Các bước khám phá tri thức

1. Tiền xử lý

- Chọn dữ liệu, trường dữ liệu phù hợp.
- Chuyển đổi kiểu dữ liệu sao cho phù hợp với giải thuật khai thác dữ liệu mà bước tiếp theo xử dụng (Xử lý sai chính tả, tiếng việt có dấu, NAN....)
- Làm sạch dữ liệu: Xóa trường dữ liệu rỗng, dư thừa hoặc dữ liệu không hợp lệ.

2. Khai thác dữ liệu.

- Khai thác dữ liệu bằng các giải thuật như: **KNN (K láng giềng), Naive Bayes, Cây quyết định,** Bagging**, Booting**.....

- Giải quyết các vấn đề: Phân lớp, hồi quy, Gom nhóm, Luật kết hợp.

3. Đánh giá kết quả.

- Đánh giá độ chính xác của giải thuật à Có thể xây dựng lại giải thuật hoặc quay lại bước tiền xử lý.

KNN(K láng giềng)

Phân lớp: Xử lý data có nhãn **rời rạc** và **liên tục**

Phương pháp này không học(Tên khác: lazy, instance-based) mà chỉ thực hiện khi có một trường hợp mới đến (Mất nhiều thời gian), kết quả phụ thuộc vào việc chọn khoảng cách sử dụng.

Mặc tốt:

- Làm việc trên nhiều loại dữ liệu khác nhau.
- Giải quyết các vấn đề về phân loại, hồi quy.
- Thành công trong việc: Tìm kiếm thông tin, nhận dạng, phân tích dữ liệu.
- Kết quả tốt, độ phức tạp của quá trình phân loại khá lớn.

Để xác định được lớp của phần tử mới đến thì:

1. **Tính Khoảng cách từ phần tử mới đến với các phần tử còn lại trong tập huấn luyện.**

   Khoảng cách:
   Kiểu số:
   Khoảng cách Minkowski:

$$d(i,j) = \sqrt[h]{|x_{i1} - x_{j1}|^h + |x_{i2} - x_{j2}|^h + \cdots + |x_{ip} - x_{jp}|^h}$$

   Nếu h = 1 thì d là khoảng cách Manhattan
   Nếu h = 2 thì d là khoảng cách Euclid

   Kiểu rời rạc:

d(i,j) = (p-m)/p Với p là tổng số biến(thuộc tính), m là số lượng trùng khớp.
Kiểu nhị phân:

Noted: Khi miền giá trị của các thuộc tính có sự chênh lệch quá lớn thì tiến hành chuẩn hóa lại toàn bộ dữ liệu.

2. **Chọn K phần tử gần nhất với phần tử mới trong tập huấn luyện.**
3. **Gán nhãn cho phần tử mới bằng nhãn "phổ biến" nhất của K láng giềng gần nhất.**

KNN trong Python (sgk/18,19,20)

PHƯƠNG PHÁP ĐÁNH GIÁ HIỆU QUẢ PHÂN LỚP

1. Nghi thức kiểm tra.
   - 1 tập train, 1 tập test (dùng để đánh giá hiệu quả của giải thuật)
   Ngoài ra trong tập train còn có giá trị validate (Tập kiểm tra trong quá trình huấn luyện)
   - Nếu không có 1 tập test sẵn ngta sử dụng **K-fold** hay **Hold-out**.

   K-fold: chia tập dữ liệu thành K phần (fold) bằng nhau, lặp lại k lần, mỗi lần sử dụng k-1 folds để học và 1 fold để kiểm tra sau đó tính trung bình k lần kiểm tra
   hold-out: đem ⅔ tập data đem đi train, ⅓ đem đi test

   Tính các giá trị: Precision, recall, accuracy, F1.

Note Meeting 10/09/2024

1. **Xác định mục tiêu đề tài.**
   Dự đoán khả năng ùn tắc giao thông ở ở thành phố New York.

2. **Tiền xử lý tiền dữ liệu.**
   - **Xóa cột ehail_fee** (Không có dữ liệu).

   - **Xóa các trường sau:**
   + *VendorID:* Mã định danh nhà cung cấp dịch vụ taxi, không ảnh hưởng đến tình trạng tắc nghẽn.
   + *Tolls_amount:* Vì khoản phí này thường cố định ở mỗi trạm thu phí nên không ảnh hưởng đến yếu tố tắc nghẽn.

+ *Extra:* Phí này cố định, phụ thu vào ban đêm hoặc giờ cao điểm, Không làm ảnh hưởng đến yếu tố tắc nghẽn.
+ *Tip_amount:* Phụ thuộc vào hành vi của khách hàng, không làm ảnh hưởng đến yếu tố tắc nghẽn.
+ *store_and_fwd_flag:* chỉ cho biết liệu dữ liệu chuyến đi đã được tạm thời lưu trữ trong thiết bị của taxi trước khi được gửi đến máy chủ hay không.
+ *mta_tax:* Thuế cố định.
+ *improvement_surcharge:* Phí cố định, không làm ảnh hưởng đến tình trạng tắc nghẽn.

