Questions to be answered in the pdf report

- **What is the name of your data?**

Titanic - Machine Learning from Disaster

- **The source of the data (which database)?**

The dataset is publicly available on **Kaggle** and originally from the **British Board of Trade's Inquiry Report**.

- **Link to the original data:**

https://www.kaggle.com/competitions/titanic/data

- **Explain the data in words:**

This dataset contains information on passengers aboard the RMS Titanic. Each row represents one passenger and includes features such as age, sex, class, fare, and whether or not they survived the disaster. It is a classic binary classification problem used to predict survival.

- **Is it a regression or classification problem?**

Classification problem — Predicting if a passenger survived (1) or not (0).

- **How many attributes?**

After preprocessing: **10 attributes (features)**, excluding the target variable.

- **How many samples?**

891 samples in the original dataset.

- **What are the properties of the data? (statistics)**

Descriptive statistics such as:

- Age: mean 29.7 years
- Fare: mean 32.2, range from 0 to 512
- Balanced gender distribution: 65% male, 35% female
- Survival rate: 38%

- **Are there any missing data? How did you fill in the missing values?**

Yes:

- Age: filled with median age.
- Embarked: filled with the most frequent value.
- Cabin: dropped due to excessive missing values.

• Visualize the data:

Visualizations included:

- Missing values heatmap
- Heatmaps for classification reports
- Accuracy comparison bar chart

• Did you normalize or standardize any of your data? Why?

Yes:

- Standardization was applied to features before training **SVM** and **KNN**, because both algorithms are sensitive to feature scale and use distance-based calculations.

• What type of preprocessing did you apply to your data? List everything and explain why.

1. Filled missing values (Age, Embarked)
2. Dropped uninformative columns (Cabin, Name, Ticket)
3. Encoded categorical features (Sex, Embarked) using one-hot encoding
4. Scaled features for SVM and KNN using StandardScaler
5. Split data into train and test sets (80/20)

• How did you divide the train and test data? What are the proportions?

Used 80% for training and 20% for testing with train_test_split().

• Apply all the machine learning models and report results. What is the best/worst model? Why?

Applied:

- Logistic Regression
- Decision Tree
- Random Forest
- Naive Bayes
- Support Vector Machine (SVM)
- K-Nearest Neighbors (KNN)

Best Model: **Random Forest** — highest accuracy due to ensemble voting.
Worst Model: **Naive Bayes** — assumed feature independence and had weaker predictive power.

• The accuracy of all models using tables and figures:

(Table already generated as model_accuracy_summary.csv, bar chart saved as model_accuracy_comparison.png)

• Advanced visual presentation:

Used seaborn to plot:

- Heatmaps of classification reports
- Confusion matrices
- Model accuracy comparison

- **Explain in 20 lines , font size 20, Font : Times New Roman, What is the reason you picked up this data? What is the importance of your data in reality, and what is the importance of your best-performing model? Is there any insight you could share from the data and the model?**

I chose the Titanic dataset because it is a well-known, structured dataset that combines both numerical and categorical features, making it ideal for testing a wide range of machine learning algorithms. The problem is simple and intuitive: predicting who survived the Titanic disaster. This task reflects a classic binary classification problem and offers clear evaluation metrics for comparing models.

The importance of this dataset lies in its historical and human-centered nature. It allows us to explore how different attributes — like age, gender, ticket class, and family connections — influenced survival chances in a real-world disaster. This kind of analysis has modern implications in designing fairer and more effective emergency protocols and systems.

Among the models I trained, **Random Forest** performed the best, likely due to its ability to handle mixed data types and capture nonlinear relationships. It aggregates decisions from many trees, reducing overfitting and improving generalization.

On the other hand, **Naive Bayes** had the weakest performance. This was expected because it assumes feature independence, which is not realistic in the Titanic dataset (e.g., Sex and Pclass are highly related).

One key insight I found is that **females and children in higher classes had the highest survival rates**, while adult males in lower classes had the lowest. This confirms historical rescue priorities.

Overall, the project shows how machine learning can extract meaningful patterns from real-world data, and why model selection and data preprocessing play a critical role in achieving accurate results.

- **Link to your code and data:**
    - All code is in the main folder: /main.ipynb
    - Original data is in /Data/original data
    - Train-test split data in /Data/2.Preprocessed data/ and /3.targets/
    - Model outputs in /Data/Results/
- **Folder Data/Results includes:**
    - Test set predictions from each model (CSV files, no features)
    - Named: predictions_<model_name>.csv