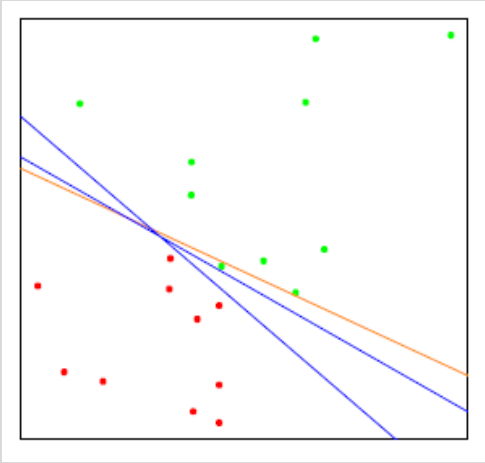


Parallel & Scalable Machine Learning

Introduction to Machine Learning Algorithms

Dr. – Ing. Morris Riedel
Adjunct Associated Professor
School of Engineering and Natural Sciences, University of Iceland
Research Group Leader, Juelich Supercomputing Centre, Germany

LECTURE 4

Unsupervised Clustering, Challenges & Solutions

January 15th, 2018
JSC, Germany

UNIVERSITY OF ICELAND
SCHOOL OF ENGINEERING AND NATURAL SCIENCES
FACULTY OF INDUSTRIAL ENGINEERING, MECHANICAL ENGINEERING AND COMPUTER SCIENCE

JÜLICH FORSCHUNGSZENTRUM

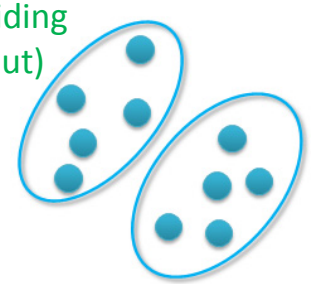
PRACE PARTNERSHIP FOR ADVANCED COMPUTING IN EUROPE
TRAINING PORTAL

Review of Lecture 3

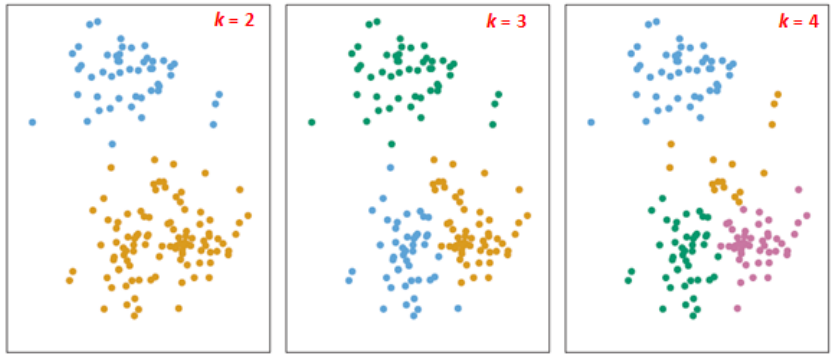
- Unsupervised Clustering
  - Input $\mathbf{x} = x_1, \ldots, x_d$
  - No output (unsupervised)
  - Data $(\mathbf{x}_1), \ldots, (\mathbf{x}_N)$
- K-Means Clustering
  - Problem of assigning K as numbers of clusters
  - Simple, works only sometimes
- DBSCAN Clustering
  - Finds arbitrary shapes and numbers of clusters
  - Still two parameters
  - Parallel version scales for big pointcloud datasets

(serial statistical computing with R tool)

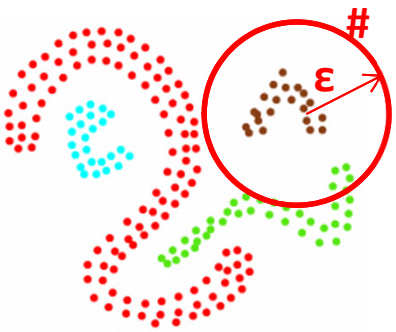
(no guiding output)

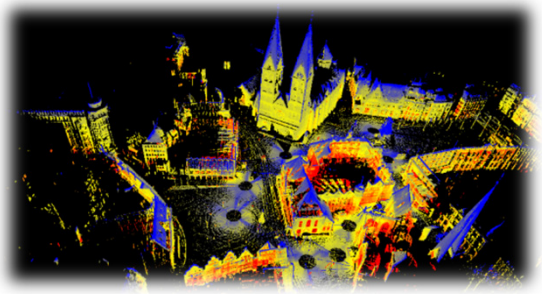


(number of cluster can be amiguities sometimes)

