BP301T · B.PHARM THEORY

BP301T Introduction to Machine Learning in Pharmaceutical Sciences

Unit V — Unsupervised Learning & Case Studies

Pills & Python Notes

From-zero teaching standard: every technical term is defined before use.

PILLS & PYTHON

Pharmacy • Data • AI • Healthcare

Instagram @pillsandpython · pillsandpython.com · YouTube @PillsAndPython

CHAPTER 5 — UNIT V: Unsupervised Learning & Practical Case Studies

1. Learning Objectives

After studying this chapter, you will be able to:

- Explain unsupervised learning and how it differs from supervised learning.
- Describe clustering and the K-means algorithm step by step.
- Explain how to choose the number of clusters (elbow method, domain knowledge).
- Interpret cluster outputs (centers, labels, sizes) in plain words.
- Carry out a mini-case study in sklearn combining regression, classification or clustering.
- Show how patient segmentation and drug grouping by properties work.

2. Definitions

Unsupervised learning. Unsupervised learning learns from data that has *no labels* — no answer column. Instead of predicting, it discovers structure: groups, patterns or unusual cases hidden in the data.

Clustering. Clustering is the main unsupervised task: automatically dividing records into groups (clusters) so that members of one group are similar to each other and different from other groups.

K-means clustering. K-means is the most widely used clustering algorithm. You choose K (the number of groups); the algorithm places K centers and repeatedly (a) assigns each record to the nearest center and (b) moves each center to the middle of its assigned records, until the groups stop changing.

Cluster center (centroid). The middle point of a cluster — the average of its members. It summarises "what a typical member of this group looks like."

Patient segmentation. Dividing patients into meaningful groups (e.g., young-healthy, elderly-high-risk) so care, monitoring or dosing can be tailored per group.

Drug grouping. Clustering drug molecules by properties (solubility, molecular weight, lipophilicity) to find families of similar compounds — useful in drug discovery and formulation.

3. Terminology

Term	Plain meaning
Label-free data	A table with no answer column — only features.
K	The number of clusters you ask K-means to find. You choose it; the algorithm does not.
Centroid	The center (average point) of a cluster.
Inertia	Total squared distance of points from their centers — measures how "tight" the clusters are (lower = tighter).
Elbow method	Trying K = 1, 2, 3... and picking the K where improvement sharply slows (the "elbow" bend).
Cluster label	The group number assigned to each record (0, 1, 2...).
Pattern recognition	The general goal: finding meaningful structure in data without being told what to look for.

4. Explanation: the concept of unsupervised learning

Technical concept. With no labels to predict, the algorithm optimises an internal quality measure — for K-means, minimising inertia (compactness of groups) — rather than matching known answers.

> **Simple explanation.** Supervised learning is a student with an answer key; unsupervised learning is a librarian given a pile of unsorted books and told "organise these sensibly" — by size, subject, author — with no instructions about the right categories.

Supervised vs unsupervised — table.

	Supervised (Units II–IV)	Unsupervised (Unit V)
Labels	Yes	No
Task	Predict the answer	Discover groups/patterns
Evaluation	Compare with true labels (accuracy, RMSE)	Internal measures (inertia) + human judgement
Example	Predict ADR yes/no	Group patients by similarity

Pharmacy example. A company has 2,000 compounds with measured solubility and molecular weight but no labels. Clustering reveals three natural families: small highly-soluble, large poorly-soluble, and in-between — guiding which formulation strategy each family needs.

> **Key takeaway.** No labels → no prediction; instead, discovery of natural groups. The human expert then names and uses the groups.

5. Clustering overview and K-means step by step

Technical concept. K-means: (1) choose K; (2) place K initial centers; (3) assign every point to the nearest center; (4) recompute each center as the mean of its points; (5) repeat steps 3–4 until assignments stop changing.

> **Simple explanation — the classroom analogy.** A teacher wants 2 groups. She picks 2 random desks as group centers. Every student sits near the closer center. Then she moves each center to the middle of its seated students. Students re-choose the nearer center; centers move again — until nobody wants to move. Groups formed!

