ML ASSIGNEMNT 2

OUR TEAM

✓ Mariam Ibrahim Mahmoud
23011146

✓ Salma Ayman
23011081

✓ Virgin Tarek
23011124

PROJECT OVERVIEW

This project focuses on predicting passenger survival on the Titanic using the K-Nearest Neighbors (KNN) algorithm. The dataset, sourced from train.csv, contains 891 passenger records with features like Age, Fare, Sex, Pclass, and Embarked. The goal is to preprocess the data, train a KNN model, evaluate its performance, and visualize class separation to understand the model's behavior.

DATA SET DESCRIPTION

The Titanic dataset is a subset of historical data collected from passengers aboard the RMS Titanic, which sank in 1912. This dataset is predicting whether a passenger survived or not.

DATA SET DESCRIPTION

columns:

PassengerId:Unique identifier for each passenger.

Survived :Survival status (0 = No, 1 = Yes). This is the target variable.

Pclass:Ticket class (1 = 1st, 2 = 2nd, 3 = 3rd).

Name:Full name of the passenger.

Sex:Gender of the passenger (male, female).

Age:Age of the passenger in years.

SibSp:Number of siblings or spouses aboard the Titanic.

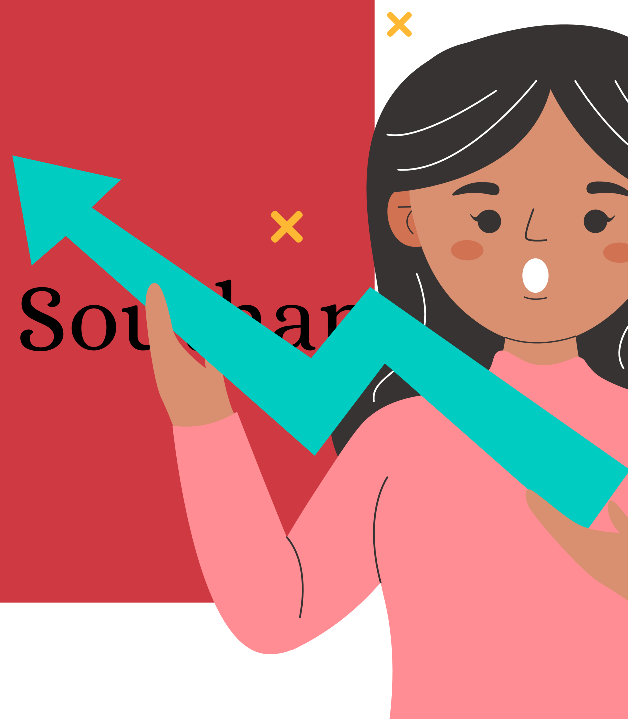
Parch"Number of parents or children aboard the Titanic.

Ticket:Ticket number.   Fare :Passenger fare (in British pounds).

Cabin:Cabin number (if known).

Embarked:Port of Embarkation (C = Cherbourg, Q = Queenstown, S = Southam

urvived, used to build classification models.



DATA PREPROCESSING

Correct Data Types:

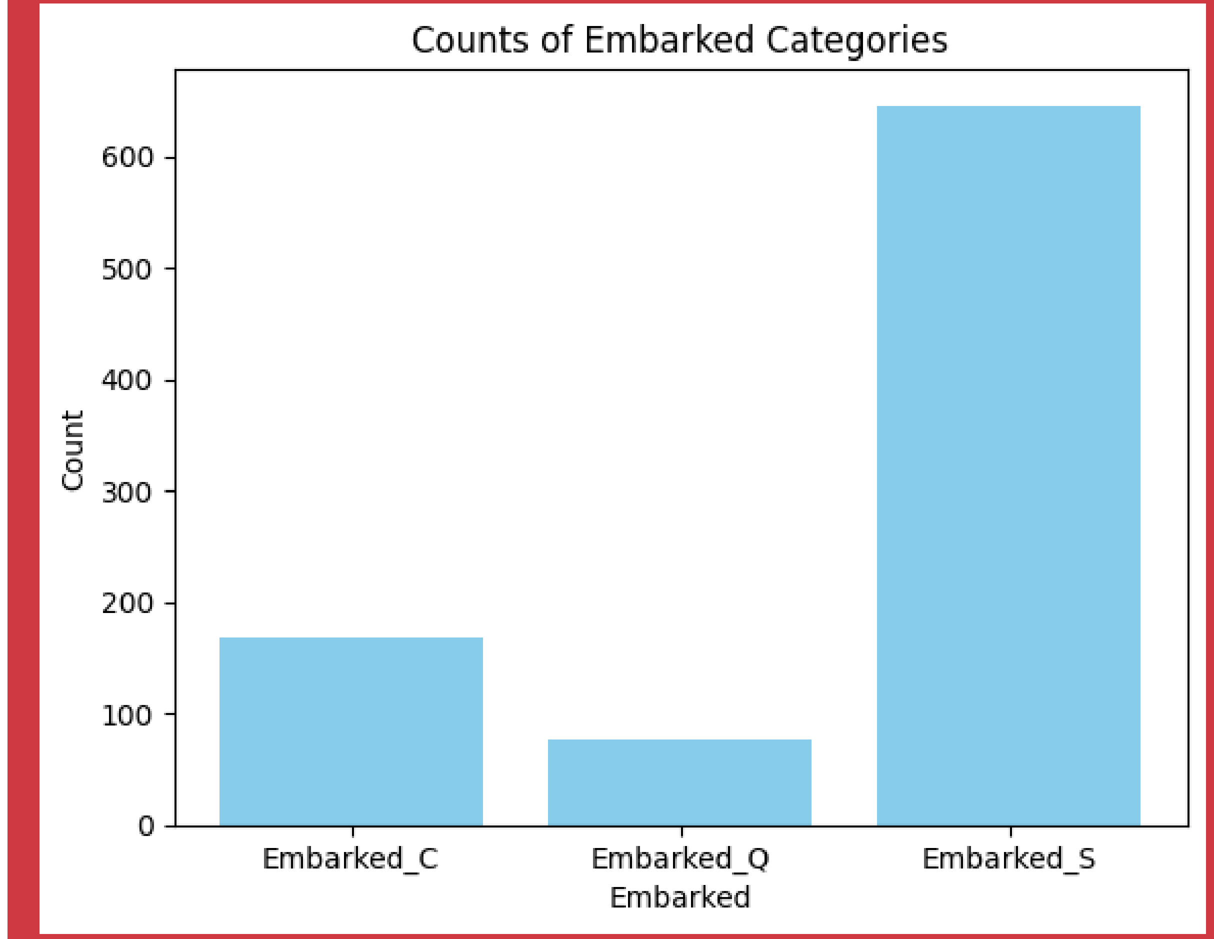
- Converted Age and Survived to integers using df['Age'].round().astype("int") and df['Survived'].round().astype("int").

