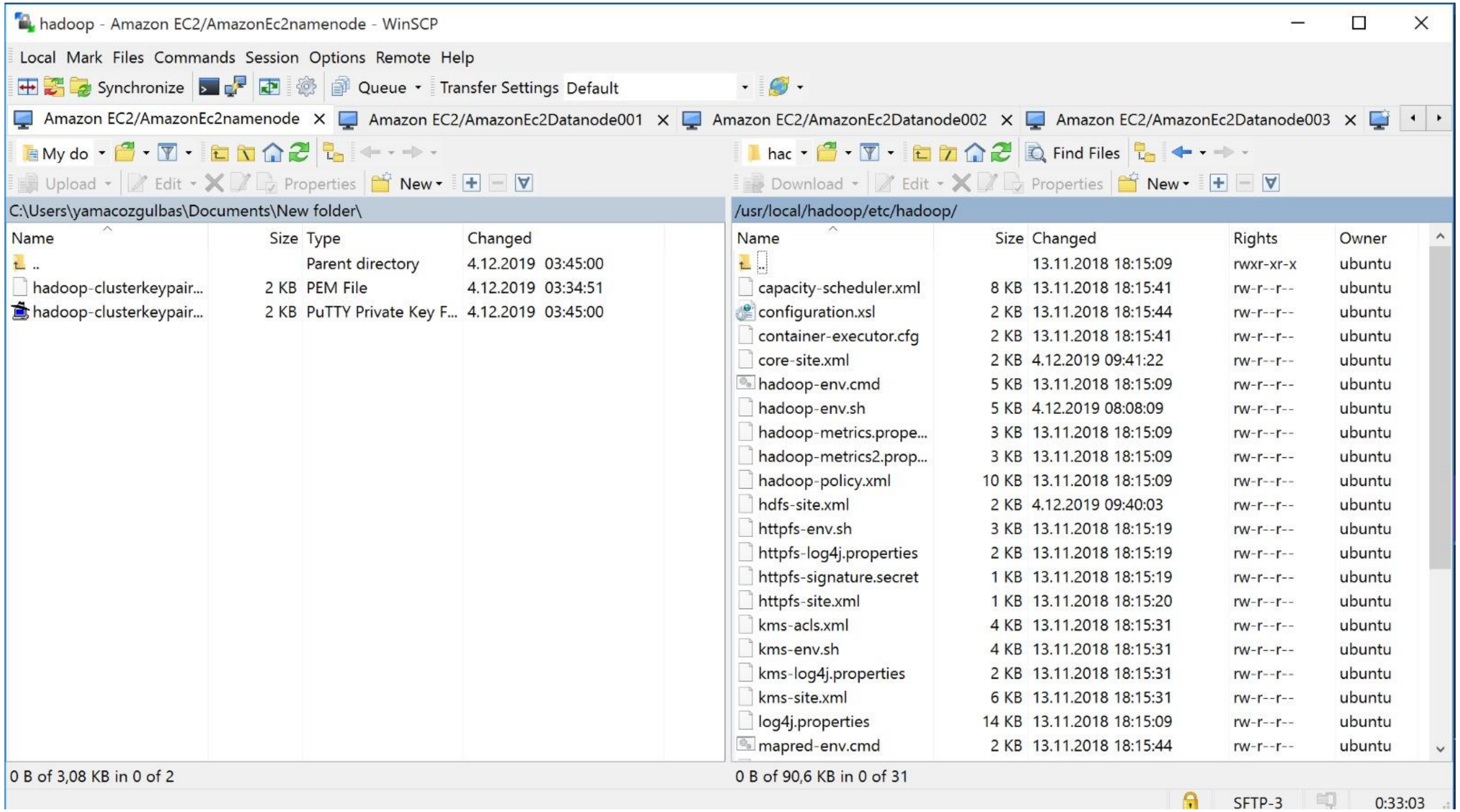
```
Using username "ubuntu".
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-1092-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 packages can be updated.
0 updates are security updates.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-28-207:~$
```

```
Using username "ubuntu".
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-1092-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 packages can be updated.
0 updates are security updates.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-31-109:~$
```

```
 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 packages can be updated.
0 updates are security updates.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-26-66:~$
```

```
 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 packages can be updated.
0 updates are security updates.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-23-200:~$
```