Step-by-step with a tiny example. Points: A(1,1), B(2,1), C(8,8), D(9,8); K = 2.

1. Initial centers: A(1,1) and C(8,8).
2. Assign: A, B nearer to (1,1); C, D nearer to (8,8).
3. New centers: mean of {A,B} = (1.5, 1); mean of {C,D} = (8.5, 8).
4. Re-assign: same groups — stop. Two clusters found: {A, B} and {C, D}.

Diagram (draw this). Four panels: (1) scattered dots with 2 star-marked initial centers; (2) dots coloured by nearest center; (3) stars moved to group middles; (4) final two coloured groups with centered stars labelled "centroids."

Flowchart (draw this). Start → "Choose K" → "Place K centers" → "Assign each point to nearest center" → "Move centers to group means" → Diamond: "Any point changed group?" → YES → back to assign; NO → "Done — report labels and centers."

Key takeaway. K-means = pick K, then repeat "assign to nearest center, move center to the middle" until stable.

6. Choosing the number of clusters

Technical concept. K is a human choice. Two guides: (a) the elbow method — plot inertia vs K and pick the bend where extra clusters stop helping much; (b) domain knowledge — how many groups are clinically meaningful.

Simple explanation. How many groups should the librarian make? Try 2, 3, 4 shelves and see where adding another shelf barely improves tidiness — that bend is the elbow. Also ask the pharmacist: "do 3 patient groups make medical sense?"

Worked example (verified by Python, hypothetical patient data). Inertia values: K=1 → 2046.2; K=2 → 310.9; K=3 → 70.5; K=4 → 41.2. The drop from K=1 to K=2 is huge (2046 → 311); after K=2 the gains shrink. The elbow is at **K = 2** — two natural patient groups.

Diagram (draw this). Line graph: X-axis K (1–4), Y-axis inertia. Steep fall from 1 to 2, then gentle slope — circle the bend at K = 2, label "elbow → choose K = 2."

Common mistake. Always picking K = 2 or 3 out of habit without checking the elbow or asking what the groups mean clinically.

Key takeaway. Elbow method + clinical sense together choose K — never K alone by guesswork.

7. Interpreting cluster outputs

Technical concept. K-means outputs: a label per record (which group) and K centroids (group averages). Interpretation = describing each centroid in domain words and checking group sizes for usefulness.

Simple explanation. The computer hands you numbered groups; *you* give them names. "Group 0: average age 29, average BMI 23 — young, healthy weight. Group 1: average age 55, average BMI 30 — older, higher BMI."

Worked example (verified by Python, hypothetical data). 10 patients [age, BMI], K = 2:

- Labels: `[0 0 0 0 1 1 1 1 1 1]` — first four patients in group 0, rest in group 1.
- Centroids: group 0 = **[28.75, 23.0]**, group 1 = **[54.67, 30.17]**.
- Interpretation: Group 0 = younger patients (avg ~29 y) with healthy BMI (~23); Group 1 = older patients (avg ~55 y) with higher BMI (~30). A hospital could schedule more frequent metabolic screening for Group 1 — that is patient segmentation in action.

Drug grouping example (verified by Python, hypothetical data). 6 compounds [solubility mg/mL, molecular weight], K = 2:

- Labels: `[0 0 1 1 0 1]`; centroids: group 0 = **[6.33, 200]**, group 1 = **[108.33, 480]**.
- Interpretation: Group 0 = small, poorly soluble molecules (may need solubility enhancement); Group 1 = larger, highly soluble molecules. Formulation teams treat each family differently.

Table: reading cluster output.

Output	What it is	How to read it
Labels	Group number per record	Who belongs where
Centroids	Group averages	"Typical member" profile — name the group from this
Sizes	Members per group	Tiny groups may be outliers or need merging
Inertia	Tightness score	Compare across K values (elbow), not as an absolute grade

> **Key takeaway.** Labels say who is where; centroids say what each group is like — the pharmacist names the groups and decides the action.