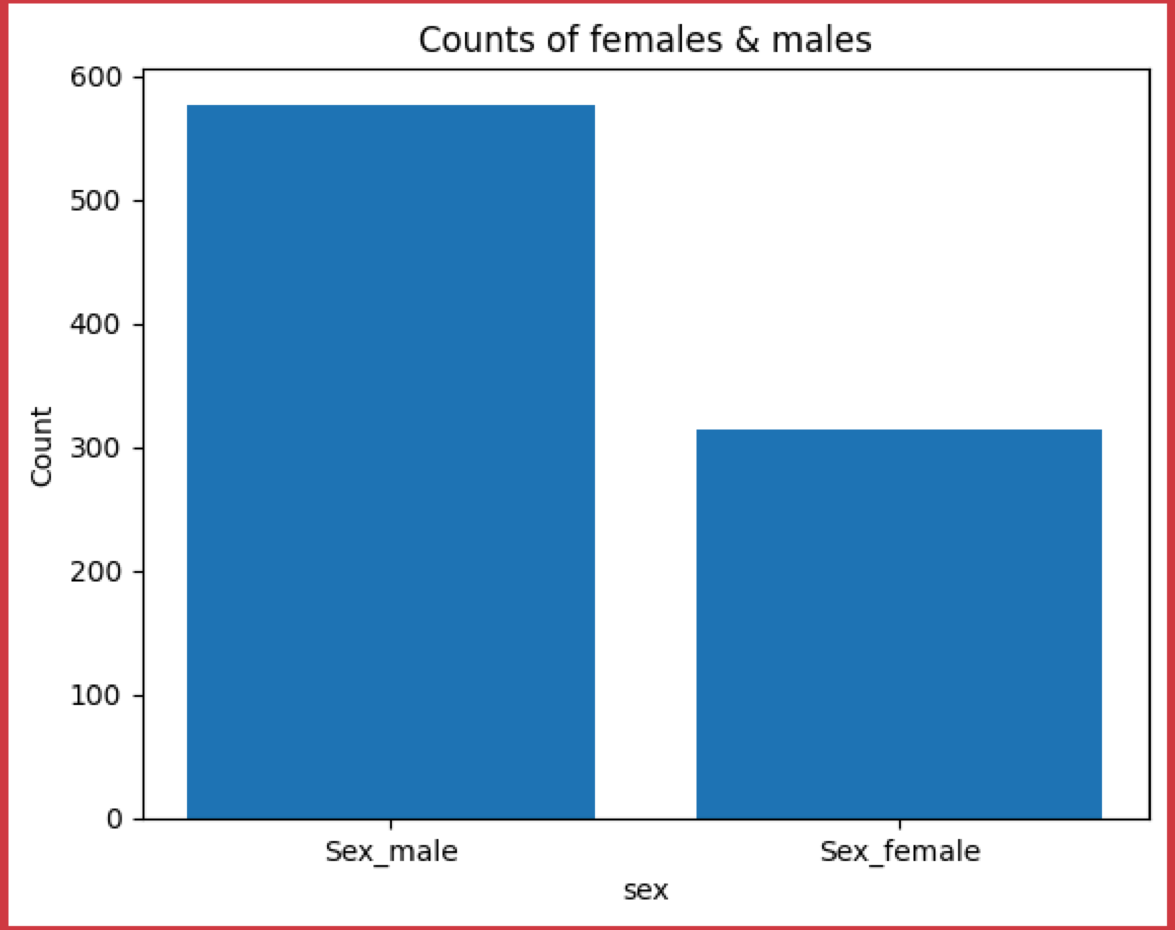
- Rationale: Ensures numerical consistency for machine learning algorithms. Survived is binary (0 or 1), and Age as an integer simplifies calculations.

