### 🎯 Task: Build, Optimize, and Compare Models for Titanic Survival Prediction

### 🧩 Objective:

Train and compare multiple machine learning models on the Titanic dataset:

* **Decision Tree Classifier**
* **Logistic Regression** (with class balancing techniques)
* Use **cross-validation** and select the best model based on a chosen **evaluation metric**

### 🪜 Step-by-Step Instructions:

#### 🔹 1. Load and Prepare the Dataset

* Load Titanic dataset (e.g., from seaborn or OpenML)
* Select relevant features:  
  ['pclass', 'sex', 'age', 'fare', 'embarked'] as features  
  ['survived'] as the target
* Handle missing values
* Encode categorical features (sex, embarked, pclass)

#### 🔹 2. Choose an Evaluation Metric

📌 **Your task:**  
Select a performance metric to **optimize for**. Choose **one** based on the business context:

* accuracy – for balanced classes
* precision, recall, or f1 – for imbalanced problems
* balanced\_accuracy – alternative to accuracy under imbalance

#### 🔹 3. Train and Tune a Decision Tree (with Grid Search + CV)

* Use GridSearchCV with 5-fold CV
* Tune:
  + max\_depth: [3, 5, 10, None]
  + min\_samples\_split: [2, 5, 10]
  + criterion: ['gini', 'entropy']
* Use a Pipeline with preprocessing
* Optimize for your chosen metric

#### 🔹 4. Train Logistic Regression – with Class Weights

* Use LogisticRegression(solver='liblinear')
* Start with class\_weight='balanced'

📌 **Then manually assign weights**, e.g.:

LogisticRegression(class\_weight={0: 0.4, 1: 0.6})



📌 **Your task:**

* Try several manual class weight combinations
* Use cross-validation to compare their scores

#### 🔹 5. Try Oversampling with Logistic Regression

Use **oversampling techniques** to handle imbalance in the training data:

* Apply SMOTENC (for mixed categorical + numerical features)
* Apply ADASYN (Adaptive Synthetic Sampling)

📌 **Your task:**

* Use imblearn.pipeline.Pipeline to combine preprocessing + oversampling + model
* Use SMOTENC for categorical columns like sex, embarked, pclass
* Use ADASYN as an alternative

from imblearn.over\_sampling import SMOTENC, ADASYN

from imblearn.pipeline import Pipeline as ImbPipeline



#### 🔹 6. Compare All Models

📊 Compare the following models:

* Best **Decision Tree**
* **Logistic Regression** with balanced class weight
* **Logistic Regression** with manual class weights
* **Logistic Regression** with SMOTENC
* **Logistic Regression** with ADASYN

📌 **Your task:**

* Evaluate all models on the **same test set**
* Report scores using your chosen metric
* Optionally: Plot confusion matrices or ROC curves
* **Select the best model**, and justify your choice