TUTORIAL 4 · STUDENT HANDOUT

Build a Star Schema and Explore the Cube with OLAP

Thursday 8 October 2026, 4:45–5:30 pm

Today: you turn 200,000 farm-season records into a star schema, then use roll-up, drill-down, slice, dice and pivot to find the 2017 drought yourself. Every task has a checkpoint number in this handout.

1. Learning outcomes

- Build dimension tables and a fact table (a **star schema**) with pandas `merge`
- Apply the five OLAP operations — **roll-up, drill-down, slice, dice, pivot** — and read the answers
- Compute all $2^3 = 8$ **cuboids** of a 3-dimension cube and show why pre-computing one pays
- Explain the **averaging trap**: why an average of averages is wrong, and how SUM / COUNT fixes it

2. What you need

Item	Details
Notebook	`W4_Tutorial4_STUDENT.ipynb`
Dataset (new this week)	`datasets/zanzibar_farm_seasons.csv` — 200,000 rows = 10,000 farms × 10 years × 2 seasons (≈ 21 MB)
Documentation	`datasets/DATA_DICTIONARY.md` — every column, and the deliberate data problems
Packages	pandas, numpy (no new installs)
From lectures	Lecture 10 (cubes, star schema, measures) and Lecture 11 (OLAP operations)

3. Plan for the 45 minutes

Time	Part	What you do	Checkpoint
4:45	A. Load & clean	replace 25 gauge errors (9999) with NaN	rain missing 2,021
4:50	B. Star schema	3 dimension tables + fact table; one star query	11 / 20 / 5; Pemba South 23.2
5:00	C. OLAP	roll-up, drill-down, slice, dice, pivot	245.6; 2.22; 175.8; 986.0
5:15	D. Cuboids	all 8 cuboids; timing; averaging trap	250 … 1 cells; 3.088 vs 2.700
5:22	E. Graded	4 questions	see rubric

4. The ideas you need — from zero

4.1 Facts, measures, dimensions and the star schema

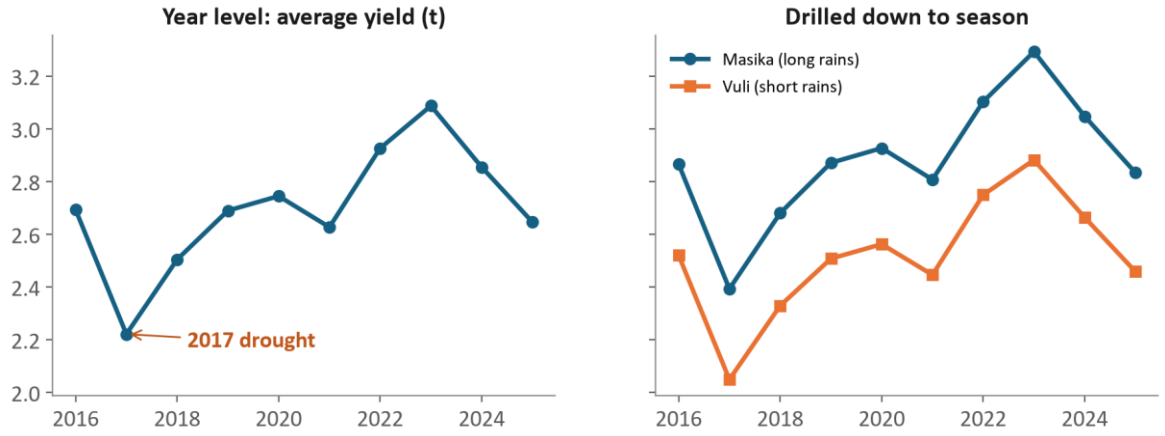
A **fact** is one event (one farm in one season). A **measure** is a number worth adding up (yield, revenue). A **dimension** is a way to group (region, crop, year). A **star schema** stores the facts in one table holding measures plus a **key** for each dimension; each dimension's descriptive text lives once, in its own small table.

Table	Rows	Columns
fact (centre)	200,000	record_id, farm_id, location_key, time_key, crop_key, yield, revenue, rainfall, price
dim_location	11	location_key, district, region
dim_time	20	time_key, year, season
dim_crop	5	crop_key, crop, crop_group

Why keys? storing 'Pemba North' 40,000 times wastes memory. In the notebook the fact table uses about 14 MB instead of 111 MB.