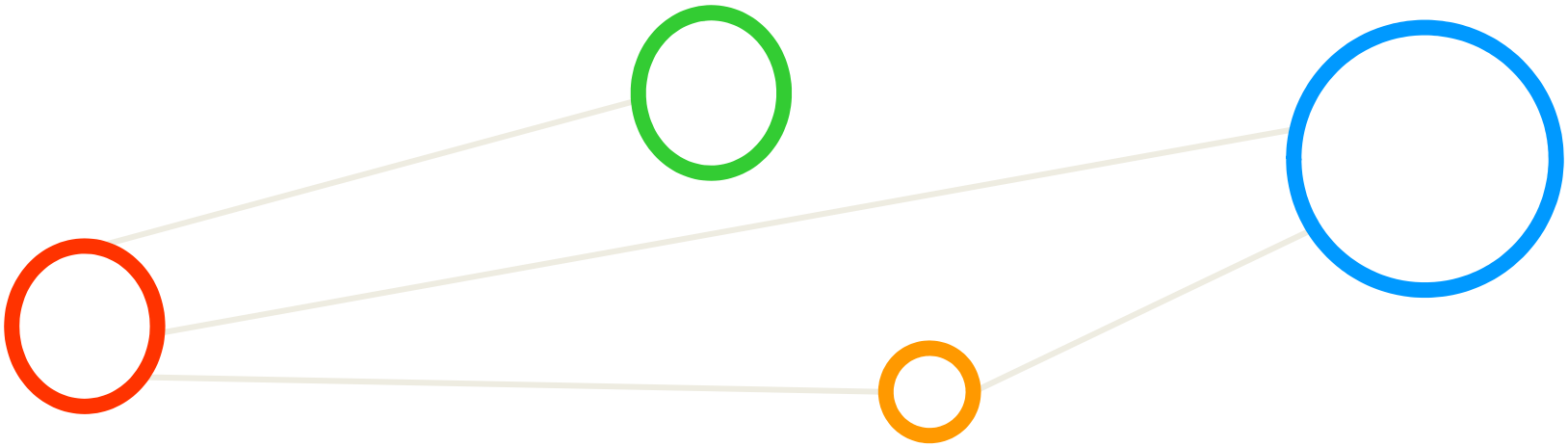


[1] An Introduction to Statistical Learning





Outline



Outline of the Course

1. Introduction to Machine Learning Fundamentals
2. PRACE and Parallel Computing Basics
3. Unsupervised Clustering and Applications
4. Unsupervised Clustering Challenges & Solutions
5. Supervised Classification and Learning Theory Basics
6. Classification Applications, Challenges, and Solutions
7. Support Vector Machines and Kernel Methods
8. Practicals with SVMs
9. Validation and Regularization Techniques
10. Practicals with Validation and Regularization
11. Parallelization Benefits
12. Cross-Validation Practicals

