

SPPU-BE-COMP-CONTENT – KSKA Git

ML UNIT 1 – PYQ Answers

➤ OCT 2022

Q1)

a) Compare machine learning vs Artificial Intelligence. [5]

Aspect	Artificial Intelligence (AI)	Machine Learning (ML)
Scope	Broad field including ML, robotics, NLP, expert systems, computer vision, etc.	Subset of AI focusing on pattern recognition, predictions, and classifications.
Goal	Mimic human intelligence for problem-solving and autonomous actions.	Learn patterns from data and improve decision-making accuracy.
Approach	Uses reasoning, logic, knowledge graphs, heuristics, and learning.	Uses supervised, unsupervised, semi-supervised, and reinforcement learning.
Data Dependency	Can work with or without large datasets (rule-based AI does not need much data).	Requires large, quality datasets for effective training.
Learning Capability	Can incorporate ML plus symbolic reasoning and manual programming.	Focused purely on data-driven learning without explicit programming.
Flexibility	Handles tasks requiring reasoning beyond pattern recognition.	Primarily limited to the scope defined by training data.
Example	Chess-playing robot using logic + ML for move predictions.	Spam detection system learning from historical email data.

b) Describe parametric and Non-parametric machine learning models. [5]

Machine learning models can be categorized into **parametric** and **non-parametric** based on how they represent and learn from data.

1. Parametric Models

- **Definition:** Models that summarize data with a fixed number of parameters, regardless of dataset size.
- **Key Features:**
 1. Assumes a specific functional form or distribution (e.g., linear, Gaussian).
 2. Model complexity is fixed; adding more data does not increase the number of parameters.

SPPU-BE-COMP-CONTENT – KSKA Git

3. Faster to train and easier to interpret.

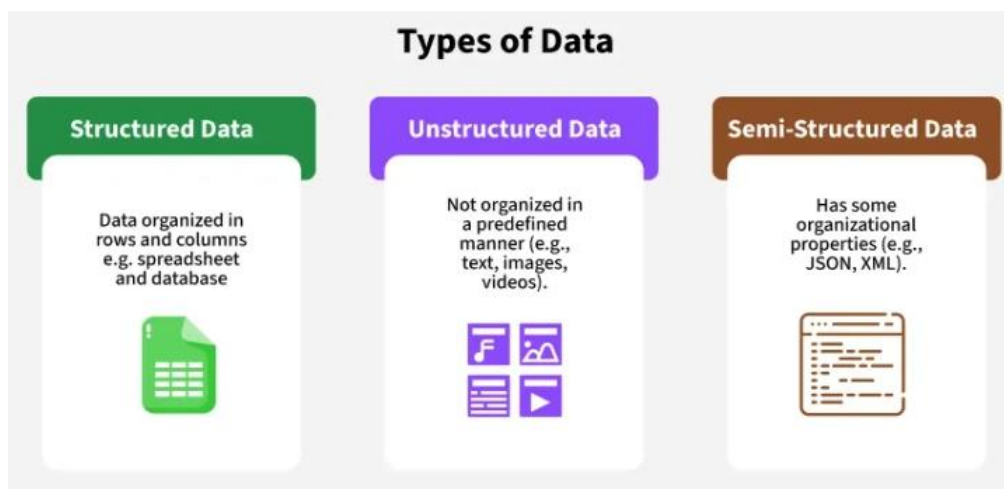
- **Advantages:** Computationally efficient, requires less data.
- **Disadvantages:** Limited flexibility, may underfit if assumption is incorrect.
- **Examples:** Linear Regression, Logistic Regression, Naïve Bayes.

2. Non-Parametric Models

- **Definition:** Models that do not assume a fixed form and can grow in complexity with more data.
- **Key Features:**
 1. Number of parameters depends on the size of the dataset.
 2. Flexible in fitting complex relationships without predefined assumptions.
 3. Can capture non-linear patterns effectively.
- **Advantages:** High flexibility, better performance on complex datasets.
- **Disadvantages:** Computationally expensive, risk of overfitting, requires more data.
- **Examples:** k-Nearest Neighbors (k-NN), Decision Trees, Random Forests.

