BP301T · B.PHARM THEORY

BP301T Introduction to Machine Learning in Pharmaceutical Sciences

Unit I — Foundations of Machine Learning

Pills & Python Notes

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

Unit I · Foundations of Machine Learning

PILLS & PYTHON

Pharmacy • Data • AI • Healthcare

Instagram @pillsandpython · pillsandpython.com · YouTube @PillsAndPython

CHAPTER 1 — UNIT I: Foundations of Machine Learning

1. Learning Objectives

After studying this chapter, you will be able to:

- Define Artificial Intelligence (AI), Machine Learning (ML) and Data Science in your own words.
- Describe the scope of AI, ML and Data Science in pharmacy and healthcare.
- List and compare the three types of machine learning: supervised, unsupervised and reinforcement learning.
- Explain what features, labels, training data and testing data mean.
- Describe how a model is built and how it makes predictions.
- Explain train-test split, overfitting and underfitting.
- Explain the bias–variance trade-off in simple terms.
- Draw and explain the basic ML workflow from data to prediction.

2. Definitions

Artificial Intelligence (AI). Artificial Intelligence is the science of making machines do tasks that normally need human intelligence — for example, recognizing a disease from an X-ray, reading a prescription, or answering a patient's question. "Artificial" means man-made; "intelligence" means the ability to learn and decide.

Machine Learning (ML). Machine Learning is a branch of AI in which the computer learns patterns from data instead of being given fixed rules by a programmer. In ordinary programming, a human writes the rule: "if temperature is above 38°C, then fever." In machine learning, the computer studies thousands of past records and discovers the rule by itself: "patients with these symptoms usually have this disease."

Data Science. Data Science is the broader field of collecting, cleaning, studying and presenting data so that useful decisions can be made. Machine learning is one powerful tool used inside data science. Think of data science as the whole hospital and machine learning as one very smart department in it.

Data. Data means facts or information collected from the real world — numbers, words, images — stored so a computer can use them. Example: a table of 500 patients with their age, blood pressure and whether they developed an adverse drug reaction (ADR).

3. Terminology (word-by-word from zero)

Term	What it means in plain words
Algorithm	A step-by-step recipe the computer follows. A cooking recipe is an algorithm; a model-building recipe is also an algorithm.
Feature	An input column used for learning — the "information we know." Example: a patient's age or a tablet's dose.
Label	The answer column the computer must learn to predict — "the answer we want." Example: whether the patient developed an ADR (yes/no). Also called the target.
Training data	The past records shown to the computer so it can learn. Like the solved examples a student practises before the exam.
Testing data	Fresh records, kept hidden during learning, used to check whether the computer really understood. Like the exam paper.
Model	The learned "brain" — a mathematical formula saved by the computer after training. You feed it features; it returns a prediction.
Prediction	The model's guess for a new case it has never seen. Example: "this new patient has a high risk of ADR."
Supervised learning	Learning with answer keys — every training record has a label.
Unsupervised learning	Learning without answer keys — the computer finds groups or patterns on its own.
Reinforcement learning	Learning by trial and reward — the computer tries actions and gets points for good results.
Overfitting	The model memorises the training data like a student who memorises the textbook but fails new questions.
Underfitting	The model learns too little — like a student who studied only one page and cannot answer anything.
Bias	Error from a model being too simple (it misses real patterns).
Variance	Error from a model being too sensitive (it changes wildly with small data changes).
Train-test split	Dividing the records into a training part and a testing part.

4. Explanation: AI, ML and Data Science — how they fit together

Technical concept. AI is the largest circle. Inside it sits machine learning. Data science overlaps with both — it is the full journey from raw data to decisions.

> **Simple explanation.** Imagine a big hospital called *Artificial Intelligence*. One department inside it is *Machine Learning* — the department where computers learn from past cases. The hospital's medical records department is *Data Science* — collecting, cleaning and organising every record so the doctors (and the ML department) can use them.