Day One – beginner

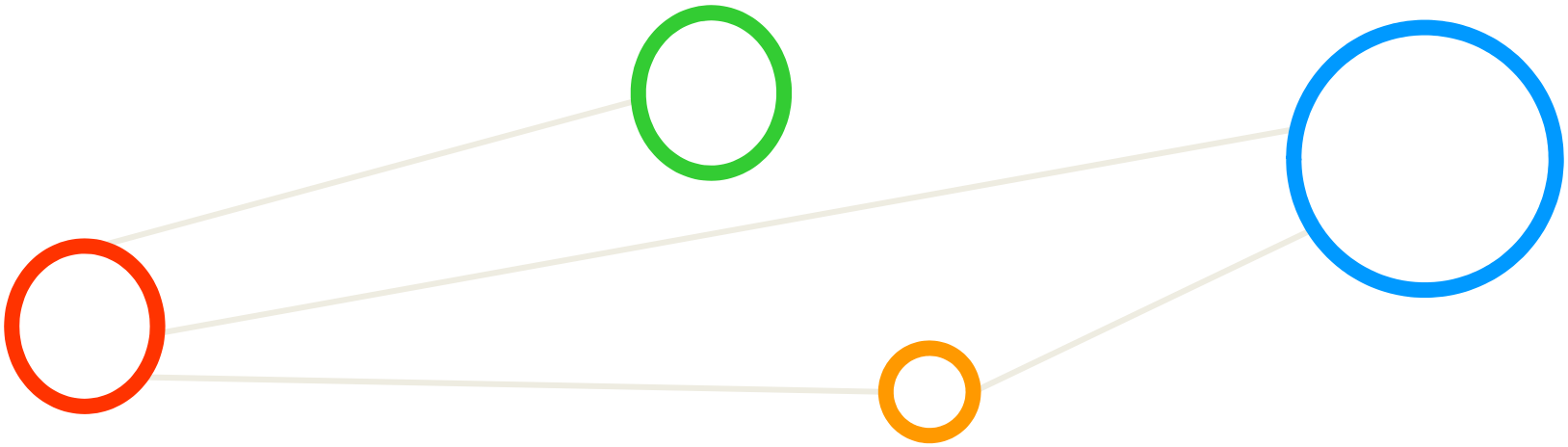
Day Two – moderate

Day Three – expert

Outline

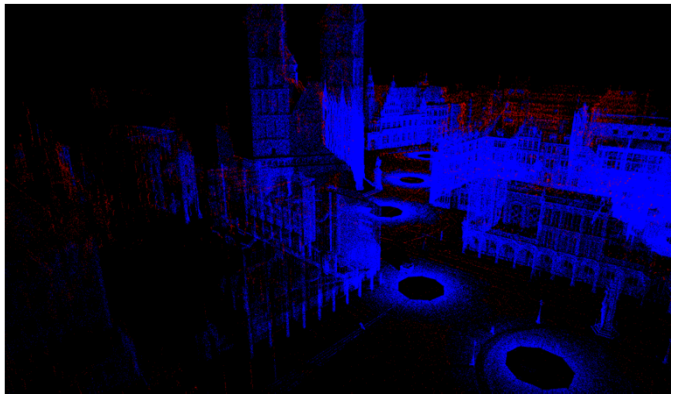
- Parallel & Scalable Clustering
  - Using HPDBSCAN for Bremen Big Dataset
  - Selected HPDBSCAN Implementation Details
  - Domain Decomposition
  - Sorting & Index
  - Hybrid Code
- Clustering Challenges & Solutions
  - Adjust Walltime to Complexity & Size
  - Change Number of Nodes
  - Change Parameters Epsilon
  - Change Parameters minPoints
  - Twitter Dataset & Interpretation

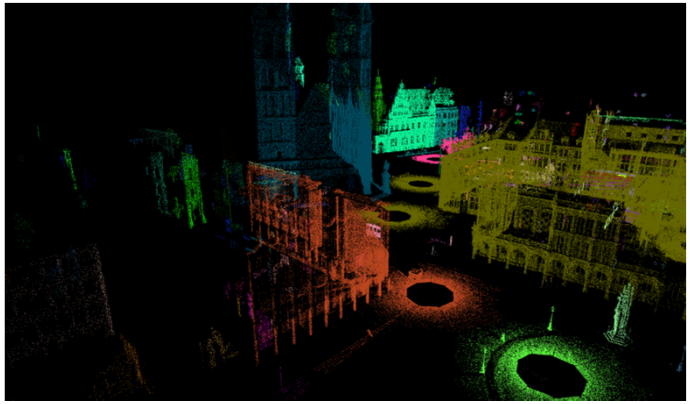
Parallel & Scalable Clustering



Bremen Dataset & Locations – Revisited

- Different clusterings of the inner city of Bremen
  - Using smart visualizations of the point cloud library (PCL)
  - Big Bremen (81 mio points) & sub sampled Small Bremen (3 mio points)





- **The Bremen Dataset is encoded in the HDF5 format (binary)**
- **You need your own copy of the file in your home directory to cluster!**

```
[train001@jrl07 bremen]$ pwd
/homea/hpclab/train001/data/bremen
[train001@jrl07 bremen]$ ls -al
total 1342208
drwxr-xr-x 2 train001 hpclab        512 Jan 14 09:58 .
drwxr-xr-x 4 train001 hpclab        512 Jan 14 08:38 ..
-rw-r--r-- 1 train001 hpclab 1302382632 Jan 14 09:56 bremen.h5
-rw-r--r-- 1 train001 hpclab   72002416 Jan 14 08:25 bremenSmall.h5
```

[3] Bremen Dataset

B2SHARE
Store and Share Research Data

Exercises – Bremen Big HPDBSCAN Runs

HPC Environment – Modules Revisited

- Module environment tool
  - Avoids to manually setup environment information for every application
  - Simplifies shell initialization and lets users easily modify their environment
- Module avail
  - Lists all available modules on the HPC system (e.g. compilers, MPI, etc.)
- Module spider
  - Find modules in the installed set of modules and more information
- Module load → needed before HPDBSCAN run
  - Loads particular modules into the current work environment, E.g.:

```
[train001@jrl12 ~]$ module load GCC

Due to MODULEPATH changes, the following have been reloaded:
  1) binutils/.2.29

The following have been reloaded with a version change:
  1) GCCcore/.5.4.0 => GCCcore/.7.2.0

[train001@jrl12 ~]$ module load ParaStationMPI/5.2.0-1
[train001@jrl12 ~]$ module load HDF5/1.8.19
```

JURECA HPC System – HPDBSCAN Job Script

```
#!/bin/bash
#SBATCH --job-name=HPDBSCAN
#SBATCH -o HPDBSCAN-%j.out
#SBATCH -e HPDBSCAN-%j.err
#SBATCH --nodes=2
#SBATCH --ntasks=4
#SBATCH --ntasks-per-node=4
#SBATCH --time=00:20:00
#SBATCH --cpus-per-task=4
#SBATCH --reservation=ml-hpc-1

export OMP_NUM_THREADS=4

# location executable
HPDBSCAN=/homea/hpclab/train001/tools/hpdbscan/dbscan

# your own copy of bremen small
BREMENSMALLDATA=/homea/hpclab/train001/bremenSmall.h5

# your own copy of bremen big
BREMENBIGDATA=/homea/hpclab/train001/bremen.h5

srun $HPDBSCAN -m 100 -e 300 -t 12 $BREMENSMALLDATA
```

