Eksploracja Danych

Wstępne przetwarzanie danych

(c) Marcin Sydow

Zawartość wykładu

- Cele wstępnego przetwarzania danych
- Brakujące dane
- Transformacje zmiennych
- Redukcja wymiarów
- Wzbogacanie danych
- Podział danych

Wstępne przetwarzanie danych

- uzupełnianie brakujących wartości
- poprawianie błędnych danych
- przekształcanie zmiennych (np. skalowalnie, standaryzacja)
- dyskretyzacja i numeracja stanów
- redukcja wymiarów
- ekstrakcja nowych cech (stworzenie nowych zmiennych)
- podział danych na treningowe, testowe i kontrolne
- operacje specjalne dla specjalnych typów danych (np. wyodrębnienie trendu i cykliczności dla szeregów czasowych, przygotowanie danych tekstowych, etc.)

Cel wstępnego przetwarzania danych

Celem jest przygotowanie danych do tego, aby algorytmy eksploracji danych zbudowały jak najlepsze modele.

Należy wziąć pod uwagę jaki typ eksperymentu będzie wykonywany:

- model deskrypcyjny: przedstawienie zależności (wzorców) ukrytych w danych
- model predykcyjny: uzupełnienie brakujących wartości interesującej nas zmiennej przewidywanej

Model deskrypcyjny

Ponieważ model deskrypcyjny ma dostarczyć wjaśnień wzorców w danych, należy ostrożnie usuwać zmienne lub przypadki.

Dane dla takich modeli mają raczej dużo zmiennych, w tym specjalnie stworzone nowe zmienne, wyprowadzone z istniejących, które mogą poprawić interpretowalność danych.

Wartości brakujące, nietypowe lub odstające mogą tu być cenną informacją i niekoniecznie należy je usuwać.

Zarówno zmienne i jak i algorytmy eksploracji w takim przypadku powinny być wysoce interpretowalne.

Model predykcyjny

W modelu predykcyjnym chodzi przede wszystkim o jak najdokładniejszą i najwiarygodniejszą predykcję interesującego atrybutu (cechy), więc obecność czy interpretowalność poszczególnych zmiennych jest podrzędnym celem.

Można np. usuwać wartości odstające, zmienne silnie skorelowane z innymi zmiennymi lub stosować algorytmy o dużej skuteczności lecz niskiej interpretowalności (ang. black-box) takie jak np. sieci neuronowe czy lasy losowe.

Uzupełnianie brakujących danych

Każdy przypadek brakujących danych może być uzupełniony na rózne sposoby:

- zastąpienie stałą
  (R: np. $NA \rightarrow 0$ w całej tabeli "oceny": oceny[is.na(oceny)] <- 0)
- zastąpienie jakąś statystyką pozycyjną (np. średnią, medianą, modą, etc.), jeśli jest to niewielka część danych (mniej niż 10%) i nie zakłóci to wyraźnie rozkładu wartości (R: `impute(e1071)`)
- usunięcie niekompletnych wierszy, szególnie jeśli w danych wierszach jest wiele brakujących wartości i nie stanowią one dużej części danych (mniej niż 10%) (R: `na.omit`)
- usunięcie niekompletnych kolumn, szczególnie jeśli usunięcie odpowiadających zmiennych nie wpłynie negatywnie na jakość modelu (R: np. `dane[,apply(dane,2,function(x) !any(is.na(x)))]`)
- uzupełnienie wartości przy użyciu modelu predykcyjnego (R: np.: `ec.knnimp(dprep)` bazuje na najbliższych sąsiadach)

Uzupełnianie danych wymaga znajomości dziedziny danych (wiedza dziedzinowa/ekspercka).

(R: zabezpieczenie zmiennej przed zmianami:

Zasada minimalizacji zmian w rozkładzie zmiennych

Przy uzupełnianiu brakujących danych należy starać się robić to w taki sposób, aby możliwie najmniej zniekształcić istniejące dane.

Można np. sprawdzać rozkłady zmiennych po uzupełnieniu danych.

Oprócz porównania graficznego (np. histogramów) zmiennych przed i po uzupełnieniu można też stosować pewne miary zgodności rozkładów.