Parametric models are simple and efficient but less flexible, while non-parametric models are flexible and adaptive but computationally heavier. The choice depends on data size, complexity, and performance needs.

c) Explain various Data formats that conform ML elements. [5]



SPPU-BE-COMP-CONTENT – KSKA Git

1. Structured Data

- **Definition:** Data organized in rows and columns, usually stored in relational databases.
- **Features:**
 1. Easily represented in tabular form.
 2. Highly organized with predefined data types.
- **Examples:** Spreadsheets, SQL tables containing numerical and categorical values.

2. Unstructured Data

- **Definition:** Data without a fixed format or organization.
- **Features:**
 1. Cannot be directly stored in traditional tables.
 2. Requires specialized preprocessing before model training.
- **Examples:** Text documents, images, audio, video.

3. Semi-Structured Data

- **Definition:** Data with partial organization using tags or markers but not fully structured.
- **Features:**
 1. Lies between structured and unstructured formats.
 2. Contains metadata for interpretation.
- **Examples:** JSON files, XML data, log files.

4. Time-Series Data

- **Definition:** Sequential data points indexed in time order.
- **Features:**
 1. Captures trends, seasonality, and temporal patterns.
 2. Requires specialized models like ARIMA, LSTM.
- **Examples:** Stock market prices, sensor readings, weather data.

5. Graph Data

SPPU-BE-COMP-CONTENT – KSKA Git

- **Definition:** Data represented as nodes (entities) and edges (relationships).
- **Features:**
 1. Captures relationships between entities.
 2. Useful for network analysis and recommendation systems.
- **Examples:** Social networks, citation networks.

Q2)

a) Explain supervised, unsupervised and semi supervised learning. [7]

1. Supervised Learning

Definition:

- The model learns from **labeled data** (input-output pairs).
- The goal is to predict the correct output for new, unseen inputs.

Key Characteristics:

Requires a **training dataset with labels** (e.g., classifications or numerical values).

Uses **input features (X)** to predict **target labels (Y)**.

Evaluated using accuracy, precision, recall, MSE, etc.

Types of Problems:

- **Classification** (discrete output, e.g., spam detection).
- **Regression** (continuous output, e.g., house price prediction).

Algorithms:

- **Linear Regression, Logistic Regression**
- **Decision Trees, Random Forest**
- **Support Vector Machines (SVM)**
- **Neural Networks (for complex tasks)**

Example:

- Predicting whether an email is **spam (1) or not (0)** using past labeled emails.

2. Unsupervised Learning

Definition:

- The model learns from **unlabeled data** (no predefined outputs).
- Discovers hidden patterns or groupings in the data.

SPPU-BE-COMP-CONTENT – KSKA Git

Key Characteristics:

Works without labeled responses.

Used for **exploratory data analysis (EDA)**.

Evaluated using clustering metrics (e.g., silhouette score).

Types of Problems:

- **Clustering** (grouping similar data points, e.g., customer segmentation).
- **Dimensionality Reduction** (reducing features while retaining information, e.g., PCA).
- **Anomaly Detection** (finding outliers).

Algorithms:

- **K-Means Clustering**
- **Hierarchical Clustering**
- **Principal Component Analysis (PCA)**
- **Autoencoders (Neural Networks for compression)**

Example:

- Grouping customers based on purchasing behaviour **without predefined categories**.

3. Semi-Supervised Learning

Definition:

- Uses a mix of **labeled and unlabeled data** (small labeled + large unlabeled dataset).
- Combines supervised and unsupervised techniques.

Key Characteristics:

Useful when **labeling data is expensive or time-consuming**.

Improves model accuracy by leveraging unlabeled data.

Approaches:

- **Self-training** (model labels its own predictions and retrains).
- **Co-training** (multiple models train on different feature sets).
- **Generative Models** (e.g., Generative Adversarial Networks - GANs).

Algorithms:

- **Label Propagation**
- **Semi-Supervised SVM**
- **Deep Learning with Pseudo-Labeling**

SPPU-BE-COMP-CONTENT – KSKA Git

Example:

- Training a **speech recognition model** with a few labeled audio samples and many unlabeled ones.

b) Describe various statistical learning approaches. [8]

Statistical learning is a framework for understanding, modeling, and predicting relationships between input variables (features) and output variables (targets) using statistical methods.