- **Định dạng lại 2 cột thời gian** (lpep_pickup_datetime, lpep_dropoff_datetime) đổi sang giờ.
- **Xóa những hàng (rows) mang giá trị null ở cột congestion_surcharge**:Vì nó là nhãn nên cần có dữ liệu thì mới dự đoán được.
- **Ở cột congestion_surcharge:** Nếu là 0 thì thay bằng 0, nếu khác 0 thì thay bằng 1.
- **Chuẩn hóa dữ liệu: chuẩn hóa các cột** `'trip_distance', 'fare_amount', 'total_amount' Vì giá giá trị trên lệch quá lớn so với các cột khác.`

3. **Đánh giá giải thuật**

store_and_fwd_flag: Thể hiện trạng trạng thái lưu dữ liệu.
Y (Yes): Điều này có nghĩa là dữ liệu chuyến đi đã được lưu trữ cục bộ trên thiết bị trong xe taxi vì không có kết nối trực tiếp (ví dụ, xe di chuyển qua khu vực mất tín hiệu hoặc không có mạng).

N (No): Điều này có nghĩa là dữ liệu chuyến đi được truyền ngay lập tức đến hệ thống trung tâm mà không cần lưu trữ tạm thời.

RatecodeID: Mã loại giá vé.

1 = Standard rate (Giá tiêu chuẩn): Giá thông thường được tính cho các chuyến đi thông thường trong thành phố.

2 = JFK: Giá vé cố định cho các chuyến đi đến và từ sân bay John F. Kennedy (JFK).

3 = Newark: Giá vé đặc biệt cho các chuyến đi đến và từ sân bay Newark (New Jersey).

4 = Nassau or Westchester: Giá vé cho các chuyến đi ra khỏi giới hạn thành phố, đến Nassau hoặc Westchester County (ngoại ô New York).

5 = Negotiated fare (Giá thỏa thuận): Giá vé đã được thỏa thuận trước giữa hành khách và tài xế cho các chuyến đi đặc biệt.

6 = Group ride (Chuyến đi nhóm): Áp dụng cho các chuyến đi chia sẻ với nhiều hành khách.

PULocationID (Pick-Up Location ID) và **DOLocationID** (Drop-Off Location ID):

Vị trí đón khách và trả khách.

VD: **1**: Financial District (Manhattan)

132: Upper East Side (Manhattan)

138: JFK Airport (Queens)

246: Battery Park (Manhattan)

fare_amount: Tổng số tiền giá vé cho chuyến đi, không bao gồm các khoản phí phụ như phí cầu đường hoặc phụ thu.

extra(chi phí phát sinh0: Các khoản phí phụ thêm như phụ thu cho giờ cao điểm, chuyến đi ban đêm, v.v.

mta_tax: Một khoản thuế cố định do Cơ quan Giao thông Metropolitan (MTA) quy định cho mỗi chuyến đi.

tip_amount: Tiền boa được đưa cho tài xế (nếu có).

tolls_amount: Số tiền trả cho các khoản phí cầu đường trong chuyến đi.

VÍ dụ:
VendorID = 2: Nhà cung cấp taxi là công ty có mã định danh là 2.

lpep_pickup_datetime = 2023-05-10 08:15:00: Chuyến đi bắt đầu vào lúc 8:15 sáng ngày 10 tháng 5, 2023.

lpep_dropoff_datetime = 2023-05-10 08:45:00: Chuyến đi kết thúc vào lúc 8:45 sáng cùng ngày, kéo dài 30 phút.

store_and_fwd_flag = N: Dữ liệu của chuyến đi này được gửi ngay lập tức đến hệ thống và không lưu trữ tạm thời.

RatecodeID = 1: Giá vé tiêu chuẩn (Standard rate).

PULocationID = 132: Vị trí đón khách tại Upper East Side, Manhattan.

DOLocationID = 246: Vị trí trả khách tại Battery Park, Manhattan.

passenger_count = 2: Có 2 hành khách trên chuyến đi.

trip_distance = 5.3: Quãng đường di chuyển là 5.3 dặm.

fare_amount = 15.5: Giá vé cho quãng đường 5.3 dặm là 15.5 USD.

extra = 0.5: Phụ thu cho thời điểm chuyến đi diễn ra (ví dụ: phụ thu ban đêm hoặc giờ cao điểm).

mta_tax = 0.5: Thuế MTA cố định.

tip_amount = 2.00: Hành khách đã boa cho tài xế 2.00 USD.

tolls_amount = 5.76: Phí cầu đường trong chuyến đi (ví dụ qua cầu hoặc đường hầm có thu phí).

ehail_fee: Trống, có nghĩa là không có phí e-hail (phí đặt taxi qua ứng dụng).

improvement_surcharge = 0.30: Phụ thu cải thiện cơ sở hạ tầng (thường là khoản phí cố định).

total_amount = 24.56: Tổng số tiền hành khách phải trả, bao gồm tất cả các khoản phí, thuế, phụ thu, và tiền boa.

payment_type = 1: Hình thức thanh toán là thẻ tín dụng (1 = thẻ tín dụng, 2 = tiền mặt).

trip_type = 1: Chuyến đi cá nhân, không phải chuyến đi nhóm.

congestion_surcharge = 2.50: Phụ thu tắc nghẽn giao thông (áp dụng khi đi vào khu vực tắc nghẽn hoặc giờ cao điểm).

VendorID	lpep_pickup_datetime	lpep_dropoff_datetime	store_and_fwd_flag	RatecodeID	PULocationID	DOLocationID	passenger_count	trip_distance	fare_amount	extra	mta_tax	tip_amount	tolls_amount	ehail_fee	improvement_surcharge	total_amount	payment_type	trip_type	congestion_surcharge
2	2023-05-10 08:15:00	2023-05-10 08:45:00	N	1	132	246	2	5.3	15.5	0.5	0.5	2.00	5.76		0.30	24.56	1	1	2.50