Czasami brak wartości okazuje się być skorelowanym z inną informacją (np. ludzie starsi mogą rzadziej podawać wiek, etc.) i dobrze jest takie ewidentne współzależności wykryć.

Można też stosować wyrafinowane pół-automatyczne metody uzupełniania brakujących danych przy pomocy modeli predykcyjnych.

Poprawianie błędnych danych

Dane mogą być błędne z różnych powodów:

- niezgodne z przyjętymi w dziedzinie regułami (np. data wypisania ze szpitala przed datą wpisania do szpitala)
- niezgodne z wiedzą dziedzinową (np. temperatura powietrza w Polsce w zimie 36 stopni Celsjusza)
- niezgodne z ogólną wiedzą (np. temperatura powietrza -500 stopni Celsjusza)

Szczególnie w przypadku modeli deskrypcyjnych zastępowanie danych błędnych powinno być konsultowane z ekspertem dziedzinowym.

Transformacje zmiennych

W fazie wstępnego przetwarzania danych zmienne mogą być poddawane rozmaitym transformacjom. Rozważa się rozmaite rodzaje transformacji w zależności m.in. od typu danych:

- zmienne numeryczne (np. rozmaite transformacje funkcyjne, dyskretyzacja)
- zmienne kategoryczne (numeracja stanów, etc.)
- nowe zmienne (tworzenie nowych zmiennych na podstawie istniejących)

Daty

Szczególnym rodzajem danych są daty. Istnieje ogromna różnorodność formatów daty.

Bardzo użytecznym narzędziem do przetwarzania formatów danych jest np. narzędzie `date` w powłoce Linuxa (Bash).

Daty mają kilka specyficznych cech, np:

- daty (właściwie time-stamp), są na ogół unikatowe (typ zmiennej monotonicznej), więc na ogół wartości ze zbioru treningowego i testowego nie będą się powtarzały
- z drugiej strony, data zawiera wiele rodzajów cykliczności (dobowy, tygodniowy, miesięczny, roczny, etc.), które mogą nieść cenne informacji i warto je wydobyć przez jawną transformację

Wartości odstające (ang. outliers)

Są to wartości, które są zdecydowanie mniejsze lub większe od większości pozostałych wartości danej zmiennej.

Typowo za wartości odstające uważa się takie, które nie mieszczą się w odległości 1.5 IQR od dolnego lub górnego kwartyla.

Wartości odstające nie są zbyt przydatne do budowania modeli predykcyjnych:

- prawdopodobieństwo ich wystąpienia w danych nieznanych jest niewielkie
- w danych treningowych występują na tyle rzadko, że algorytmy eksploracji danych nie są na ogół w stanie wychwycić wzorców ich występowania

Dlatego w modelach predykcyjnych wartości odstające nie są na ogół brane pod uwagę (mogą być traktowane podobnie jak w przypadku danych błędnych lub brakujących)

Skalowanie

Skalowanie zmiennych oznacza funkcyjną transformację zmiennej numerycznej polegającą na poddaniu jej działaniu pewnej matematycznej funkcji w taki sposób, żeby:

- transformacja była monotoniczna (czyli zachowująca porządek wartości) i różnowartościowa
- wartości po transformacji były w ustalonym przedziale (np. [0,1])
- (jeśli to możliwe) nie zmienić rozkładu danych

Cele skalowania zmiennych

Powody normalizacji/skalowania mogą być różnorakie np:

- niektóre algorytmy eksploracji danych są wrażliwe na bezwzględną wartość zmiennej (np. większe wartości mają większy wpływ na algorytm niż mniejsze), a więc normalizacja niweluje taki, często arbitralny wpływ
- (w przypadku niektórych transformacji) łatwiejsza interpretowalność danych nie wymagająca znajomości dziedziny (nie trzeba znać zakresu wartości w dziedzinie, aby ocenić jak wysoka jest dana wartość, etc.). Z drugiej strony, transformowane wartości mogą być mniej zrozumiałe dla eksperta dziedzinowego.
- w przypadku skalowania zmieniającego rozkład może chodzić np. o to, żeby:
  - uszczegółowić przypadki graniczne, tzn. blisko wartości średnich (amplifikacja)
  - odzwierciedlić pewne elementy wiedzy dziedzinowej (np. multiplikatywność zmiennej a nie jej addytywność)

