Data Mining and Discovery Report

Artificial Neural Networks vs Decision Tree Classifiers and Random Forests

1. Introduction

This report compares Artificial Neural Networks (ANNs), Decision Tree Regression and Random Forests Regression for predicting Carbon Monoxide levels using the UCI Air Quality dataset. It explores model accuracy, complexity, training time, and usability after data preprocessing, highlighting strengths and applications of each method in real-world data mining tasks.

2. Dataset and Preprocessing

We used the Air Quality dataset from the UCI Machine Learning Repository. It contains hourly records of air pollution, including chemical concentrations like CO, NOx, and NMHC.

Preprocessing Steps:

- Removed columns which are not required
- Handling missing values
- Handling duplicate values

Feature Selection:
To improve model performance and reduce complexity, we used a technique called Recursive Feature Elimination (RFE). This method helps select the most important features by repeatedly building a model and removing the least important feature each time.

Train Test Split:
To train and evaluate the model properly, we split the data into training and testing sets. We used 80% of the data for training the model and 20% for testing its performance.

Feature Scaling:
Before training the model, we applied standardization to the input features using StandardScaler. This scaling method transforms the data so that each feature has a mean of 0 and a standard deviation of 1. It helps the model learn better.

3. Models Used

Artificial Neural Networks (ANNs) are a type of machine learning model inspired by the structure and functioning of the human brain.

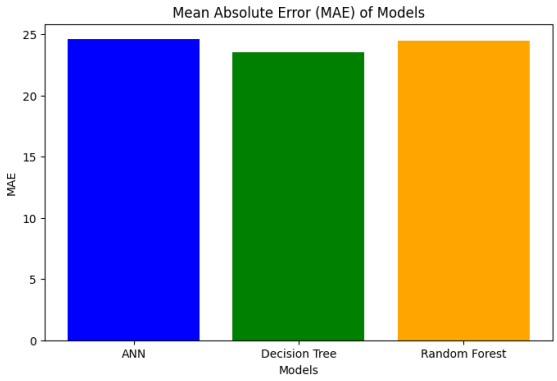
Decision Tree Regression is a type of machine learning model used to predict continuous values (like temperature, prices, etc.). It works by splitting the data into smaller and smaller parts based on rules. At each step, it chooses the best feature and value to divide the data, so the predictions become more accurate.

Random Forest Regression is a machine learning method used to make predictions based on numerical data. It works by combining the results of many individual decision trees to make a final prediction.

4. Comparison

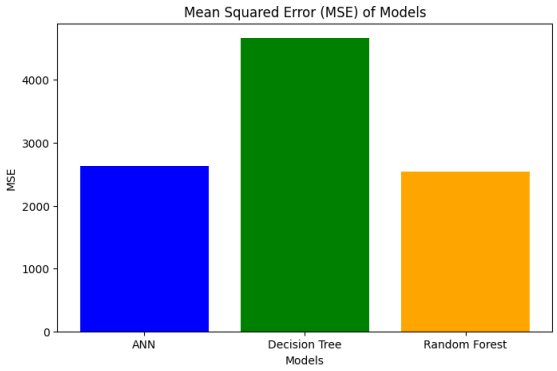
Model	MAE	MSE	RMSE	Loss
ANN	25.28	2633.04	51.31	2633.04
Decision Tree	23.50	4665.66	68.31	4665.66
Random Forest	24.47	2538.15	50.38	2538.15

Table 1: Comparison of Model Metrics

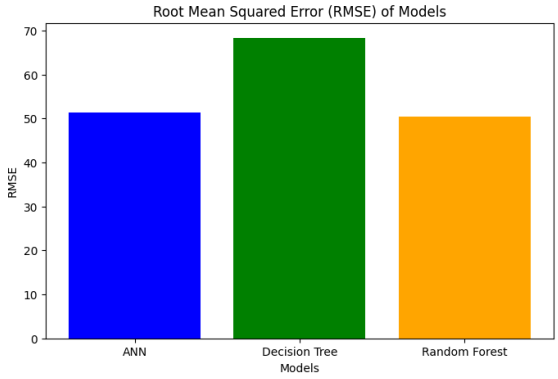


(a) Comparison of MAE

(b) Comparison of loss



(c) Comparison of MSE



(d) Comparison of RMSE

Figure 1: Comparison of evaluation metrics for All Models

5. Conclusion

Best performing model: Random Forest

- It has the **lowest MSE (2538.15)** and **lowest RMSE (50.38)**, indicating better prediction accuracy.
- Although its MAE (24.47) is slightly higher than the Decision Tree's MAE (23.50), the Random Forest's lower MSE and RMSE make it more reliable for minimizing large errors.

Why Random Forest performed well:

- Random Forest is an ensemble method that combines multiple decision trees, reducing overfitting and improving generalization.
- It captures complex relationships in the data better than a single Decision Tree.

References

1. archive.ics.uci.edu. (n.d.). *UCI Machine Learning Repository: Air Quality Data Set*. [online] Available at: https://archive.ics.uci.edu/ml/datasets/Air+Quality

2. Pandas (2024). pandas documentation — pandas 1.0.1 documentation. [online] pandas.pydata.org. Available at: https://pandas.pydata.org/docs/

3. NumPy (2022). NumPy Documentation. [online] numpy.org. Available at: https://numpy.org/doc/.

4. TensorFlow (2019). TensorFlow. [online] TensorFlow. Available at: https://www.tensorflow.org/.

5. scikit-learn. (2025). sklearn.metrics. [online] Available at: https://scikit-learn.org/stable/api/sklearn.metrics.html#regression-metrics.

6. scikit-learn. (2025). DecisionTreeRegressor. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html

7. scikit-learn. (2025). RandomForestRegressor. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html