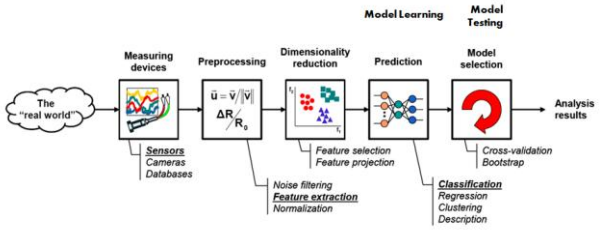
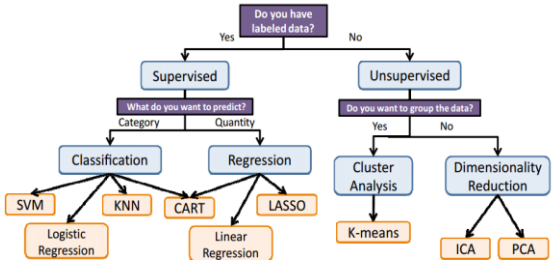
UIT
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

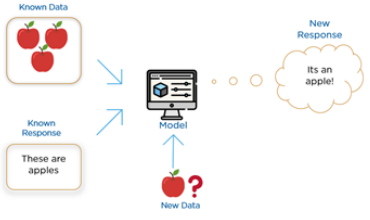
NHẬN DẠNG
PATTERN RECOGNITION

Phân loại - Classification

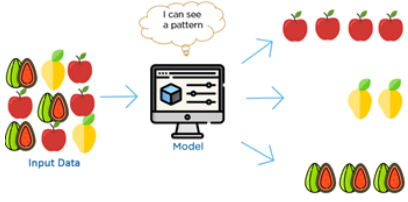
QUÁ TRÌNH HOC MÁY CƠ BẢN

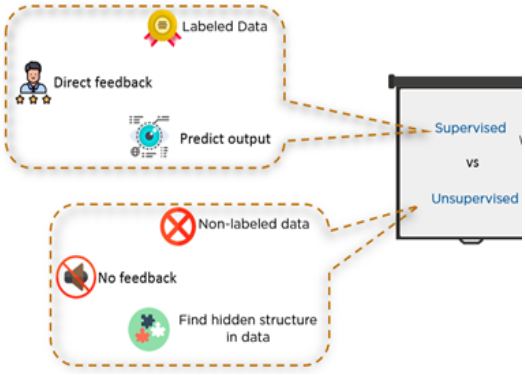






VS





Phương pháp phân lớp
Dựa trên phương pháp học

- Học có giám sát:
  - Các lớp của dữ liệu học đã biết,
  - Mục đích: tìm ánh xạ từ không gian đặc trưng sang không gian lớp sao cho chi phí nhỏ nhất.
  - Dễ mất tính tổng quát hóa vì tính "quá khớp" (overfitting).
- Học không giám sát:
  - Các lớp của dữ liệu chưa biết,
  - Mục đích: gói cụm các mẫu thành nhóm sao cho các mẫu trong 1 nhóm khác nhau ít và các mẫu khác nhóm khác nhau nhiều.
  - Số cụm có thể là đã biết hoặc chưa biết.
- Học tăng cường:
  - Các lớp chưa biết khi bắt đầu học.
  - Việc lan truyền ngược sẽ hiệu chỉnh hành động đã học.

Phương pháp phân lớp
Dựa trên phương pháp

- Phương pháp thống kê (Bayesian):
  - Đặc trưng thay đổi ngẫu nhiên với xác suất nào đó.
  - Nhận dạng dựa trên cực tiểu ước lượng sai số.
  - Ước lượng của hàm phân bố xác suất không chắc chắn.
- Phương pháp hình học:
  - Không gian đặc trưng được chia thành các phần sao cho mỗi phần đại diện cho 1 lớp nào đó.
  - Một số phương pháp thuộc nhóm này: biệt số tuyến tính Fisher, máy hỗ trợ vector…
- Phương pháp mạng neuron:
  - Sử dụng "hộp đen" để biến đổi từ không gian đặc trưng sang không gian lớp.
  - Ví dụ: mạng MLP (multi-layer perceptron), ánh xạ tự tổ chức,…