1. Regression Analysis

- **Purpose:** Models the relationship between a dependent (target) variable and one or more independent (predictor) variables.
- **Types:**
 - **Linear Regression:** Fits a straight line (e.g., predicting house prices based on square footage).
 - **Multiple Regression:** Uses several predictors (e.g., predicting salary based on education, experience, and location).
 - **Nonlinear Regression:** Captures complex relationships (e.g., polynomial regression).
- **Evaluation:** R^2 , Mean Squared Error (MSE).

2. Classification

- **Purpose:** Assigns data points to discrete categories (classes).
- **Methods:**
 - **Logistic Regression:** Predicts probabilities (e.g., spam detection).
 - **Linear Discriminant Analysis (LDA):** Finds linear separations between classes.
 - **Support Vector Machines (SVM):** Maximizes margin between classes.
- **Evaluation:** Accuracy, Precision, Recall, F1-Score.

3. Bayesian Methods

- **Purpose:** Uses probability theory to update beliefs based on observed data.
- **Key Technique:**
 - **Naïve Bayes:** Assumes feature independence (e.g., document classification).
 - **Bayesian Networks:** Models dependencies between variables.
- **Advantage:** Handles uncertainty well.

4. Clustering (Unsupervised Learning)

- **Purpose:** Groups similar data points without predefined labels.
- **Algorithms:**
 - **K-Means:** Partitions data into k clusters (e.g., customer segmentation).

SPPU-BE-COMP-CONTENT – KSKA Git

- **Hierarchical Clustering:** Creates a tree of clusters (dendrogram).
 - **DBSCAN:** Identifies dense regions (works well with noise).
- **Evaluation:** Silhouette Score, Elbow Method.

5. Principal Component Analysis (PCA)

- **Purpose:** Reduces dimensionality while preserving variance.
- **Steps:**
 1. Standardize data.
 2. Compute covariance matrix.
 3. Extract eigenvectors (principal components).
- **Use Case:** Image compression, feature selection.

6. Hypothesis Testing

- **Purpose:** Validates assumptions about data using statistical tests.
- **Common Tests:**
 - **t-test:** Compares means of two groups.
 - **ANOVA:** Extends t-test to multiple groups.
 - **Chi-square test:** Checks independence in categorical data.
- **Example:** Testing if a new drug is more effective than a placebo.

7. Time-Series Analysis

- **Purpose:** Models sequential data with temporal dependencies.
- **Methods:**
 - **ARIMA (AutoRegressive Integrated Moving Average):** Captures trends and seasonality.
 - **Exponential Smoothing:** Weighted averages of past observations.
- **Example:** Stock price forecasting, weather prediction.

8. Survival Analysis

- **Purpose:** Predicts time until an event (e.g., failure, death).
- **Techniques:**
 - **Kaplan-Meier Estimator:** Non-parametric survival curves.
 - **Cox Proportional Hazards Model:** Assesses impact of covariates.
- **Example:** Predicting patient survival rates in clinical trials.

SPPU-BE-COMP-CONTENT – KSKA Git

➤ SEP 2023

Q1)

a) Compare Machine Learning with traditional programming. Discuss types of Machine Learning with suitable examples. [5]

Traditional Programming: Programmer writes explicit rules (logic) to process input and produce output.

Machine Learning (ML): System learns rules automatically from data without explicit programming.

Aspect	Traditional Programming	Machine Learning
Approach	Logic/rules manually coded by humans.	Model learns patterns from data.
Data Dependency	Works without large datasets; relies on rules.	Requires data for training and prediction.
Flexibility	Needs reprogramming for new tasks.	Can adapt to new patterns without full reprogramming.
Example	Tax calculation software using fixed rules.	Email spam filter learning from labeled examples.

Types of Machine Learning

1. Supervised Learning

- **Definition:** Model is trained on labeled data (input-output pairs).
- **Goal:** Predict outcomes for new inputs.
- **Examples:**
 - Classification: Disease diagnosis (Yes/No).
 - Regression: Predicting stock prices.

