**Car Sales Forecasting Models**

our target is the car price (a regression problem). Below are the performance metrics for each model:

### ****Model Performance Summary****

#### **1. Overview of Models:**

This report compares the performance of four models used for predicting the target variable. The models evaluated include:

* **Random Forest (RF)**
* **Gradient Boosting (GB)**
* **XGBoost (XGB)**
* **Decision Tree (DT)**

#### **2. Performance Metrics:**

The following metrics were used to evaluate each model:

* **R²**: Indicates the proportion of variance explained by the model.
* **MAE (Mean Absolute Error)**: Measures the average of absolute errors between predicted and actual values.
* **MSE (Mean Squared Error)**: Represents the average of the squared errors.
* **MAPE (Mean Absolute Percentage Error)**: Measures the percentage difference between predicted and actual values.

#### **3. Model Performance Overview:**

**Random Forest (RF)**

* **R²**: **0.9973** (highest)
* **MAE**: **0.0147** (lowest)
* **MAPE**: **0.0588** (lowest)
* **MSE**: **0.0027** (tied for lowest)
* **Training Time**: 105 seconds (longer)
* **Summary**: The Random Forest model provides the best overall accuracy, with the highest R² and lowest error metrics. However, it has longer training time, making it less suitable for situations where speed is crucial.

**Gradient Boosting (GB)**

* **R²**: **0.9973** (same as Random Forest)
* **MAE**: **0.0300**
* **MAPE**: **0.1832**
* **MSE**: **0.0027** (tied for lowest)
* **Training Time**: 28 seconds (faster than RF)
* **Summary**: Gradient Boosting achieves near-identical performance to Random Forest in terms of R² and MSE but has slightly higher MAE and MAPE. It is a better choice when faster training is needed, making it suitable for production environments.

**XGBoost (XGB)**

* **R²**: **0.9954** (slightly lower than RF and GB)
* **MAE**: **0.0341**
* **MAPE**: **0.1999**
* **MSE**: **0.0045**
* **Training Time**: 3 seconds (fastest)
* **Summary**: XGBoost is the fastest model but performs slightly worse than Random Forest and Gradient Boosting, especially in terms of consistency. It may need further fine-tuning for better stability, but it is a good choice for applications requiring rapid training.

**Decision Tree (DT)**

* **R²**: **0.9906** (lowest)
* **MAE**: **0.0279**
* **MAPE**: **0.1753**
* **MSE**: **0.0094**
* **Training Time**: Variable, but typically fast
* **Summary**: Decision Tree underperforms in all metrics, with the lowest R² and highest MSE. It may suffer from overfitting and is not recommended for final deployment, though it can still be useful for interpretability.

#### **4. Model Comparison Table:**

| **Model** | **R²** | **MAE** | **MAPE** | **MSE** | **Verdict** |
| --- | --- | --- | --- | --- | --- |
| **Random Forest** | **0.9973** | **0.0147** | **0.0588** | **0.0027** | **Best Overall** |
| **Gradient Boosting** | **0.9973** | 0.0300 | 0.1832 | **0.0027** | **Best Trade-off** |
| **XGBoost** | 0.9954 | 0.0341 | 0.1999 | 0.0045 | **Needs Tuning** |
| **Decision Tree** | 0.9906 | 0.0279 | 0.1753 | 0.0094 | **Not Recommended** |

#### **5. Final Recommendations:**

* **Random Forest (RF)** is recommended if **accuracy** is the primary goal, and the training time is not a concern.
* **Gradient Boosting (GB)** is the best option if you need a **faster model** with performance similar to Random Forest.
* **XGBoost (XGB)** is suited for scenarios where **speed** is critical, but it may require further tuning to improve consistency.
* **Decision Tree (DT)** is not recommended for final deployment due to its lower performance, though it can be useful for understanding feature importance.