Example. A pharmaceutical company collects 10 years of tablet dissolution test data (data science collects and cleans it). A machine learning model learns from this data which formulation factors control dissolution speed (machine learning finds the pattern). Together, this AI system warns scientists when a new batch may dissolve too slowly (artificial intelligence serving pharmacy).

> **Diagram (draw this).** Three overlapping circles. Largest circle labelled "Artificial Intelligence — machines doing intelligent tasks." A circle inside it labelled "Machine Learning — computers learning from data." A third circle overlapping both labelled "Data Science — collecting, cleaning and studying data." In the overlap write: "ML models trained on data-science pipelines are AI systems."

Key takeaway. AI is the goal (intelligent machines), ML is one method to reach it (learning from data), and data science is the discipline that prepares and studies the data.

5. Classification: the three types of machine learning

Technical concept. Machine learning is classified by what kind of "teaching material" the computer receives: labelled examples, unlabelled data, or rewards.

Simple explanation with examples.

1. **Supervised learning — learning with an answer key.** Every training record has the correct answer attached. The computer practises predicting the answer and checks itself. *Pharmacy example:* past records of patients with dose, age (features) and whether they got an ADR (label). The computer learns to predict ADR risk for new patients.
2. **Unsupervised learning — learning without an answer key.** The records have no labels. The computer must find natural groups or patterns itself. *Pharmacy example:* the computer studies 1,000 drug molecules (solubility, molecular weight) and groups similar molecules together without anyone telling it the groups.
3. **Reinforcement learning — learning by trial, error and reward.** The computer (called an agent) tries actions in an environment and receives rewards for good actions and penalties for bad ones. Over many trials it learns the best strategy. *Pharmacy example:* a robotic arm in a warehouse learns the fastest way to pick and pack medicine boxes by trial and reward.

Table: comparison of the three types.

Point	Supervised learning	Unsupervised learning	Reinforcement learning
Labels	Yes — every record has the answer	No — no answers given	No answers; reward signal instead
Goal	Predict the label for new data	Find groups/patterns	Learn the best action strategy
Example algorithms	Linear regression, logistic regression, decision trees	K-means clustering	(Used in robotics, game playing)
Pharmacy example	Predict ADR yes/no	Group patients with similar profiles	Robot learns to dispense tablets
Covered in this course	Units II, III, IV	Unit V	Only the concept (Unit I)

Flowchart — which type should I use? (draw this). Start: "Do you have labelled data (answers)?" → YES → "Is the answer a number (price, dissolution %)? → YES → Supervised: regression. NO → Is it a category (yes/no, disease type)? → Supervised: classification." From the first question → NO → "Do you want to find groups or patterns? → YES → Unsupervised: clustering." "Do you want a machine to learn actions by reward? → YES → Reinforcement learning."

Key takeaway. Supervised = answers given (predict); unsupervised = no answers (find groups); reinforcement = learn by reward.

6. Features and labels, training data and testing data

Technical concept. A dataset is a table. Columns used as inputs are features; the column to be predicted is the label. Rows are divided into training data (for learning) and testing data (for honest checking).

> **Simple explanation.** Think of a pharmacist's register. Each row is one patient. Columns like age, weight and dose are *features* — the facts we know. The last column, "ADR happened: yes/no," is the *label* — the answer we want the computer to learn to predict.

Example table (a tiny hypothetical dataset).

Patient	Age (years) — feature	Dose (mg) — feature	ADR (yes=1/no=0) — label
1	25	100	0
2	68	400	1
3	45	200	0
4	72	450	1

Here we have 2 features (age, dose) and 1 label (ADR). Training data = the rows the computer studies. Testing data = rows kept hidden until the exam.

> **Why two sets? Analogy.** A student practises with solved question papers (training data). If you test the student on the *same* solved paper, a high score proves nothing — the student may have memorised the answers. The real exam (testing data) must be a *new* paper. In ML, testing on the training data is cheating ourselves; testing on fresh data shows the truth.

> **Key takeaway.** Features = inputs we know; label = answer to predict; train to learn, test to verify honestly.