2. Unsupervised Learning

- **Definition:** Model works on unlabeled data to find patterns or structure.
- **Goal:** Discover hidden relationships.
- **Examples:**
 - Clustering: Customer segmentation.

SPPU-BE-COMP-CONTENT – KSKA Git

- Dimensionality Reduction: PCA for data compression.

3. Semi-Supervised Learning

- **Definition:** Uses small labeled dataset + large unlabeled dataset.
- **Goal:** Improve learning efficiency when labeling is costly.
- **Example:** Speech recognition with limited labeled transcripts.

4. Reinforcement Learning

- **Definition:** Agent learns by interacting with an environment, receiving rewards or penalties.
- **Goal:** Learn optimal strategies through trial and error.
- **Example:** Game-playing AI (e.g., AlphaGo).

b) What are various Statistical Learning Approaches?

ALREADY DONE !

c) Explain different dataformats used in Machine Learning. [5]

ALREADY DONE !

Q2)

a) What is Machine Learning? Explain applications of Machine Learning in data science. [5]

Machine Learning is a subset of Artificial Intelligence that enables systems to **learn patterns from data** and improve their performance on tasks **without being explicitly programmed**.

Key Features:

1. Data-driven approach to prediction and decision-making.
2. Improves accuracy with more training data.
3. Uses algorithms like regression, decision trees, neural networks, etc.

Applications of ML in Data Science

1. Predictive Analytics

- Uses historical data to forecast future trends.
- **Example:** Predicting product demand or stock market prices.

2. Natural Language Processing (NLP)

SPPU-BE-COMP-CONTENT – KSKA Git

- Analyzes and understands human language data.
- **Example:** Sentiment analysis of customer reviews.

3. Image and Video Analysis

- Processes visual data for classification and detection tasks.
- **Example:** Facial recognition in security systems.

4. Recommendation Systems

- Suggests items to users based on preferences and behavior.
- **Example:** Movie recommendations on Netflix.

5. Fraud Detection

- Identifies unusual patterns in financial transactions.
- **Example:** Detecting credit card fraud in banking.

Machine Learning powers **data-driven insights** in Data Science by enabling automated prediction, pattern recognition, and intelligent decision-making across various domains.

b) Explain Geometric Model and Probabilistic Model with suitable examples.[5]

1. Geometric Model

Definition:

Represents data as points in a geometric space (e.g., vectors) and makes predictions based on **spatial relationships** (distances, boundaries).

Key Features:

Data represented as vectors in multi-dimensional space.

Uses **distance metrics** (Euclidean, Manhattan, cosine similarity).

Decision boundaries are geometric (lines, planes, hyperplanes).

Examples:

- **k-Nearest Neighbors (k-NN):** Classifies a point based on the majority class of its k closest neighbors.
- **Support Vector Machine (SVM):** Finds the optimal hyperplane to separate classes with maximum margin.

2. Probabilistic Model

Definition:

Uses **probability distributions** to model uncertainty and makes predictions via statistical inference.

SPPU-BE-COMP-CONTENT – KSKA Git

Key Features:

Quantifies uncertainty using probabilities.

Can be:

- **Generative** (models joint probability $P(X,Y)$, e.g., Naïve Bayes).
- **Discriminative** (models conditional probability $P(Y/X)$, e.g., Logistic Regression).

