Báo cáo Bài tập lớn

Môn học: Học máy (Machine Learning) - CO3117

Dự đoán giá xe Audi đã qua sử dụng

GVHD: ThS. Võ Thanh Hùng

DANH SÁCH THÀNH VIÊN

Lê Phương Vũ	2313954
Nguyễn Thanh Lộc	2311958
Đặng Quốc Bảo	2210200
Nguyễn Trọng Thắng	1915244

Nội dung báo cáo

1. Thành viên và Phân công (Workload)

STT	Họ và Tên	Nhiệm vụ chính
1	Lê Phương Vũ	SVM, Tuning Hyperparams
2	Nguyễn Thanh Lộc	Tiền xử lý, Hồi quy tuyến tính
3	Đặng Quốc Bảo	Tiền xử lý, Random Forest
4	Nguyễn Trọng Thắng	MLP, Voting, So sánh tổng hợp

Bảng: Bảng phân công công việc

2.1 Bối cảnh và Vấn đề

Bối cảnh:

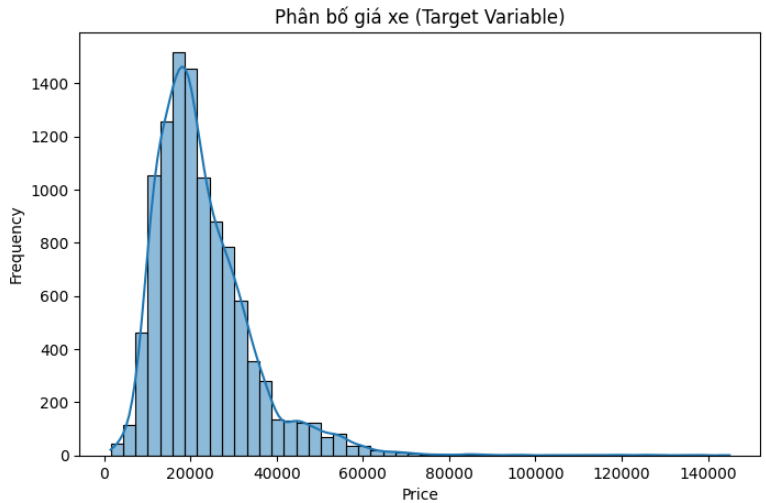
- Thị trường xe cũ (Used Car) tăng trưởng mạnh mẽ.
- Việc định giá xe thường dựa trên cảm tính hoặc kinh nghiệm chủ quan.
- Dữ liệu lớn (Big Data) cho phép định lượng hóa giá trị xe dựa trên thông số kỹ thuật.

Mục tiêu đề tài:

- Xây dựng mô hình Học máy để **dự đoán giá xe Audi cũ** (biến mục tiêu: `price`).
- So sánh hiệu quả giữa các thuật toán cơ bản (Linear, SVM) và nâng cao (Random Forest, MLP, Voting).

3.1 Giới thiệu bộ dữ liệu (Dataset)

- **Nguồn:** Kaggle (Audi Used Car Dataset).
- **Kích thước:** 10,668 mẫu.
- **Input (X):**
  - Số: `year`, `mileage`, `tax`, `mpg`, `engineSize`.
  - Phân loại: `model`, `transmission`, `fuelType`.
- **Output (y):** `price` (Giá xe - £).



Hình: Phân phối giá xe Audi

3.2 Tiền xử lý dữ liệu (Preprocessing)

Quy trình xử lý dữ liệu chuẩn hóa cho tất cả mô hình:

1. **Làm sạch:** Kiểm tra null, loại bỏ xe có giá trị ngoại lai (Outliers).
2. **Mã hóa (Encoding):**
   - One-Hot Encoding cho các biến phân loại (`model`, `transmission`, `fuelType`).
3. **Chuẩn hóa (Scaling):**
   - Sử dụng `StandardScaler` cho các biến số thực để hỗ trợ SVM và MLP hội tụ tốt hơn.
4. **Chia tập dữ liệu:** Train (80%) - Test (20%) với `random_state=42`.