- **Job submit using command: sbatch <jobscript>**
- **Remember your <jobid> that is returned from the sbatch command**
- **Show status of the job then with: squeue –u <your-user-id>**

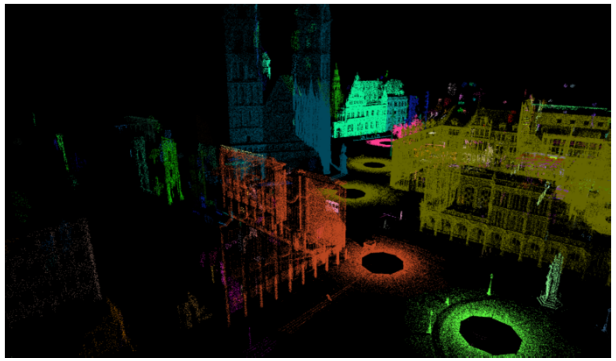
(parameters of DBSCAN and file to be clustered)

- **Note the tutorial reservation with –reservation=ml-hpc-1 just valid for today on JURECA**

JURECA HPC System – HPDBSCAN Check Outcome

```
[train001@jrl07 jsc_mpi]$ more HPDBSCAN-4629640.out
Calculating Cell Space...
        Computing Dimensions...  [OK] in 0.001657
        Computing Cells...       [OK] in 0.029877
        Sorting Points...        [OK] in 0.174414
        Distributing Points...   [OK] in 0.113745
DBSCAN...
        Local Scan...            [OK] in 58.095238
        Merging Neighbors...     [OK] in 0.005433
        Adjust Labels ...        [OK] in 0.004473
        Rec. Init. Order ...     [OK] in 0.559311
        Writing File ...         [OK] in 0.008467
Result...
        65      Clusters
        2973821 Cluster Points
        26179   Noise Points
        2953129 Core Points
Took: 59.111594s
```

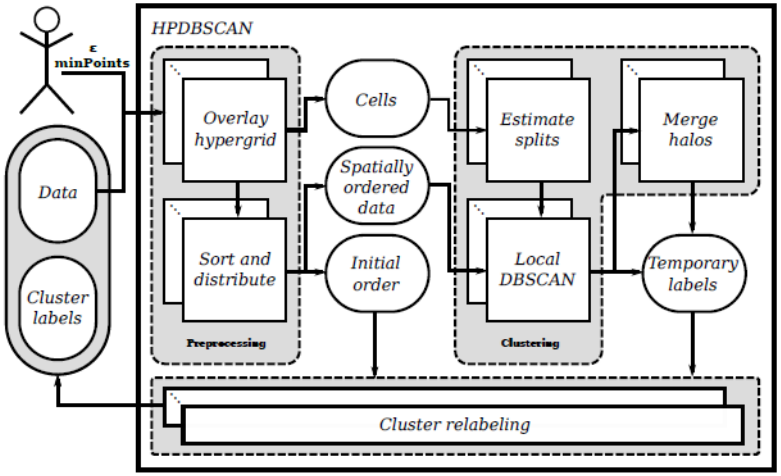
- **The outcome of the clustering process is written directly into the HDF5 file using cluster IDs and noise IDs**



```
[train001@jrl07 ~]$ ls -al
total 1124800
drwxr-x---  13 train001 hpclab      32768 Jan 14 08:47 .
drwxr-xr-x 302 root     sys         32768 Mar 25  2013 ..
-rw-------   1 train001 hpclab       7547 Jan 14 08:28 .bash_history
-rw-r--r--   1 train001 hpclab         18 Jan  8 08:58 .bash_logout
-rw-r--r--   1 train001 hpclab        176 Jan  8 08:58 .bash_profile
-rw-r--r--   1 train001 hpclab        124 Jan  8 08:58 .bashrc
drwxr-xr-x   3 train001 hpclab        512 Jan 14 00:28 bin
-rw-r--r--   1 train001 hpclab 1079412312 Jan 14 08:39 bremen.h5.h5
-rw-r--r--   1 train001 hpclab   72002416 Jan 14 08:47 bremenSmall.h5.h5
```

HPDBSCAN – Smart Domain Decomposition Example

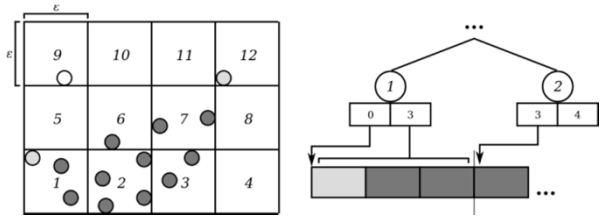
- Parallelization Strategy
  - Chunk data space equally
  - Overlay with hypergrid
  - Apply cost heuristic
  - Redistribute points (data locality)
  - Execute DBSCAN locally
  - Merge clusters at chunk edges
  - Restore initial order
- Data organization
  - Use of HDF5
  - Cluster Id / noise ID stored in HDF5 file