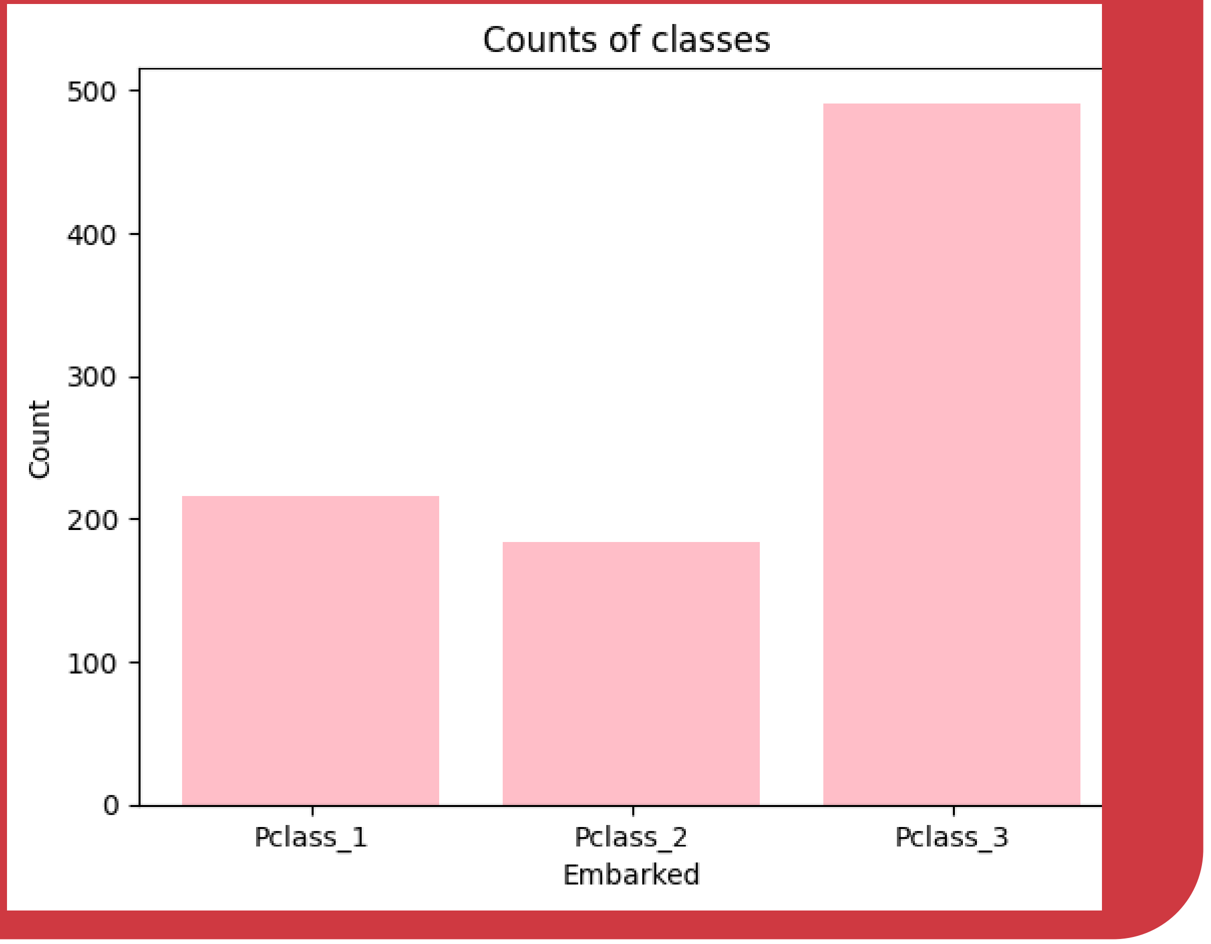
DATA PREPROCESSING

- Categorical to Numerical Conversion:
  - Used pd.get_dummies() to one-hot encode categorical features (Pclass, Sex, Embarked), creating columns like Pclass_1, Pclass_2, Pclass_3, Sex_male, Sex_female, Embarked_C, Embarked_Q, Embarked_S.
  - Dropped redundant columns (Embarked_Q, Sex_female) as they don’t add new information (e.g., Sex_female is redundant with Sex_male).
  - Rationale: KNN requires numerical inputs. One-hot encoding transforms categorical variables into a format suitable for distance calculations, and dropping redundant columns reduces dimensionality.

EXPLORATORY ANALYSIS







DATA PREPROCESSING

Feature Selection and Data Splitting

- **Feature Selection:**

Selected features: Age, SibSp, Parch, Fare, Pclass_1, Pclass_2, Pclass_3, Sex_male, Embarked_C, Embarked_S.

Target: Survived.

Rationale: These features are relevant for predicting survival. Dropped columns like PassengerId and Name as they’re not predictive.

