1. What did you do to prepare the data?

First, I examined the dataset, including the data types and the first five rows, to understand its structure. I found that some values were missing or invalid, such as ?, *. These prevented proper data conversion and caused missing values issues. I used ChatGPT to debug and find the best way to clean and process the data properly.

Then I handled missing values by filling categorical variables, such as node-caps, with the mode, while numerical variables like age and tumor-size were filled with the median. Additionally, I removed duplicate rows to ensure data integrity. I also applied One-Hot Encoding to categorical variables, specifically menopause and breast-quad.

2. What insights did you get from your data preparation?

One key insight from the data preparation process was that it is an iterative and step-by-step procedure—any mistake in one step can lead to errors in subsequent steps, affecting the final model performance.

Another important finding was the class imbalance issue, where recurrence cases (class=1) were significantly fewer than non-recurrence cases (class=0). This imbalance could potentially affect the model's ability to predict recurrence accurately.

3.What procedure did you use to train the model?

I first split the dataset into 80% training and 20% testing sets. Since the dataset was imbalanced, I used stratified sampling to ensure that the distribution of the class variable remained consistent in both the training and testing sets.

The KNN model was trained using default parameters. For optimized version of KNN, I used GridSearchCV to perform cross-validation and find the best values for k, weights, and metric. The Logistic Regression was trained with a maximum iteration limit of 1000 to ensure proper convergence.

4.How does the model perform to predict the class?

```
Best KNN parameters: {'metric': 'euclidean', 'n_neighbors': 3, 'weights': 'uniform'}
Best cross-validation score: 0.4030

Model Performance Comparison:
                 Model  Accuracy    Recall  Precision  F1 Score
0                  KNN  0.613333  0.208333   0.333333  0.256410
1       Optimized KNN  0.586667  0.333333   0.347826  0.340426
2  Logistic Regression  0.613333  0.166667   0.307692  0.216216

Detailed Classification Reports:
```

```
Detailed Classification Reports:

KNN:
              precision    recall  f1-score   support

           0       0.68      0.80      0.74        51
           1       0.33      0.21      0.26        24

    accuracy                           0.61        75
   macro avg       0.51      0.51      0.50        75
weighted avg       0.57      0.61      0.58        75


Optimized KNN:
              precision    recall  f1-score   support

           0       0.69      0.71      0.70        51
           1       0.35      0.33      0.34        24

    accuracy                           0.59        75
   macro avg       0.52      0.52      0.52        75
weighted avg       0.58      0.59      0.58        75


Logistic Regression:
              precision    recall  f1-score   support

           0       0.68      0.82      0.74        51
           1       0.31      0.17      0.22        24

    accuracy                           0.61        75
   macro avg       0.49      0.50      0.48        75
weighted avg       0.56      0.61      0.57        75
```

5.How confident are you in the model?

I think the models demonstrated limited confidence in predicting recurrence cases due to the imbalanced dataset. Both Logistic Regression and KNN tended to predict non-recurrence (class=0) more frequently, leading to low recall scores for recurrence cases.
While the optimized KNN model slightly improved recall, the overall model performance suggests that additional improvements are necessary to enhance prediction accuracy for recurrence cases.