WinSCP


hadoop - Amazon EC2/AmazonEc2namenode - WinSCP
Local Mark Files Commands Session Options Remote Help
Synchronize
Queue
Transfer Settings Default
Amazon EC2/AmazonEc2namenode
Amazon EC2/AmazonEc2Datanode001
Amazon EC2/AmazonEc2Datanode002
Amazon EC2/AmazonEc2Datanode003
My do
Upload
Edit
Properties
New
hac
Find Files
Download
Edit
Properties
New

C:\Users\yamacozgulbas\Documents\New folder\

Name	Size	Type	Changed
..		Parent directory	4.12.2019 03:45:00
hadoop-clusterkeypair...	2 KB	PEM File	4.12.2019 03:34:51
hadoop-clusterkeypair...	2 KB	PuTTY Private Key F...	4.12.2019 03:45:00

0 B of 3,08 KB in 0 of 2

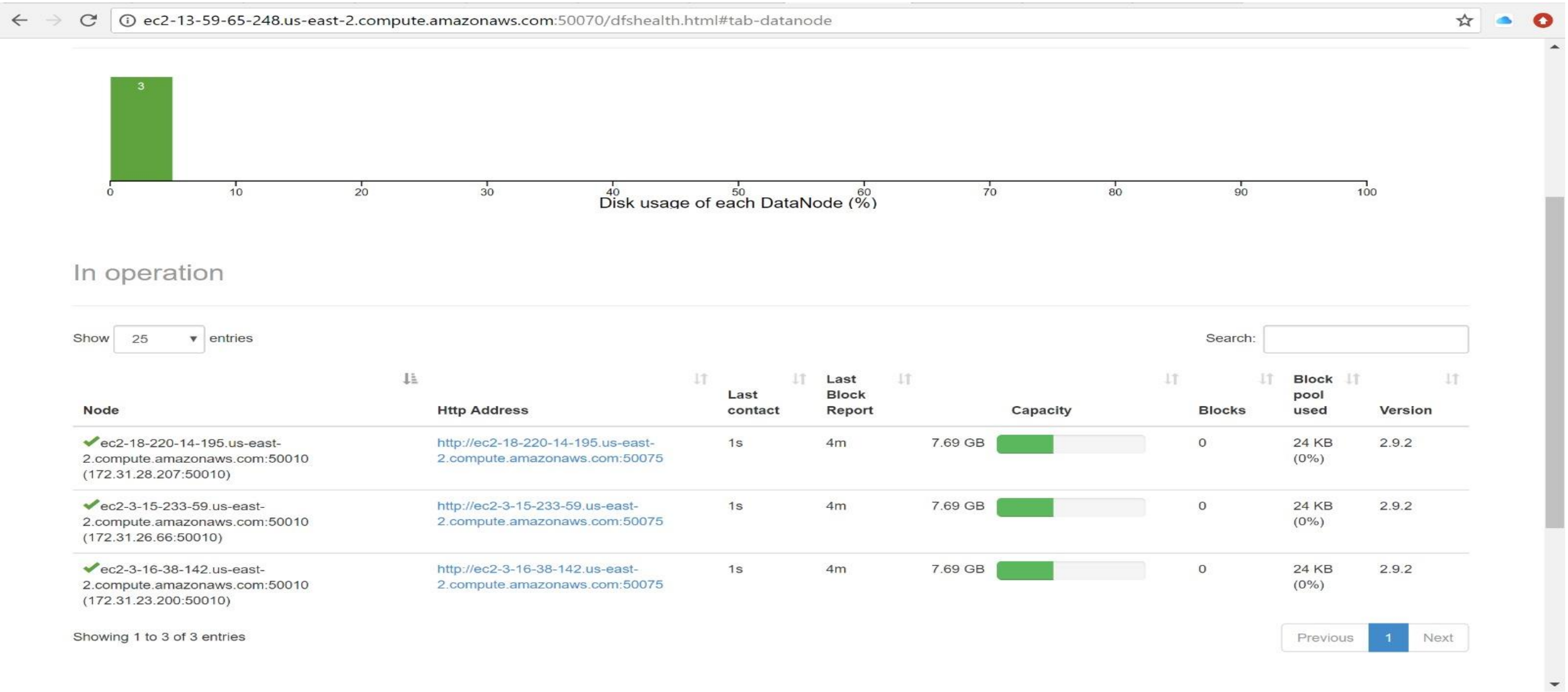
/usr/local/hadoop/etc/hadoop/

Name	Size	Changed	Rights	Owner
..		13.11.2018 18:15:09	rwxr-xr-x	ubuntu
capacity-scheduler.xml	8 KB	13.11.2018 18:15:41	rw-r--r--	ubuntu
configuration.xsl	2 KB	13.11.2018 18:15:44	rw-r--r--	ubuntu
container-executor.cfg	2 KB	13.11.2018 18:15:41	rw-r--r--	ubuntu
core-site.xml	2 KB	4.12.2019 09:41:22	rw-r--r--	ubuntu
hadoop-env.cmd	5 KB	13.11.2018 18:15:09	rw-r--r--	ubuntu
hadoop-env.sh	5 KB	4.12.2019 08:08:09	rw-r--r--	ubuntu
