



DAILY WORK
REPORT
TR-02

INFOWIZ

18 JUNE 2024

Day 12: Decision Trees

Summary: Today, we delved into decision trees, a versatile machine learning algorithm known for its ability to perform both classification and regression tasks. We explored the foundational concepts, practical implementation using Python with scikit-learn, and applications of decision trees in real-world scenarios.

Key Learnings:

1. Introduction to Decision Trees:

- Defined decision trees as hierarchical structures composed of nodes (decision points) and edges (branches) that lead to outcomes (leaf nodes).
- Discussed their intuitive nature for decision-making, capturing complex relationships in data through a series of binary splits based on feature values.

2. Building Decision Trees:

- Explored the process of constructing decision trees using recursive partitioning:
 - **Splitting Criteria:** Learned about criteria such as Gini impurity and entropy used to evaluate the quality of splits and determine optimal feature selection.
 - **Tree Pruning:** Discussed techniques like pruning to prevent overfitting by simplifying the tree structure, balancing model complexity and predictive accuracy.

3. Implementation in Python:

- Preprocessed the dataset to handle missing values, encode categorical variables, and ensure data readiness for decision tree modeling.
- Utilized scikit-learn's `DecisionTreeClassifier` and `DecisionTreeRegressor` classes to build decision tree models for classification and regression tasks, respectively.
- Visualized decision tree structures to interpret model decisions and understand feature importance in predicting outcomes.

4. Practical Application:

- Applied decision trees to a real-world dataset, such as predicting customer churn or classifying iris flower species based on petal and sepal measurements.
- Explored methods for optimizing decision tree performance, adjusting hyperparameters like maximum depth, minimum samples per leaf, and splitting criteria.

Today's session provided a solid foundation in decision tree algorithms, highlighting their interpretability, flexibility, and robust performance across various machine learning tasks. The hands-on implementation in Python equipped us with practical skills to leverage decision trees for data-driven decision-making and predictive modeling.