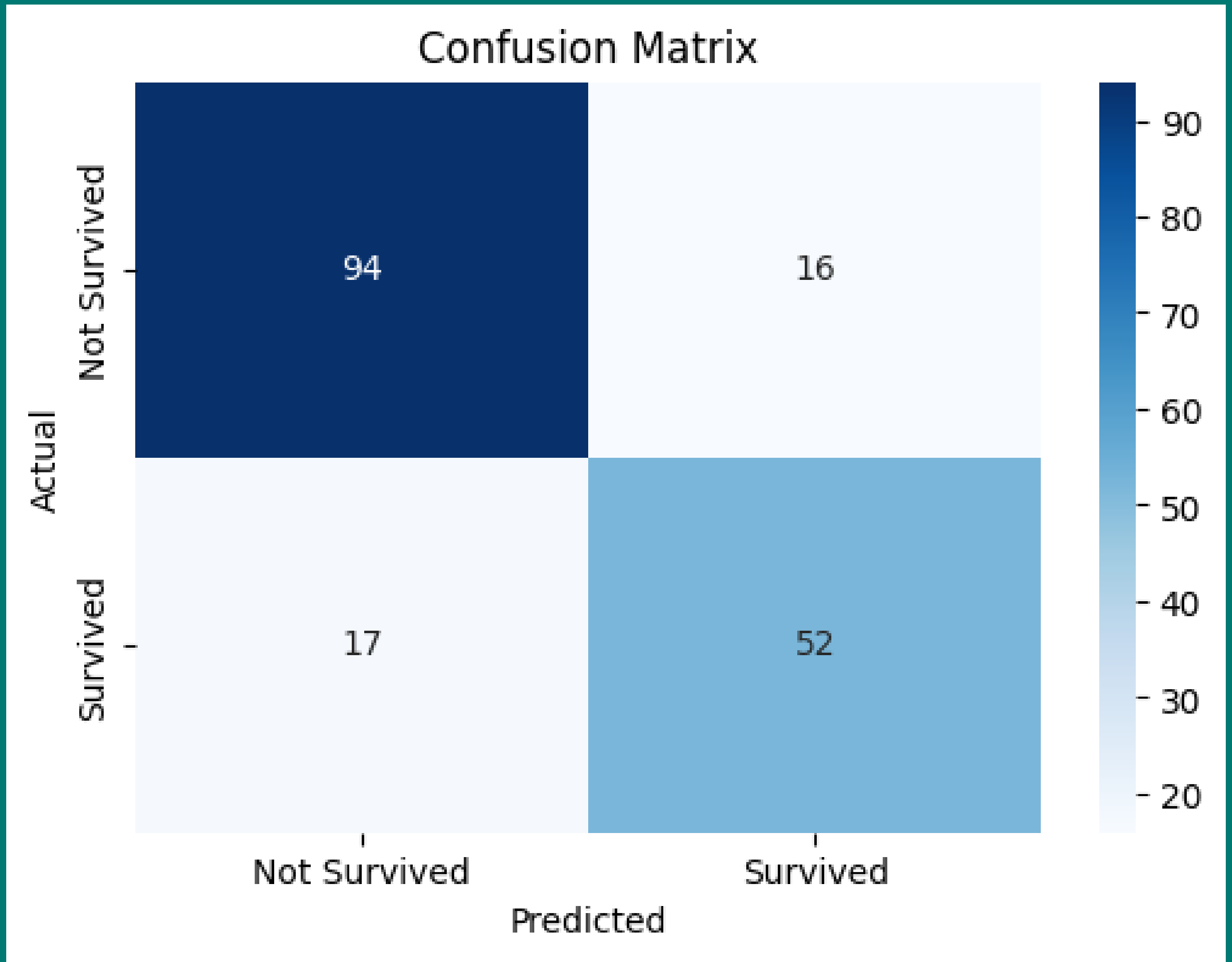
- **Split the data** into 60% training, 20% validation, and 20% test sets using train_test_split.

Rationale: Splitting ensures the model is trained, validated, and tested on separate data to evaluate generalization and avoid overfitting.