Phương pháp phân lớp
Dựa trên phương pháp

- Dựa trên mô hình:
  - Các lớp được đại diện bởi mẫu tham chiếu nào đó.
  - Nhận dạng dựa trên việc tìm mẫu tham chiếu gần nhất.
- Phương pháp sử dụng cú pháp:
  - Các lớp được đại diện bởi cú pháp được xây dựng từ mẫu nguyên thủy.
  - Nhận dạng bằng việc kiểm tra xem đầu vào có thể sinh ra được từ cú pháp có sẵn không.
- Phương pháp dựa trên kết cấu:
  - Các lớp được đại diện bởi đồ thị hoặc cấu trúc tương tự.
  - Nhận dạng dựa trên quá trình khớp đồ thị.

Supervised learning

Regression vs Classification

9

KNN (K nearest neighbors – láng giềng)

- ***K-nearest neighbors*** (k-NN) là một trong số các phương pháp phổ biến trong học máy. Vài tên gọi khác như:
  - Instance-based learning
  - Lazy learning
  - Memory-based learning
- Ý tưởng của phương pháp
  - Không xây dựng một mô hình (mô tả) rõ ràng cho hàm mục tiêu cần học.
  - Quá trình học chỉ lưu lại các dữ liệu huấn luyện.
  - Việc dự đoán cho một quan sát mới sẽ dựa vào các hàng xóm gần nhất trong tập học.
- Do đó k-NN là một phương pháp phi tham số (nonparametric methods)

10

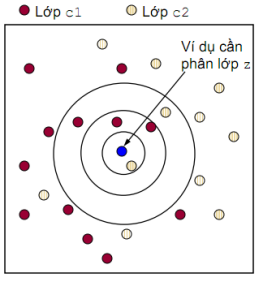
Parametric vs. Nonparametric models

Almost all models for machine learning have "parameters" or "weights" that need to be learned.

Parametric Models	Nonparametric models
The number of parameters is constant, or independent of the number of training examples.	The number of parameters grows with the number of training examples.

K nearest neighbors – láng giềng

- Xét 1 láng giềng gần nhất
  → Gán z vào lớp c2
- Xét 3 láng giềng gần nhất
  → Gán z vào lớp c1
- Xét 5 láng giềng gần nhất
  → Gán z vào lớp c1



12

Classification với KNN (K nearest neighbors – láng giềng)

- Hai thành phần chính:
  - Độ đo tương đồng (similarity measure/distance) giữa các đối tượng.
  - Các hang xóm sẽ dùng vào việc phán đoán.
- Bộ phân lớp: Chia không gian thuộc tính thành nhiều vùng
  - Mỗi vùng được gắn với 1 nhãn lớp (class label)
  - *Ranh giới quyết định* chia tách các vùng quyết định
- Các phương pháp phân lớp xây dựng mô hình có dạng:

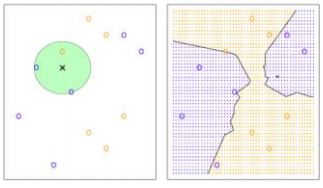
$$Pr(Y \mid X)$$

13

Classification với KNN (K nearest neighbors – láng giềng)

- Bộ phân lớp KNN
  - Việc dự đoán lớp cho mẫu X là *lớp phổ biến nhất giữa K láng giềng gần nhất* (trong tập học)
  - Mô hình phân lớp:

$$Pr(X \text{ belongs to class } Y) \approx \frac{\#(\text{neighbors of } X \text{ in class } Y)}{K}$$



14

Classification với KNN (K nearest neighbors – láng giềng)

- Mỗi ví dụ học $\boldsymbol{x}$ được biểu diễn bởi 2 thành phần:
  - Mô tả của ví dụ: $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, trong đó $x_i \in R$
  - Nhãn lớp : $c \in C$, với C là tập các nhãn lớp được xác định trước
- Giai đoạn học
  - Đơn giản là lưu lại các ví dụ học trong tập học: $\boldsymbol{D}$
- Giai đoạn phân lớp: Để phân lớp cho một ví dụ (mới) $\boldsymbol{z}$
  - Với mỗi ví dụ học $\boldsymbol{x} \in \boldsymbol{D}$, tính khoảng cách giữa $\boldsymbol{x}$ và $\boldsymbol{z}$
  - Xác định tập $NB(\boldsymbol{z})$ – các láng giềng gần nhất của $\boldsymbol{z}$
    → Gồm $\boldsymbol{k}$ ví dụ học trong $\boldsymbol{D}$ gần nhất với $\boldsymbol{z}$ tính theo một hàm khoảng cách d
  - Phân $\boldsymbol{z}$ vào lớp chiếm số đông (the majority class) trong số các lớp của các ví dụ trong $NB(\boldsymbol{z})$

