CA6001 课程 A4 开卷资料（双语知识点提纲）

一、Supervised Learning（监督学习）

1. Core Features（核心特征）

- **Input**: Labeled data (input (X) + target output (Y))（输入：标记数据（输入(X)+目标输出(Y)））
- **Feedback**: Has ground truth (direct evaluation of prediction accuracy)（反馈：有真实标签（直接评估预测准确率））
- **Goal**: Learn (X→Y) mapping for future prediction（目标：学习(X→Y)映射以进行未来预测）

2. Subtypes & Algorithms（子类型与算法）

(1) Regression（回归）

- **Output**: Real/continuous value（输出：实数/连续值）
- **Key Algorithms**:
  - **Linear Regression（线性回归）**：
    - Model: ($f_{w,b}(x) = wx + b$)（模型公式）
    - Cost Function (MSE): ($J(w,b) = \frac{1}{2m}\sum_{i=1}^m (f_{w,b}(x^{(i)}) - y^{(i)})^2$)（成本函数：均方误差）
    - Goal: Minimize ($J(w,b)$) via gradient descent（目标：通过梯度下降最小化成本函数）
  - **Logistic Regression（逻辑回归）**：
    - **Non-linear Relationship（非线性关系）**: Learns non-linear boundaries（学习非线性边界）
    - **Activation Function（激活函数）**: Sigmoid function（Sigmoid函数）
      - Formula: ($g(z) = \frac{1}{1+e^{-z}}$)，where ($z = wx + b$)（公式：($g(z)$)将输出压缩至[0,1]）
    - **Threshold（阈值）**: (≥ 0.5) → Class 1; (< 0.5) → Class 0（二分类阈值）

(2) Classification（分类）

- **Output**: Categorical value（输出：分类值）
- **Key Algorithms**:
  - Decision Tree（决策树）: Splits data via feature thresholds（通过特征阈值分割数据）
  - Random Forest（随机森林）: Ensemble of decision trees (reduces overfitting)（决策树集成，减少过拟合）
  - K-Nearest Neighbors (KNN)（K近邻）: Classifies via majority vote of (k) nearest samples（通过(k)个近邻的多数投票分类）
  - Support Vector Machine (SVM)（支持向量机）: Finds hyperplane to maximize class margin（寻找超平面以最大化类别间隔）
  - Naive Bayes（朴素贝叶斯）: Based on Bayes' theorem (fast, for text classification)（基于贝叶斯定理，适用于文本分类）

二、Unsupervised Learning（无监督学习）

1. Core Features（核心特征）

- **Input**: Unlabeled data (only (X))（输入：无标记数据（仅(X)））
- **Feedback**: No ground truth（反馈：无真实标签）
- **Goal**: Discover hidden patterns in data（目标：发现数据中的隐藏模式）

2. Subtypes & Algorithms（子类型与算法）

(1) Clustering（聚类）

- **Goal**: Group similar data points（目标：将相似数据分组）
- **Key Algorithms**:
  - K-Means（K均值）: Partitions data into (k) clusters (centroid-based)（将数据分为(k)个簇，基于质心）
  - GMM (Gaussian Mixture Model)（高斯混合模型）: Probabilistic clustering (handles overlapping clusters)（概率型聚类，处理重叠簇）

(2) Dimensionality Reduction（降维）

- **Goal**: Reduce feature count while preserving key information（目标：减少特征数同时保留关键信息）
- **Key Algorithms**:
  - PCA (Principal Component Analysis)（主成分分析）: Projects data to orthogonal principal components（将数据投影到正交主成分）
  - SVD (Singular Value Decomposition)（奇异值分解）: Factorizes matrix into 3 sub-matrices (for data compression)（矩阵分解为3个子矩阵，用于数据压缩）

(3) Association（关联）

- **Goal**: Find frequent item relationships (e.g., "buy A → buy B")（目标：发现频繁项关联）
- **Key Algorithms**:
  - Apriori: Finds frequent itemsets via "prune infrequent" rule（通过"剪枝不频繁项"寻找频繁项集）
  - FP-Growth (Frequent Pattern Growth): Builds FP-tree to mine frequent patterns (more efficient)（构建FP树挖掘频繁模式，更高效）