Typy transformacji zmiennych numerycznych

Przykładowe transformacje:

- normalizacja min-max
- normalizacja eksponencjalna (funkcją sigmoidalną)
- standaryzacja (ang. z-score)
- logarytmizacja
- odwrotność (np. podobieństwo $\leftrightarrow$ odległość)
- pierwiastkowanie
- funkcje cyklometryczne (np. arcus sinus)

Normalizacja min-max

Jest to jedna z najprostszych metod skalowania zmiennych:

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

Własności:

- liniowość
- monotoniczność
- niezmienność kształtu rozkładu (poza skalowaniem liniowym)
- zakres [0,1] (ale tylko dla danych treningowych!)
- prostota

Normalizacja eksponencjalna

$$z(x) = \frac{1}{1 + e^{-\alpha \cdot x}}$$

$\alpha > 0$ jest parametrem: im wyższy tym bardziej "stromy" wykres (większa amplifikacja)[1]
(R: `x = seq(-3,3,0.1); plot(1/(1+exp(-(2*x))))`)

Własności:

- monotoniczność
- zakres (0,1) - dla wszystkich możliwych wartości (nawet spoza zbioru treningowego!)
- nieliniowość (zmiana kształtu rozkładu)
- nieograniczoność dziedziny
- amplifikacja (wzmocnienie różnic) dla wartości średnich

[1] z uwagi na kształt funkcja ta nazywana jest sigmoidalną, jest też używana jako funkcja aktywacji w ciągłych neuronach

Standaryzacja (ang. z-score)

Celem standaryzacji zmiennej jest modyfikacja rozkładu tak aby:

- miał wartość średnią 0
- miał odchylenie standardowe 1

$$z(x) = \frac{x - mean(x)}{sd(x)}$$

Własności:

- przekształcenie liniowe i monotoniczne
- brak zmiany kształtu rozkładu (poza przeskalowaniem liniowym)

(R: `scale`)

Logarytmowanie

$$z(x) = log_b(x)$$

(gdzie $b > 0, b \neq 1$ jest parametrem, np. $b = e$ lub $b = 2$)

Logarytmowanie może być pożądane, jeśli zmienna ma charakter multiplikatywny (np. częstotliwość dźwięku, przyrost ceny akcji) a chcemy uzyskać zmienną o charakterze addytywnym.

W szczególności, zmienna losowa ma rozkład logarytmicznie normalny jeśli jej logarytm $ln(X)$ ma rozkład normalny.

Gdy zmienna przyjmuje wartości nieujemne (włącznie z 0), można dodać 1, np:

$$z(x) = log_b(x + 1)$$

Odwrotność

Czasem przydatna jest transformacja odwrotna:

$$z(x) = \frac{1}{x}$$

(dla x dodatnich)

lub:

$$z(x) = \frac{1}{x+1}$$

(dla x nieujemnych)

Jest to przydatne np. przy przechodzeniu z "podobieństwa" do "odległości" i odwrotnie

Dyskretyzacja (kwantyzacja) zmiennych numerycznych

Dyskretyzacja to operacja zamiany zmiennej numerycznej na odpowiadającą jej zmienną kategoryczną poprzez zdefiniowanie pewnej funkcyjnej zależności pomiędzy dawnymi wartościami (numerycznymi) a nowymi (kategorycznymi).

Na ogół przy zmniejszeniu (na ogół) liczby możliwych przyjmowanych wartości.