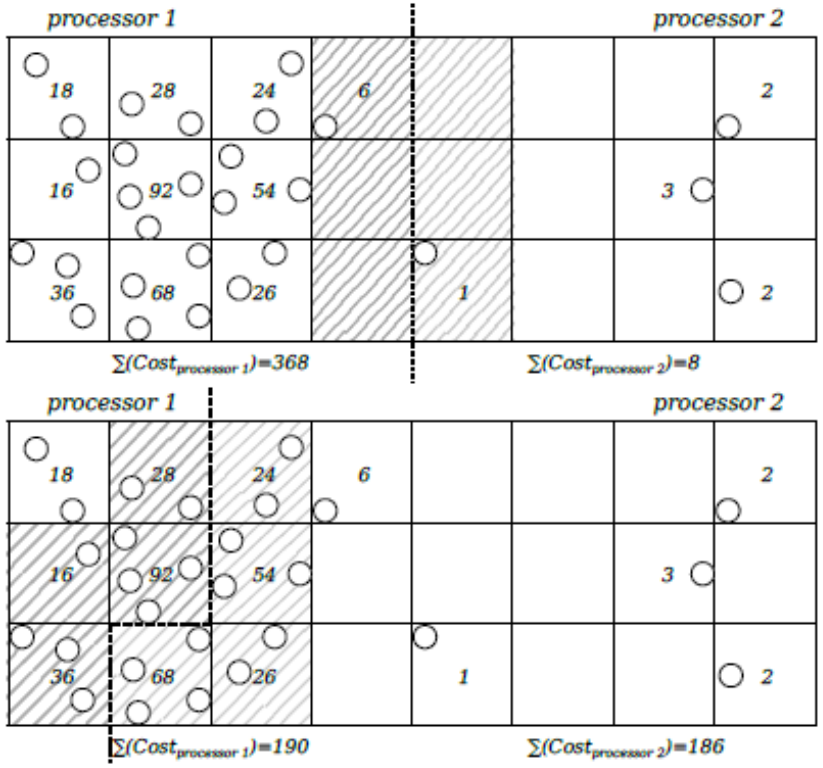


[2] M.Goetz, M. Riedel et al., 'HPDBSCAN – Highly Parallel DBSCAN', MLHPC Workshop at Supercomputing 2015

HPDBSCAN – Domain Decomposition

- Parallelization Strategy
  - Chunk data space equally
  - Overlay with hypergrid
  - Apply cost heuristic
  - Redistribute points (data locality)
  - Execute DBSCAN locally
  - Merge clusters at chunk edges
  - Restore initial order
- Data organization
  - Use of HDF5
  - Cluster Id / noise ID stored in HDF5 file

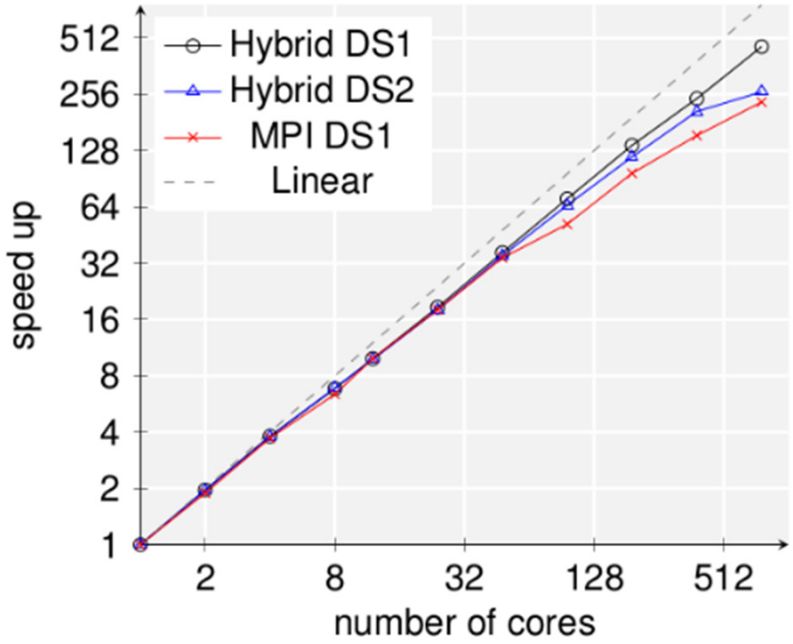




[2] M.Goetz, M. Riedel et al., 'HPDBSCAN – Highly Parallel DBSCAN', MLHPC Workshop at Supercomputing 2015

