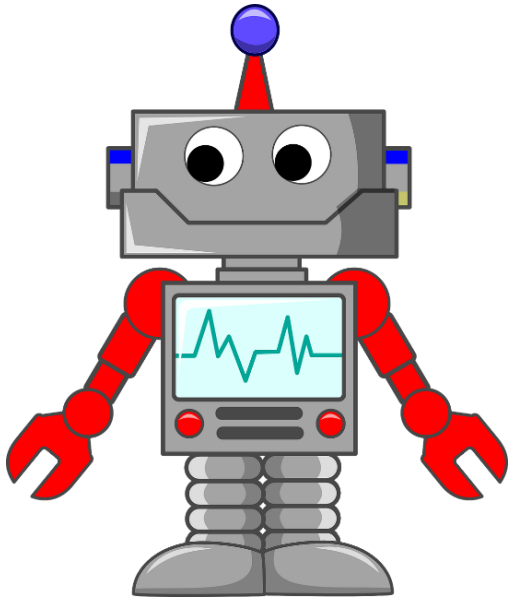
ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

CS116 – LẬP TRÌNH PYTHON CHO MÁY HỌC

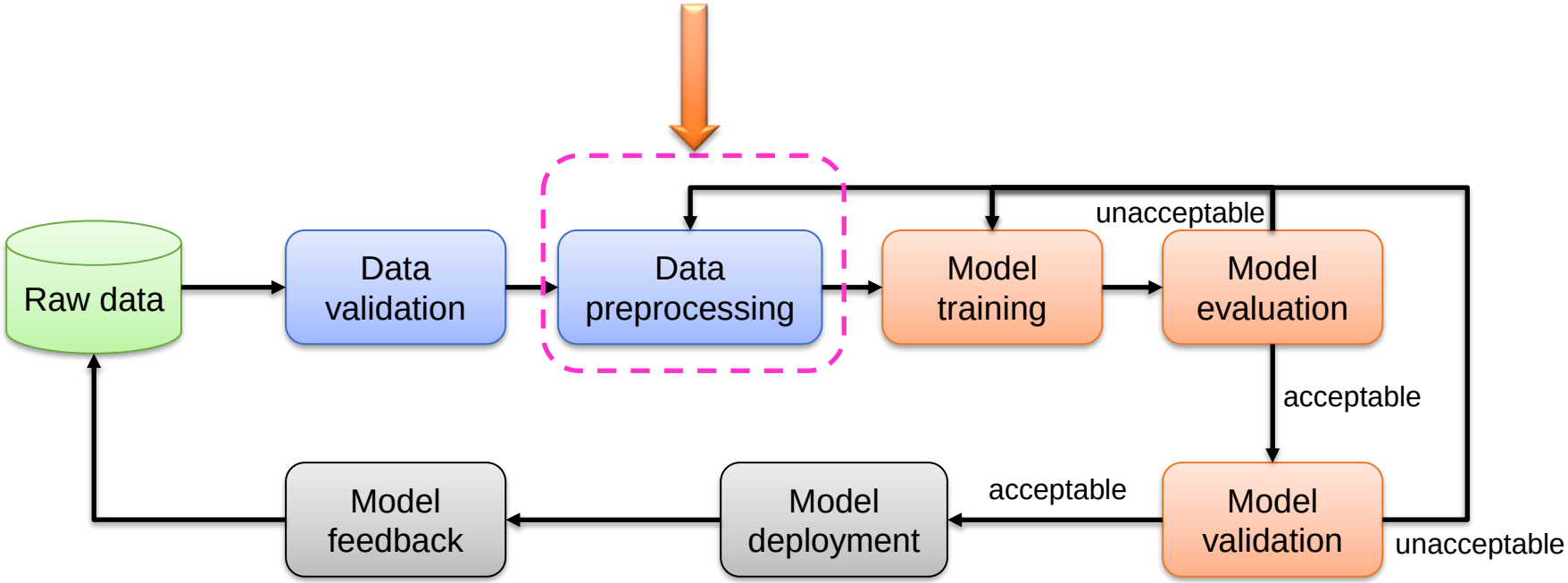
BÀI 04

TIỀN XỬ LÝ DỮ LIỆU



TS. Nguyễn Vinh Tiệp

Vị trí của bài học



NỘI DUNG

1. PHÁT HIỆN & XỬ LÝ DỮ LIỆU BỊ THIẾU
2. PHÁT HIỆN & XỬ LÝ DỮ LIỆU NGOẠI LỆ
3. TẠO ĐẶC TRƯNG MỚI – FEATURE EXTRACTION
4. BIẾN ĐỔI ĐẶC TRƯNG – FEATURE TRANSFORMATION
5. CHỌN LỰA ĐẶC TRƯNG – FEATURE SELECTION

Nhắc lại: Phát hiện dữ liệu bị thiếu

- Trong pandas, ta có thể sử dụng hàm isnull() / isna() để kiểm tra bảng / cột bị thiếu dữ liệu hay không

name	sales
Markus	34000
Edward	42000
William	NaN
Emma	52000
Sofia	NaN

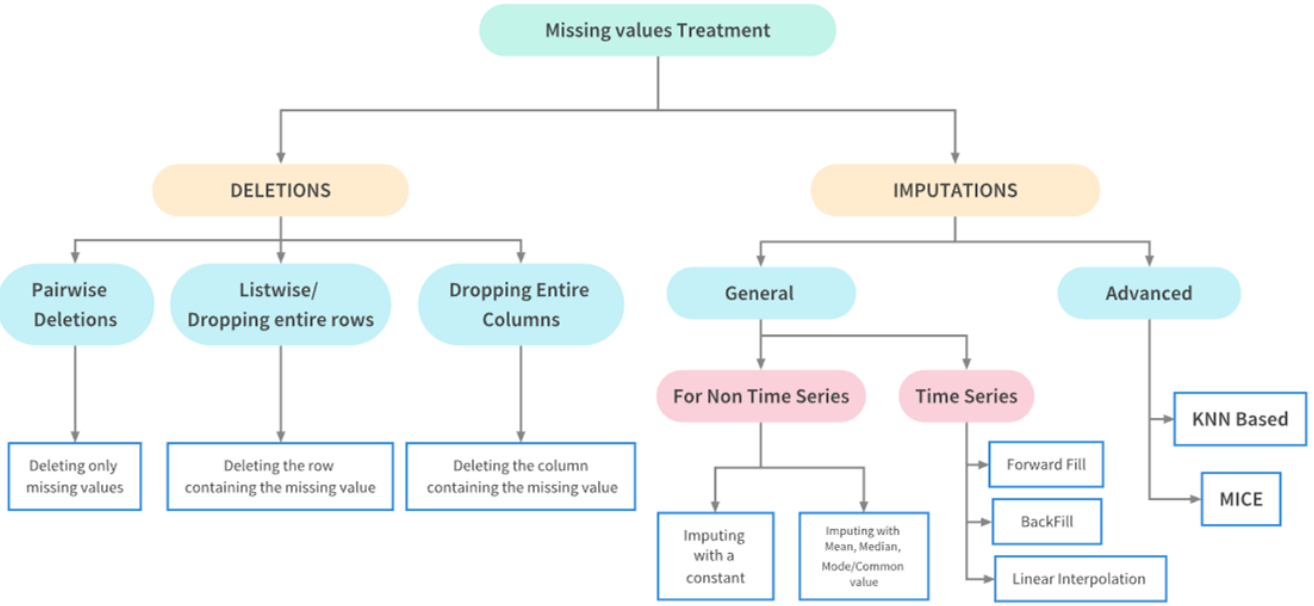
.isnull()

name	sales
FALSE	FALSE
FALSE	FALSE
FALSE	**TRUE**
FALSE	FALSE
FALSE	**TRUE**

Xử lý dữ liệu bị thiếu

Đơn giản, nhưng có thể làm mất dữ liệu quan trọng

- Có ba cách tiếp cận chính:
  - Loại bỏ hàng hoặc cột có tỉ lệ dữ liệu bị thiếu lớn (vd: 50%)
  - Thay thế đơn biến, đa biến, chuỗi thời gian: sklearn-imputation
  - Sử dụng các mô hình máy học để dự đoán
- Cách tiếp cận khác: tạo cột mới chứa thông tin có giá trị bị thiếu



Nguồn: https://www.kaggle.com/code/parulpandey/a-guide-to-handling-missing-values-in-python/notebook