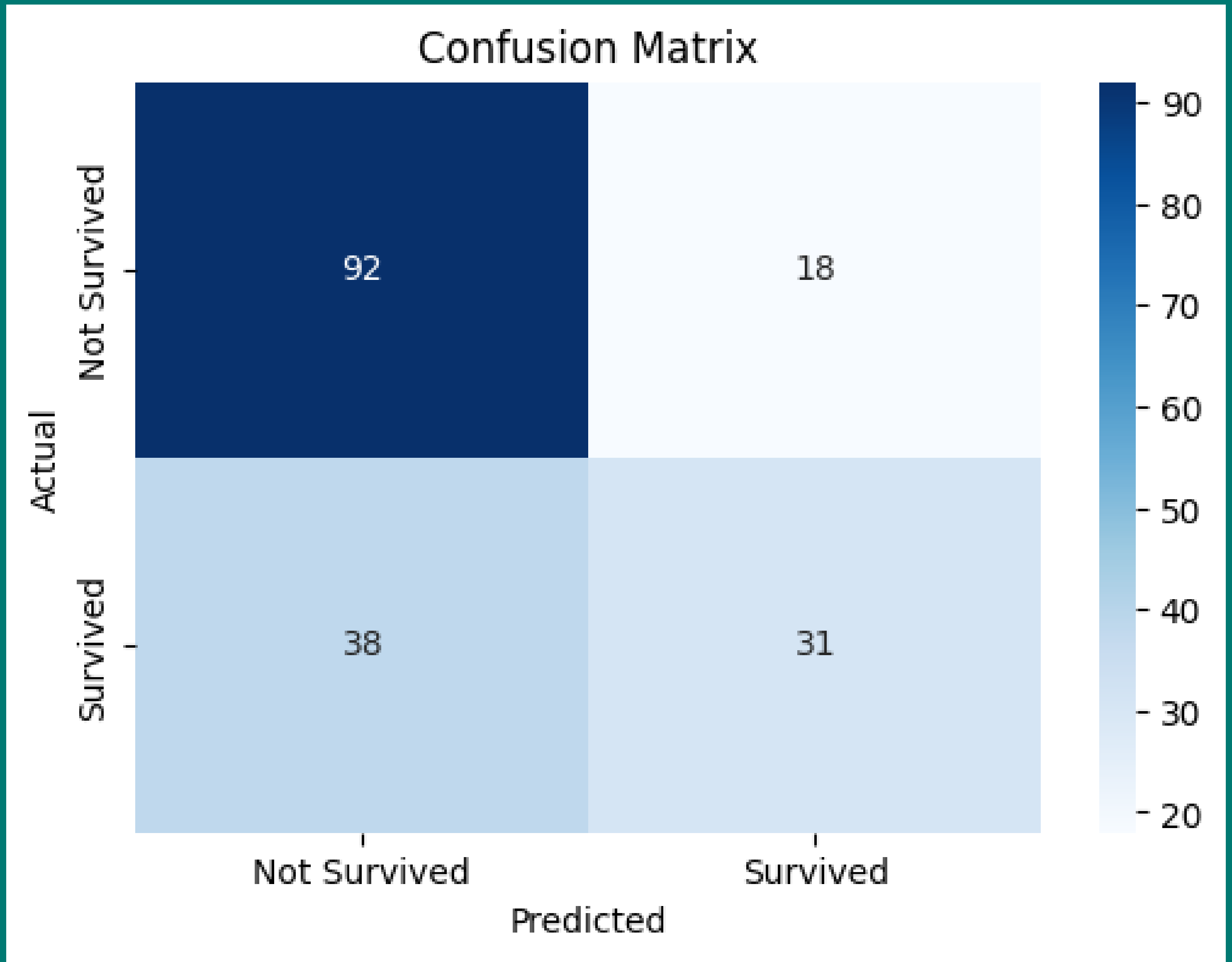
-

SCALING NUMERICAL DATA USING STANDARD SCALER LIBRARY

KNN RELIES ON DISTANCE CALCULATIONS, SO FEATURES MUST BE ON THE SAME SCALE. FITTING THE SCALER ONLY ON THE TRAINING DATA PREVENTS DATA LEAKAGE FROM VALIDATION/TEST SETS.



AFTER



BEFORE

5. KNN MODEL TRAINING AND HYPERPARAMETER TUNING

- HYPERPARAMETER TUNING:
- TESTED K VALUES FROM 1 TO 20 USING THE VALIDATION SET.
- FOR EACH K, TRAINED A KNN MODEL, PREDICTED ON X_VAL_SCALED, AND CALCULATED ACCURACY.
- FOUND THE BEST K=5 WITH A VALIDATION ACCURACY OF 83.15%.
- RATIONALE: TUNING K OPTIMIZES KNN'S PERFORMANCE BY BALANCING UNDERFITTING (LARGE K) AND OVERFITTING (SMALL K).

5. KNN MODEL TRAINING AND HYPERPARAMETER TUNING

- HYPERPARAMETER TUNING:
- TESTED K VALUES FROM 1 TO 20 USING THE VALIDATION SET.
- FOR EACH K, TRAINED A KNN MODEL, PREDICTED ON X_VAL_SCALED, AND CALCULATED ACCURACY.
- FOUND THE BEST K=5 WITH A VALIDATION ACCURACY OF 83.15%.
- RATIONALE: TUNING K OPTIMIZES KNN’S PERFORMANCE BY BALANCING UNDERFITTING (LARGE K) AND OVERFITTING (SMALL K).

5. KNN MODEL TRAINING AND CROSS-VALIDATION

- CROSS-VALIDATION:
- PERFORMED 5-FOLD CROSS-VALIDATION ON THE COMBINED TRAINING DATA (X_TEMP_SCALED, Y_TEMP).
- RESULTS: CV ACCURACIES = [0.769, 0.769, 0.852, 0.810, 0.824], AVERAGE CV ACCURACY = 80.49%, STANDARD DEVIATION = 3.2%.
- RATIONALE: CROSS-VALIDATION PROVIDES A MORE ROBUST ESTIMATE OF MODEL PERFORMANCE BY TRAINING/TESTING ON DIFFERENT DATA SUBSETS, REDUCING THE RISK OF OVERFITTING TO A SINGLE VALIDATION SPLIT.

6. ADDRESSING CLASS IMBALANCE WITH SMOTE

- SMOTE OVERSAMPLING:
- APPLIED SMOTE TO THE TRAINING DATA TO BALANCE THE CLASSES (SURVIVED=0 VS. SURVIVED=1).
- RETRAINED KNN WITH K=5 ON THE SMOTE-BALANCED DATA.
- RATIONALE: THE DATASET IS IMBALANCED (577 NOT SURVIVED VS. 314 SURVIVED IN THE FULL DATA). SMOTE CREATES SYNTHETIC SAMPLES FOR THE MINORITY CLASS (SURVIVED=1) TO REDUCE BIAS TOWARD THE MAJORITY CLASS, POTENTIALLY IMPROVING RECALL FOR SURVIVED.

Model Evaluaiton

for test set

Without SMOTE:

- Evaluated the best KNN (k=5) on X_test_scaled.
- Results:
  - Test Accuracy: 81.56%.
  - Classification Report:
    - Not Survived: Precision 0.85, Recall 0.85, F1-Score 0.85 (Support 110).
    - Survived: Precision 0.76, Recall 0.75, F1-Score 0.76 (Support 69).
    - Overall Accuracy: 81.56%.

