Decision Tree and KNN Practice Questions

PART A: Data Exploration and Preparation

A1. Initial Data Investigation

1. Load the dataset and display basic information about its structure.
   - How many rows and columns are in the dataset?
   - What are the data types of each column?
   - Are there any missing values? If so, how would you handle them?
2. Create a summary of the target variable (final_grade).
   - What percentage of students achieved "High" vs "Low" grades?
   - Is the dataset balanced? Why does this matter for machine learning?

A2. Descriptive Statistics

3. Calculate descriptive statistics for all numerical features:
   - Mean, median, standard deviation, min, max for each numerical column
   - Identify any potential outliers using the IQR method
   - Which numerical feature has the highest variability?
4. Create frequency tables for all categorical features:
   - What is the most common previous grade?
   - Which socioeconomic status category is most represented?
   - How are students distributed across extracurricular participation levels?

A3. Data Visualization and Relationships

Create appropriate visualizations to explore feature distributions:

- Histograms for numerical features
- Bar charts for categorical features
- Box plots comparing numerical features across the target variable

6. Investigate correlations between features:
   - Create a correlation matrix for numerical features
   - Which numerical features are most strongly correlated with each other?
   - Use cross-tabulations to explore relationships between categorical features

A4. Data Preprocessing

7. Prepare the data for machine learning:
   - Encode categorical variables using appropriate methods (explain your choices)
   - Scale numerical features (why is this important for KNN but not Decision Trees?)
   - Create feature matrix (X) and target vector (y)
8. Split the data:
   - Divide into training (70%) and testing (30%) sets
   - Use stratification to maintain class distribution

- Set a random state for reproducibility

PART B: Decision Tree Implementation and Analysis

B1. Basic Decision Tree Model

9. Build a basic decision tree classifier:
   - Train on the training set using default parameters
   - Make predictions on the test set
   - Calculate accuracy, precision, recall, and F1-score
   - Create and interpret the confusion matrix
10. Visualize the decision tree:
    - Plot the tree structure (limit depth to 3 for clarity)
    - Identify the root node split - which feature is used and why?
    - Trace the decision path for a high-performing and low-performing student

B2. Decision Tree Parameter Tuning

11. Experiment with tree depth:
    - Train trees with max_depth = [3, 5, 7, 10, None]
    - Plot training and validation accuracy vs. depth
    - Identify the optimal depth and explain the bias-variance tradeoff
12. Tune other hyperparameters:
    - Test different values for min_samples_split [2, 5, 10, 20]
    - Test different values for min_samples_leaf [1, 5, 10, 15]
    - Use cross-validation to find the best combination
    - Report the best parameters and their performance

B3. Feature Importance Analysis

13. Analyze feature importance:
    - Extract and visualize feature importance scores
    - Which are the top 3 most important features?
    - Compare importance scores between different tree configurations
    - Do the results align with your intuition about student performance?
14. Create a simplified model:
    - Build a new tree using only the top 5 most important features
    - Compare performance with the full-feature model
    - Discuss the trade-offs between model complexity and performance

PART C: K-Nearest Neighbors Implementation and Analysis

C1. Basic KNN Model

15. Build a KNN classifier:
    - Start with k=5 and Euclidean distance
    - Train on the scaled training data
    - Calculate the same performance metrics as for Decision Tree
    - Compare the confusion matrix with the Decision Tree results
16. Impact of feature scaling:
    - Train KNN models with and without feature scaling
    - Compare their performance
    - Explain why scaling affects KNN but not Decision Trees

C2. Parameter Optimization

17. Find optimal k value:
    - Test k values from 1 to 21 (odd numbers only)
    - Plot accuracy vs. k for both training and validation sets
    - Identify the optimal k and explain the bias-variance tradeoff
    - What happens when k is too small or too large?
18. Distance metric comparison:
    - Compare Euclidean, Manhattan, and Minkowski distances
    - Test with different p values for Minkowski (p=1, 1.5, 2, 3)
    - Which distance metric works best for this dataset?

C3. Advanced KNN Analysis

19. Analyze computational complexity:
    - Measure training and prediction times for different k values
    - How does the dataset size affect KNN performance?
    - Compare computational costs with Decision Tree
20. Feature impact on KNN:
    - Systematically remove each feature and measure performance impact
    - Which features are most critical for KNN predictions?
    - How does this compare to Decision Tree feature importance?

PART D: Model Comparison and Evaluation

D1. Performance Comparison

21. Create a comprehensive comparison:
    - Build a table comparing both models' best performance metrics
    - Include accuracy, precision, recall, F1-score for both classes
    - Calculate and compare ROC curves and AUC scores
    - Which model performs better overall?
22. Error analysis:
    - Identify samples that both models predict incorrectly
    - Find samples where models disagree in their predictions
    - Analyze patterns in misclassified students - any common characteristics?

D2. Model Interpretability

23. Interpretability comparison:
    - Explain how you would interpret a Decision Tree prediction to a teacher
    - Explain how you would interpret a KNN prediction to a teacher
    - Which model provides better insights for educational interventions?
24. Business impact analysis:
    - If you were a school administrator, which model would you prefer and why?
    - Discuss the consequences of false positives vs. false negatives
    - How would you present these models' insights to non-technical stakeholders?

PART E: Advanced Analysis and Real-World Considerations

E1. Cross-Validation and Stability

25. Implement robust evaluation:
    - Perform 5-fold cross-validation for both models
    - Calculate mean and standard deviation of performance metrics
    - Which model is more stable across different data splits?
26. Learning curves:
    - Plot learning curves showing performance vs. training set size
    - Start with 100 samples and increase to full dataset
    - How much data does each algorithm need to achieve good performance?

E2. Ethical and Practical Considerations

27. Bias and fairness analysis:
    - Check if model performance varies across socioeconomic status groups
    - Are there any signs of unfair bias in the predictions?
    - How would you address any identified biases?
28. Real-world deployment:
    - What additional features might improve model performance?

- How would you monitor model performance in production?
- What are the privacy and ethical implications of using such models in schools?

B1. Feature Engineering

29. Create new features:
    - Engineer features like study_efficiency = study_hours / screen_time
    - Create interaction features between categorical variables
    - Test if these improve model performance

B2. Ensemble Methods

30. Combine models:
    - Create a voting classifier using both Decision Tree and KNN
    - Implement a simple ensemble that uses Decision Tree for interpretable cases and KNN for others
    - Compare ensemble performance with individual models