7. Concept of model building and prediction

Technical concept. Model building (training) means the algorithm adjusts its internal numbers using the training data until its predictions match the labels as closely as possible. Prediction means feeding new features into the finished model to get an answer.

> **Simple explanation.** Building a model is like a new pharmacist learning from a senior: "When elderly patients take high doses, watch for ADRs." The senior's advice is the *model* — a compact rule learned from experience. *Prediction* is using that advice on tomorrow's new patient.

> **Flowchart of model building (draw this).** Box 1: "Collect data (patient records)." → Box 2: "Clean data (fix errors, missing values)." → Box 3: "Choose features and label." → Box 4: "Split into training and testing data." → Box 5: "Train the model on training data (the algorithm learns the pattern)." → Box 6: "Test the model on testing data (check predictions vs real labels)." → Decision diamond: "Good enough?" → NO → "Adjust and retrain" (arrow back to Box 5). → YES → Box 7: "Use the model for new predictions."

Example. Using the 4-patient table above: the algorithm studies the 4 rows and learns a model (e.g., "if age > 60 and dose > 350 mg, ADR risk is high"). Then a new patient arrives — age 70, dose 420 mg — and the model predicts "ADR likely."

Key takeaway. Training = learning the pattern from old data; prediction = applying the learned pattern to new cases.

8. Train-test split

Technical concept. The dataset is randomly divided — commonly 70–80% for training and 20–30% for testing. The test set is locked away and touched only once, at the end.

Simple explanation. Out of 100 patient records, let the computer study 80 (training) and examine it on the remaining 20 (testing). The split is random so both sets fairly represent all kinds of patients.

Common mistake. Testing the model on the same data it trained on and celebrating 100% accuracy. That is like letting students write the exam from their own notebooks — the score is meaningless.

Key takeaway. Train on most of the data; test once on the hidden remainder; never let the model peek at the test set during training.

9. Overfitting and underfitting

Technical concept. Overfitting: the model is too complex — it learns the noise (random quirks) of the training data along with the real pattern, so it performs well on training data but poorly on new data. Underfitting: the model is too simple — it cannot capture even the real pattern, so it performs poorly on both training and new data.

Simple explanation. Imagine fitting a shirt. *Overfitting* is a shirt stitched exactly to every bump of one person — it fits that person perfectly but nobody else. *Underfitting* is a shapeless sack — it fits nobody well. We want a well-tailored shirt: fits the body shape (real pattern), not every wrinkle (noise).

Example. Predicting drug dissolution rate from tablet hardness:

- *Underfit model:* "Dissolution is always 50%." Too simple — wrong for nearly every tablet.
- *Overfit model:* a wiggly curve passing exactly through all 10 training points, including one faulty measurement. It scores perfectly on training data but fails on new tablets.
- *Good model:* a smooth line through the general trend — slightly wrong on training data, but reliable on new tablets.

Diagram (draw this). Three panels showing data points with: (a) a straight flat line far from points — label "Underfitting: too simple"; (b) a wild zigzag curve touching every point — label "Overfitting: memorised noise"; (c) a smooth curve following the trend — label "Good fit: captures the real pattern."

Key takeaway. Underfitting = learned too little (bad everywhere). Overfitting = memorised too much (good on training, bad on new data). Aim for the balanced middle.

10. Bias–variance trade-off (conceptual explanation)

Technical concept. Prediction error comes from three parts: bias (error from wrong/simple assumptions), variance (error from sensitivity to the particular training sample), and irreducible noise. Making a model simpler reduces variance but increases bias; making it more complex reduces bias but increases variance. The best model balances the two.

> **Simple explanation.** Think of two students:

- *High-bias student* studies only the chapter headings. Whatever the question paper, this student gives the same shallow answers — consistently wrong (high bias), but answers never change much (low variance).
- *High-variance student* memorises every word of one particular textbook edition. On questions from that book, perfect; on a slightly different paper, the answers collapse — the performance swings wildly (high variance).

You cannot have both zero bias and zero variance with limited data. A very simple model has high bias but low variance; a very complex model has low bias but high variance. *Trade-off* means: improving one side worsens the other, so we choose the sweet middle.