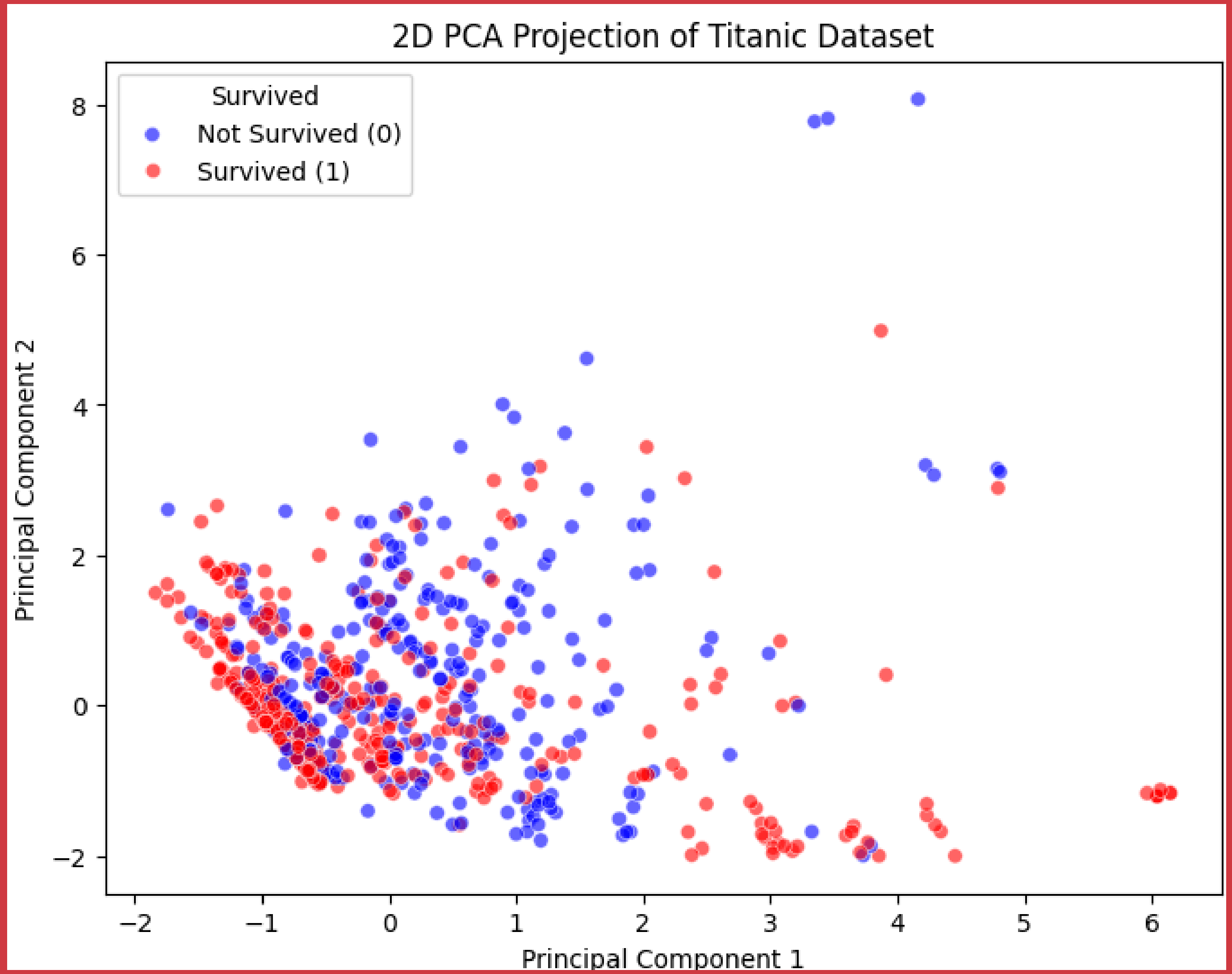
with Smote

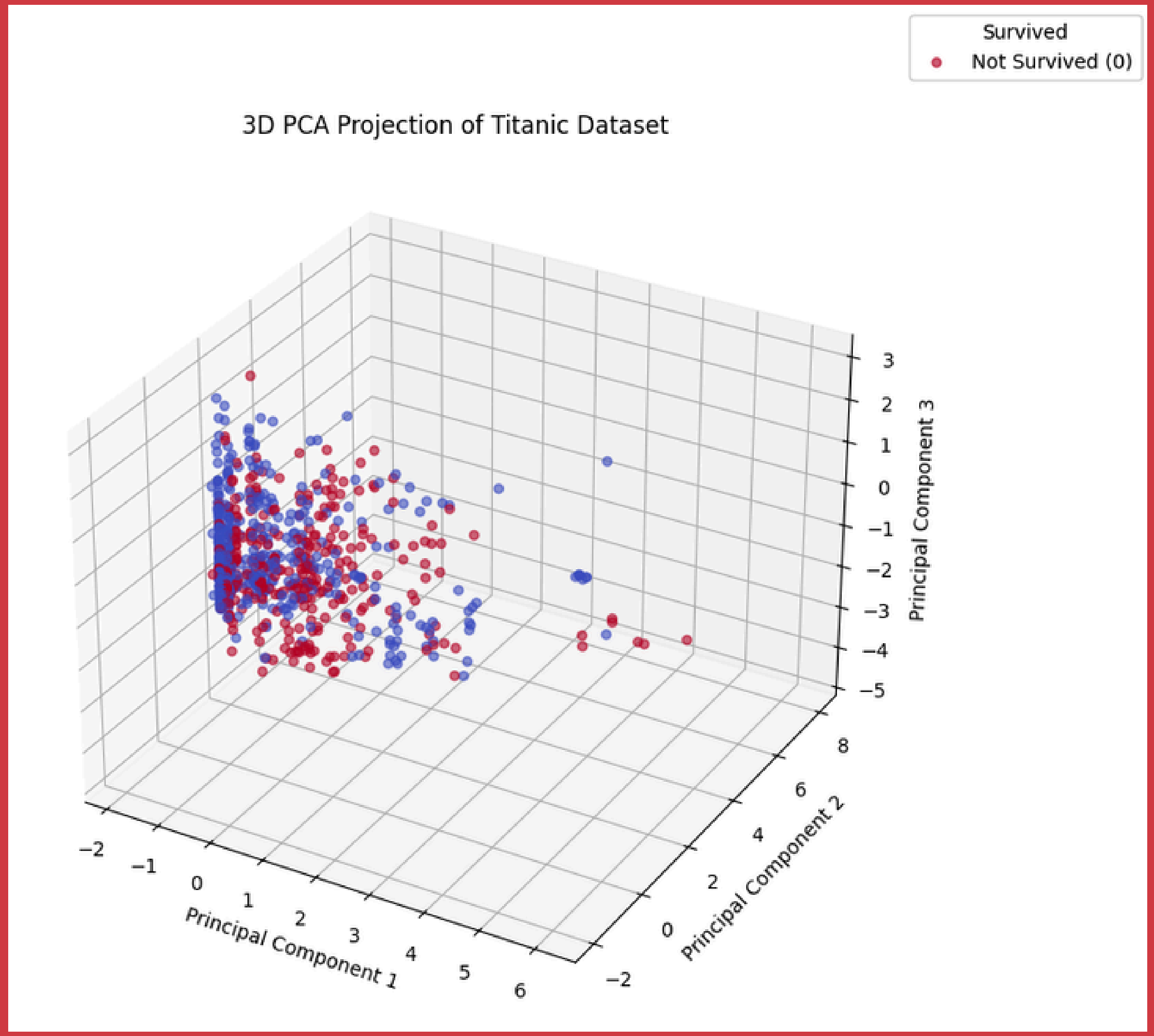
- Test Accuracy: 79.89%.
- Classification Report:
  - Not Survived: Precision 0.86, Recall 0.81, F1-Score 0.83 (Support 110).
  - Survived: Precision 0.72, Recall 0.78, F1-Score 0.75 (Support 69).
  - Overall Accuracy: 79.89%.