hadoop-metrics.prope...	3 KB	13.11.2018 18:15:09	rw-r--r--	ubuntu
hadoop-metrics2.prop...	3 KB	13.11.2018 18:15:09	rw-r--r--	ubuntu
hadoop-policy.xml	10 KB	13.11.2018 18:15:09	rw-r--r--	ubuntu
hdfs-site.xml	2 KB	4.12.2019 09:40:03	rw-r--r--	ubuntu
httpfs-env.sh	3 KB	13.11.2018 18:15:19	rw-r--r--	ubuntu
httpfs-log4j.properties	2 KB	13.11.2018 18:15:19	rw-r--r--	ubuntu
httpfs-signature.secret	1 KB	13.11.2018 18:15:19	rw-r--r--	ubuntu
httpfs-site.xml	1 KB	13.11.2018 18:15:20	rw-r--r--	ubuntu
kms-acls.xml	4 KB	13.11.2018 18:15:31	rw-r--r--	ubuntu
kms-env.sh	4 KB	13.11.2018 18:15:31	rw-r--r--	ubuntu
kms-log4j.properties	2 KB	13.11.2018 18:15:31	rw-r--r--	ubuntu
kms-site.xml	6 KB	13.11.2018 18:15:31	rw-r--r--	ubuntu
log4j.properties	14 KB	13.11.2018 18:15:09	rw-r--r--	ubuntu
mapred-env.cmd	2 KB	13.11.2018 18:15:44	rw-r--r--	ubuntu

0 B of 90,6 KB in 0 of 31

SFTP-3 0:33:03

Amazon Web Service



ec2-13-59-65-248.us-east-2.compute.amazonaws.com:50070/dfshealth.html#tab-datanode
3
0
10
20
30
40
50
60
70
80
90
100
Disk usage of each DataNode (%)

In operation

Show 25 entries

Search:

Node	Http Address	Last contact	Last Block Report	Capacity		Blocks	Block pool used	Version
✓ec2-18-220-14-195.us-east-2.compute.amazonaws.com:50010 (172.31.28.207:50010)	http://ec2-18-220-14-195.us-east-2.compute.amazonaws.com:50075	1s	4m	7.69 GB	🟩	0	24 KB (0%)	2.9.2
✓ec2-3-15-233-59.us-east-2.compute.amazonaws.com:50010 (172.31.26.66:50010)	http://ec2-3-15-233-59.us-east-2.compute.amazonaws.com:50075	1s	4m	7.69 GB	🟩	0	24 KB (0%)	2.9.2
✓ec2-3-16-38-142.us-east-2.compute.amazonaws.com:50010 (172.31.23.200:50010)	http://ec2-3-16-38-142.us-east-2.compute.amazonaws.com:50075	1s	4m	7.69 GB	🟩	0	24 KB (0%)	2.9.2

Showing 1 to 3 of 3 entries

Previous 1 Next

http://ec2-13-59-65-248.us-east-2.compute.amazonaws.com:50070/dfshealth.html#tab-overview

http://ec2-3-16-38-142.us-east-2.compute.amazonaws.com:50075/datanode.html

http://ec2-3-15-233-59.us-east-2.compute.amazonaws.com:50075/datanode.html

http://ec2-18-220-14-195.us-east-2.compute.amazonaws.com:50075/datanode.html

Datasets

Wisconsin Breast Cancer Diagnosis: A labelled dataset classifying breast cancer tumours as malignant or benign with 30 features

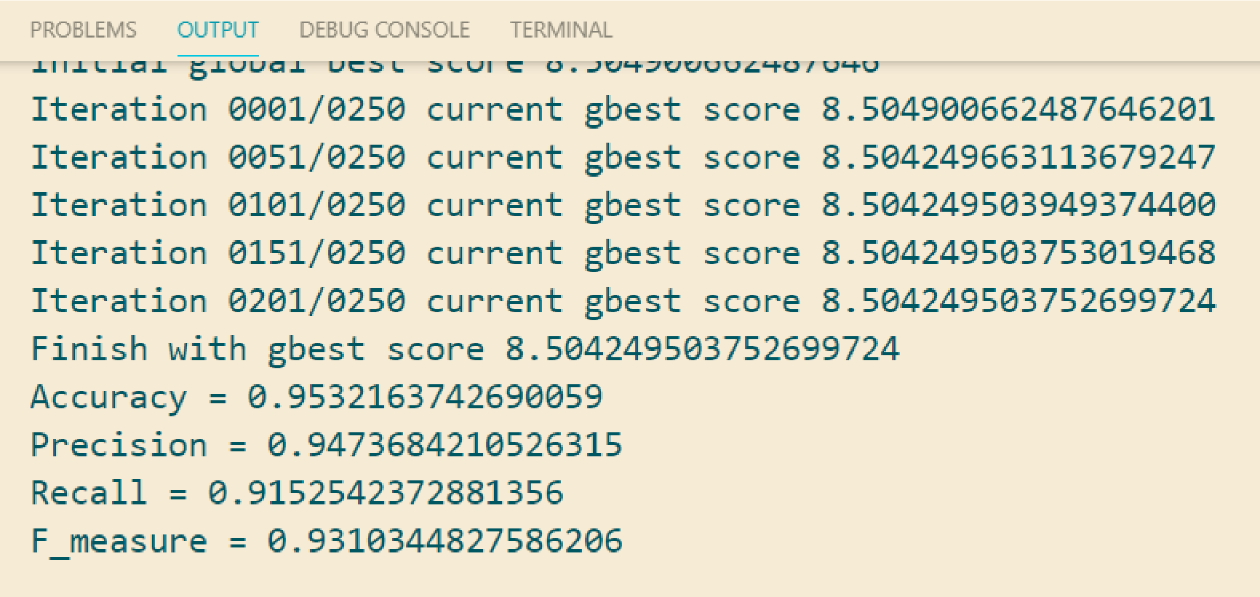
https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

Iris: A labelled dataset with three classes and four features.