> **Diagram (draw this).** An X-axis labelled "model complexity (simple → complex)." Two curves: "Bias" falling from high to low; "Variance" rising from low to high. A U-shaped "Total error" curve with its lowest point marked "Best balance — sweet spot."

Table.

	High bias (underfit)	High variance (overfit)
Model	Too simple	Too complex
Training performance	Poor	Excellent
New-data performance	Poor	Poor
Behaviour	Misses real patterns	Memorises noise
Fix direction	Make the model more flexible	Make the model simpler / use more data

> **Key takeaway.** Bias = error from being too rigid; variance = error from being too jumpy. Total error is lowest at the balance point — that is the trade-off.

11. Overview of the basic ML workflow

Technical concept. The standard workflow: define the problem → collect data → clean and prepare data → select features → split train/test → choose a model → train → evaluate → improve → deploy and monitor.

> **Simple explanation.** It is exactly like opening a new pharmacy counter, step by step: decide what service you will offer (define the problem), gather the medicines and records (collect data), arrange the shelves neatly (clean data), pick which records the computer may use (select features), keep an exam paper aside (train-test split), choose your method (choose a model), train the staff (train), test their work (evaluate), correct mistakes (improve), and finally open the counter to the public while watching quality daily (deploy and monitor).

Flowchart (draw this — the master flowchart of Unit I).

1. Define the problem — "What should the model predict?" (e.g., ADR risk)
2. Collect data — hospital records, lab results
3. Clean and prepare data — fix missing values and errors
4. Select features and label — choose useful input columns and the answer column
5. Split data — training set (learn) and testing set (honest check)
6. Choose a model — regression, classification, clustering…
7. Train the model — the algorithm learns from training data
8. Evaluate on test data — measure accuracy or error
9. Improve if needed — adjust and retrain
10. Deploy and monitor — use on new data; keep checking performance

Pharmacy relevance. Every later unit of this course (regression for dissolution, classification for ADR, trees for treatment outcome, clustering for patient groups) follows exactly this workflow. Learn it once; apply it everywhere.

India context. Indian hospitals and pharmaceutical companies generate enormous data — prescriptions, lab reports, clinical trials. The government's Digital India and Ayushman Bharat digital health records are creating large datasets. Pharmacists who understand the ML workflow can help build Indian tools for ADR monitoring, drug quality prediction and affordable healthcare AI.

12. Key points

- AI = intelligent machines; ML = computers learning from data; Data Science = collecting, cleaning and studying data.
- Three ML types: supervised (labels → predict), unsupervised (no labels → find groups), reinforcement (rewards → learn actions).
- Features are inputs; the label is the answer to predict.
- Train on training data; honestly test on hidden testing data.
- Underfitting = too simple (bad everywhere); overfitting = memorised noise (good on training, bad on new data).
- Bias–variance trade-off: simple models have high bias, complex models have high variance; balance them.
- The ML workflow runs from problem definition → data → training → evaluation → deployment.

13. Common confusions

- **AI vs ML:** ML is a *part* of AI, not a separate competing field. All ML is AI, but not all AI is ML (a rule-based expert system is AI without ML).
- **Data Science vs ML:** Data science is the whole journey with data; ML is one powerful method used in that journey.
- **Feature vs label:** The feature is what you *know* (age); the label is what you *want to know* (ADR yes/no). Students often swap them.
- **Training accuracy vs real performance:** High training accuracy alone means nothing — only test accuracy on unseen data counts.
- **Reinforcement learning is not in the practical syllabus:** you must know its concept (Unit I) but the demonstrations in this course use supervised and unsupervised learning.

14. Common mistakes

- Testing the model on the training data and reporting that as the model's quality.
- Using all data for training and keeping nothing for testing.
- Choosing an extremely complex model for a tiny dataset (invites overfitting).
- Forgetting that the test set must stay hidden until final evaluation.
- Thinking "more complex model = always better."