HPDBSCAN – Scaling

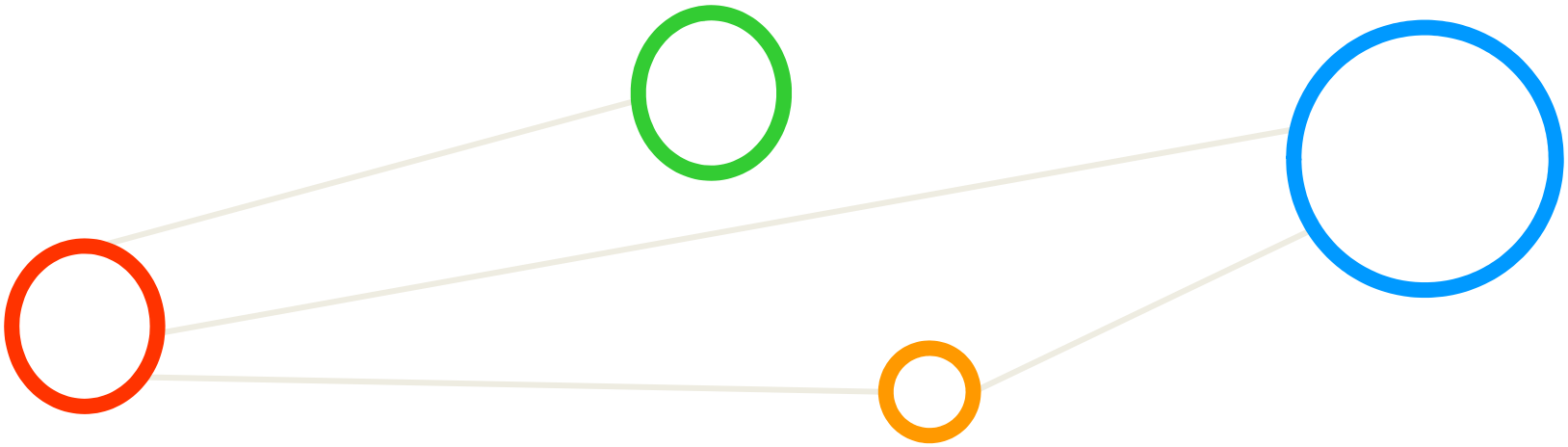
- Parallelization Strategy
  - Chunk data space equally
  - Overlay with hypergrid
  - Apply cost heuristic
  - Redistribute points (data locality)
  - Execute DBSCAN locally
  - Merge clusters at chunk edges
  - Restore initial order
- Data organization
  - Use of HDF5
  - Cluster Id / noise ID stored in HDF5 file



(DS1 = Bremen; DS2 = Twitter)

[2] M.Goetz, M. Riedel et al.,
'HPDBSCAN – Highly Parallel DBSCAN',
MLHPC Workshop at Supercomputing 2015

Clustering Challenges & Solutions



Bremen Big Dataset – ‘Running against the Wall‘ (1)

- Configured walltime
  - 1:00 hour in jobscript; 2 nodes (4 tasks per node)
- Check job status (shortly before the hour)

```
[train001@jrl07 hpdbscan]$ squeue -u train001
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
           4629896     batch HPDBSCAN train001  R      59:36      2 jrc[1250-1251]
```

- Job gets automatically cancelled by scheduler

```
[train001@jrl07 hpdbscan]$ squeue -u train001
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
           4629896     batch HPDBSCAN train001 CG    1:00:27      2 jrc[1250-1251]
[train001@jrl07 hpdbscan]$ squeue -u train001
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
[train001@jrl07 hpdbscan]$ squeue -u train001
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
```

- **In parallel & scalable machine learning one needs to adjust the walltimes of jobs to the complexity in processing time and/or size of the dataset (cf. Bremen small vs. Bremen big)**
- **Determining the right amount of walltime is not easy and mostly be best obtained by test runs**
- **The required walltime depends on the number of used nodes (and tasks) and is directly linked**

Bremen Big Dataset – ‘Running against the Wall‘ (2)

- Check outcome of the job

```
[train001@jrl07 hpdbscan]$ more HPDBSCAN-4629896.out
Calculating Cell Space...
        Computing Dimensions... [OK] in 0.043040
        Computing Cells...      [OK] in 0.157041
        Sorting Points...       [OK] in 1.041985
        Distributing Points...  [OK] in 2.126353
DBSCAN...
        Local Scan...
```

- Check error report of the job

```
[train001@jrl07 hpdbscan]$ more HPDBSCAN-4629896.err
HDF5-DIAG: Error detected in HDF5 (1.8.19) MPI-process 0:
  #000: H5F.c line 772 in H5Fclose(): not a file ID
    major: Invalid arguments to routine
    minor: Inappropriate type
error: *** step 4629896 CANCELLED DUE TO TIME LIMIT ***
srun: Job step aborted: Waiting up to 6 seconds for job step to finish.
srun: error: jrc1250: tasks 0-1: Terminated
```

