

Indian Institute of Technology Madras
ZANZIBAR CAMPUS

Z3003 · DATA MINING: SCALING TO LARGE DATA

Clustering at Scale: Mini-Batch k-Means, BIRCH and DBSCAN

Wednesday 11 November 2026 · 50 minutes

Dr. Innocent Nyalala | BS Semester 5 (DSAI) | IIT Madras Zanzibar

By the End of Today You Can

IIT Madras Zanzibar

1. Explain the two walls: memory and shape
2. Describe mini-batch k-means and its trade-off
3. Describe BIRCH: one pass, (N, LS, SS) summaries
4. Run DBSCAN by hand: core, border, noise
5. Choose a clustering method for a problem

Today's Plan (45 minutes + 5 minutes questions)

IIT Madras Zanzibar

minutes	part	what you get
0 – 5	Two walls	memory and shape
5 – 14	Mini-batch k-means	learn from small batches; 1 million farms
14 – 22	BIRCH	one pass, additive summaries
22 – 36	DBSCAN	density by hand; shapes; unusual farms
36 – 45	Choosing, exercise, summary	
45 – 50	Questions	

Two Walls: Memory and Shape

IIT Madras Zanzibar

Wall 1: memory

k-means reads all data every round;
hierarchical stores n^2 distances

Wall 2: shape

k-means assumes round clusters;
forces outliers into a cluster

Mini-batch k-means

small batches → memory

BIRCH

summarise in one pass → memory

DBSCAN

density → shape + outliers

Mini-Batch k-Means: Learn From Small Batches

IIT Madras Zanzibar

1	**Take a batch**	e.g. 10,000 farms
2	**Assign**	each farm to its nearest centre
3	**Nudge**	move each centre a little toward its new farms
4	**Slow down**	centres move less as they see more farms (running mean)
5	**Repeat**	next batch: works on an endless stream

Update: centre ← centre + (farm − centre) ÷ n, n = farms this centre has seen (Week 3's running mean).

A Million Farms, 10,000 at a Time

IIT Madras Zanzibar

farms seen	vs full k-means on all 1,000,000
50,000	0.03% worse
100,000	0.01% worse
200,000	no measurable difference
300,000	no measurable difference

Memory: 1%
one batch 0.32 MB vs 32 MB

Honest note
fits in memory? full k-means takes 2.7 s

Nearly identical clusters, with 1% of the memory, on a stream that never ends.

BIRCH: Summarise in One Pass

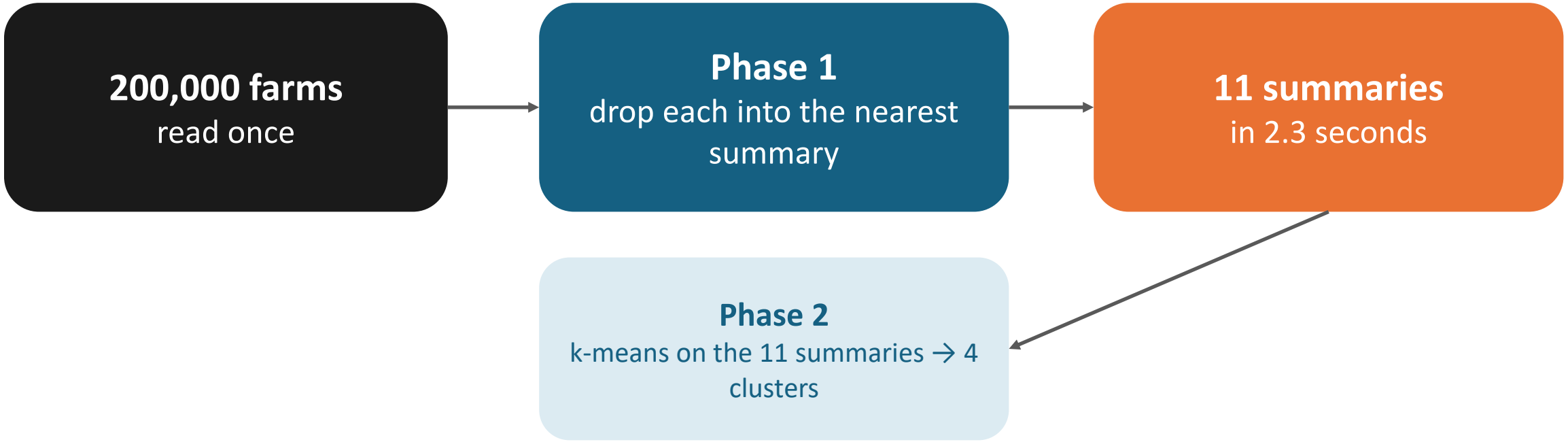
IIT Madras Zanzibar

CF = (N, LS, SS) = (count, sum, sum of squares)

	N	LS	SS
points 2, 3, 4	3	2 + 3 + 4 = 9	4 + 9 + 16 = 29
centre = LS ÷ N		9 ÷ 3 = 3	
merge with (2, 11, 61)	3 + 2 = 5	9 + 11 = 20	29 + 61 = 90

Summaries merge by ADDING: N, LS, SS are distributive (Week 4). One pass over the data.