15. Mnemonic

"S-U-R" for the three types — **S**upervised (answers **S**upplied), **U**nsupervised (answers **U**nseen), **R**einforcement (**R**eward-driven). **"T-E-M"** for model building — **T**rain, **E**valuate, **M**onitor. **"B-V" balance:** "**B**ias is being too **B**lind (simple); **V**ariance is being too **V**olatile (complex)."

16. Infographic concept box — Unit I

Design note for the PDF: one-page visual summary with icons.

- Title: "Machine Learning in One Page"
- Three columns with icons: Supervised (answer key icon) → "predict"; Unsupervised (magnifier over groups) → "discover groups"; Reinforcement (trophy) → "learn by reward."
- Center: the workflow wheel — Problem → Data → Clean → Split → Train → Test → Deploy.
- Bottom strip: the balance scale — Bias ⚖ Variance with "sweet spot" marked.
- Footer line: "Features in, prediction out — tested on data the model never saw."

17. Exam summary

Unit I builds the vocabulary of the whole course. AI is the big field; machine learning learns patterns from data; data science prepares and studies data. Supervised learning uses labelled data to predict (regression for numbers, classification for categories); unsupervised learning finds groups without labels; reinforcement learning learns from rewards. A dataset's input columns are features and the answer column is the label. Data is split into training data (learning) and testing data (honest checking). A model is the learned pattern; prediction applies it to new cases. Underfitting means too simple, overfitting means memorised noise, and the bias–variance trade-off says we must balance simplicity against flexibility. The ML workflow runs from defining the problem to deploying and monitoring the model.

18. One-minute revision

- AI ⊃ ML; Data Science = data journey.
- Supervised = labels (predict). Unsupervised = no labels (groups). Reinforcement = rewards (actions).
- Feature = input; label = answer.
- Train = learn; test = honest exam on unseen data.
- Underfit = too simple; overfit = memorised noise.
- Bias ↓ as complexity ↑; variance ↑ as complexity ↑ — balance them.
- Workflow: problem → data → clean → split → train → evaluate → deploy.

19. Exam questions

Short answers (2–3 marks):

1. Define Artificial Intelligence and Machine Learning.
2. What is the difference between a feature and a label? Give one pharmacy example.
3. Why is testing data kept separate from training data?
4. Define overfitting and underfitting in one line each.
5. What is the bias–variance trade-off?

Long answers (5–10 marks): 6. Explain the three types of machine learning with one pharmacy example for each. 7. Describe the basic ML workflow with a flowchart. Illustrate it with an ADR-prediction example. 8. Explain overfitting and underfitting with diagrams. How does the bias–variance trade-off guide model selection?

MCQs: 9. Which of the following is supervised learning? a) K-means clustering b) Predicting ADR risk from labelled patient records c) Robot learning by reward d) Finding customer groups *Answer: b* 10. The column to be predicted is called the: a) feature b) label c) algorithm d) workflow *Answer: b* 11. Overfitting means the model: a) is too simple b) memorises training noise and fails on new data c) has high bias d) needs no data *Answer: b* 12. In the bias–variance trade-off, increasing model complexity: a) increases bias and decreases variance b) decreases bias and increases variance c) decreases both d) has no effect *Answer: b*

Viva questions: 13. If your model scores 99% on training data but 60% on test data, what is wrong and what will you do? 14. Give one example each of supervised and unsupervised learning from a hospital pharmacy.

Application-based: 15. A company wants to predict tablet dissolution rate from formulation data, and also wants to group 2,000 drug molecules by similarity without any labels. Which ML types will you use for each task? Justify.

20. References (Unit I)

1. Géron, A. *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow*. O'Reilly Media. (Latest edition as per syllabus.)
2. James, G., Witten, D., Hastie, T. and Tibshirani, R. *An Introduction to Statistical Learning with Applications in Python*. Springer. Available at: statlearning.com.
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.
4. Walters, S. J., Campbell, M. J. and Machin, D. Medical Statistics: A Textbook for the Health Sciences. Wiley-Blackwell.
5. Géron, A. Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow. O'Reilly Media.