Xử lý dữ liệu bị thiếu – Các PP thay thế

Phương pháp	Cách thực hiện	Đặc điểm
Thay thế đặc trưng đơn biến (mean/median/mode)	Thay thế giá trị còn thiếu bằng giá trị trung bình, trung vị hoặc giá trị xuất hiện thường xuyên nhất của một biến	- Đơn giản - Giá trị khó phản ánh đúng
Thay thế giá trị hằng số	Thay thế giá trị còn thiếu bằng một giá trị không đổi. Ví dụ: “NaN" đối với các biến phân loại	- Đơn giản - Giá trị khó phản ánh đúng
Thay thế bằng phương pháp K-Nearest Neighbors	Thay thế giá trị bị thiếu bằng giá trị trung bình hoặc tổng trọng số của K láng giềng trong không gian đặc trưng	- Chính xác hơn - Có thể tốn kém về chi phí tính toán với tập dữ liệu lớn
Phép nội suy tuyến tính	Thay thế giá trị bị thiếu bằng giá trị được nội suy tuyến tính dựa trên các điểm dữ liệu không bị thiếu lân cận	- Giả sử mối quan hệ tuyến tính giữa các điểm dữ liệu - Có thể không phù hợp với mọi loại dữ liệu
Thay thế bằng phương pháp hồi qui	Ước tính giá trị còn thiếu bằng cách khớp mô hình hồi quy sử dụng các biến khác làm yếu tố dự đoán	- Chính xác hơn - Có thể gây ra hiện tượng đa cộng tuyến và quá khớp nếu các đặc trưng có mối tương quan cao với các đặc trưng khác
Thay thế dựa trên mô hình	Sử dụng mô hình máy học để ước tính các giá trị còn thiếu dựa trên dữ liệu được quan sát	- Chính xác cao - Có thể phức tạp hơn và tốn kém hơn về mặt tính toán

NỘI DUNG

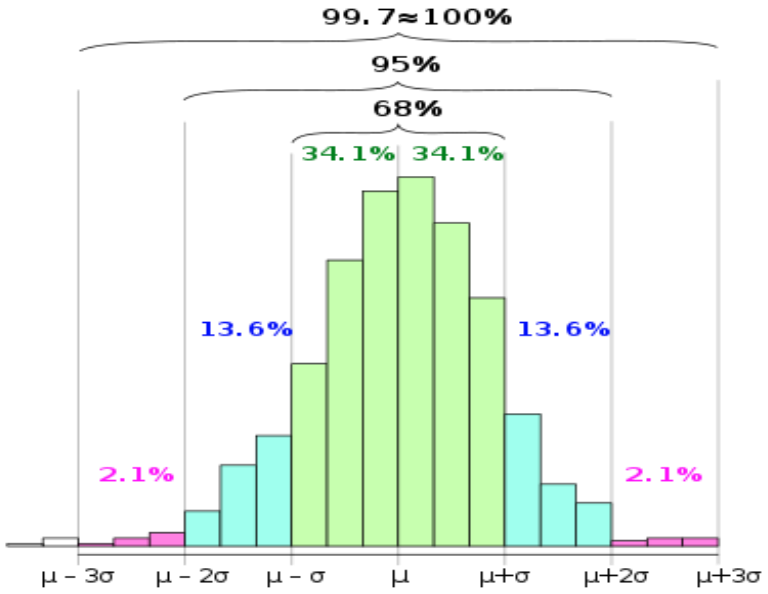
1. PHÁT HIỆN & XỬ LÝ DỮ LIỆU BỊ THIẾU
2. PHÁT HIỆN & XỬ LÝ DỮ LIỆU NGOẠI LỆ
3. TẠO ĐẶC TRƯNG MỚI – FEATURE EXTRACTION
4. BIẾN ĐỔI ĐẶC TRƯNG – FEATURE TRANSFORMATION
5. CHỌN LỰA ĐẶC TRƯNG – FEATURE SELECTION

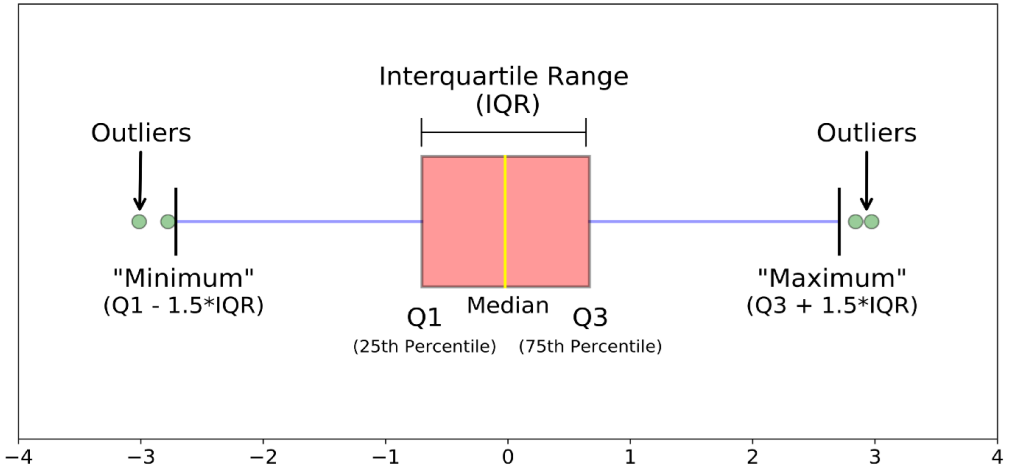
Phát hiện ngoại lệ

- Có hai cách tiếp cận:
  - Phương pháp thống kê (xem nội dung EDA)
  - Tự động phát hiện ngoại lệ
    - Phương pháp Local Outlier Factor
    - Phương pháp Isolation Forest
    - Phương pháp EllipticEnvelope
    - Phương pháp One-class SVM
- Công cụ tự động: CleanLab tìm OOD , phát hiện vấn đề dữ liệu

Nhắc lại: Phương pháp thống kê với EDA

- Phương pháp thống kê:
  - **Phương pháp tính trung bình và độ lệch chuẩn**: để xác định các giá trị ngoại lệ (với dữ liệu dạng Gaussian hoặc tương tự Gaussian)
  - **Phương pháp Interquartile Range (IQR):** để xác định các giá trị ngoại lệ với dữ liệu phân phối không phải Gaussian





Xử lý dữ liệu ngoại lệ

- Tương tự như xử lý dữ liệu bị thiếu:
  - Loại bỏ
  - Thay thế đơn giản
  - Sử dụng mô hình dự đoán

Một số thao tác làm sạch dữ liệu khác

```
# Re indexing
data.set_index(‘column', inplace = True)
data.reset_index(drop = True)

# Re-formatting
data[‘column'] = data[‘column'].astype(int)

# Correcting inconsistent data
data[‘column'].replace(old_value, new_value, inplace = True)
```

Một số thao tác làm sạch dữ liệu khác

```
# Remove duplicates
data.drop_duplicates()

# Drop unnecessary columns
data.drop(columns = [list cols], axis = 1)

# Drop/Filter unnecessary rows
data.drop([0, 1], inplace = True)
data[data[‘column_filter'] == ‘abc']
```

NỘI DUNG

1. PHÁT HIỆN & XỬ LÝ DỮ LIỆU BỊ THIẾU
2. PHÁT HIỆN & XỬ LÝ DỮ LIỆU NGOẠI LỆ
3. TẠO ĐẶC TRƯNG MỚI – FEATURE EXTRACTION
4. BIẾN ĐỔI ĐẶC TRƯNG – FEATURE TRANSFORMATION
5. CHỌN LỰA ĐẶC TRƯNG – FEATURE SELECTION

Tạo đặc trưng mới

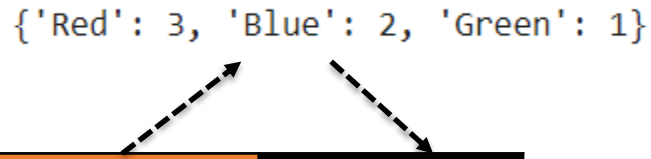
- Biến đổi toán học giữa các đặc trưng đã có

Quảng đường = Vận tốc x Thời gian

Nhân viên	Vận tốc	Thời gian	Q. đường
Nhân viên A	7	8	56
Nhân viên B	9	10	90
Nhân viên C	11	6	66
Nhân viên D	20	4	80
Nhân viên E	10	3	30

Tạo đặc trưng mới (2)