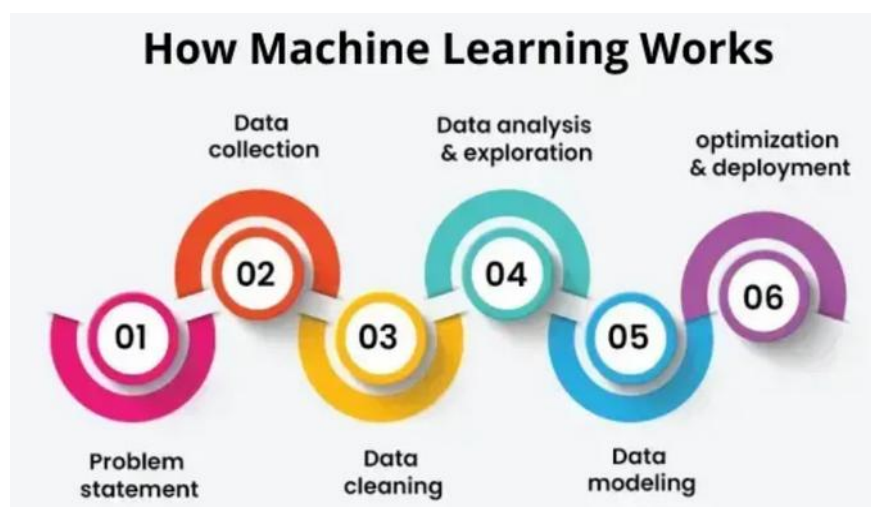
Examples:

- **Naïve Bayes:** Predicts class probabilities using Bayes' theorem (assumes feature independence).
- **Hidden Markov Model (HMM):** Models sequential data with probabilistic state transitions (e.g., speech recognition).

Comparison Summary

Aspect	Geometric Model	Probabilistic Model
Core Idea	Spatial relationships	Uncertainty & likelihoods
Decision Basis	Distances/boundaries	Probability distributions
Examples	k-NN, SVM	Naïve Bayes, HMM

c) How machine learning model works? Explain various steps involved.[5]



SPPU-BE-COMP-CONTENT – KSKA Git

1. Data Collection

- **Purpose:** Gather relevant data for training.
- **Sources:** Databases, APIs, sensors, web scraping, etc.
- **Example:** Collecting customer purchase history for a recommendation system.

2. Data Preprocessing

- **Cleaning:** Handle missing values, outliers, and noise.
- **Transformation:** Normalize, scale, or encode categorical data.
- **Feature Engineering:** Select/extract meaningful features (e.g., calculating "purchase frequency" from raw data).

3. Model Training

- **Algorithm Selection:** Choose based on problem type (e.g., SVM for classification, Linear Regression for prediction).
- **Training Phase:** Feed processed data to the algorithm to learn patterns.
- **Example:** Training a decision tree to classify emails as spam/ham.

4. Model Evaluation

- **Metrics:** Accuracy, Precision, Recall (for classification); MSE, R^2 (for regression).
- **Validation:** Use a test dataset to check performance.
- **Improvement:** Tune hyperparameters (e.g., learning rate, tree depth).

5. Deployment & Monitoring

- **Deploy:** Integrate the model into applications (APIs, mobile apps).
- **Monitor:** Track real-world performance and retrain with new data.
- **Example:** A fraud detection model in a bank flags transactions and updates weekly.

➤ SEP 2024

Q1)

a) Describe in detail different Machine Learning models used. [5]

Machine Learning models are categorized based on how they represent and process data to learn patterns for prediction or classification.

1. Geometric Models

- **Concept:** Represent data points as vectors in a geometric space and use spatial relationships for classification or prediction.
- **Features:** Uses distances, angles, and boundaries like lines or planes.
- **Examples:**
 - **k-Nearest Neighbors (k-NN):** Classifies based on nearest data points.

SPPU-BE-COMP-CONTENT – KSKA Git

- **Support Vector Machines (SVM):** Finds optimal separating hyperplane.

2. Probabilistic Models

- **Concept:** Represent data and predictions in terms of probability distributions.
- **Features:** Handle uncertainty and use statistical inference for decision-making.
- **Examples:**
 - **Naïve Bayes:** Applies Bayes' theorem with independence assumption.
 - **Hidden Markov Models (HMM):** Models sequential data probabilistically.

3. Logical Models

- **Concept:** Use logical rules, decision paths, and if-then conditions for classification or regression.
- **Features:** Easy to interpret, suitable for rule-based decisions.
- **Examples:**
 - **Decision Trees:** Splits data based on feature conditions.
 - **Rule-Based Classifiers:** Apply explicit logic rules.

4. Ensemble Models

- **Concept:** Combine multiple models to improve performance.
- **Features:** Reduce variance and bias.
- **Examples:** Random Forests, Gradient Boosting Machines (GBM).

