



# **KHOA HỌC DỮ LIỆU ĐỒ ÁN CUỐI KỲ**

## **DỰ ĐOÁN TRÊN TẬP DỮ LIỆU LAPTOP**

GV: Thầy Trần Trung Kiên

18120379 – Võ Thị Hiếu

18120408 – Trần Ngọc Lan Khanh

# Nội dung

---

- Giới thiệu đồ án
- Thu thập dữ liệu
- Khám phá dữ liệu
- Tiền xử lý dữ liệu
- Xây dựng mô hình
- Đánh giá kết quả
- Tổng kết
- Tham khảo

# Giới thiệu đề án

---

- Câu hỏi: dự đoán giá của laptop dựa trên các thông số kỹ thuật của laptop?
- Input: Các thông số kỹ thuật của laptop
- Output: Giá laptop dự đoán
- Lợi ích: đem lại thông tin cần thiết cho người muốn mua laptop, với các thông số kỹ thuật cho trước thì giá laptop có thể rơi vào phân khúc giá nào.

# Thu thập dữ liệu

- Thu thập dữ liệu trên trang: <https://phongvu.vn/>

The screenshot shows the homepage of PhongVu.vn. The top navigation bar is blue with white text and icons for 'Hệ thống Showroom', 'Tư vấn mua hàng: 1800 6867', 'CSKH: 1800 6865', 'Tin công nghệ', and 'Xây dựng cấu hình'. Below this is a white header with the PhongVu.VN logo, a search bar with the placeholder 'Nhập từ khoá cần tìm', and icons for 'Khuyến mãi', 'Đơn hàng', 'Đăng nhập', and 'Giỏ hàng' (with a red notification badge). The main content area has a red background with a large central banner for 'LỄ HỘI GAMING GEAR' featuring a keyboard, mouse, and headset, with text 'Nâng trang bị tăng trải nghiệm' and 'Giảm đến 50%'. To the left is a vertical menu with categories like 'Điện máy - Điện gia...', 'Laptop & Macbook', 'Tivi - Màn hình TV', 'Điện thoại & Thiết bị...', 'PC - Máy tính đồng bộ', 'Màn hình máy tính', 'Linh kiện máy tính', 'Hi-End Gaming', 'Thiết bị ngoại vi', 'Thiết bị âm thanh', 'Máy ảnh - Máy quay...', 'Thiết bị văn phòng', and 'Thiết bị mạng - An...'. To the right are two smaller promotional tiles: 'BUILD PC' with a computer tower and monitor, and 'TIN CÔNG NGHỆ' with a robot head.

# Thu thập dữ liệu

---

- Dữ liệu bao gồm 782 dòng, có 31 thuộc tính bao gồm:
- URL, Name, Cost, Thương hiệu, Bảo hành, Màu sắc, Series laptop, Part-number, Thế hệ CPU, CPU, Chip đồ họa, RAM, Màn hình, Lưu trữ, Số cổng lưu trữ tối đa, Kiểu khe M.2 hỗ trợ, Cổng xuất hình, Cổng kết nối, Kết nối không dây, Bàn phím, Hệ điều hành, Kích thước, Pin, Khối lượng, Bảo mật, Đèn LED trên máy, Phụ kiện đi kèm, Tính năng, Mic, Ổ đĩa quang, Mô tả bảo hành.
- Chia thành các tập train, tập validate và tập test

# Thu thập dữ liệu

---

- Dữ liệu có các vấn đề như sau:
- Dữ liệu chứa các giá trị thiếu. → loại bỏ các cột có tỷ lệ thiếu hơn 10%
- Một số cột có thể không mang lại ý nghĩa cho việc dự đoán.
- Cột Cost có nhiều giá trị khác nhau → dán nhãn cột Cost, rút gọn số lượng giá trị dự đoán để giảm độ lỗi

# Chuyển đổi dữ liệu thô

---

- Cột Cost có định dạng 'xx.yy.zz\_đồng', thay dấu '.' thành khoảng trắng và bỏ 'đồng' để tách lấy giá trị số nhằm mục đích tính toán.

URL	Name	Cost	Thương hiệu	Bán bao nhiêu	Màu sắc
https://ph	Laptop AS	6.390.000đ	ASUS	24	Vàng
https://ph	Laptop AS	5.990.000đ	ASUS	24	Xanh
https://ph	Laptop AC	6.990.000đ	ACER	12	Đen

# Tiền xử lý dữ liệu

---

Xóa bỏ các thuộc tính:

- ‘Bảo hành’: không có ý nghĩa trong việc dự đoán
- ‘Bảo mật’: Số lượng giá trị thiếu hơn 10%, chỉ có hai giá trị là “Vận tay” và “Khuôn mặt”
- ‘Phụ kiện đi kèm’, ‘Tính năng’, ‘Mic’, ‘Ổ đĩa quang’, ‘Mô tả bảo hành’: giá trị thiếu hơn 10% và không có nhiều ý nghĩa trong việc dự đoán



# Tiền xử lý dữ liệu

---

- Class ColAdderDropper để xóa và gán lại giá trị cho thuộc tính “Name” dựa vào “num\_top\_titles”.
- Numerical: Điền giá trị thiếu bằng giá trị trung bình của thuộc tính
- Categorical:
  - ☐ Điền giá trị thiếu bằng giá trị phổ biến nhất của thuộc tính
  - ☐ Sử dụng One Hot Encoder

# Xây dựng mô hình

---

- Chọn mô hình MLPRegressor, mô hình này được sklearn hỗ trợ
- Với MLP sau khi chọn được mô hình với num\_top\_titles tốt nhất sẽ dùng mô hình đó huấn luyện lại full\_pipeline trên X\_df và y\_sr để ra được mô hình cụ thể cuối cùng
- Các thiết lập cho mô hình:
  - ☐ solver: 'lbfgs'
  - ☐ learning\_rate: 'adaptive'

# Đánh giá kết quả

---

Kết quả model Neural net:

## TESTING

```
In [9]:  test_df = pd.read_csv("test.csv")  
         test_y_sr = test_df["Cost"]  
         test_X_df = test_df.drop("Cost", axis = 1)
```

```
In [10]:  # test_X_df = pd.read_csv("test.csv")  
         pred_y = full_pipeline.predict(test_X_df)  
         test_err = (1 - full_pipeline.score(test_X_df, test_y_sr))*100  
         test_err
```

```
Out[10]: 13.826577392354157
```

# Tổng kết

---

- Thực hành lại quy trình một bài toán Khoa học dữ liệu
- Biết cách sử dụng phương pháp Regression cho bài toán dự đoán.
- Biết được MLPRegressor

# Tham khảo

---

[https://scikit-learn.org/stable/modules/generated/sklearn.neural\\_network.MLPRegressor.html](https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html)

File BT03-TienXuLy\_MoHinhHoa