Cele dyskretyzacji

Cele mogą być rozmaite, np:

- uproszczenie danych w zamian za częściową utratę informacji (szczególnie, jeśli zmienna przyjmuje b.dużo różnych wartości)
- zmniejszenie "rozdzielczości" zmiennej
- wychwycenie bardziej zgrubnych wzorców
- "podpowiedzenie" algorytmom (przy użyciu wiedzy dziedzinowej), że pewne przedziały wartości mają istotne znaczenie dziedzinowe (np. niepełnoletniość, godzina policyjna, etc.)
- podział danych na podzbiory, aby zwiększyć korelację ze zmienną przewidywaną
- wykorzystanie algorytmów pracujących tylko na danych kategorycznych
- wyeliminowanie wartości odstających

Sposoby dyskretyzacji

Na ogół dyskretyzacja dokonywana jest metodą przedziałową (przynależność do określonego przedziału wartości równoważna jest otrzymaniu danej wartości kategorycznej)

- przedziały równej szerokości
- przedziały o równej liczbie wartości (zmienia rozkład w kierunku jednostajnego)
- maksymalizacja wpływu na zmienną decyzyjną/przewidywaną (np. za pomocą minimalizacji entropii)
- przedziały o konkretnych wartościach brzegowych (zgodnie z wiedzą dziedzinową, np. wiek < 18, etc.)

Dyskretyzacja za pomocą grupowania

Dyskretyzacji można też dokonać za pomocą algorytmu grupującego (ang. clustering) - wtedy wartość kategoryczna wyznaczona jest przez przynależność do odpowiedniej grupy.

Podejście takie jest bardziej wyrafinowane niż metoda przedziałowa, gdyż przy obliczaniu nowej wartości może uwzględniać wartości innych zmiennych.

Uogólnianie (zmiennych kategorycznych)

Jeśli zmienna kategoryczna przybiera bardzo dużą liczbę wartości (szczególnie w porównaniu z liczbą przypadków), to może to stanowić problem dla algorytmów eksploracji danych z wuagi na trudne (lub kosztowne obliczeniowo[2]) wykrycie zależności.
Problemowi takiemu można zaradzić poprzez np.:

- uogólnianie: odwzorowanie wielu różnych wartości w jedną, bardziej ogólną (wymaga to wiedzy dziedzinowej), np: miasto -> powiat, kwartał -> rok, etc.
- ignorowanie rzadziej występujących stanów
- zastępowanie wartości dyskretnych ciągłymi i traktowanie jako zmiennej numerycznej (numerowanie stanów)

[2] liczba możliwych zależności jest wykładniczą funkcją liczby możliwych wartości

Numerowanie stanów

Jest to operacja w pewnym sensie odwrotna do dyskretyzacji.
Niektóre algorytmy wymagają wartości numerycznych.
Ponadto, można w ten sposób oddać pewną wiedzę dziedzinową (np. uporządkowanie stanów, etc.)

Kodowanie zmiennych

Występują też m.in. następujące metody:

- kodowanie binarne (zastąpienie jednej zmiennej o k wartościach k zmiennymi binarnymi, tzw. indykatorami - tylko jeden indykator może być "1", pozostałe są "0"). Wadą jest większa liczba zmiennych, ale niektóre algorytmy lepiej przy takim kodowaniu działają.
- kodowanie wiele-do-wielu (wymaga pewnej kreatywności i wiedzy dziedzinowej), np. zamiast nazwy miasta można podać wielkość miasta (małe, średnie, duże) i oprócz tego np. część kraju (np. wschodnia, zachodnia, etc.)

Przestrzeń atrybutów

Przestrzeń atrybutów, to sposób patrzenia na dane jako na punkty (wektory) w wielo-wymiarowej przestrzeni, gdzie każda zmienna reprezentuje inny wymiar.

Niektóre dane rzeczywiste mogą zawierać bardzo dużo zmiennych (np. dane bio-medyczne).

Problem wysokiej liczby wymiarów powoduje rozmaite trudności algorytmiczne i matematyczne i został nazwany umownie "przekleństwem wymiarowości" (ang. curse of dimensionality).

Istnieją różne techniki redukcji liczby wymiarów.

