Politechnika Białostocka

Sztuczna inteligencja

Sprawozdanie z projektu końcowego

Temat: Prezentacja możliwości biblioteki scikit-learn w projektach informatycznych, wymagających użycia drzew decyzyjnych.

Wykonujący projekt: **Adam Bajguz**
Magdalena Kalisz

Studia dzienne
Kierunek: Informatyka
Semestr: IV Grupa zajęciowa: **PS1**
Prowadzący ćwiczenie: **mgr inż. Dariusz Jankowski**

2 czerwca 2018
Data wykonania projektu

.....
Data i podpis prowadzącego

Spis treści

1 Wstęp

Drzewa decyzyjne to nieparametryczna nadzorowana metoda uczenia, która może być stosowana zarówno do klasyfikacji, jak i regresji. Pierwszy artykuł, który przedstawia podejście do klasyfikacji znane z drzew decyzyjn, pochodzi z 1959 roku [1]. Z kolei pierwszy algorytm drzew regresyjnych został obublikowany w roku 1963 [2]. Drzewa decyzyjne są jednym z najczęściej wykorzystywanych narzędzi do klasyfikacji danych i prognozowania z uwagi na fakt, że wiedza odkryta przez drzewo decyzyjne jest zilustrowana w hierarchicznej strukturze. Technika drzew decyzyjnych, czy też klasyfikacyjnych, pozwala m.in. na [3, 4]:

- wyznaczenie zasad decyzyjnych opisujących reguły przypisywania obiektów do wyróżnionych klas (zasady odwołują się do wartości atrybutów opisujących obiekty),
- analizę zbioru obiektów opisywanych przez przyjęty zestaw atrybutów, której celem jest doskonalenie podziału obiektów na jednorodne klasy.

1.1 Drzewa decyzyjne w teorii decyzji i uczeniu maszynowym

Drzewa decyzyjne to również graficzny sposób wspierania procesu decyzyjnego. Drzewo stosowane jest w teorii decyzji i ma szereg zastosowań. Może zarówno rozwiązać problem decyzyjny, jak i stworzyć plan. Metoda drzew decyzyjnych sprawdza się również w momencie rozwiązywania problemów decyzyjnych z wieloma rozgałęziającymi się wariantami oraz podejmowania decyzji w warunkach ryzyka. Drzewa znalazły zastosowanie w takich dziedzinach jak botanika i medycyna, a nawet ekonomia, gdyż są w stanie ułatwiać i usprawniać komputerowe wspomaganie procesu podejmowania decyzji [2, 5].

1.2 Idea drzew decyzji

Idea drzew opiera się na rekursywnym podziale danych na coraz to mniejsze sterty w celu jak najlepszego dopasowania. Drzewa decyzyjne kodują zestaw reguł if-else, które mogą być używane do przewidywania zmiennej docelowej danych funkcji danych. Reguły if-else są tworzone przy użyciu zestawu danych treningowych w celu zaspokojenia jak największej liczby instancji danych treningowych. Początkowo próbka (węzeł macierzysty, korzeń) dzielona jest na dwa lub więcej podzbiorów (węzły potomne). Natomiast węzeł optymalny wyszukuje się na podstawie wszystkich punktów węzłowych dla każdej zmiennej. Następnie proces jest powtarzany dla każdego węzła potomnego, a te podczas dzielenia traktowane są jak węzły macierzyste. Węzeł, którego nie można już podzielić nazywamy liściem, bądź węzłem końcowym, a liczbę liści - wielkością drzewa [3, 4, 6, 7, 8].

1.3 Drzewa klasyfikacyjne i regresyjne

Zaletą drzew decyzyjnych jest to, że mogą modelować dowolny typ funkcji do klasyfikacji lub regresji, czego inne techniki nie potrafią. Do rozwiązywania problemów zarówno regresyjnych, jaki i klasyfikacyjnych służy metoda CART (rodział 5.1). Metoda ta powstała w roku 1984 i opiera się na dwóch typach drzew [6]:

a) drzewa klasyfikacyjne - służą porządkowaniu klas, charakteryzuje je kategoryczna zmienna zależna, której wartość (czyli przynależność przypadku do klasy, grupy) chcemy poznać na podstawie znajomości wartości jednej lub większej liczby predykcyjnych zmiennych ciągłych oraz, ewentualnie zmiennych kategorialnych (Rys.);

   Rysunek 1: Idea drzew klasyfikacyjnych

b) drzewa regresyjne - służą do przewidywania wartości zmiennej ciągłej, na podstawie znajomości wartości jednej lub większej liczby predykcyjnych zmiennych ciągłych (Rys.).