https://www.kaggle.com/uciml/iris

Results

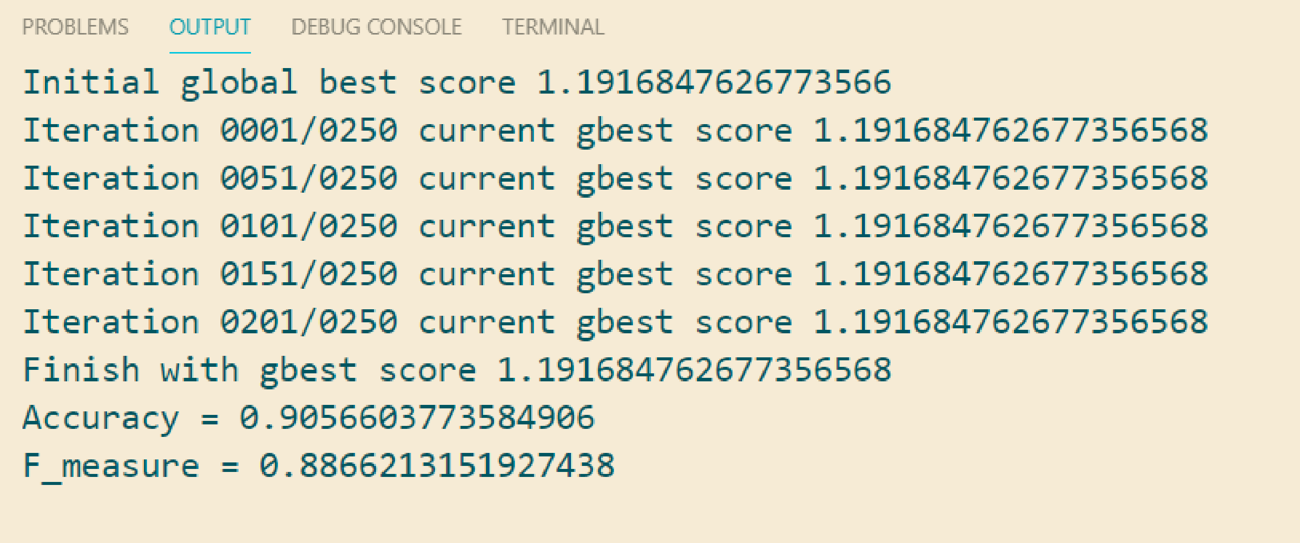
Wisconsin Breast Cancer Diagnosis



```
Initial global best score 8.504900662487646
Iteration 0001/0250 current gbest score 8.504900662487646201
Iteration 0051/0250 current gbest score 8.504249663113679247
Iteration 0101/0250 current gbest score 8.504249503949374400
Iteration 0151/0250 current gbest score 8.504249503753019468
Iteration 0201/0250 current gbest score 8.504249503752699724
Finish with gbest score 8.504249503752699724
Accuracy = 0.9532163742690059
Precision = 0.9473684210526315
Recall = 0.9152542372881356
F_measure = 0.9310344827586206
```

Results

Iris



```
Initial global best score 1.1916847626773566
Iteration 0001/0250 current gbest score 1.191684762677356568
Iteration 0051/0250 current gbest score 1.191684762677356568
Iteration 0101/0250 current gbest score 1.191684762677356568
Iteration 0151/0250 current gbest score 1.191684762677356568
Iteration 0201/0250 current gbest score 1.191684762677356568
Finish with gbest score 1.191684762677356568
Accuracy = 0.9056603773584906
F_measure = 0.8866213151927438
```

Conclusion

KMPSO implementation was successful with promising results. Both datasets yielded high accuracy and f1 scores. Additional testing is needed to determine limitations of KMPSO and comparison of performance against other optimization algorithms.

Future Work

- Perform clustering on much larger and complex data sets
- Perform clustering using CRO and compare the performance of the two optimization algorithms
- Implement clustering algorithm on a cloud platform using Hadoop or Apache Spark

Thank You
Any Questions?