- Đếm tần số xuất hiện



Value	Color	Color_count
100	Red	3
150	Red	3
50	Blue	2
200	Red	3
100	Green	1
100	Blue	2

Tạo đặc trưng mới (3)

- Tổng hợp đặc trưng theo nhiều cột

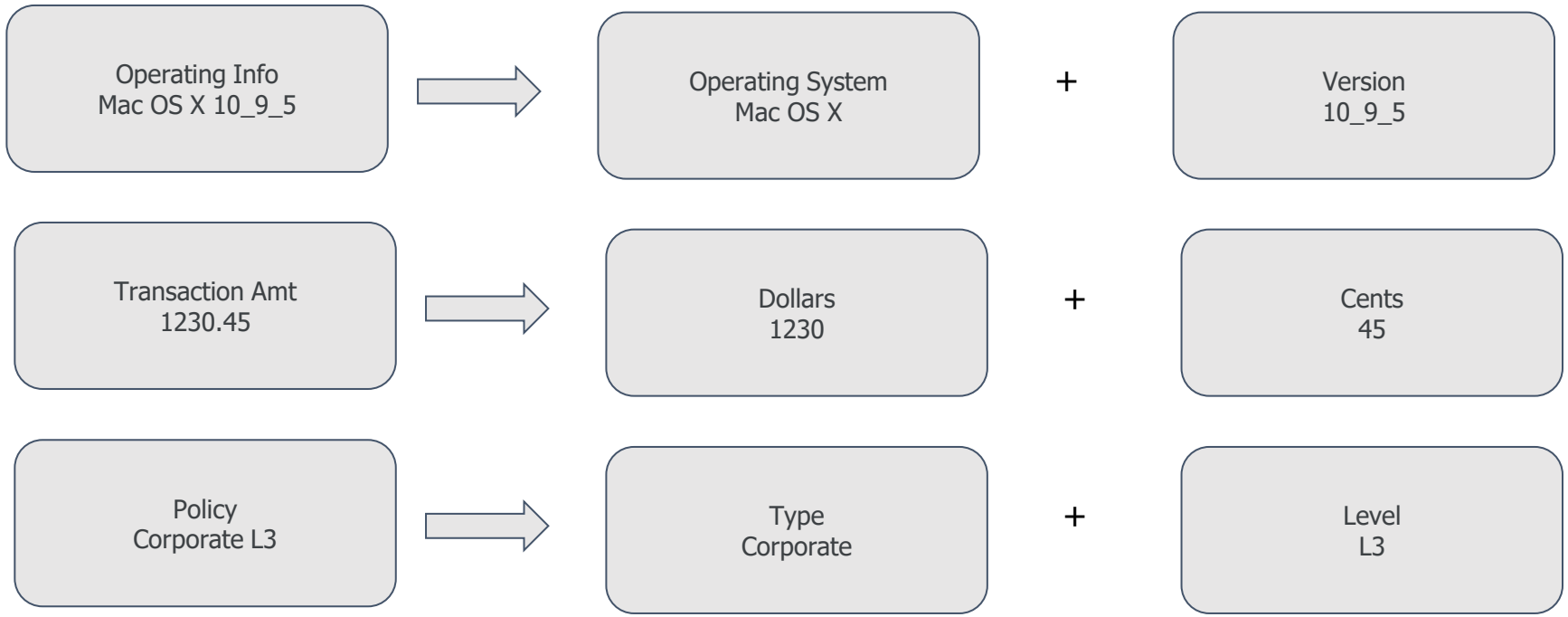
Bus	Car	Motorbike	use_vehicle
0	0	0	0
0	0	1	1
0	0	0	0
1	0	0	1
0	1	1	1

Phân rã đặc trưng

- Một số đặc trưng ở dạng chuỗi phức tạp, nhưng có cấu trúc

➔ Có thể phân rã ra thành nhiều đặc trưng.

- Ví dụ: “06122450” → Năm: 2006, hệ: chính quy, khoa: KHMT, STT: 450



Một số ví dụ khác

Tổng hợp đặc trưng

- Có thể tạo đặc trưng tổng hợp từ nhiều đặc trưng thành phần

Make	Type	Make_Type
Toyota	Sedan	Toyota_Sedan
Audi	Sedan	Audi_Sedan
Honda	Crossover	Honda_Crossover
Honda	Hatchback	Honda_Hrossover
Toyota	SUV	Honda_SUV
Mercedes	Sedan	Honda_Sedan

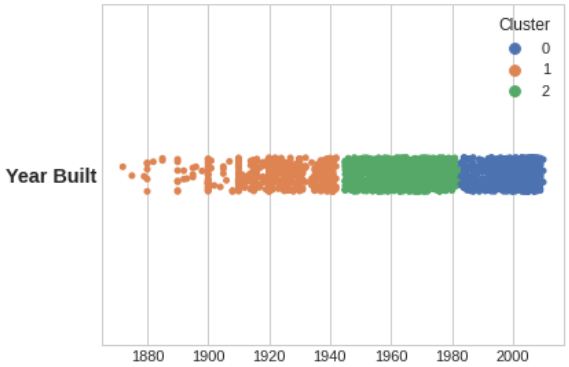
Tổng hợp theo nhóm

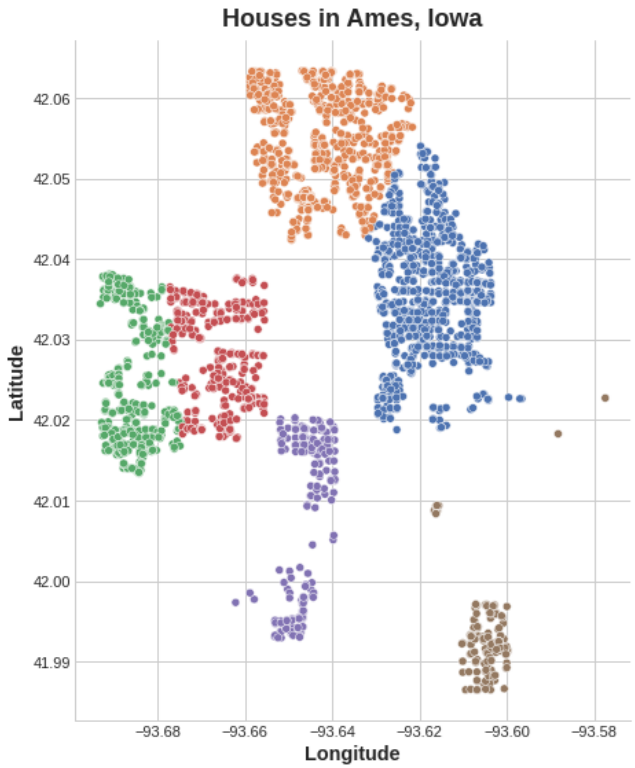
- Tổng hợp thông tin trên nhiều dòng dữ liệu, thực hiện theo nhóm
- Sử dụng groupby, tổng hợp theo “mean”, “max”, “min”…

City	Salary	AvgSalary
Danang	10	12.000000
HCM	20	13.333333
Hanoi	15	15.000000
HCM	8	13.333333
HCM	12	13.333333
Hanoi	15	15.000000
Danang	14	12.000000

Đặc trưng cụm

- Dựa trên phân cụm của một / một số đặc trưng trong dữ liệu





City	Salary	Cluster
Danang	10	1
HCM	20	0
Hanoi	15	0
HCM	8	1
HCM	12	1
Hanoi	15	0
Danang	14	0
Danang	35	2
Hanoi	30	2
HCM	5	1

Cụm: lương trung bình

Cụm: lương cao

Nguồn: https://www.kaggle.com/code/ryanholbrook/clustering-with-k-means

Đặc trưng thành phần chính PCA

- Các thành phần chính của dữ liệu có thể mang lại nhiều thông tin hơn các đặc trưng ban đầu ➔ phân tích thành phần chính

sepal length	sepal width	petal length	petal width	PCA1	PCA2
-0.900681	1.019004	-1.340227	-1.315444	-2.264703	0.480027
-1.143017	-0.131979	-1.340227	-1.315444	-2.080961	-0.674134
-1.385353	0.328414	-1.397064	-1.315444	-2.364229	-0.341908
-1.506521	0.098217	-1.283389	-1.315444	-2.299384	-0.597395
-1.021849	1.249201	-1.340227	-1.315444	-2.389842	0.646835
...	...	...	...	...	...
1.038005	-0.131979	0.819596	1.448832	1.870503	0.386966
0.553333	-1.282963	0.705921	0.922303	1.564580	-0.896687
0.795669	-0.131979	0.819596	1.053935	1.521170	0.269069

