

Technology for Informatics

Kmeans and MapReduce Parallelism

June 19 2013

Geoffrey Fox

gcf@indiana.edu

<http://www.infomall.org/>

Associate Dean for Research, School of Informatics
and Computing

Indiana University Bloomington

2013

Big Data Ecosystem in One Sentence

Use **Clouds** running **Data Analytics Collaboratively**
processing **Big Data** to solve problems in
X-Informatics (or e-X)

X = Astronomy, Biology, Biomedicine, Business, Chemistry, Climate,
Crisis, Earth Science, Energy, Environment, Finance, Health,
Intelligence, Lifestyle, Marketing, Medicine, Pathology, Policy, Radar,
Security, Sensor, Social, Sustainability, Wealth and Wellness with
more fields (physics) defined implicitly
Spans Industry and Science (research)

Education: **Data Science** see recent New York Times articles
<http://datascience101.wordpress.com/2013/04/13/new-york-times-data-science-articles/>



Climate Informatics
network

How Wealth Informatics can help
with your financial freedom?



Xinformatics

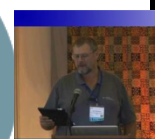
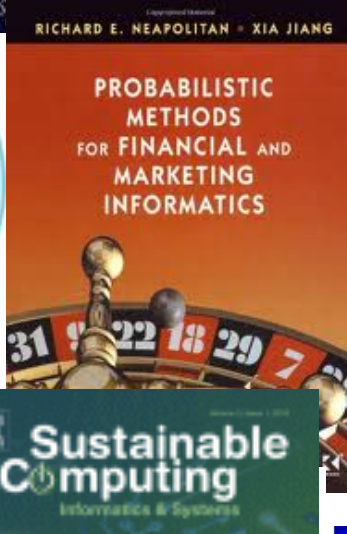
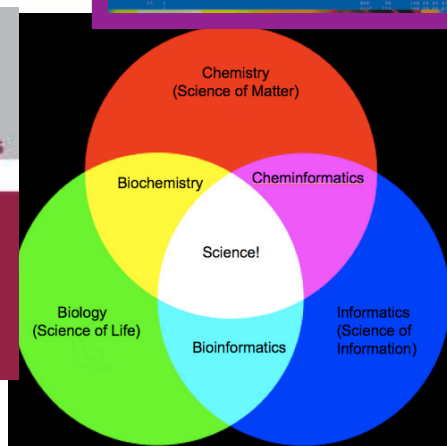
Biomedical Informatics

Computer Applications in Health Care
and Biomedicine

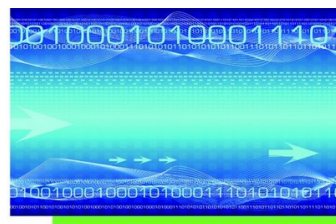
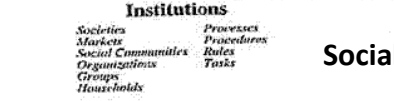
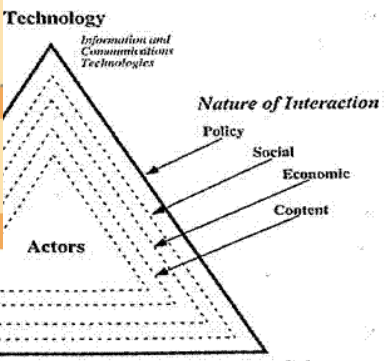
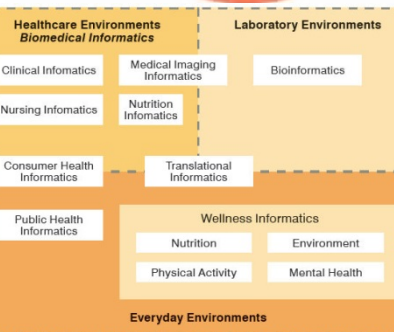
AstroInformatics2012

Redmond, WA, September 10 - 14, 2012

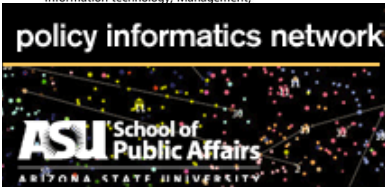
Journal of
Pathology
Informatics



Opportunities and Challenges in Crisis Informatics



Noella Penelope Greer (Ed.)
Business Informatics
Information technology, Management