8. Mini-case study: patient segmentation + drug grouping (sklearn, verified)

Case background (hypothetical). A district hospital pharmacy wants to (a) segment 10 diabetic-clinic patients by age and BMI for targeted counselling, and (b) group 6 candidate compounds by solubility and molecular weight for formulation planning. One workflow, two unsupervised tasks.

```
import numpy as np
from sklearn.cluster import KMeans

# Part A: patient segmentation
patients = np.array([
    [25, 22], [28, 23], [30, 24], [32, 23],   # younger, healthy BMI
    [55, 31], [60, 33], [58, 30], [62, 34],   # older, higher BMI
    [45, 26], [48, 27]])                       # middle-aged, overweight

model = KMeans(n_clusters=2, random_state=42, n_init=10)
model.fit(patients)
print("Patient groups:", model.labels_)
print("Group profiles (age, BMI):",
      np.round(model.cluster_centers_, 2))
```

Output:

```
Patient groups: [0 0 0 0 1 1 1 1 1 1]
Group profiles (age, BMI): [[28.75 23.  ]
 [54.67 30.17]]
```

Explanation: `KMeans(n_clusters=2)` asks for 2 groups; `.fit(patients)` runs the assign-and-move loop. `random_state=42` makes the start repeatable; `n_init=10` runs 10 different starts and keeps the best (avoids a bad random start). Result: Group 0 = younger, healthy-BMI patients → standard counselling; Group 1 = older, higher-BMI patients → intensive lifestyle + medication review.

```
# Part B: drug grouping by properties
drugs = np.array([
    [5, 180], [8, 220], [6, 200],      # small, poorly soluble
    [120, 460], [95, 500], [110, 480]  # larger, highly soluble
])
drug_model = KMeans(n_clusters=2, random_state=7, n_init=10)
drug_model.fit(drugs)
print("Drug groups:", drug_model.labels_)
print("Group profiles (solubility, MW):",
      np.round(drug_model.cluster_centers_, 2))
```

Output:

```
Drug groups: [0 0 0 1 1 1]
Group profiles (solubility, MW): [[  6.33 200.  ]
 [108.33 480.  ]]
```

Explanation: Group 0 = low solubility (~6.3 mg/mL), low MW (~200) → candidates for solubility-enhancement techniques (solid dispersions, micronisation). Group 1 = high solubility (~108 mg/mL), higher MW (~480) → straightforward conventional formulations. The same K-means tool serves both clinical and formulation questions.

> **Case conclusion.** One algorithm, two pharmacy wins: patients segmented for targeted care, compounds grouped for formulation strategy — no labels needed anywhere.

9. Key points

- Unsupervised learning = no labels; discovers groups and patterns.
- Clustering groups similar records; K-means is the standard algorithm.
- K-means loop: choose K → assign to nearest center → move centers to means → repeat until stable.
- Choose K with the elbow method (inertia bend) plus clinical/domain sense.
- Interpret outputs via centroids ("typical member") and label lists; the expert names the groups.
- Applications: patient segmentation, drug grouping by properties.

10. Common confusions

- **Clustering is not classification:** classification predicts known labels; clustering discovers unknown groups.
- **K is chosen by you**, not found by the algorithm — a favourite viva trap.
- **Cluster numbers are arbitrary:** "group 0" today could be "group 1" tomorrow — only the membership matters.
- **Inertia always falls as K rises** (K = n gives inertia 0) — so "lower inertia" alone never justifies more clusters; use the elbow.

11. Common mistakes

- Forgetting `n_init`/`random_state` and getting different answers every run.
- Picking K by habit instead of the elbow + domain sense.
- Treating clusters as diagnoses — they are *suggestions* for the expert, not verdicts.
- Clustering on features with wildly different scales without standardisation (e.g., age 25–70 vs income in lakhs) — distances get dominated by the big-scale feature.