Hai thành phần chính từ 4 thành phần ban đầu

NỘI DUNG

1. PHÁT HIỆN & XỬ LÝ DỮ LIỆU BỊ THIẾU
2. PHÁT HIỆN & XỬ LÝ DỮ LIỆU NGOẠI LỆ
3. TẠO ĐẶC TRƯNG MỚI – FEATURE EXTRACTION
4. BIẾN ĐỔI ĐẶC TRƯNG – FEATURE TRANSFORMATION
5. CHỌN LỰA ĐẶC TRƯNG – FEATURE SELECTION

Tại sao cần Biến đổi đặc trưng

- **Yêu cầu loại dữ liệu đầu vào của mô hình:**
  - Nhiều mô hình yêu cầu dữ liệu dạng số, trong khi đặc trưng có thể ở dạng khác nhau
  - Biến đổi dữ liệu từ dạng khác về dạng số → mô hình có thể chạy được

- **Giả định về dữ liệu đầu vào của mô hình:**
  - Nhiều mô hình máy học đặt giả định về phân bố và tỉ lệ (scale) của dữ liệu đầu vào
  - Biến đổi từ dữ liệu gốc về các tỉ lệ / phân bố giả định của mô hình (normalize / scale dữ liệu) → chính xác hơn, học nhanh hơn

Tại sao cần Biến đổi đặc trưng

- **Vấn đề dữ liệu nhiễu:**
  - Các giá trị nhiễu có thể ảnh hưởng lớn đến hiệu quả mô hình
  - Biến đổi log transform, robust scaler → giảm sự ảnh hưởng dữ liệu nhiễu

- **Vấn đề giải thích kết quả:**
  - Đặc trưng có giá trị liên tục có thể làm mô hình khó hiểu / giải thích
  - Binning transformation → chia khoảng giá trị → mỗi khoảng có một ý nghĩa

- **Vấn đề quan hệ phi tuyến giữa các đặc trưng**
  - Quan hệ phi tuyến làm cho mô hình hóa và giải thích trở nên khó khăn hơn
  - Biến đổi để chuyển về dạng tuyến tính: log transform → đơn giản hơn
  - Ex: $Y = b * \exp(a*X) \rightarrow \log(Y) = \log(b) + a*X$

Biến đổi đặc trưng

- Biến đổi dữ liệu **dạng số**:
  - Min-Max scaling

	RoomService	FoodCourt	ShoppingMall	Spa	VRDeck
0	0.0	0.0	0.0	0.0	0.0
1	109.0	9.0	25.0	549.0	44.0
2	43.0	3576.0	0.0	6715.0	49.0
3	0.0	1283.0	371.0	3329.0	193.0
4	303.0	70.0	151.0	565.0	2.0

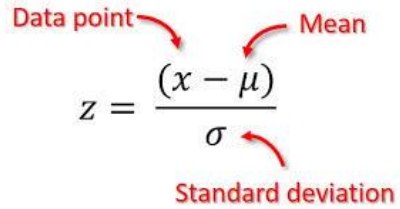
$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

	RoomService	FoodCourt	ShoppingMall	Spa	VRDeck
0	0.000000	0.000000	0.000000	0.000000	0.000000
1	0.007608	0.000302	0.001064	0.024500	0.001823
2	0.003001	0.119948	0.000000	0.299670	0.002030
3	0.000000	0.043035	0.015793	0.148563	0.007997
4	0.021149	0.002348	0.006428	0.025214	0.000083

Biến đổi đặc trưng

- Biến đổi dữ liệu **dạng số**:
  - Min-Max scaling
  - Standardization (Z-score scaling)

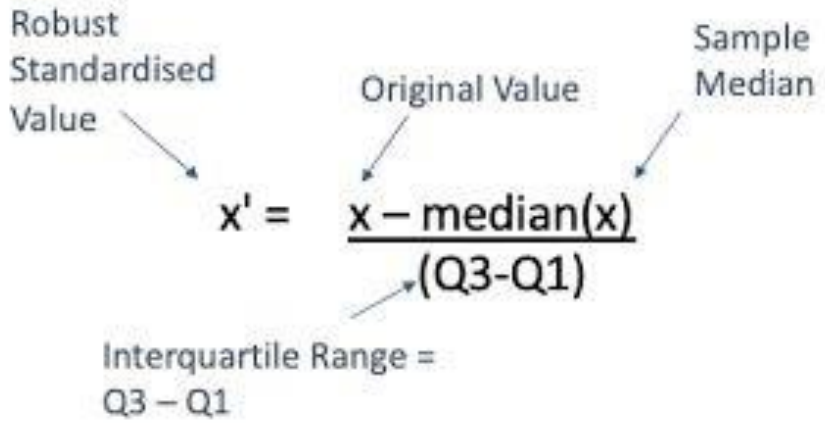
	RoomService	FoodCourt	ShoppingMall	Spa	VRDeck
0	0.0	0.0	0.0	0.0	0.0
1	109.0	9.0	25.0	549.0	44.0
2	43.0	3576.0	0.0	6715.0	49.0
3	0.0	1283.0	371.0	3329.0	193.0
4	303.0	70.0	151.0	565.0	2.0



	RoomService	FoodCourt	ShoppingMall	Spa	VRDeck
0	-0.337025	-0.284274	-0.287317	-0.273736	-0.266098
1	-0.173528	-0.278689	-0.245971	0.209267	-0.227692
2	-0.272527	1.934922	-0.287317	5.634034	-0.223327
3	-0.337025	0.511931	0.326250	2.655075	-0.097634
4	0.117466	-0.240833	-0.037590	0.223344	-0.264352

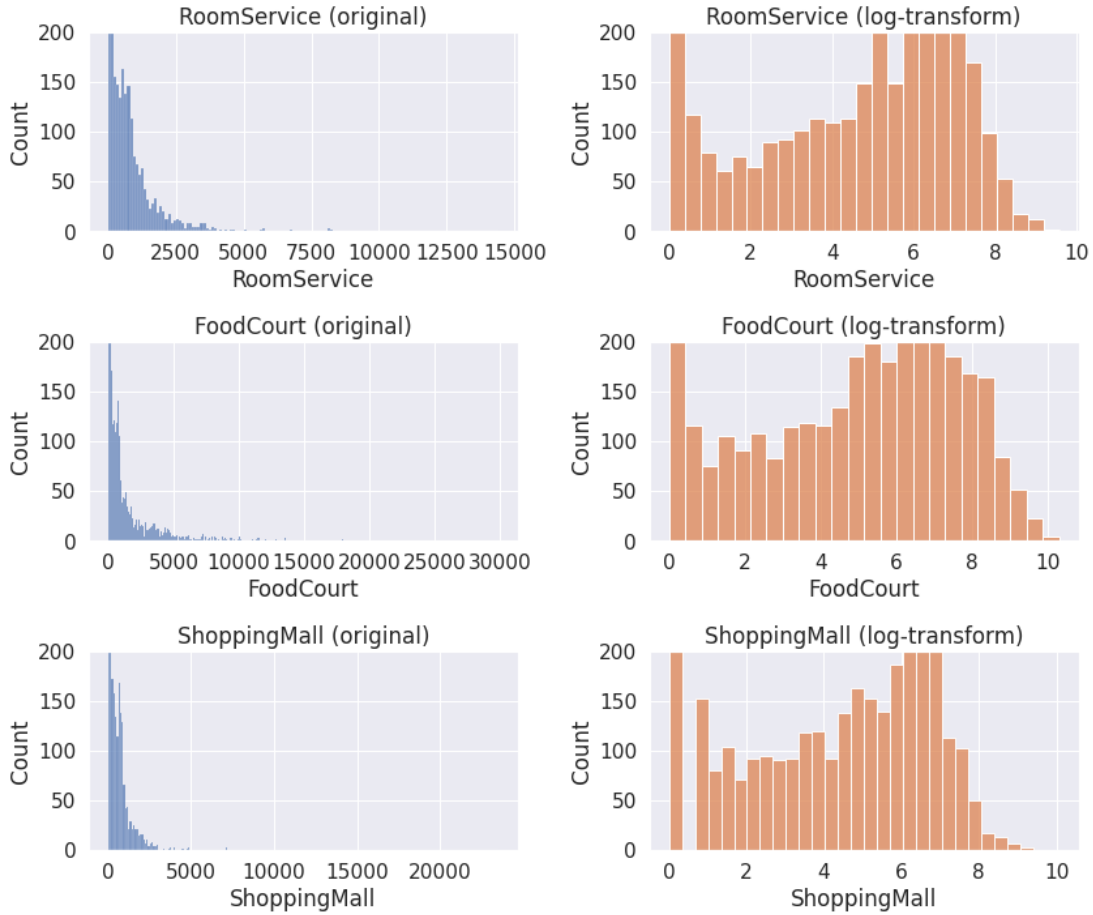
Biến đổi đặc trưng

- Biến đổi dữ liệu **dạng số**:
  - Min-Max scaling
  - Standardization (Z-score scaling)
  - Robust Scaler