- **The partial result of clustering when terminated is not useful and should be not used anymore**
- **In case of termination by scheduler even HDF problems might occur that corrupt the file**

Exercises – Increasing Number of Nodes

JURECA HPC System – HPDBSCAN Job Script

```
#!/bin/bash
#SBATCH --job-name=HPDBSCAN
#SBATCH -o HPDBSCAN-%j.out
#SBATCH -e HPDBSCAN-%j.err
#SBATCH --nodes=2
#SBATCH --ntasks=4
#SBATCH --ntasks-per-node=4
#SBATCH --time=00:20:00
#SBATCH --cpus-per-task=4
#SBATCH --reservation=ml-hpc-1

export OMP_NUM_THREADS=4

# location executable
HPDBSCAN=/homea/hpclab/train001/tools/hpdbscan/dbscan

# your own copy of bremen small
BREMENSMALLDATA=/homea/hpclab/train001/bremenSmall.h5

# your own copy of bremen big
BREMENBIGDATA=/homea/hpclab/train001/bremen.h5

srun $HPDBSCAN -m 100 -e 300 -t 12 $BREMENSMALLDATA
```

- **Job submit using command: sbatch <jobscript>**
- **Remember your <jobid> that is returned from the sbatch command**
- **Show status of the job then with: squeue –u <your-user-id>**

(parameters of DBSCAN and file to be clustered)

- **Note the tutorial reservation with –reservation=ml-hpc-1 just valid for today on JURECA**

Exercises – Changing Epsilon & MinPoints Parameters

JURECA HPC System – HPDBSCAN Job Script

```
#!/bin/bash
#SBATCH --job-name=HPDBSCAN
#SBATCH -o HPDBSCAN-%j.out
#SBATCH -e HPDBSCAN-%j.err
#SBATCH --nodes=2
#SBATCH --ntasks=4
#SBATCH --ntasks-per-node=4
#SBATCH --time=00:20:00
#SBATCH --cpus-per-task=4
#SBATCH --reservation=ml-hpc-1

export OMP_NUM_THREADS=4

# location executable
HPDBSCAN=/homea/hpclab/train001/tools/hpdbscan/dbscan

# your own copy of bremen small
BREMENSMALLDATA=/homea/hpclab/train001/bremenSmall.h5

# your own copy of bremen big
BREMENBIGDATA=/homea/hpclab/train001/bremen.h5

srun $HPDBSCAN -m 100 -e 300 -t 12 $BREMENSMALLDATA
```

(parameters of DBSCAN and file to be clustered)

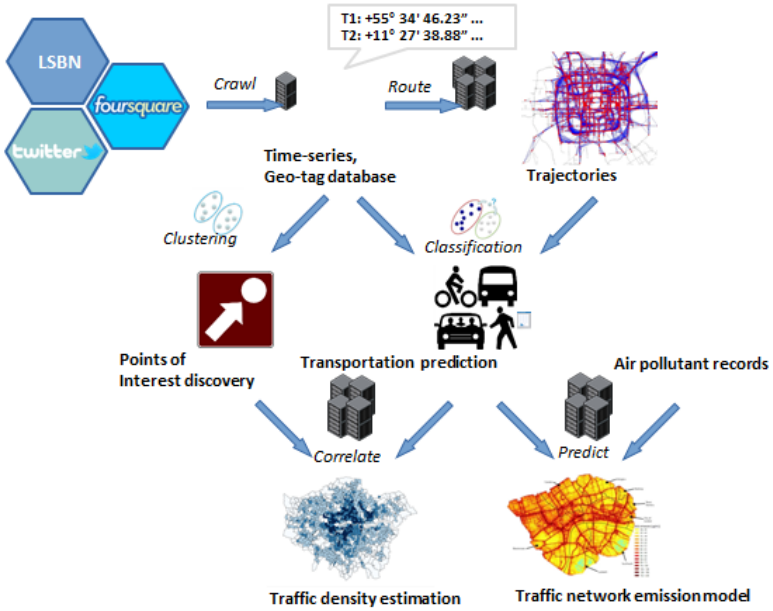
- **Job submit using command: sbatch <jobscript>**
- **Remember your <jobid> that is returned from the sbatch command**
- **Show status of the job then with: squeue –u <your-user-id>**