4.2 The five OLAP operations

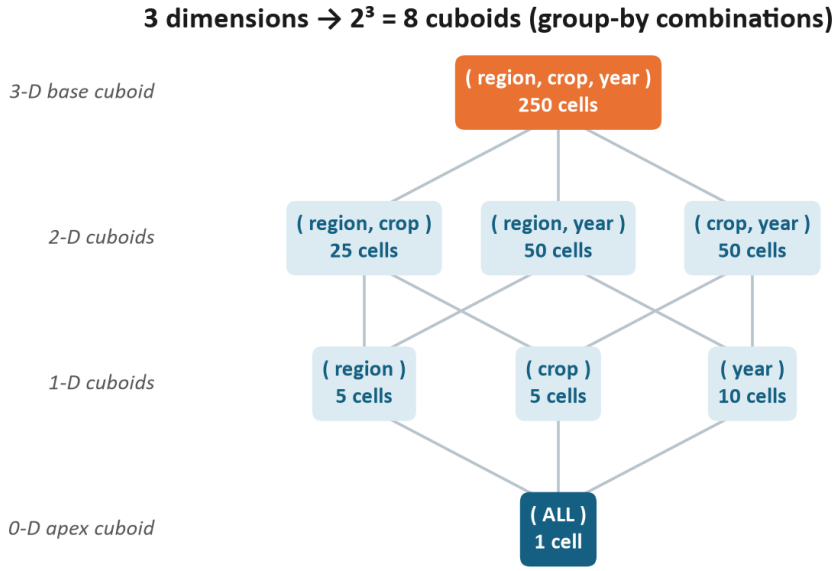
Operation	Meaning	pandas	Our example
Roll-up	up a hierarchy / drop a dimension	`groupby` the higher level	district → region
Drill-down	down a hierarchy / add a dimension	`groupby` more columns	year → (year, season)
Slice	one dimension = one value	`df[df.year == 2017]`	the drought year
Dice	several dimensions restricted	`isin(...)` & `between(...)`	Pemba × Clove × 2021–25
Pivot	swap rows and columns	`pivot_table(index=…, columns=…)`	crop × year ↔ year × crop



Drill-down finds the story: 2017's low yield appears in BOTH seasons — an island-wide drought.

4.3 Cuboids and the averaging trap

With 3 dimensions each can be in or out of a group-by: 2 × 2 × 2 = **8 cuboids**, from the base cuboid (region, crop, year: 250 cells) to the apex (1 cell: the grand total). A coarse cuboid can be computed from a finer one because SUM is **distributive**. AVG is **algebraic**: never average averages — combine SUMs and COUNTs, then divide.



5. Checkpoints — compare after every part

Part	Task	☑ You should see
A	clean 9999 codes	25 errors; rainfall missing 2,021; yield missing 8,111
B1	dimension tables	11 locations, 20 time periods, 5 crops
B2	fact table	(200000, 9); ≈ 14 MB vs ≈ 111 MB
B3	clove revenue by region, 2023	Pemba South largest ≈ 23.2 $M
C1	roll-up	123.0 + 122.5 = 245.6 $M (Pemba South)

C2	drill-down	2017: 2.22 t; Masika 2.39, Vuli 2.05
C3	slice 2017	clove ≈ 1.56 t in every region
C4	dice	11,862 rows with revenue; 175.8 $M
C5	pivot with ALL	grand total ≈ 986.0 $M
D1	8 cuboids	250, 25, 50, 50, 5, 5, 10, 1 cells
D2	pre-computed cuboid	same answer: True; faster
D3	averaging trap	3.088 t (wrong) vs 2.700 t (true)

6. Graded exercise (Part E) and rubric

#	Task	What 'complete' looks like	Marks
E1	Roll up crop → crop_group	cash crops 844.6 $M; food crops 141.4 $M	2
E2	Iceberg: cells with ≥ 100 records	1,100 cells, 1,000 kept	2
E3	Bitmap AND: Clove & Pemba South	14,940 rows, using & on True/False arrays	2
E4	Explain the 2017 dip with OLAP	correct order (roll-up → drill-down → slice) and at least one quoted number	4

7. Common errors and fixes

Symptom	Cause	Fix
merge gives more than 200,000 rows	duplicate keys in a dimension table	use drop_duplicates() when building it
KeyError: 'region' after merging	the column was dropped from the fact table	merge the dimension table back first
Averages look too high in 2017 checks	9999 codes not cleaned	re-run Part A
pivot_table very slow	too many columns in index	select only the columns you need

8. Submission

- Run all cells so outputs are visible
- Save as **YourName_W4_Tutorial4.ipynb**; upload before **5:30 pm**
- Mini-project: dataset approval checkpoint is **Friday 16 October**