BIRCH in Two Phases



One pass, tiny memory. Threshold = how close a point must be to join a summary.

DBSCAN: Clusters Are Crowded Places

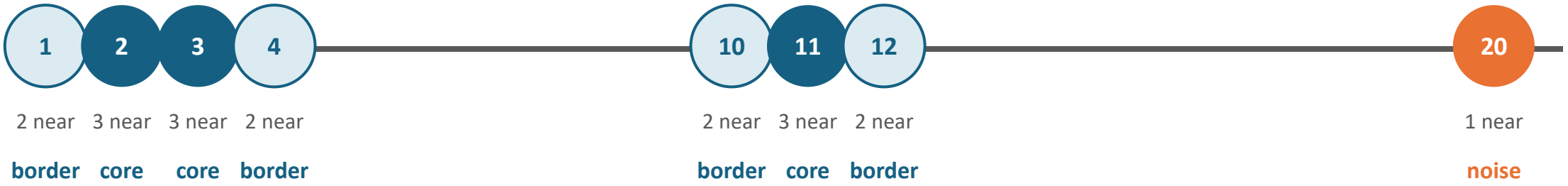
IIT Madras Zanzibar

1	**Core point**	≥ minPts points within ε (itself included): the middle of a crowd
2	**Border point**	within ε of a core point, but not core itself: edge of town
3	**Noise**	neither: an isolated house, an outlier
4	**Clusters**	core points within ε join up, with their borders. No k needed!

Like a map: towns are crowded places; isolated houses belong to no town.

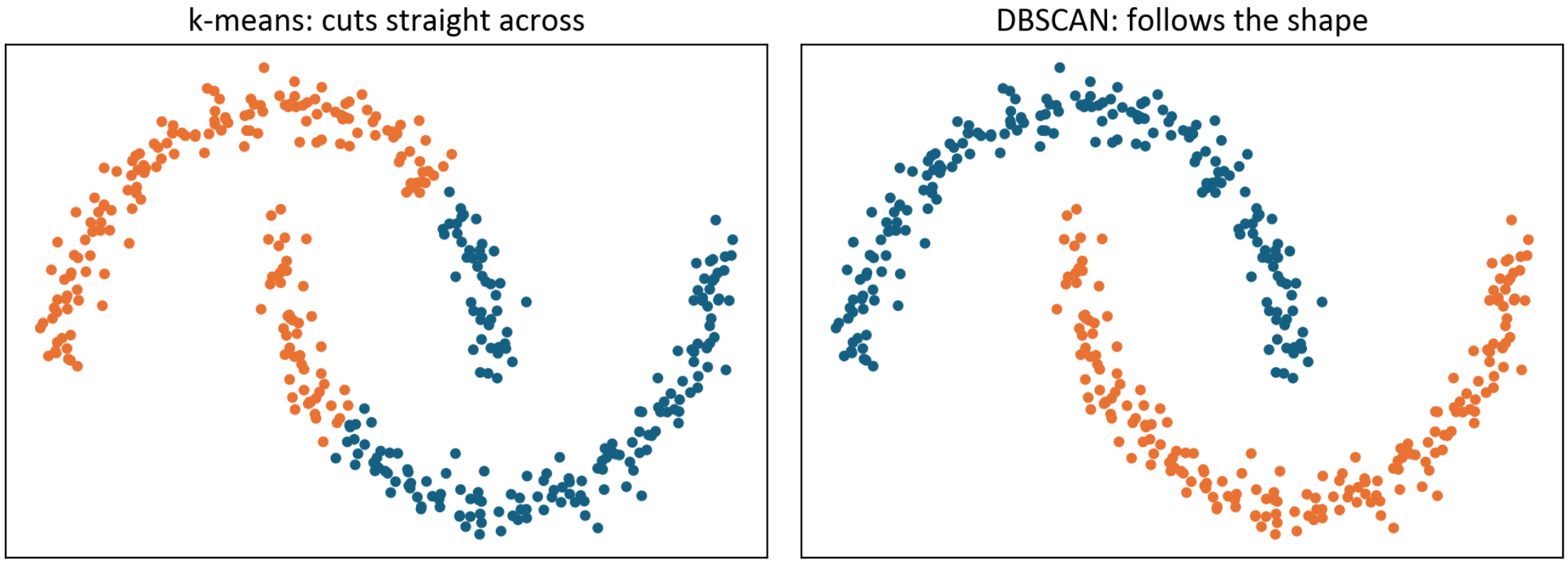
DBSCAN by Hand on a Line

Points: 1, 2, 3, 4, 10, 11, 12, 20 ε = 1.5 minPts = 3 (count the point itself)



Clusters: {1, 2, 3, 4} and {10, 11, 12}. Noise: 20.
DBSCAN found the number of clusters by itself.