Biến đổi đặc trưng

- Biến đổi dữ liệu **dạng số**:
  - Min-Max scaling
  - Standardization (Z-score scaling)
  - Robust Scaler
  - Log Transform



Biến đổi đặc trưng

- Biến đổi dữ liệu **dạng số**:
  - Min-Max scaling
  - Standardization (Z-score scaling)
  - Robust Scaler
  - Log Transform
  - Rời rạc hóa (Discretization hay binning)

	PassengerId	Age
0	0001_01	39.0
1	0002_01	24.0
2	0003_01	58.0
3	0003_02	33.0
4	0004_01	16.0

	PassengerId	Age_group
0	0001_01	Age_31-50
1	0002_01	Age_18-25
2	0003_01	Age_51+
3	0003_02	Age_31-50
4	0004_01	Age_13-17

Biến đổi đặc trưng

- Biến đổi dữ liệu **dạng danh mục (phân loại)**:
  - One-hot encoding

	PassengerId	HomePlanet
0	0001_01	Europa
1	0002_01	Earth
2	0003_01	Europa
3	0003_02	Europa
4	0004_01	Earth

	is_Earth	is_Europa	is_Mars
0	0	1	0
1	1	0	0
2	0	1	0
3	0	1	0
4	1	0	0

Biến đổi đặc trưng

- Biến đổi dữ liệu **dạng danh mục (phân loại)**:
  - One-hot encoding
  - Ordinal encoding

	PassengerId	Age_group
0	0001_01	Age_31-50
1	0002_01	Age_18-25
2	0003_01	Age_51+
3	0003_02	Age_31-50
4	0004_01	Age_13-17

Age_group	Age_group_encode
Age_31-50	5.0
Age_18-25	3.0
Age_51+	6.0
Age_31-50	5.0
Age_13-17	2.0

Biến đổi đặc trưng

- Biến đổi dữ liệu **dạng danh mục (phân loại)**:
  - One-hot encoding
  - Ordinal encoding
  - Label encoding

	PassengerId	HomePlanet
0	0001_01	Europa
1	0002_01	Earth
2	0003_01	Europa
3	0003_02	Europa
4	0004_01	Earth

	PassengerId	HomePlanet	HomePlanet_enc
0	0001_01	Europa	0
1	0002_01	Earth	1
2	0003_01	Europa	0
3	0003_02	Europa	0
4	0004_01	Earth	1

Biến đổi đặc trưng

- Biến đổi dữ liệu **dạng danh mục (phân loại)**:
  - One-hot encoding
  - Ordinal encoding
  - Label encoding
  - Target Encoding

	HomePlanet	Transported
0	Europa	False
1	Earth	True
2	Europa	False
3	Europa	False
4	Earth	True
5	Earth	True

	HomePlanet	Transported	HomePlanet_target_en
0	Europa	False	0.658846
1	Earth	True	0.423946
2	Europa	False	0.658846
3	Europa	False	0.658846
4	Earth	True	0.423946
5	Earth	True	0.423946

NỘI DUNG

1. PHÁT HIỆN & XỬ LÝ DỮ LIỆU BỊ THIẾU
2. PHÁT HIỆN & XỬ LÝ DỮ LIỆU NGOẠI LỆ
3. TẠO ĐẶC TRƯNG MỚI – FEATURE EXTRACTION
4. BIẾN ĐỔI ĐẶC TRƯNG – FEATURE TRANSFORMATION
5. CHỌN LỰA ĐẶC TRƯNG – FEATURE SELECTION

Tại sao cần Chọn lựa đặc trưng

- **Vấn đề độ chính xác của mô hình:**
  - Các đặc trưng không liên quan và dư thừa làm mô hình bị nhiễu
  - Chỉ chọn đặc trưng phù hợp → giảm nhiễu → tăng độ chính xác

- **Vấn đề overfitting:**
  - Mô hình phức tạp hấp thụ các đặc trưng nhiễu nhiều hơn mô hình đơn giản
  - Loại bỏ đặc trưng nhiễu → mô hình đơn giản hơn → tránh overfitting

Tại sao cần chọn đặc trưng

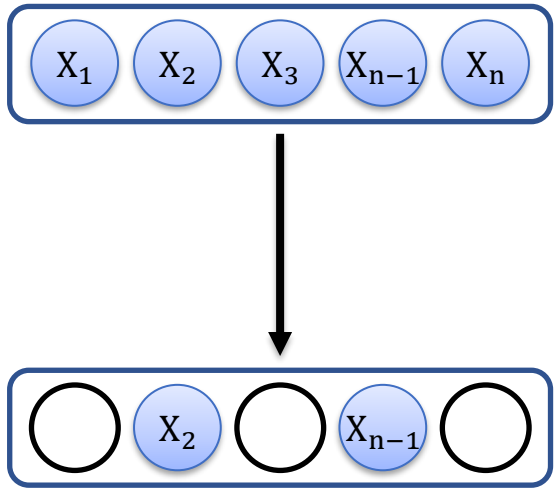
- **Vấn đề thời gian và chi phí huấn luyện:**
  - Nhiều đặc trưng → mô hình phức tạp → tốn chi phí tính toán và thời gian
  - Chọn đặc trưng quan trọng nhất →giảm chi phí và thời gian

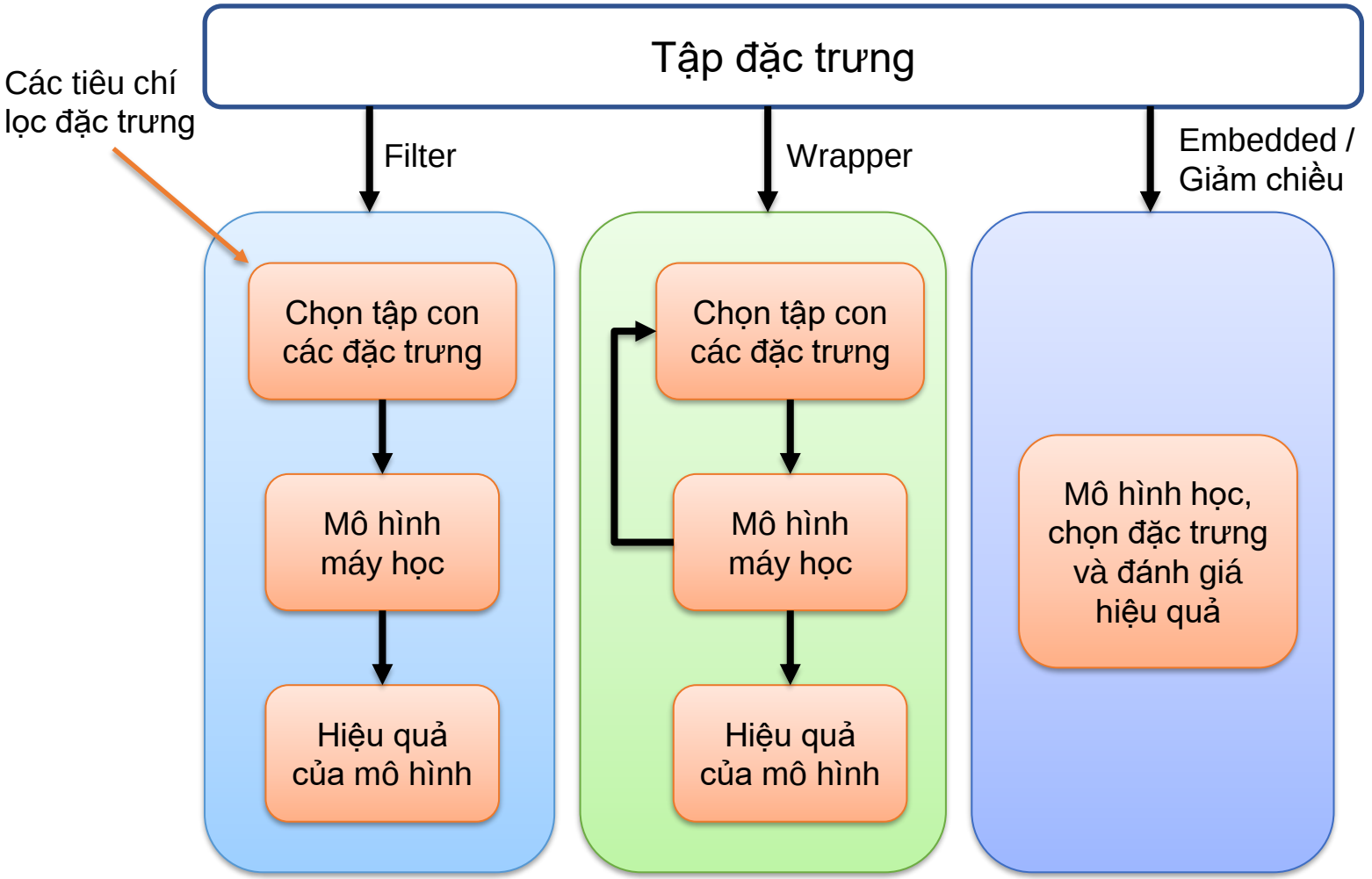
- **Vấn đề khả năng giải thích của mô hình:**
  - Mô hình quá nhiều đặc trưng → khó giải thích (cho khách hàng)
  - Chỉ chọn những đặc trưng quan trọng → dễ giải thích và hiểu lý do ra quyết định của mô hình

