Predicting Titanic Survival: A Comparative Analysis of ML Algorithms

Deadline: 5/10/2024

1 Objective

The objective of this assignment is to analyze the Titanic dataset and develop predictive models using three different machine learning algorithms: K-Nearest Neighbors (KNN), Naive Bayes, and Support Vector Machine (SVM).You will compare the performance of these algorithms in predicting the survival status of passengers aboard the Titanic.

Please note that the team consists of four to five members.

2 Dataset

The dataset provided for this assignment is the Titanic dataset, which contains information about the passengers aboard the Titanic, including their demographic details, ticket information, cabin class, survival status, etc.

Variable	Definition	Key
survival	Survival (Label)	0 = No, 1 = Yes
pclass	Ticket class	1 = 1st, 2 = 2nd, 3 = 3rd
sex	Sex	
Age	Age in years	
sibsp	# of siblings / spouses aboard the Titanic	
parch	# of parents / children aboard the Titanic	
ticket	Ticket number	
fare	Passenger fare	
cabin	Cabin number	

embarked	Port of Embarkation	C = Cherbourg, Q = Queenstown, S = Southampton

Table 1: Variable Definitions.

3 Tasks

In this section, we outline the specific tasks that need to be performed as part of this assignment. These tasks are designed to guide you through the process of analyzing the Titanic dataset and developing predictive models using various machine learning algorithms.

3.1 Data Exploration and Preprocessing

1. Perform exploratory data analysis (EDA) to gain insights into the dataset. Summarize the key characteristics and distributions of the variables.

2. Handle missing values, outliers, and categorical variables appropriately.

3.2 K-Nearest Neighbors (KNN)

1. Implement the KNN algorithm using a suitable library (*e.g.*, scikit-learn) with a range of *k* values.

2. Train the model on the training set and evaluate its performance using appropriate evaluation metrics (*e.g.*, accuracy, precision, recall, F1-score).

3. Experiment with different distance metrics and discuss their impact on the model's performance.

4. Select the best *k* value based on performance metrics and apply the model to the testing set for predictions.

3.3 Naive Bayes

1. Implement the Naive Bayes algorithm using a suitable library (*e.g.*, scikitlearn).

2. Train the model on the training set and evaluate its performance using appropriate evaluation metrics.

3. Apply the trained model to the testing set for predictions.

3.4 Support Vector Machine (SVM)

1. Implement the SVM algorithm using a suitable library (e.g., scikit-learn) with various hyperparameters.
2. Train the model on the training set and evaluate its performance using appropriate evaluation metrics (e.g., accuracy, precision, recall, F1-score).
3. Experiment with different kernel functions (e.g., linear, polynomial, radial basis function) and regularization parameters to assess their impact on the model's performance.
4. Tune the hyperparameters (e.g., C, gamma) using techniques like grid search or randomized search to optimize the model's performance.
5. Select the best combination of hyperparameters based on performance metrics and apply the tuned model to the testing set for predictions.

4 Comparative Analysis

1. Compare the performance of the three algorithms (KNN, Naive Bayes, and SVM) based on the evaluation metrics obtained.
2. Discuss the strengths and weaknesses of each algorithm in the context of the Titanic dataset.
3. Provide insights into which algorithm performs better in predicting the survival status of passengers.

5 Deliverables

Provide a printed Jupyter notebook containing the following information:

1. A comprehensive report documenting the analysis, implementation, and evaluation of each algorithm.
2. Visualizations, tables, confusion metrics, and plots to support the findings.
3. A discussion section summarizing the comparative analysis and the insights gained.

6 Bonus:

Artificial Neural Networks (ANN)

1. Implement an ANN model using a suitable library (*e.g.*, scikit-learn).
2. Design the architecture of the neural network, including the number of layers, activation functions, and optimization algorithm.
3. Train the model on the training set and evaluate its performance using appropriate evaluation metrics.
4. Apply the trained model to the testing set for predictions.

Use Markdown cells to provide explanations and discussions.