15

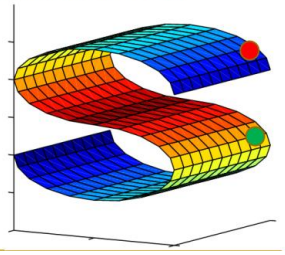
Regression với KNN (K nearest neighbors – láng giềng)

- Mỗi ví dụ học $\boldsymbol{x}$ được biểu diễn bởi 2 thành phần:
  - Mô tả của ví dụ: $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, trong đó $x_i \in R$
  - Giá trị đầu ra mong muốn: $y_x \in R$ (là một số thực)
- Giai đoạn học
  - Đơn giản là lưu lại các ví dụ học trong tập học $\boldsymbol{D}$
- Giai đoạn dự đoán: Để dự đoán giá trị đầu ra cho ví dụ $\boldsymbol{z}$
  - Đối với mỗi ví dụ học $\boldsymbol{x} \in \boldsymbol{D}$, tính khoảng cách giữa $\boldsymbol{x}$ và $\boldsymbol{z}$
  - Xác định tập $NB(\boldsymbol{z})$ – các láng giềng gần nhất của $\boldsymbol{z}$
    → Gồm $\boldsymbol{k}$ ví dụ học trong $\boldsymbol{D}$ gần nhất với $\boldsymbol{z}$ tính theo một hàm khoảng cách d
  - Dự đoán giá trị đầu ra đối với $\boldsymbol{z}$: $y_z = \frac{1}{k}\sum_{x \in NB(z)} y_x$

16

Các yếu tố quan trọng trong KNN

- Hàm khoảng cách
  - Mỗi hàm sẽ tương ứng với một cách nhìn về dữ liệu.
  - Vô hạn hàm!!!
  - Chọn hàm nào?



17

KNN (K nearest neighbors – láng giềng)

- Hàm tính khoảng cách ***d***
  - Đóng vai trò rất quan trọng trong phương pháp học dựa trên các láng giềng gần nhất
  - Thường được xác định trước, và không thay đổi trong suốt quá trình học và phân loại/dự đoán
- Lựa chọn hàm khoảng cách ***d***
  - *Các hàm khoảng cách hình học*: Dành cho các bài toán có các thuộc tính đầu vào là kiểu số thực ($x_i \in R$)
  - *Hàm khoảng cách Hamming*: Dành cho các bài toán có các thuộc tính đầu vào là kiểu nhị phân ($x_i \in \{0,1\}$)

18

KNN (K nearest neighbors – láng giềng)

- Các hàm tính khoảng cách hình học (Geometry distance functions)
  - Hàm Minkowski (p-norm): $d(x,z) = \left(\sum_{i=1}^{n} |x_i - z_i|^p \right)^{1/p}$
  - Hàm Manhattan ($p = 1$): $d(x,z) = \sum_{i=1}^{n} |x_i - z_i|$
  - Hàm Euclid ($p = 2$): $d(x,z) = \sqrt{\sum_{i=1}^{n} (x_i - z_i)^2}$
  - Hàm Chebyshev ($p = \infty$): $d(x,z) = \lim_{p \to \infty} \left(\sum_{i=1}^{n} |x_i - z_i|^p \right)^{1/p} = \max_i |x_i - z_i|$

19

KNN (K nearest neighbors – láng giềng)

- Hàm khoảng cách Hamming
  - Đối với các thuộc tính đầu vào là kiểu nhị phân ($\{0,1\}$)
  - Ví dụ: $\boldsymbol{x} = (0,1,0,1,1)$

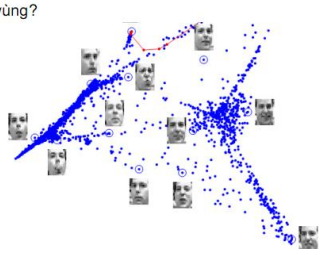
$$d(x,z) = \sum_{i=1}^{n} Difference(x_i, z_i)$$

$$Difference(a,b) = \begin{cases} 1, if\ (a \neq b) \\ 0, if\ (a = b) \end{cases}$$

20

Các yếu tố quan trọng trong KNN

- Chọn tập láng giềng $NB(z)$
  - Chọn bao nhiêu láng giềng?
  - Giới hạn chọn theo vùng?