Một số kỹ thuật chọn đặc trưng

01	**Phương pháp Filter**	• Correlation coefficient: Pearson,.. • Variance Threshold • Missing value ratio; Mutual Information
02	**Phương pháp Wrapper**	• Forward Selection • Backward Elimination • Recursive Feature Elimination (RFE)
02	**Phương pháp Embedded**	• LASSO, Ridge Regression, Elastic Net • Tree-based: Random Forest, GBM
04	**Phương pháp giảm chiều**	• Component/Factor based: Factor Analysis, PCA, ICA • Projection based: t-SNE, UMAP

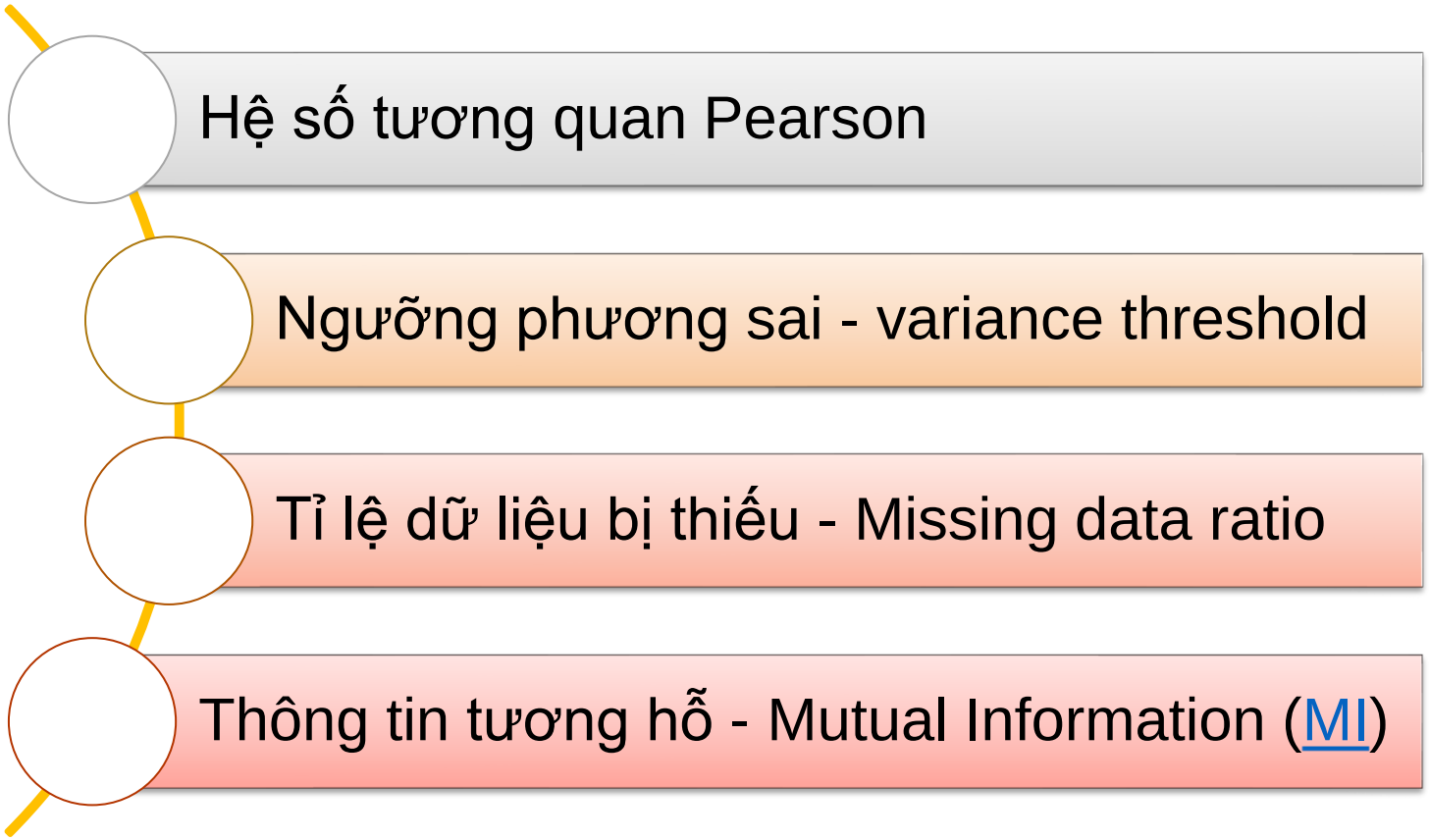
Một số kỹ thuật chọn đặc trưng





Phương pháp 1: Filter

- Áp dụng một loại chỉ số để loại bỏ các đặc trưng không liên quan hoặc dư thừa



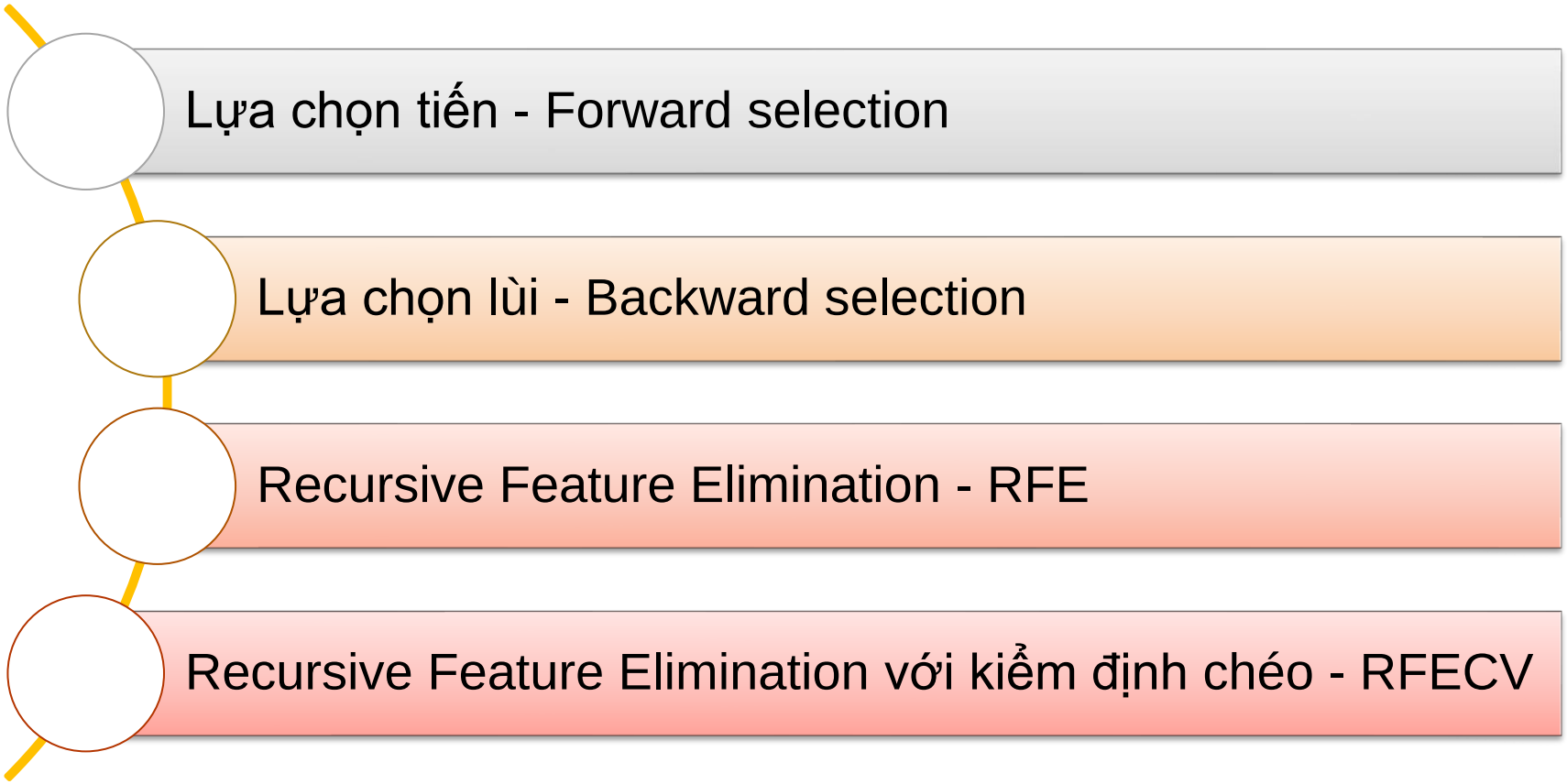