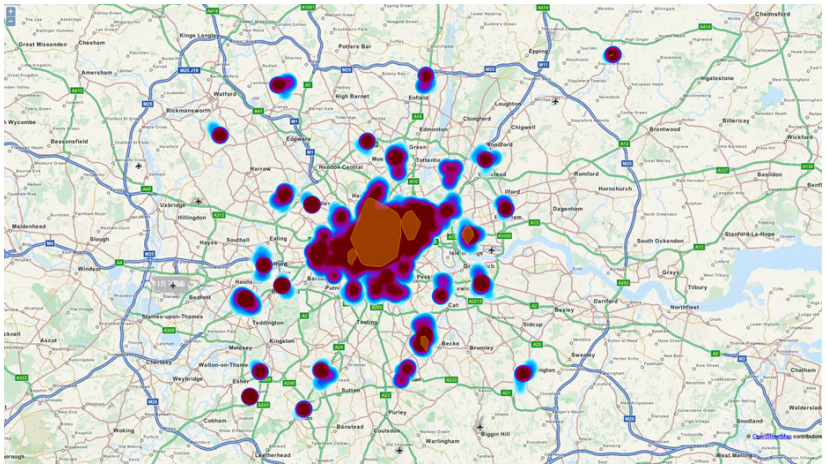
- **Note the tutorial reservation with –reservation=ml-hpc-1 just valid for today on JURECA**

Exercises – Twitter Dataset

Twitter Dataset & Locations – Revisited

- Twitter streaming API data
  - Containts 1% of all geo-tagged of the UK in June 2014 (e.g. London)





```
[train001@jrl04 twitter]$ pwd
/homea/hpclab/train001/data/twitter
[train001@jrl04 twitter]$ ls -al
total 317312
drwxr-xr-x 2 train001 hpclab       512 Jan 14 23:00 .
drwxr-xr-x 8 train001 hpclab       512 Jan 14 22:06 ..
-rw-r--r-- 1 train001 hpclab 265636608 Jan 13  2017 twitter.h5
-rw-r--r-- 1 train001 hpclab  59272032 Jan 13  2017 twitterSmall.h5
```

- **The Twitter Dataset is encoded in the HDF5 format (binary)**
- **You need your own copy of the file in your home directory to cluster!**

[4] Twitter Dataset

B2SHARE
Store and Share Research Data

JURECA HPC System – HPDBSCAN Job Script

```
#!/bin/bash
#SBATCH --job-name=HPDBSCAN
#SBATCH -o HPDBSCAN-%j.out
#SBATCH -e HPDBSCAN-%j.err
#SBATCH --nodes=4
#SBATCH --ntasks=4
#SBATCH --ntasks-per-node=4
#SBATCH --time=01:00:00
#SBATCH --cpus-per-task=4
#SBATCH --reservation=ml-hpc-1

export OMP_NUM_THREADS=4

# location executable
HPDBSCAN=/homea/hpclab/train001/tools/hpdbscan/dbscan

# your own copy of bremen small
TWITTERSMALLDATA=/homea/hpclab/train001/twitterSmall.h5

# your own copy of bremen big
TWITTERBIGDATA=/homea/hpclab/train001/twitter.h5

srun $HPDBSCAN -m 40 -e 0.0001 -t 12 $TWITTERSMALLDATA
```

(parameters of DBSCAN and file to be clustered)

- **Job submit using command: sbatch <jobscript>**
- **Remember your <jobid> that is returned from the sbatch command**
- **Show status of the job then with: squeue –u <your-user-id>**

- **Note the tutorial reservation with –reservation=ml-hpc-1 just valid for today on JURECA**

JURECA HPC System – HPDBSCAN Check Outcome

```
[train001@jrl04 hpdbscan]$ more HPDBSCAN-4632910.out
Calculating Cell Space...
        Computing Dimensions... [OK] in 0.002393
        Computing Cells...      [OK] in 0.498816
        Sorting Points...       [OK] in 0.891462
        Distributing Points...  [OK] in 2.576206
DBSCAN...
        Local Scan...           [OK] in 1.375779
        Merging Neighbors...    [OK] in 0.000586
        Adjust Labels ...       [OK] in 0.013686
        Rec. Init. Order ...    [OK] in 0.640681
        Writing File ...        [OK] in 0.006232
Result...
        8976    Clusters
        906807  Cluster Points
        2797544 Noise Points
        757369  Core Points
Took: 6.189666s
```

- **The outcome of the clustering process is written directly into the HDF5 file using cluster IDs and noise IDs**

Lecture Bibliography

Lecture Bibliography

- [1] An Introduction to Statistical Learning with Applications in R,
  Online: http://www-bcf.usc.edu/~gareth/ISL/index.html
- [2] M.Goetz, M. Riedel et al.,'HPDBSCAN – Highly Parallel DBSCAN', Proceedings of MLHPC Workshop at Supercomputing 2015,
  Online: http://www.wikicfp.com/cfp/servlet/event.showcfp?eventid=46948
- [3] B2SHARE, 'HPDBSCAN Benchmark test files', contains Bremen dataset
  Online: http://hdl.handle.net/11304/6eacaa76-c275-11e4-ac7e-860aa0063d1f
- [4] B2SHARE, 'HPDBSCAN Benchmark test files', contains Twitter dataset
  Online: http://hdl.handle.net/11304/6eacaa76-c275-11e4-ac7e-860aa0063d1f

Slides Available at http://www.morrisriedel.de/talks