三、Learning Evaluation（学习评估）

1. Dataset Splitting（数据集划分）

- **Train Set**: For model training（训练集：用于模型训练）
- **Validation Set**: For hyperparameter tuning (e.g., adjust (k) in KNN)（验证集：用于超参数调优）
- **Test Set**: For final performance evaluation (unseen data)（测试集：用于最终性能评估，未见过的数据）

2. Cross-Validation（交叉验证）

- **k-fold Cross-Validation**: Split data into (k) folds; train on (k-1) folds, test on 1 fold (repeat (k) times)（将数据分为(k)折；用(k-1)折训练，1折测试，重复(k)次）
- **Purpose**: Ensure robust performance (avoids overfitting to test set)（目的：确保性能鲁棒性，避免过拟合测试集）

3. Core Evaluation Metrics（核心评估指标）

(1) Confusion Matrix（混淆矩阵）

	Predicted Positive	Predicted Negative
Actual Positive	TP (True Positive)	FN (False Negative)
Actual Negative	FP (False Positive)	TN (True Negative)

(2) Key Metrics（关键指标）

- **Accuracy（准确率）**: $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$
  - Note: Not suitable for imbalanced data（注意：不适用于不平衡数据）
- **Precision（精确率）**: $Precision = \frac{TP}{TP+FP}$
  - Purpose: Minimize false positives (e.g., disease diagnosis: avoid misdiagnosing healthy people)（目的：减少假阳性，如疾病诊断：避免健康人被误诊）
- **Recall（召回率）**: $Recall = \frac{TP}{TP+FN}$
  - Purpose: Minimize false negatives (e.g., cancer screening: avoid missing patients)（目的：减少假阴性，如癌症筛查：避免漏诊患者）
- **F1-Score**: $F1 = 2\times\frac{Precision\times Recall}{Precision+Recall}$
  - Purpose: Balance precision and recall (good for imbalanced data)（目的：平衡精确率和召回率，适用于不平衡数据）

四、Reinforcement Learning（强化学习）

1. Core Definition（核心定义）

- **Agent（智能体）**: Learner/decision-maker that interacts with environment（与环境交互的学习者/决策者）
- **Goal**: Maximize cumulative reward in the environment（目标：最大化环境中的累积奖励）
- **Key Cycle**: (State(S) → Action(A) → Reward(R) → Next State(S'))（核心循环：状态→行动→奖励→下一状态）

2. Implementation Steps（实现步骤）

1. Define/Create the Environment（定义/创建环境）
2. Specify the Reward Function（指定奖励函数：将行动作为智能体性能指标）
3. Training and Validation（训练与验证：给智能体训练策略）
4. Define and Validate the Policy（定义并验证策略）
5. Implement the Policy（执行策略）

3. Exploration-Exploitation Tradeoff（探索-利用权衡）

- **Exploration**: Try new actions (reduce uncertainty about environment)（探索：尝试新行动，减少环境不确定性）
- **Exploitation**: Choose known best actions (maximize immediate reward)（利用：选择已知最佳行动，最大化即时奖励）

- **Strategy**: **ε-Greedy Strategy**
  - Rule: With probability (ε) → Explore (random action); with probability (1-ε) → Exploit (best action)（规则：以概率(ε)探索，(1-ε)利用）

4. Key Algorithms（关键算法）

(1) Q-Learning

- **Type**: Off-policy (uses past experiences, not just current policy)（类型：离策略，使用过去经验而非仅当前策略）
- **Core**: Estimate (Q(S,A)) (value of action (A) in state (S))（核心：估计状态-行动对的价值(Q(S,A))）
- **Update Rule**: (Q(S,A) = Q(S,A) + α[R + γ\max_{A'}Q(S',A') - Q(S,A)])
  - (α): Learning rate (0<α≤1)（学习率）
  - (γ): Discount factor (0≤γ≤1, prioritizes future rewards)（折扣因子，重视未来奖励）

(2) SARSA

- **Type**: On-policy (uses actions from current policy)（类型：在策略，使用当前策略产生的行动）
- **Core**: Learns via (S→A→R→S'→A') (current state-action to next state-action)（核心：通过当前状态-行动到下一状态-行动学习）