ML models can be **geometric, probabilistic, logical, or ensemble-based**, and the choice depends on the nature of the problem, data type, and accuracy requirements.

b) What are the main differences between supervised learning and reinforcement learning? [5]

Supervised Learning (SL): Model learns from labeled training data (input-output pairs) to make predictions.

Reinforcement Learning (RL): Agent learns by interacting with an environment, receiving rewards or penalties to maximize long-term gain.

Aspect	Supervised Learning	Reinforcement Learning
--------	---------------------	------------------------

SPPU-BE-COMP-CONTENT – KSKA Git

Data Requirement	Requires labeled dataset (input-output pairs).	No labeled dataset; learns from trial-and-error feedback.
Goal	Minimize prediction error on unseen data.	Maximize cumulative reward over time.
Learning Process	Learns mapping between inputs and outputs directly.	Learns optimal policy (sequence of actions).
Feedback Type	Direct feedback via correct output labels.	Indirect feedback via reward/penalty signals.
Examples	Email spam classification, price prediction.	Game-playing AI (Chess, AlphaGo), robotic navigation.

Key Points:

1. **Supervised Learning** relies on a fixed dataset with correct answers, while **Reinforcement Learning** learns by exploring actions and receiving rewards.
2. **Supervised Learning** aims for accuracy in predictions, whereas **Reinforcement Learning** seeks optimal decision-making over time.
3. **Reinforcement Learning** involves sequential decision-making, while **Supervised Learning** typically deals with independent data points.

c) Explain geometric models and its types [5]

Geometric models in machine learning leverage geometric concepts to represent data and relationships. These models can be categorized based on their **structure** and the **type of data** they handle.

1. Linear Models: Linear models assume a linear relationship between input features and the output. They can be represented geometrically as hyperplanes in a multi-dimensional space.

Examples:

- Linear Regression: Predicts a continuous output based on linear combinations of input features.
- Logistic Regression: Used for binary classification, representing decision boundaries as linear hyperplanes.

2. Non-Linear Models: Non-linear models capture more complex relationships by using non-linear functions. These models can represent curves and other complex shapes in the feature space.

Examples:

- Polynomial Regression: Extends linear regression by adding polynomial terms to capture non-linear relationships.

SPPU-BE-COMP-CONTENT – KSKA Git

- Support Vector Machines (SVM): Can use kernel functions to create non-linear decision boundaries.

3. Geometric Deep Learning Models: These models extend traditional deep learning to non-Euclidean domains, such as graphs and manifolds. They leverage geometric structures to process data.

Examples:

- Graph Neural Networks (GNNs): Designed to work with graph-structured data, capturing relationships between nodes.
- Convolutional Neural Networks (CNNs): While primarily used for grid-like data (images), they can also be adapted for geometric data.

4. Manifold Learning Models: Manifold learning focuses on understanding the underlying structure of high-dimensional data by assuming that it lies on a lower-dimensional manifold.

Examples:

- t-Distributed Stochastic Neighbor Embedding (t-SNE): A technique for visualizing high-dimensional data by reducing it to two or three dimensions while preserving local structures.
- Isomap: Combines classical MDS with geodesic distances to preserve the manifold structure.

5. Probabilistic Geometric Models: These models incorporate uncertainty and probabilistic reasoning into geometric representations.

Examples:

- Gaussian Mixture Models (GMMs): Represent data as a mixture of multiple Gaussian distributions, allowing for flexible modeling of complex data distributions.
- Bayesian Networks: Use directed acyclic graphs to represent probabilistic relationships among variables.

6. Topological Models: Topological data analysis (TDA) focuses on the shape and structure of data rather than its specific values. It uses concepts from topology to analyze the connectivity and relationships within data.

Examples:

- Persistent Homology: A method that studies the multi-scale topological features of a dataset, capturing features that persist across multiple scales. It helps in identifying clusters, holes, and voids in data.
- Mapper Algorithm: A technique that creates a simplicial complex from high-dimensional data, allowing for visualization and analysis of the data's topological structure.