21

KNN – Chuẩn hóa miền giá trị thuộc tính

- Hàm tính khoảng cách Euclid: $d(x,z)=\sqrt{\sum_{i=1}^{n}(x_i-z_i)^2}$
- Giả sử mỗi ví dụ được biểu diễn bởi 3 thuộc tính: `Age`, `Income` (cho mỗi tháng), và `Height` (đo theo mét)
  - $\boldsymbol{x}$ = (`Age`=20, `Income`=12000, `Height`=1.68)
  - $\boldsymbol{z}$ = (`Age`=40, `Income`=1300, `Height`=1.75)
- Khoảng cách giữa $\boldsymbol{x}$ và $\boldsymbol{z}$
  - $d(\boldsymbol{x},\boldsymbol{z}) = [(20-40)^2 + (12000-1300)^2 + (1.68-1.75)^2]^{0.5}$
  - Giá trị khoảng cách bị quyết định chủ yếu bởi giá trị khoảng cách (sự khác biệt) giữa 2 ví dụ đối với thuộc tính `Income`

    → Vì: Thuộc tính `Income` có miền giá trị rất lớn so với các thuộc tính khác
- Cần phải chuẩn hóa miền giá trị (đưa về cùng một khoảng giá trị)
  - Khoảng giá trị [0,1] thường được sử dụng
  - Đối với mỗi thuộc tính `i`: $x_i := x_i / \max(x_j)$

22

KNN – Trọng số thuộc tính

- Hàm khoảng cách Euclid: $d(x,z)=\sqrt{\sum_{i=1}^{n}(x_i-z_i)^2}$
  - Tất cả các thuộc tính có cùng (như nhau) ảnh hưởng đối với giá trị khoảng cách
- **Các thuộc tính khác nhau** có thể (nên) có **mức độ ảnh hưởng khác nhau** đối với giá trị khoảng cách
- Cần phải tích hợp (đưa vào) các giá trị trọng số của các thuộc tính trong hàm tính khoảng cách
  - w_i là trọng số của thuộc tính `i`: $d(x,z)=\sqrt{\sum_{i=1}^{n}w_i(x_i-z_i)^2}$
- Làm sao để xác định các giá trị trọng số của các thuộc tính?
  - Dựa trên các tri thức cụ thể của bài toán (vd: được chỉ định bởi các chuyên gia trong lĩnh vực của bài toán đang xét)
  - Bằng một quá trình tối ưu hóa các giá trị trọng số (vd: sử dụng một tập học để học một bộ các giá trị trọng số tối ưu)

23

KNN – Ưu điểm

- Độ phức tạp tính toán của quá trình training là bằng 0.
- Việc dự đoán kết quả của dữ liệu mới rất đơn giản.
- Không cần giả sử gì về phân phối của các class.
- KNN cũng có thể sử dụng cho bài toán Regression

24

KNN – Nhược điểm

- KNN rất nhạy cảm với nhiễu khi K nhỏ.
- K càng lớn thì độ phức tạp cũng sẽ tăng
- Việc lưu toàn bộ dữ liệu trong bộ nhớ cũng ảnh hưởng tới hiệu năng

25

KNN – Ví dụ minh họa

- https://github.com/tiepvupsu/tiepvupsu.github.io/blob/master/assets/knn/KNN.ipynb

26

Tài liệu tham khảo

- https://www.coursera.org/learn/machine-learning?
- https://machinelearningcoban.com/
- https://machinelearningmastery.com/
- Slide Machine learning TQ Khoat , ĐHBK HN
- Slide Machine learning Thầy Tùng, ĐH Thủy Lợi
- Lý thuyết nhận dạng – Ngô Hữu Phúc

27

28