12. Mnemonic

"A-M-R" — **A**ssign to nearest center, **M**ove centers to means, **R**epeat till stable. **"Elbow = the bend where more shelves stop tidying the library."**

13. Infographic concept box — Unit V

- Title: "Unsupervised Learning in One Page"
- Top: librarian icon — "no labels? organise by similarity."
- Center: 4-panel K-means loop (scatter → assign colours → move stars → final groups).
- Right: elbow curve with the bend circled at K = 2.
- Bottom: two result cards — "Patient groups: young-healthy / older-higher-BMI → targeted counselling" and "Drug groups: small-poorly-soluble / large-soluble → formulation strategy."
- Footer: "No answers given — the data reveals its own groups; the pharmacist names them."

14. Exam summary

Unsupervised learning works without labels to discover structure; clustering is its main task. K-means takes a chosen K, places K centers, and repeats two steps — assign each record to the nearest center, then move each center to its group's mean — until stable. K is chosen by the elbow method (the bend in the inertia-vs-K curve) together with domain knowledge. Outputs are labels (who is where) and centroids (typical-member profiles), which the expert interprets and names. In pharmacy this gives patient segmentation (e.g., younger-healthy vs older-higher-BMI groups for targeted care) and drug grouping by properties (e.g., small poorly-soluble vs larger soluble molecules for formulation planning). In sklearn, `KMeans(n_clusters=K).fit(X)` does it all.

15. One-minute revision

- Unsupervised = no labels → discover groups.
- K-means: assign → move centers → repeat.
- K chosen by YOU: elbow method + clinical sense.
- Labels = membership; centroids = group profile.
- Pharmacy: patient segmentation, drug grouping.
- sklearn: `KMeans(n_clusters=2, random_state=42, n_init=10).fit(X)`.

16. Exam questions

Short answers:

1. Define unsupervised learning. How does it differ from supervised learning?
2. List the steps of the K-means algorithm.
3. What is a centroid? What is inertia?
4. How does the elbow method choose K?

Long answers:

5. Explain K-means clustering step by step with a small numerical example and diagrams.
6. Describe how to choose the number of clusters and how to interpret cluster outputs, with a patient-segmentation example.
7. Present a mini-case study using clustering on a healthcare dataset: state the problem, the features, the K-means output and the pharmacy action for each group.

MCQs:

8. K-means is:

a) supervised classification b) unsupervised clustering c) reinforcement learning d) a regression metric *Answer: b*

9. In K-means, K is:

a) found automatically b) chosen by the user c) always 2 d) equal to accuracy *Answer: b*

10. The elbow method uses:

a) accuracy vs K b) inertia vs K c) precision vs K d) depth vs K *Answer: b*

11. A centroid is:

a) the first data point b) the mean (center) of a cluster c) an outlier d) the label column *Answer: b*

Viva:

12. Your K-means gives different groups on every run. What two parameters will you fix and why?
13. You clustered patients into 3 groups. How would you convince a doctor these groups are meaningful?

Application-based:

14. A formulation team has 50 new compounds with solubility and molecular weight data but no labels. Design a clustering analysis: state K-selection method, what outputs you will report, and what action each resulting group suggests.

17. References (Unit V)

1. James, G., Witten, D., Hastie, T. and Tibshirani, R. *An Introduction to Statistical Learning with Applications in Python*. Springer. Available at: statlearning.com.
2. Géron, A. *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow*. O'Reilly Media.
3. Simon, G. J. and Aliferis, C. *Artificial Intelligence and Machine Learning in Health Care and Medical Sciences: Best Practices and Pitfalls*. Springer.

COMBINED REFERENCES (all units)

As listed in the official BP301T syllabus (preferably latest editions):

1. James, G., Witten, D., Hastie, T. and Tibshirani, R. An Introduction to Statistical Learning with Applications in Python. Springer. Available at: statlearning.com.
2. Simon, G. J. and Aliferis, C. Artificial Intelligence and Machine Learning in Health Care and Medical Sciences: Best Practices and Pitfalls. Springer.
3. Brown, N. Artificial Intelligence in Drug Discovery. Royal Society of Chemistry.

1. Walters, S. J., Campbell, M. J. and Machin, D. Medical Statistics: A Textbook for the Health Sciences. Wiley-Blackwell.
2. Géron, A. Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow. O'Reilly Media.