7. Geometric Transformations: These models utilize geometric transformations to manipulate data for various tasks, such as data augmentation or feature extraction.

Examples:

- Affine Transformations: Include scaling, rotation, and translation, which can be applied to images or spatial data to enhance model robustness.

SPPU-BE-COMP-CONTENT – KSKA Git

- Homographies: Used in computer vision to relate different views of the same scene, allowing for tasks like image stitching and perspective correction.

Q2)

a) Compare Artificial intelligence and Machine learning. [5]

ALREADY DONE !

b) Describe the various steps involved in developing a machine learning application. [5]

Steps Involved in Developing a Machine Learning Application

Developing an ML application follows a systematic process to ensure that the solution is accurate, reliable, and deployable in a real-world environment. The key steps are:

1. **Problem Definition & Requirement Analysis**
 - Clearly define the objective of the application and the problem it aims to solve.
 - Understand business goals, success metrics, and constraints.
 - Example: Predicting customer churn in a telecom company.
2. **Data Collection & Understanding**
 - Gather relevant data from multiple sources such as databases, sensors, APIs, or web scraping.
 - Perform exploratory data analysis (EDA) to understand patterns, trends, and anomalies in the dataset.
3. **Data Preprocessing & Cleaning**
 - Handle missing values, outliers, and inconsistent formats.
 - Normalize, standardize, and encode data for model readiness.
 - Ensures the data quality is suitable for training models.
4. **Feature Engineering & Selection**
 - Create new informative features from existing data.
 - Select only relevant features to improve model performance and reduce complexity.
5. **Model Selection & Training**
 - Choose the appropriate algorithm (e.g., Decision Trees, Neural Networks, SVM) based on the problem type.
 - Train the model using historical data while tuning hyperparameters for optimal results.
6. **Model Evaluation & Validation**

SPPU-BE-COMP-CONTENT – KSKA Git

- Assess performance using metrics like accuracy, precision, recall, F1-score, or RMSE.
- Use cross-validation to ensure the model generalizes well to unseen data.

7. Deployment & Integration

- Implement the model into the application environment (web app, mobile app, API).
- Ensure it works seamlessly with existing systems and user interfaces.

8. Monitoring, Maintenance & Updates

- Continuously monitor performance and retrain with fresh data to maintain accuracy.
- Apply updates when the business needs or data patterns change.

These steps ensure that the ML application is not only technically accurate but also aligned with real-world requirements. A well-developed ML application delivers consistent, reliable results while adapting to evolving data and business goals.

c) Describe grouping and grading models. [5]

1. Grouping Models (Unsupervised Learning)

Grouping models, also called **clustering techniques**, are unsupervised learning methods that organize unlabeled data into meaningful clusters based on similarity patterns.

Key Characteristics:

- No predefined output labels required
- Discovers hidden structures in data
- Uses similarity/distance metrics (Euclidean, Manhattan, cosine)

Common Algorithms:

- **K-Means:** Partitions data into K spherical clusters
- **Hierarchical:** Creates nested clusters (dendrogram visualization)
- **DBSCAN:** Forms clusters based on density (handles noise well)

Applications:

- Customer segmentation for targeted marketing
- Document clustering in NLP
- Anomaly detection in network security

2. Grading Models (Supervised Learning)

Grading models are supervised approaches that classify or rank items into predefined categories or ordered levels.

SPPU-BE-COMP-CONTENT – KSKA Git

Key Characteristics:

- Requires labeled training data
- Predicts discrete classes or continuous scores
- Can handle ordinal (ranked) categories

Common Approaches:

- **Classification Models:**
 - Logistic regression, Random Forests (for categorical grading)
 - Example: Pass/Fail student evaluation
- **Ordinal Regression Models:**
 - Proportional Odds Model (for ordered categories)
 - Example: Product rating prediction (1-5 stars)
- **Learning-to-Rank Models:**
 - Listwise, pairwise approaches for ranking
 - Example: Search engine result ordering

Applications:

- Credit risk assessment (AAA to D ratings)
- Academic performance grading
- Quality control in manufacturing

“Check / Verify Answer – Read at Your Own Risk”