4.1 Hồi quy tuyến tính & SVM

Linear Regression (Baseline):

- Mô hình đơn giản, dễ giải thích.
- **Kết quả:** $R^2 \approx 0.896$.
- *Nhược điểm:* Không bắt được các quan hệ phi tuyến phức tạp.

Support Vector Machine (SVR):

- Sử dụng Kernel RBF để ánh xạ dữ liệu.
- **Kết quả:** $R^2 \approx 0.851$.
- *Hạn chế:* RMSE cao (4736), nhạy cảm với nhiễu ở các dòng xe giá trị cao.

4.2 Random Forest Regressor

Cấu hình:

- `n_estimators = 100` (Số lượng cây).
- `random_state = 42`.

Ưu điểm:

- Xử lý tốt dữ liệu dạng bảng và mối quan hệ phi tuyến.
- Giảm Overfitting nhờ cơ chế Bagging.

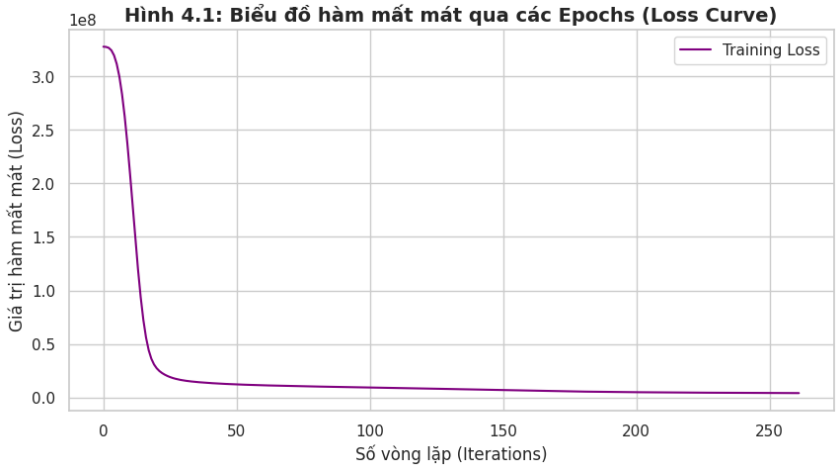
Kết quả thực nghiệm

Đây là mô hình đơn lẻ tốt nhất với $R^2 = 0.9654$ và **MAPE** $\approx$ **7%**.

4.3 Deep Learning (MLP)

Kiến trúc mạng (Multi-layer Perceptron):

- Input Layer: Số chiều đặc trưng sau One-hot.
- Hidden Layers: 2 lớp (100 neuron, 50 neuron) + ReLU.
- Optimizer: Adam ($\alpha = 0.001$).

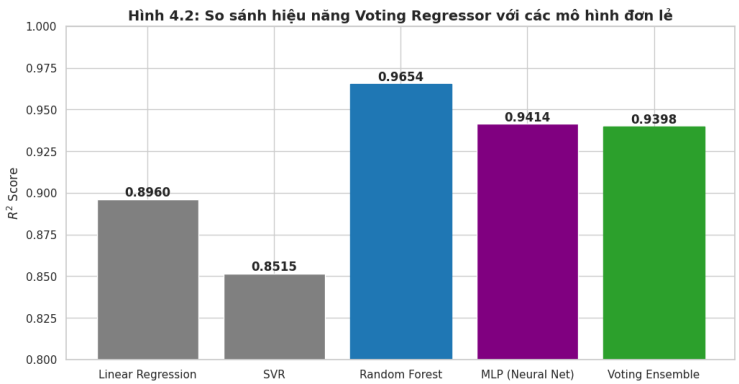


Hình: Biểu đồ Loss Curve (Hội tụ sau 100 epochs)

4.4 Ensemble Learning (Voting Regressor)

Ý tưởng: Kết hợp sức mạnh của 4 "chuyên gia" (Linear, SVM, RF, MLP) để giảm phương sai.

$$\hat{y}_{final} = \frac{1}{N} \sum \hat{y}_{model}$$



Kết quả:

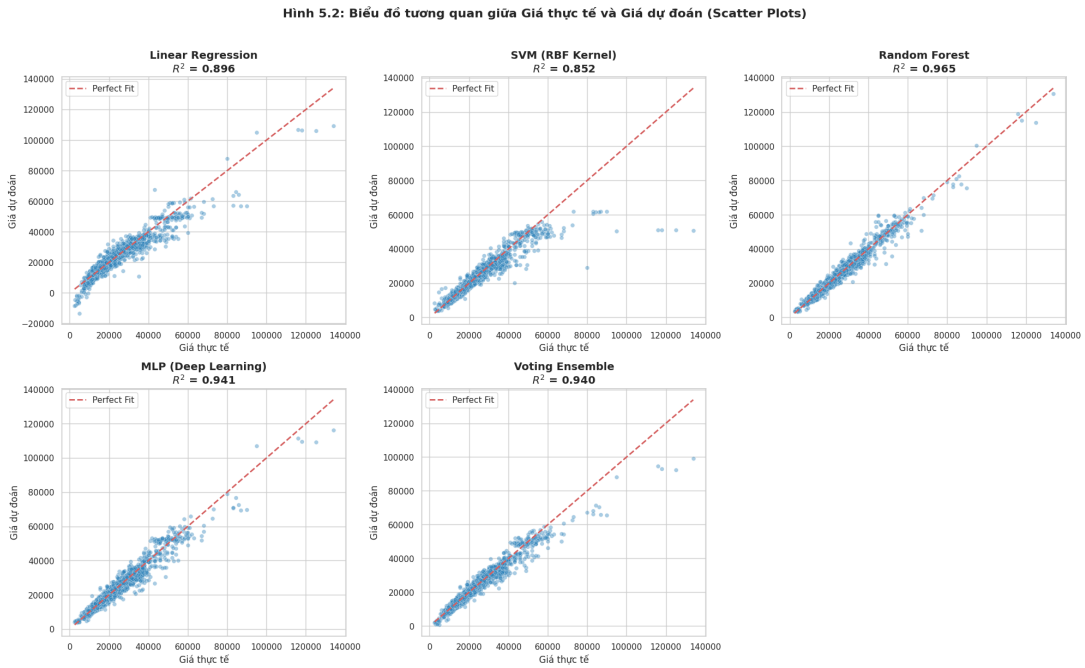
- $R^2 \approx 0.94$.
- Độ ổn định cao hơn mô hình đơn lẻ.

5.1 Bảng tổng hợp kết quả

Mô hình	R2 Score	RMSE	MAPE (%)
Random Forest	**0.9654**	**2285.93**	**7.07%**
Voting Ensemble	0.9487	3016.17	8.14%
MLP (Deep Learning)	0.9414	2975.71	8.89%
Linear Regression	0.8960	3963.67	13.32%
SVM (RBF)	0.8515	4736.74	8.73%

Bảng: So sánh hiệu năng trên tập kiểm thử (Test Set)

5.2 Phân tích trực quan



Hình: Biểu đồ phân tán: Giá trị Thực tế vs. Dự đoán

**Random Forest và Voting cho các điểm dữ liệu bám sát đường chéo nhất.*

Kết luận và Hướng phát triển

Kết luận:

- **Random Forest** là thuật toán hiệu quả nhất cho bài toán này ($R^2 > 96\%$).
- Kỹ thuật **Voting** và **Deep Learning** (MLP) cho kết quả rất tốt, tăng độ ổn định so với các mô hình cơ bản.
- Linear Regression phù hợp làm baseline nhưng hạn chế ở các quan hệ phi tuyến.

Hướng phát triển:

- Mở rộng dữ liệu sang các hãng xe khác (BMW, Mercedes).
- Thử nghiệm các thuật toán Boosting (XGBoost, LightGBM).
- Tối ưu hóa siêu tham số (Hyperparameter Tuning) kỹ hơn.

CẢM ƠN THẦY
VÀ CÁC BẠN ĐÃ LẮNG NGHE!