DBSCAN Finds Shapes k-Means Cannot



Real non-round clusters: farms along a river, shops along a road, cases along a coast.

DBSCAN on Our Farms: Finding Unusual Farms

IIT Madras Zanzibar

4 dense groups

found without choosing k

6 unusual farms

labelled noise: median size 8.5 ha
(most farms ≈ 1 ha)

Next step

check: recording error,
or something worth learning?

k-means forces every farm into a cluster. DBSCAN can say: this farm fits nowhere.

Clustering at Scale in Python

IIT Madras Zanzibar

```
1  from sklearn.cluster import MiniBatchKMeans, Birch, DBSCAN
2  mb = MiniBatchKMeans(n_clusters=4, batch_size=10_000)
3  for batch in stream_of_farms():           # one batch at a time
4      mb.partial_fit(batch)
5  br = Birch(threshold=0.9, n_clusters=4).fit(Z_big)          # one pass
6  db = DBSCAN(eps=0.5, min_samples=10).fit(Z)
7  print("unusual farms:", (db.labels_ == -1).sum())          # label -1 = noise
```

line	what it does
2 – 4	learn from a stream
5	BIRCH: one pass
6	DBSCAN
7	count noise

DBSCAN label −1 = noise. That is how unusual farms are found.

Choosing a Clustering Method

IIT Madras Zanzibar

method	needs k?	shapes	outliers	scale
k-means	yes	round	forced in	large, in memory
mini-batch k-means	yes	round	forced in	streams; bigger than memory
hierarchical (Ward)	no: cut later	round-ish	forced in	small (n^2 memory)
BIRCH	phase 2	round-ish	absorbed	very large, one pass
DBSCAN	no (ε, minPts)	ANY	labelled noise	large, with an index

Start with k-means. Odd shapes or outliers → DBSCAN. Too big → mini-batch or BIRCH. Levels → hierarchical.

Scaled Clustering in the Real World

IIT Madras Zanzibar

Mobile operators

mini-batch k-means on millions

Satellite images

billions of pixels → land types

Fraud

DBSCAN noise = unusual payments

Epidemiology

density-based hotspots

Ride-hailing

pickup hotspots from GPS

Agriculture

unusual farms flagged for a visit

Summarise or sample to scale; use density to find shapes and outliers.

Your Turn (5 minutes, pairs)

IIT Madras Zanzibar

Points: 0, 1, 2, 6, 7, 8, 9, 15 ε = 1.5 minPts = 3 (count the point itself)

1. For each point, count the points within ε
2. Label each point core, border or noise
3. List the clusters DBSCAN finds
4. BIRCH: give the CF (N, LS, SS) of {6, 7, 8, 9} and its centre

Solution, Step by Step

IIT Madras Zanzibar

step	working	answer
counts	0:2 · 1:3 · 2:2 · 6:2 · 7:3 · 8:3 · 9:2 · 15:1	
core	≥ 3 points within 1.5	1, 7, 8
border	near a core point	0, 2, 6, 9
noise	neither	15
clusters	cores 7 and 8 are within 1.5: one group	{0,1,2} and {6,7,8,9}
BIRCH CF	N 4 · LS 30 · SS 36+49+64+81 = 230	centre 7.5

Summary

IIT Madras Zanzibar

1	**Mini-batch k-means**	small batches; same clusters, 1% of the memory
2	**BIRCH**	one pass; additive (N, LS, SS): 200,000 farms → 11 summaries
3	**DBSCAN**	dense regions of any shape; outliers = noise (6 unusual farms)
4	**Choose by need**	size → mini-batch / BIRCH; shape / outliers → DBSCAN

Next: the whole course as one data mining pipeline, and the handover to Part 2.

Questions (5 minutes)

IIT Madras Zanzibar

Q: How do I choose ε?

A: Sort each point's distance to its k-th neighbour; pick ε at the bend.

Q: Is DBSCAN always better?

A: No: struggles with very different densities; needs a good ε.

Q: Does mini-batch match full k-means?

A: Nearly; the gap shrinks as it sees more data.

Q: Can BIRCH find odd shapes?

A: Not well: its summaries are round, like k-means.