Conclusion:

Evaluating on the test set gives an unbiased estimate of performance on unseen data. SMOTE slightly lowered accuracy but improved recall for Survived (from 75% to 78%), addressing the class imbalance.

8. VISUALIZATION AND INTERPRETATION





- COMBINED X_TEMP_SCALED AND X_TEST_SCALED FOR VISUALIZATION.
- APPLIED PCA TO REDUCE FEATURES TO 2D AND 3D:
  - 2D PCA: EXPLAINED VARIANCE RATIO = [0.314, 0.256] (57% VARIANCE).
  - 3D PCA: EXPLAINED VARIANCE RATIO = [0.314, 0.256, 0.140] (71% VARIANCE).

8. VISUALIZATION AND INTERPRETATION

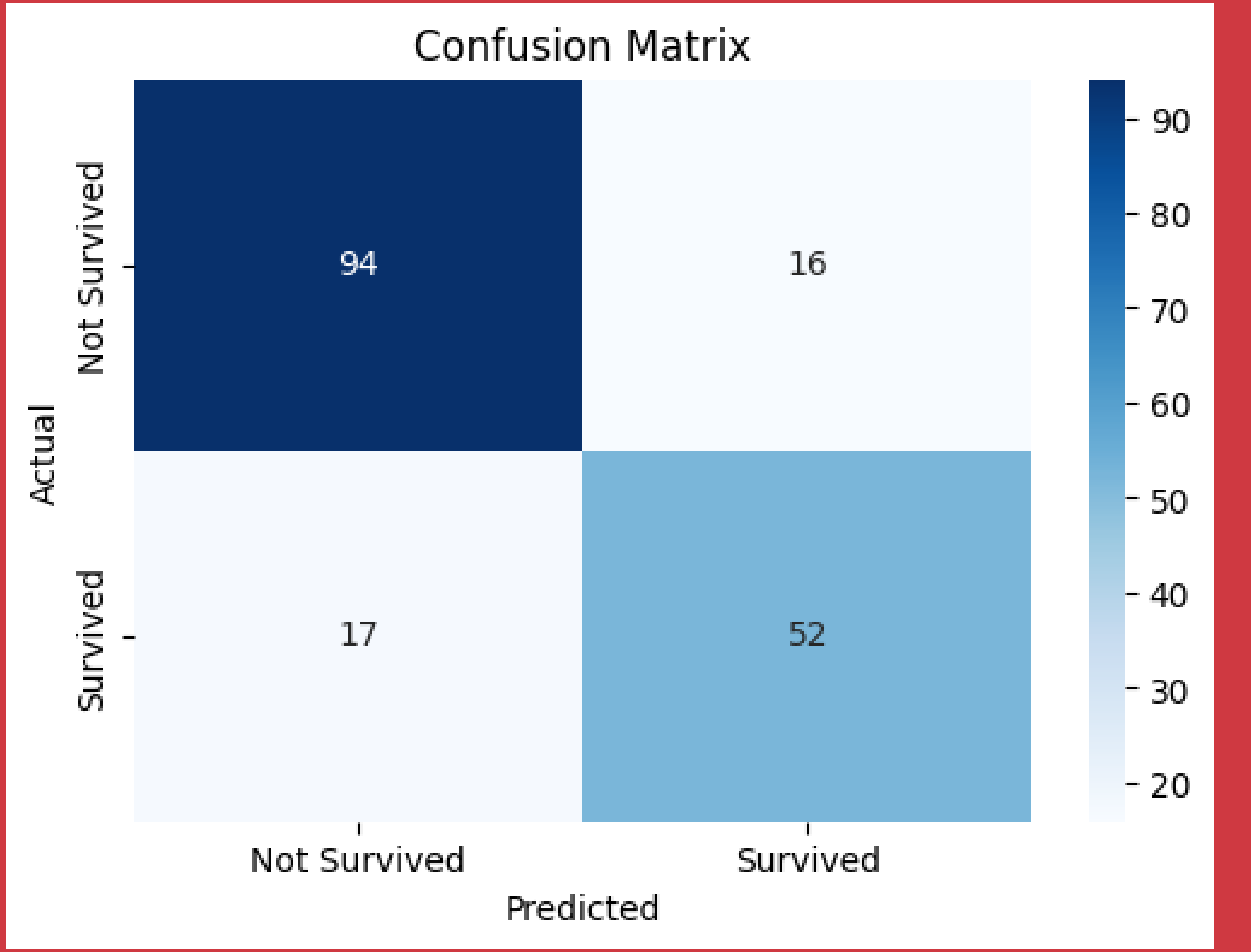
The confusion matrix is structured as follows:*

True Negatives (TN): 94 passengers were correctly predicted as "Not Survived."