Phương pháp 1: Filter

- Nhận xét:

ƯU ĐIỂM	KHUYẾT ĐIỂM
- Nhanh, do chọn đặc trưng nhưng không cần huấn luyện - Dễ hiểu, dễ thực hiện	- Thiếu sự tương tác giữa các đặc trưng - Có thể bỏ lỡ tập đặc trưng tối ưu - Có khả năng xóa thừa dữ liệu

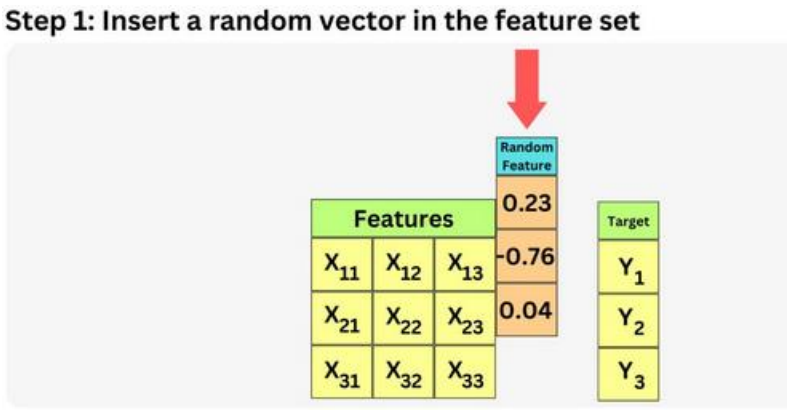
Phương pháp 2: Wrapper

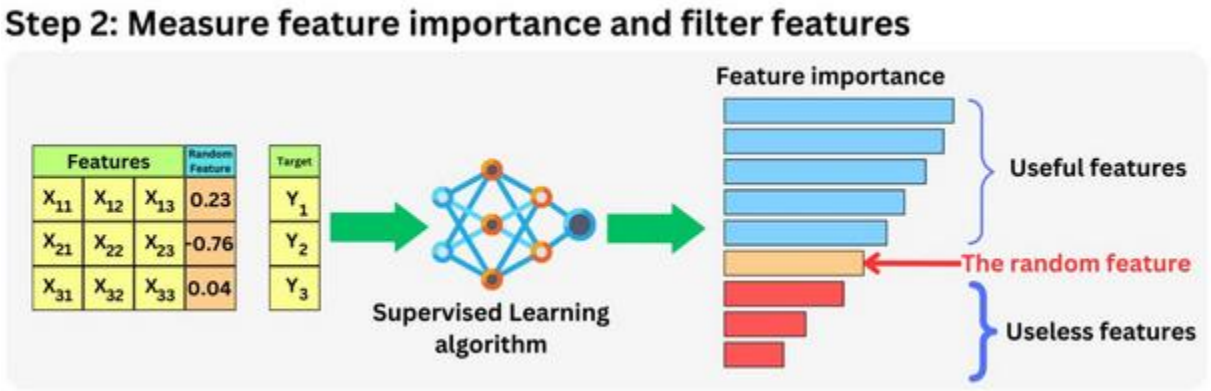
- Sử dụng mô hình dự đoán để đánh giá hiệu quả các tập hợp con đặc trưng

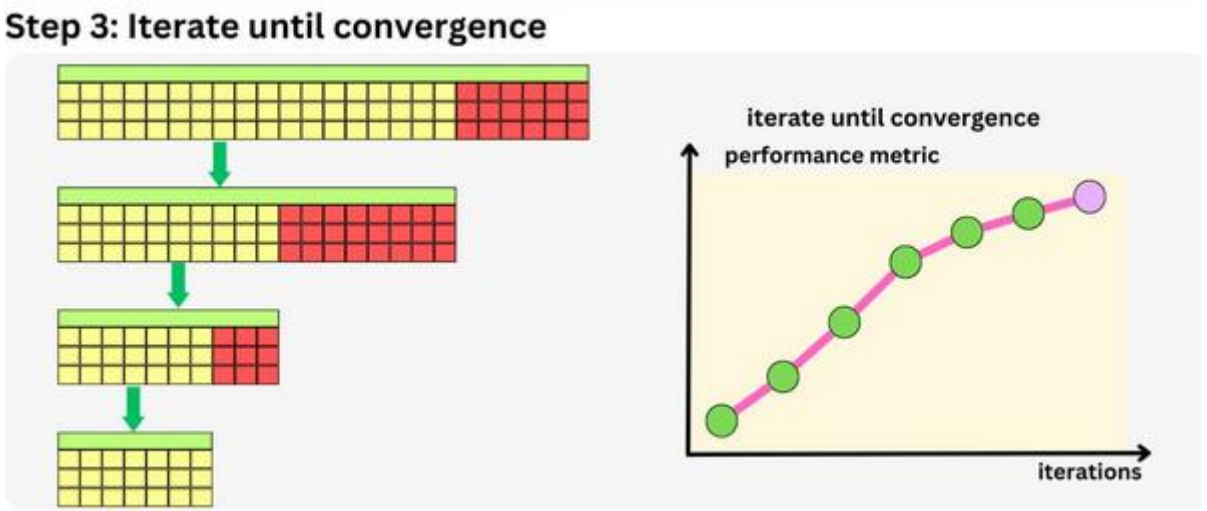


Phương pháp 2: Wrapper với Random Bar

- So sánh mức độ quan trọng của đặc trưng với đặc trưng ngẫu nhiên







Nguồn: https://www.linkedin.com/feed/update/urn:li:activity:7059555375821307904?utm_source=share&utm_medium=member_desktop