Przekleństwo wymiarowości (ang. curse of dimensionality)

Im większa liczba wymiarów, tym bardziej mogą dawać się we znaki m.in. następujące problemy algorytmiczne i matematyczne:

- coraz większa minimalna liczba przypadków niezbędna, aby uchwycić jakiekolwiek zależności w danych (zauważmy, że np. przez 2 punkty w 3 wymiarach przechodzi nieskończenie wiele płaszczyzn, etc.)
- coraz większa liczba kombinacji zmiennych (i kombinacji wartości tych zmiennych)
- coraz większy promień odległości musi być wzięty pod uwagę, aby objąć ustaloną część przestrzeni.
- tym łatwiej przetrenować model

Redukcja wymiarów

Aby zredukować liczbę wymiarów można stosować m.in. następujące techniki:

- usuwanie niektórych zmiennych
- analiza składowych głównych (PCA - ang. principal component analysis)

Usuwanie zmiennych

Przy operacji usuwania zmiennych należy:

- konsultować wiedzę dziedzinową
- usuwać w pierwszej kolejności te zmienne, które mają niską wartość informacyjną (są redundantne), co można sprawdzać np. za pomocą miar korelacji.

Analiza składowych głównych (PCA - principal component analysis)

Metoda składowych głównych jest matematyczną techniką macierzową mającą na celu transformację przestrzeni atrybutów do przestrzeni o niższej liczbie wymiarów w taki sposób, że:

- automatycznie tworzone są nowe "wymiary" (zmienne) będące kombinacjami istniejących wymiarów
- pozostawia się tylko zmienne, które mają największą "zmienność", czyli niosą najwięcej informacji

(technika PCA wymaga odrębnego omówienia i wykracza poza materiał niniejszego wykładu)

Równoważenie danych

Technika ta ma znaczenie w przypadku gdy:

- liczności przypadków odpowiadające różnym klasom (kategoriom) są niezrównoważone, co może dać w efekcie np. model stały o wysokiej dokładności, ale niskiej F-mierze (tzw. paradoks dokładności)
- rozkład przypadków w danych daleko odbiega od sytuacji rzeczywistej co może zaburzyć model

Dane można równoważyć np.:

- poprzez usunięcie części przypadków "większościowych"
- "nadpróbkowanie" przypadków "mniejszościowych" (ang. over-sampling)

Dodawanie zmiennych

Aby podnieść jakość modeli obliczanych przez niektóre algorytmy eksploracji danych, można dodać nowe zmienne obliczone na podstawie istniejących zmiennych.

Np. w modelu regresji liniowej można sztucznie dodać do modelu kwadraty, iloczyny par zmiennych i wyższe potęgi do modelu, co może znacznie rozszerzyć elastyczność i dokładność modelu, przy wszystkich zastrzeżeniach odnośnie wady, jaką jest wzrost liczby wymiarów.

Pomocna jest konsultacja wiedzy dziedzinowej.

Podział danych

Podział danych wykonuje się w celu uniknięcia przetrenowania oraz w celu oszacowania jakości zbudowanych modeli w przypadku danych nieznanych.

- dane treningowe (uczenie modeli)
- dane ewaluacyjne (ewaluacja, parametryzacja i selekcja modeli)
- dane kontrolne/testowe (ostateczna ewaluacja modeli)

Na ogół stosuje się podział danych w proporcjach ok. 70%,20%,10% lub zbliżonych.

Specjalne przypadki

Podział na dane treningowe/testowe i ewaluacyjne musi uwzględniać specyfikę zadania, np:

- szeregi czasowe (na ogół dzieli się dane wg cezury czasowej: wcześniejsze to treningowe, późniejsze to testowe, aby uniknąć niepełności danych i maksymalnie odwzorować realne zadanie)
- wykrywanie oszustw (ang. fraud detection) (należy uwzględnić integralność danych, np. nie dzielić operacji z danego konta pomiędzy testowe i treningowe, etc.)

Podsumowanie

- Cele wstępnego przetwarzania danych
- Brakujące dane
- Transformacje zmiennych
- Redukcja wymiarów
- Wzbogacanie danych
- Podział danych

Przykładowe pytania/zadania/problemy

- wymień cele i fazy wstępnego przetwarzania danych
- wymień metody uzupełniania brakujących danych
- wymień rodzaje, cele i techniki transformacji zmiennych
- co to jest przekleństwo wymiarowości?
- wymień cele i techniki wzbogacania danych
- opisz zagadnienie podziału danych

Dziękuję za uwagę.