False Positives (FP): 16 passengers were incorrectly predicted as "Survived" when they did not survive.

False Negatives (FN): 17 passengers were incorrectly predicted as "Not Survived" when they actually survived.

True Positives (TP): 52 passengers were correctly predicted as "Survived."



8. VISUALIZATION AND INTERPRETATION

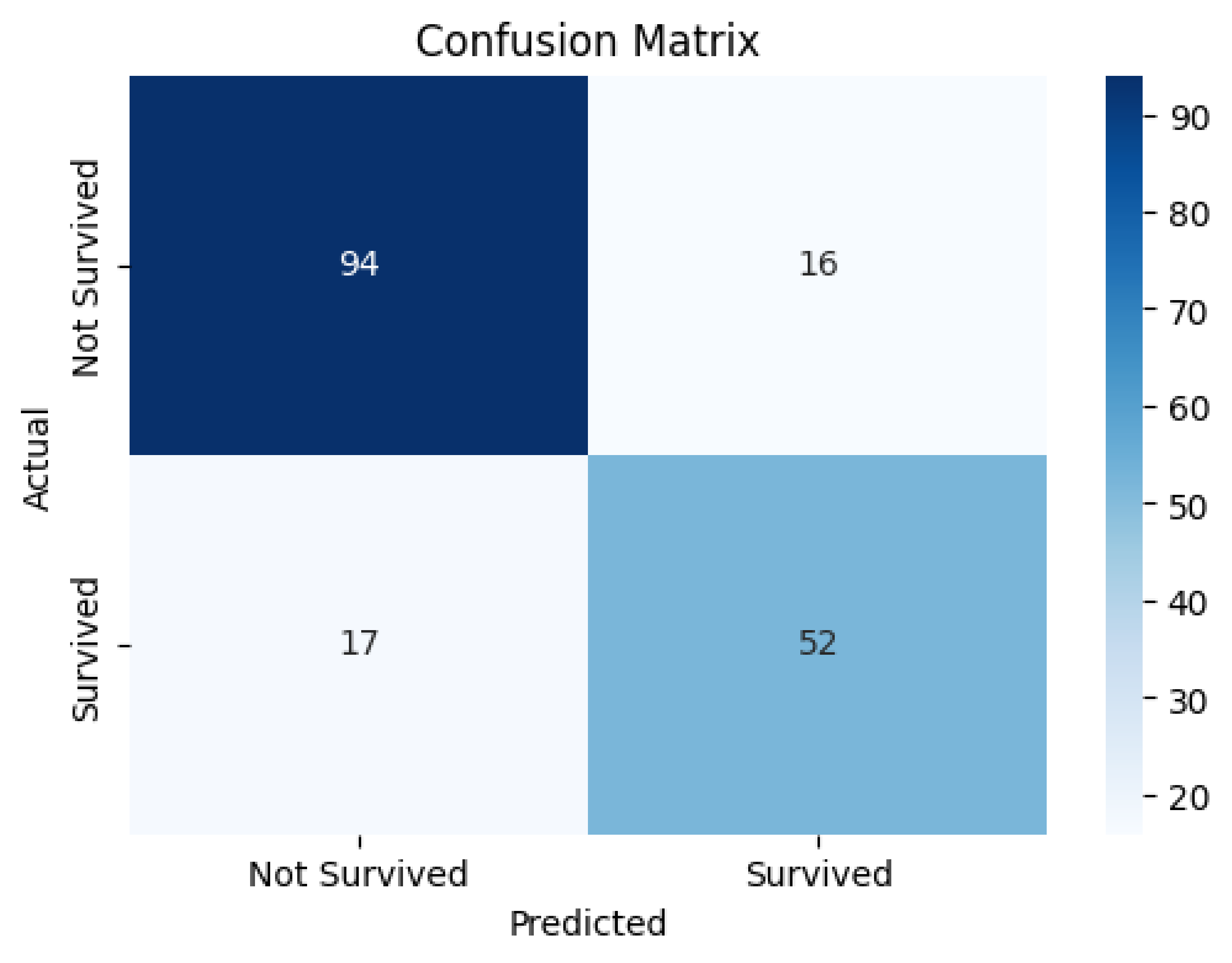
Metrics Analysis (Supporting the Patterns):

Accuracy (0.8156): The overall accuracy is 81.56%, which is solid and reflects the model's good performance, though it's influenced by its stronger prediction of the majority class ("Not Survived").

Precision for Survived (0.7647): Of all passengers predicted to survive, 76.47% actually survived. This indicates reasonable reliability in survival predictions but suggests some room for enhancement.

Recall for Survived (0.7536): Only 75.36% of actual survivors were correctly identified, highlighting a moderate FN rate and the model's challenge with this class, though it's better than a lower recall would indicate.

F1-Score for Survived (0.7591): The F1-score balances precision and recall, and the value for "Survived" (0.7591) reflects a decent balance, indicating the model performs reasonably well in identifying survivors despite the class imbalance.



8. VISUALIZATION AND INTERPRETATION

Key Findings

Model Performance: The KNN model achieved a Test Accuracy of 81.56% without SMOTE and 79.89% with SMOTE. SMOTE improved recall for Survived (75% to 78%) but slightly lowered overall accuracy due to more false positives.

Class Separation: PCA plots revealed significant overlap between Survived=0 and Survived=1, explaining KNN’s performance limit. The overlap suggests KNN struggles in dense regions, while some separation along PC2 indicates predictive features.

Class Imbalance: The dataset’s imbalance (more Not Survived) biased KNN toward Not Survived, but SMOTE helped by improving Survived recall.