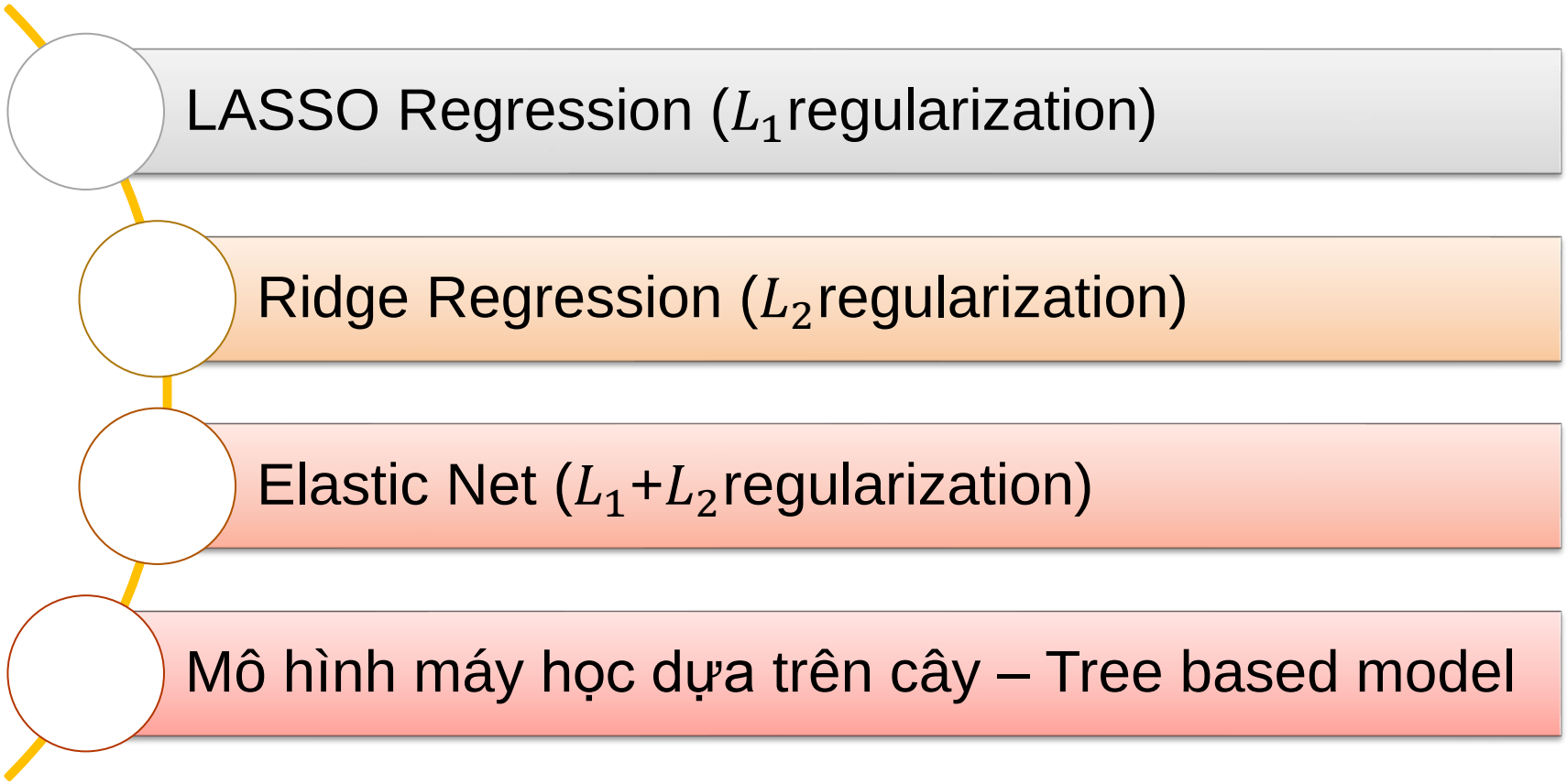
Phương pháp 2: Wrapper

- Nhận xét

ƯU ĐIỂM	KHUYẾT ĐIỂM
- Có sự tương tác giữa các đặc trưng - Tập con đặc trưng tối ưu theo mô hình	- Chi phí tính toán lớn - Dễ bị overfitting - Phức tạp hơn so với phương pháp Filter

Phương pháp 3: Embedded model

- Chọn lựa đặc trưng là một phần của quá trình học của mô hình



Phương pháp 3: Embedded model

- Nhận xét

ƯU ĐIỂM	KHUYẾT ĐIỂM
- Hiệu quả tính toán cao hơn Wrapper - Tính tổng quát cao hơn, có sự tương tác giữa các đặc trưng và tham số của mô hình	- Khả năng giải thích đặc trưng thấp hơn so với PP Filter - Có khả năng overfit khi mô hình phức tạp và tập dữ liệu nhỏ

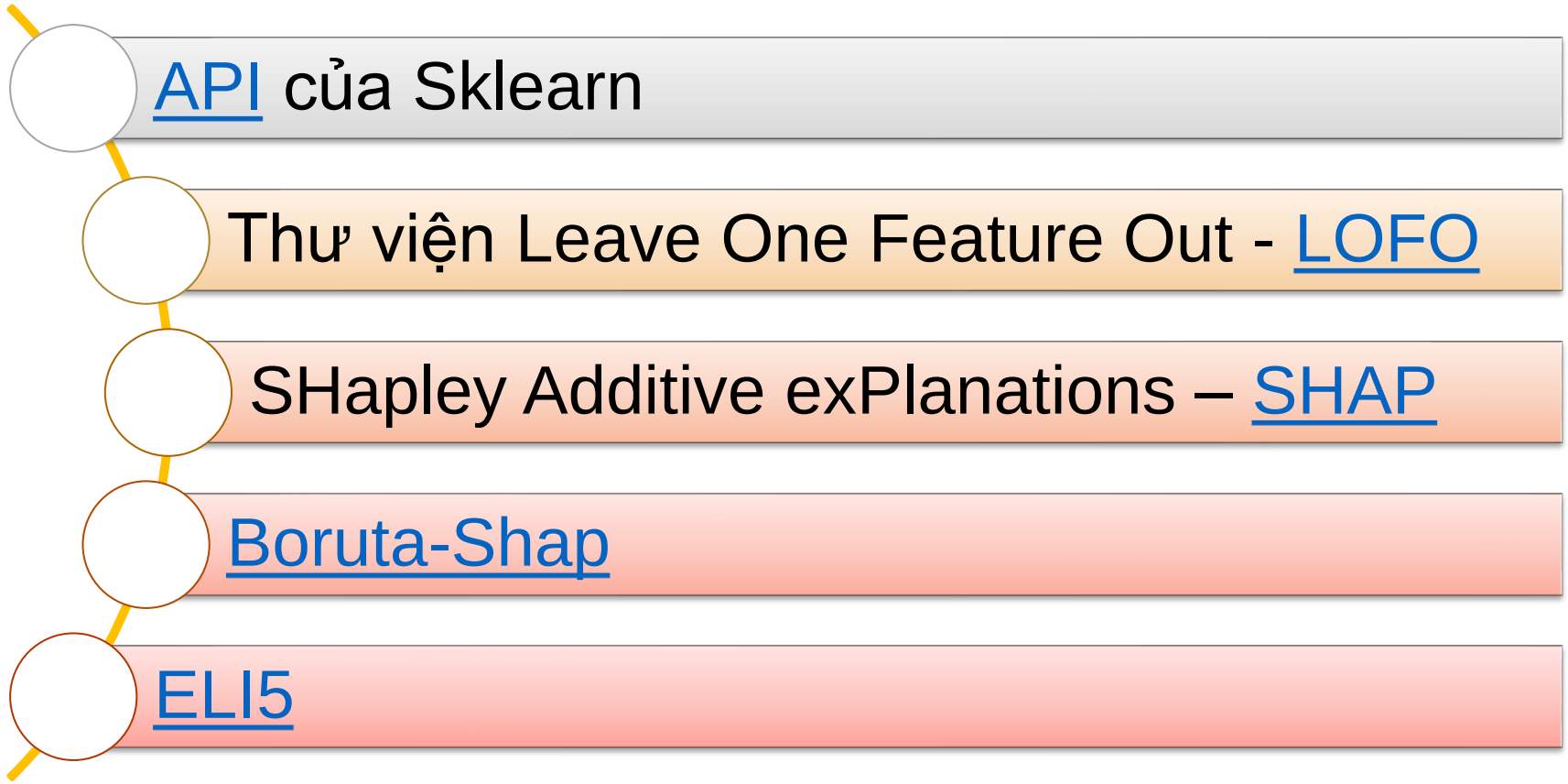
Phương pháp 4: Giảm chiều dữ liệu

- Dựa trên nền tảng máy học không giám sát: PCA, ICA, tSNE,…

ƯU ĐIỂM	KHUYẾT ĐIỂM
- Hiệu quả tính toán cao (do biến đổi tuyến tính) - Có thể trực quan hóa: 2D hoặc 3D - Có thể loại bỏ được các đặc trưng nhiễu (đặc trưng có phương sai thấp)	- Khả năng giải thích đặc trưng thấp - Không phù hợp với các loại dữ liệu phân loại (categorical) - Chuyển đặc trưng sang không gian khác → không xác định được tập con các đặc trưng quan trọng ở không gian gốc

Một số công cụ chọn lựa đặc trưng

- Các phương pháp tiếp cận trên được cài đặt, hỗ trợ trong các API, thư viện sau:



BÀI QUIZ VÀ HỎI ĐÁP