USC Center For Energy Informatics

Home Research Publications Smar



About the Center
Welcome to the Center For Energy Informatics (CEI) at USC, an Organized Research Unit (ORU) housed in the [Viterbi School of Engineering](#). Energy Informatics is the application of inf

Lifestyle Informatics

Applications of LI
How is the training classified
Occupation Pr
Further study
Student at the
Watch the mov
Studying Abro

Admission and registration
VU Honours Programme

ENVIRONMENTAL INFORMATICS

Lifestyle Informatics: Let people I

The study Lifestyle Informatics is about s
this bachelor including applied psycholog
knowledge about language and informatic
short better. Lifestyle Informatics: let peo
[Lifestyle Informatics](#)

combine
body,
healthier,
[aining](#)

BACHELOR-VOORLICHTINGSDAG
ZATERDAG 3 NOVEMBER

LOOP EEN DAG MEE MET EEN STUDENT

MapReduce Kmeans in Python

Slightly Changed Sequential version

- We change to allow MapReduce implementation
- Overall iteration over starting positions
 - Initialize Centroids
 - Iterate until converged
 - Call **map** to find association and do first step in find centroids as sum over point vectors and distortion
 - **Reduce** step: Divide summed centers by points in center to get centroid
 - Do other book keeping
 - Delete zero size clusters
 - Check convergence
- Return best solution based on Quality criterion

Parallel MapReduce Version

```
lseven = np.empty([tot], dtype=bool)
for i in arange(tot):
    lseven[i] = (i%2 == 0);
obs1 = compress(lseven, obs, 0)
obs2 = compress(logical_not(lseven), obs, 0)
```

```
avg_dist = []
diff = thresh+1.
while diff > thresh:
    #
    if Parallelism == 1:
        code_book, NumPointsinClusters, distortsum, distortmax, NumPoints = Kmeans_map(obs, code_book)
    if Parallelism == 2:
        # Can be Parallel Map Operations
        code_book1, NumPointsinClusters1, distortsum1, distortmax1, NumPoints1 = Kmeans_map(obs1, code_book)
        code_book2, NumPointsinClusters2, distortsum2, distortmax2, NumPoints2 = Kmeans_map(obs2, code_book)
        #
        # Following are 4 Reduction Operations
        # Note maps include local reductions
        code_book = np.add( code_book1, code_book2)
        NumPointsinClusters = np.add( NumPointsinClusters1, NumPointsinClusters2)
        distortsum = distortsum1 + distortsum2
        distortmax = np.maximum(distortmax1, distortmax2)
        NumPoints = NumPoints1 + NumPoints2
        #
        code_book = compress(np.greater(NumPointsinClusters, 0), code_book, 0)
        # remove code_books that didn't have any members
        #
        j = 0
        nc = code_book.shape[0]
        for i in arange(nc):
            if NumPointsinClusters[i] > 0:
                code_book[j,:] = code_book[i,:] / NumPointsinClusters[i]
                j = j + 1
```

Map

Reduce

Map for Kmeans

```
def Kmeans_map(obs, code_book):
    No = obs.shape[0]
    nc = code_book.shape[0]
    # nc is current number of clusters (may decrease if zero clusters last iteration)
    #
    #compute membership and distances between obs and code_book
    obs_code, distort = vq(obs, code_book)
    distortsum = np.sum(distort)
    distortmax = np.amax(distort)
    #
    # vq returns an indexing array obs_code mapping rows of obs (the points) to code_book (the centroids)
    # distort is an array of length No that has difference between observation and chosen centroid
    # vq stands for vector quantization and is provided in SciPy
    #
    VectorDimension = obs.shape[1]
    NewCode_Book = np.zeros([nc, VectorDimension])
    NumPointsinClusters = np.zeros([nc])
    for i in arange(nc):
        #          Loop over clusters labelled with i
        cell_members = compress(equal(obs_code, i), obs, 0)
        NumPointsinClusters[i] = cell_members.shape[0]
        #          Extract Points in this Cluster; extract points whose quantization label is i
        #
        NewCode_Book[i] = np.sum(cell_members, 0)
        #          Calculate centroid of i'th cluster
    return NewCode_Book, NumPointsinClusters, distortsum, distortmax, No
```

- This routine takes Points in obs and Centroids in code_book and associates each point with nearest centroid
- It calculates Non normalized centroids (sum over vectors in cluster), Number of points in each cluster and convergence measures (mean and max of distance between points and centers)