HyperionDev

Supervised Learning – Decision Trees

Model Answer Approach

Visit our website

Auto-graded task

In this task, we start by loading the Titanic dataset using the `pd.read_csv()` function from pandas, which converts the CSV file into a DataFrame. Next, we preprocess the data by selecting relevant variables, handling missing values, and converting categorical data into a numerical format using functions like `drop()`, `get_dummies()`, and `fillna()`.

We then build a decision tree using `DecisionTreeClassifier` from scikit-learn. The data is split into training, development, and test sets using t`rain_test_spli`t to train the model, tune its hyperparameters, and evaluate its performance on unseen data.

The decision tree is trained using different values of `max_depth`, which controls the tree's complexity and helps prevent overfitting. We visualise the tree and evaluate model performance on the development set using accuracy as a metric.

To tune the `max_depth` parameter, we plot training and development accuracies on the same graph. The model with the best accuracy on the development set is selected and tested on the final test data, achieving approximately 79.9% accuracy.

Some potential pitfalls include overfitting, which is controlled by tuning `max_depth` and handling missing values. We used mean imputation for missing ages, but other strategies could also be applied. Furthermore, selecting the right features is crucial for the model's performance, and careful feature selection is important.