   Rysunek 2: Idea drzew regresyjnych [http://www.statystyka.az.pl/analiza-skupien/metoda-cart.php]

2 Budowa drzew decyzyjnych

Drzewo decyzyjne buduje modele klasyfikacji lub regresji w postaci struktury drzewa, rozkłada zbiór danych na coraz mniejsze podzbiory. Oznacza to, że drzewem decyzyjnym jest graf-drzewo, które składa się z korzenia, węzłów, krawędzi oraz liści. Liście to węzły, z których nie wychodzą już żadne krawędzie. Korzeń drzewa tworzony jest przez wybrany atrybut natomiast poszczególne gałęzie reprezentują wartości

tego atrybutu (Rys.). Drzewa decyzyjne charakteryzują się strukturą hierarchiczną. Oznacza to, że w kolejnych krokach zbiór obiektów jest dzielony, poprzez odpowiedzi na pytania o wartości wybranych cech lub ich kombinacji liniowych. W algorytmach konstrukcji drzew jednym z kluczowych elementów jest wybór kolejności cech, według których, na poszczególnych etapach, będzie dokonywany podział zbioru obiektów. Technika drzew decyzyjnych to uzupełnienie metod klasycznych. Przykładem może tu być analiza dyskryminacyjna. Hierarchiczność podejmowania decyzji jest cechą, która wyróżnia drzewo decyzyjne od innych metod [7, 9].

3 Cechy drzew decyzyjnych

Drzewa decyzyjne i uczenie się drzewa decyzyjnego razem stanowią prosty i szybki sposób uczenia się funkcji, która mapuje dane x na wyniki y, gdzie x może być mieszanką zmiennych jakościowych i liczbowych, a y może być kategoryczne dla klasyfikacji lub numeryczne dla regresji [6].

Największą zaletą drzew decyzyjnych jest to, że mogą modelować dowolny typ funkcji do klasyfikacji lub regresji, czego inne techniki nie potrafią. Ponadto drzewa decyzyjne są uważane za metodę nieparametryczną, co oznacza nie mają one żadnych założeń co do rozkładu danych i struktury klasyfikatora. Drzewa decyzyjne zapewniają również znacznie szybsze trenowanie w porównaniu z prostymi sieciami neuronowymi przy porównywalnej wydajności (złożoność czasowa drzew decyzyjnych jest funkcją zależną od liczby cech oraz wierszy w zestawie danych, podczas gdy dla sieci neuronowych jest funkcją zależną od liczby cech, wierszy w zestawie danych, warstw ukrytych oraz węzłów w każdej ukrytej warstwie). Wadą drzew decyzyjnych jest podatność na przeuczenie. Zatem drzewa decyzyjne są używane najczęściej w przypadku bardzo dużych zbiorów danych, które są uważane za dobrze reprezentujące rzeczywistość. Niektóre algorytmy przycinania drzew są używane do rozwiązania problemu przeuczenia [3, 6, 9, 10].

4 Rodzaje kryteriów podziału

W procecie budowy drzewa decyzyjengo należy wielokrotnie dokonać podziału zbioru danych, tj. należy zadać więcej niż jedno pytanie: co jest cechą, od której powinniśmy zacząć (węzeł główny) i w jakiej kolejności powinniśmy zadawać pytania (budować węzły wewnętrzne), to znaczy używać opisowych cech, aby podzielić zbiór danych? W związku z tym przydatne byłoby zmierzenie "informatywności" funkcji i wykorzystanie tej funkcji z największą "informacyjnością" jako cechą, która powinna być używana do dzielenia danych.

