Module 2: Machine Learning — 10 Lessons (Paragraph Edition)

With Key Terms and Lesson Quick■Check MCQs

Lesson 1 — Types of Learning

Machine learning problems fall into three families. In supervised learning, models learn a mapping from inputs to known labels (for example, predicting prices or classifying images). In unsupervised learning, algorithms discover structure without labels, such as grouping similar samples or compressing features. In reinforcement learning, an agent interacts with an environment and learns a policy that maximizes long■term reward through trial and error.

Key terms: supervised, unsupervised, reinforcement

Lesson 2 — The ML Data Pipeline

The standard workflow turns raw information into reliable predictions: first assemble a dataset; then preprocess it by cleaning missing or extreme values, encoding categories, and scaling numerics; next train a model by minimizing a loss on the training split; and finally run prediction (inference) on unseen data to evaluate and deploy.

Key terms: dataset, preprocessing, inference

Lesson 3 — Linear Regression

Linear regression models the relationship between a numeric target and one or more features using a straight line. Training selects parameters—slope and intercept—that minimize the mean squared error between true and predicted values; solutions can use least squares or iterative gradient descent. Once fitted, the line generalizes to forecast values for new inputs.

Key terms: slope, intercept, MSE

Lesson 4 — Classification

Classification assigns each input to a discrete class. A simple baseline is k■nearest neighbors (k■NN): for a new point, find its k closest labeled neighbors in feature space and return the majority label; larger k usually smooths the decision boundary but may blur fine structure. Alternatives like decision trees learn simple, interpretable rules from data and can capture nonlinear patterns.

Key terms: classification, k■NN, boundary

Lesson 5 — Overfitting vs Underfitting

Generalization depends on model complexity. Underfitting occurs when the model is too simple to capture patterns, leading to high errors on both training and test data. Overfitting happens when the model memorizes noise, yielding very low training error but poor test performance. The goal is a balance—often helped by regularization, simpler models, and proper validation.

Key terms: bias, variance, generalization

Lesson 6 — Train/Validation/Test & Cross■Validation

To estimate real■world performance, split data into train, validation, and test sets. The model learns on the train split, hyperparameters are chosen on the validation split (or via k■fold cross■validation for stability), and the held■out test set is used once at the end to report an unbiased estimate. Always fit preprocessing steps only on training data to avoid leakage.

Key terms: validation, cross■validation, holdout

Lesson 7 — Evaluation Metrics

Different tasks require different metrics. For regression, common choices include MAE, MSE/RMSE, and R^2 to summarize error and explained variance. For classification, accuracy can be misleading with class imbalance, so we track precision, recall, and F1, and examine the confusion matrix; threshold■based curves like ROC and PR help compare classifiers.

Key terms: precision, recall, F1

Lesson 8 — Feature Engineering & Preprocessing

Feature engineering transforms raw inputs into informative signals. Typical steps include scaling numeric features, encoding categorical variables, and imputing missing values; for text, bag■of■words or TF■IDF create count■based representations. To prevent leakage, compute all preprocessing parameters using only the training split, then apply them consistently to validation and test.

Key terms: scaling, encoding, imputation

Lesson 9 — Model Selection & Hyperparameter Tuning

Model selection chooses a configuration that performs best on validation data. Hyperparameters—such as k in k■NN or tree depth—are tuned via grid or random search combined with cross■validation. Learning and validation curves diagnose whether collecting more data, simplifying the model, or regularizing would likely improve generalization.

Key terms: hyperparameters, cross■validation, search

Lesson 10 — Clustering & Dimensionality Reduction

Unsupervised exploration reveals structure without labels. K■means iteratively assigns points to the nearest centroids and updates those centroids to minimize within■cluster variance; the elbow or silhouette methods help choose k. Dimensionality reduction with PCA projects data onto directions of maximal variance to visualize or simplify downstream modeling.

Key terms: k■means, PCA, clusters

Lesson Quick■Check MCQs

Lesson 1 — Types of Learning

Q: Which task is most suitable for reinforcement learning?

A. Grouping similar samples without labels

B. Training an agent to maximize long■term reward through trial and error

C. Predicting prices from labeled examples

D. Compressing features into fewer dimensions

Answer: Training an agent to maximize long■term reward through trial and error

Lesson 2 — The ML Data Pipeline

Q: What is the correct order of the pipeline described in the lesson?

A. Preprocessing → Dataset → Training → Prediction

B. Dataset → Training → Preprocessing → Prediction

C. Dataset → Preprocessing → Training → Prediction

D. Training → Dataset → Prediction → Preprocessing

Answer: Dataset → Preprocessing → Training → Prediction

Lesson 3 — Linear Regression

Q: Which quantity is minimized during training to fit the line?

A. Cross■entropy

B. Mean squared error

C. Hinge loss

D. Kullback–Leibler divergence

Answer: Mean squared error

Lesson 4 — Classification

Q: Increasing k in k■NN typically has what effect on the decision boundary?

A. Makes it more jagged (higher variance)

B. Smooths it (lower variance)

C. Always increases training accuracy without test impact

D. Guarantees underfitting on every dataset

Answer: Smooths it (lower variance)

Lesson 5 — Overfitting vs Underfitting

Q: Which pattern indicates overfitting as defined in the lesson?

A. Low training error and low test error

B. High training error and high test error

C. Low training error but high test error

D. High training error but low test error

Answer: Low training error but high test error

Lesson 6 — Train/Validation/Test & Cross■Validation

Q: According to the lesson, when should the held■out test set be used?

A. For selecting hyperparameters

B. Only once at the end to estimate generalization

C. For computing scaling parameters

D. To augment the training data when data is scarce

Answer: Only once at the end to estimate generalization

Lesson 7 — Evaluation Metrics

Q: Which metric combines precision and recall into a single number?

A. Accuracy

B. ROC■AUC

C. F1 score

D. R^2

Answer: F1 score

Lesson 8 — Feature Engineering & Preprocessing

Q: Which practice can cause data leakage as described in the lesson?

A. Fitting a scaler on the training set then applying to validation/test

B. One■hot encoding categorical variables

C. Imputing missing values using training■set statistics only

D. Fitting a scaler on the entire dataset before splitting

Answer: Fitting a scaler on the entire dataset before splitting

Lesson 9 — Model Selection & Hyperparameter Tuning

Q: Which is a hyperparameter according to the lesson?

A. The learned weight w in linear regression

B. The number of neighbors k in k■NN

C. The predicted probability for a test sample

D. The residual error for a training point

Answer: The number of neighbors k in k■NN

Lesson 10 — Clustering & Dimensionality Reduction

Q: What is the primary function of PCA as defined in the lesson?

A. Assign points to nearest centroids iteratively

B. Project data onto directions of maximum variance

C. Train a deep neural network

D. Guarantee higher classification accuracy

Answer: Project data onto directions of maximum variance