5. RLHF (Reinforcement Learning with Human Feedback)（基于人类反馈的强化学习）

- **3 Steps**:
  1. **Supervised Fine-tuning (SFT)**: Train LLM with human-labeled data（监督微调：用人类标记数据训练大语言模型）
  2. **Train Reward Model (RM)**: Train model to rank LLM outputs (align with human preference)（训练奖励模型：训练模型对LLM输出排序，对齐人类偏好）
  3. **Policy Optimization (PPO)**: Optimize LLM policy via RM (maximize human-aligned reward)（策略优化：通过奖励模型优化LLM策略，最大化人类对齐奖励）

五、Deep Learning（深度学习）

1. Core Concepts（核心概念）

- **Deep Neural Network**: Neural network with ≥2 hidden layers (learns hierarchical features)（深度神经网络：≥2个隐藏层的网络，学习分层特征）
- **Feature Learning**: Automatically extracts features (no manual feature engineering)（特征学习：自动提取特征，无需手动特征工程）

2. Propagation Mechanisms（传播机制）

(1) Forward Propagation（前向传播）

- **Direction**: Input Layer → Hidden Layers → Output Layer（方向：输入层→隐藏层→输出层）
- **Calculation**: (z = wx + b) → (a = g(z)) ((g) = activation function)（计算：权重×输入+偏置→激活函数处理）
- **Purpose**: Compute prediction (for inference/training)（目的：计算预测值，用于推理/训练）

(2) Backward Propagation（反向传播）

- **Direction**: Output Layer → Hidden Layers → Input Layer（方向：输出层→隐藏层→输入层）
- **Calculation**: Use **Chain Rule** to compute gradients of loss w.r.t (w) and (b)（计算：通过链式法则计算损失对权重(w)和偏置(b)的梯度）
- **Purpose**: Update (w) and (b) to minimize loss (via gradient descent)（目的：更新权重和偏置以最小化损失）

3. Activation Functions（激活函数）

- **Core Role**: Introduce non-linearity (enables learning complex patterns)（核心作用：引入非线性，使网络能学习复杂模式）
- **Common Functions**:
  - **Sigmoid**: ($g(z) = \frac{1}{1+e^{-z}}$) (output [0,1]; used in binary classification)
  - **ReLU (Rectified Linear Unit)**: ($g(z) = \max(0,z)$) (fast computation; reduces vanishing gradient)
  - **Tanh (Hyperbolic Tangent)**: ($g(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$) (output [-1,1]; centered at 0)

4. Fully Connected (Dense) Layers（全连接层）

- **Structure**: All neurons in layer (l) connect to all neurons in layer (l+1)（结构：第(l)层所有神经元连接到第(l+1)层所有神经元）
- **Roles**:
  - Feature Transformation: Map high-dimensional features to low-dimensional space（特征转换：将高维特征映射到低维空间）
  - Output Generation: Use Softmax (for multi-class classification) to output class probabilities（输出生成：用Softmax生成多分类概率）

5. Regularization (Overfitting Control)（正则化：过拟合控制）

- **Overfitting**: Model performs well on train set but poorly on test set（过拟合：模型在训练集表现好，测试集表现差）
- **Solutions**:
  - **Dropout**: Randomly deactivate some neurons during training (reduces co-dependency)（训练时随机停用部分神经元，减少共依赖）
  - **L2 Regularization**: Add ($\lambda\sum w^2$) to loss (penalizes large weights)（在损失中添加权重平方和，惩罚大权重）

6. Applications & Challenges（应用与挑战）

- **Applications**:
  - Vision: Object detection, facial recognition（视觉：目标检测、人脸识别）
  - NLP: Translation, sentiment analysis（自然语言处理：翻译、情感分析）
  - Healthcare: Disease diagnosis, drug discovery（医疗：疾病诊断、药物发现）
- **Challenges**:
  - High data/computational cost (needs GPU/TPU)（高数据/计算成本，需GPU/TPU）
  - Black-box interpretability (hard to explain decisions)（黑箱可解释性，难以解释决策）
  - Overfitting & generalization issues（过拟合与泛化问题）