4.1 Współczynnik Giniego

Współczynnik Giniego lub indeks Giniego to miara koncentracji (nierównomierności) rozkładu zmiennej losowej. Nazwa współczynnika pochodzi od nazwiska jego twórcy, włoskiego statystyka Corrado Giniego. Jest on wykorzystywany przez algorytm CART do mierzenia tego, jak często losowo wybrany element z zestawu byłby niewłaściwie oznaczany, gdyby był losowo oznaczony zgodnie z rozkładem etykiet w podzbiorze. Gini index można obliczyć, sumując prawdopodobieństwo p_i elementu o etykiecie i wybieranej razy prawdopodobieństwo $\sum_{k \neq i} p_k = 1 - p_i$ o pomyłce w kategoryzacji tego przedmiotu. Osiąga minimum (zero), gdy wszystkie przypadki w węźle należą do jednej kategorii docelowe [9, 11].

4.2 Information gain

Information gain (wzmocnienie informacji) - atrybutem podziału jest ten, który ma maksymalny przyrost informacji. Aby móc obliczyć przyrost informacji, konieczne jest wprowadzenie terminu entropii zbioru danych. Entropia zbioru danych służy do pomiaru nierównomierności zbioru danych,a w przypadku drzew decyzyjnych jest używana jako miernik informatywności. Termin entropia (w teorii informacji) pochodzi od Claude'a E. Shannona. Idea entropii jest w uproszczeniu następująca: wyobraź sobie, że masz pudełko, które zawiera 100 białych kulek. Zestaw kulek w pudełku można uznać za całkowicie równomierny, ponieważ zawieraja tylko białe kulki (zbiór kulek ma entropię 0, tj. zero nierównomierności). Jeżeli 30 z tych kulek zostało by zastąpionych przez szare, a 20 przez czarne, to gdyby teraz zechcieć wyjąć jedną kulkę z pudełka to prawdopodobieństwo otrzymania białej kulki spadło by z 1,0 do 0,5. Oznacza to, że nierównomierność wzrosła, równomierność zmniejszyła się, a entropia wzrosła. Podsumowując im bardziej nierównomierny jest zbiór danych, tym wyższa entropiam, a im mniej nierównomierny zbiór danych, tym niższa entropia [3, 9]

Tabela 1: Porównanie popularnych algorytmów budowania drzew decyzyjnych

Algorytm	Typ danych	Metoda podziału danych numerycznych
CHAID [14]	Kategoryczne	Nie dotyczy
ID3 [4]	Kategoryczne	Nie dotyczy
C4.5 [15]	Kategoryczne, numeryczne	Brak ograniczeń
C5.0	Kategoryczne, numeryczne	Brak ograniczeń
CART [6]	Kategoryczne, numeryczne	Podziały binarne

4.3 Gain ratio

Gain ratio (współczynnik wzmocnienia) - wybiera atrybut o najwyższym wzroście informacji do liczby współczynników wartości wejściowych. Liczba wartości wejściowych to liczba odrębnych wartości atrybutu występującego w zbiorze treningowym. Jest on stosunkiem przyrostu informacji do wewnętrznej informacji. Jest stosowany w celu zmniejszenia błędu w kierunku atrybutów o wielu wartościach, biorąc pod uwagę liczbę i rozmiar oddziałów przy wyborze atrybutu [9, 11, 12].

5 Algorytmy budowania drzew decyzyjnych

Dostępnych jest kilka algorytmów służących do klasyfikacji i analizy segmentacji. Wszystkie te algorytmy zasadniczo realizują to samo zadanie: dzieląc dane na kolejne podgrupy, analizują wszystkie zmienne w zbiorze danych, by znaleźć zmienną zapewniającą najlepszą klasyfikację lub predykcję. Proces jest rekursywny, a grupy są dzielone na coraz mniejsze jednostki aż do ukończenia drzewa (zgodnie z określonym kryterium zatrzymania). Zmienne przewidywane i wejściowe używane do budowania drzewa mogą być ilościowe (przedział liczbowy) lub jakościowe, w zależności od algorytmu. Jeśli zmienna przewidywana jest ilościowa, generowane jest drzewo regresji; jeśli zmienna przewidywana jest jakościowa, generowane jest drzewo klasyfikacji. Popularne algorytmy dzielenia obejmują minimalizację Gini Impurity (używanego przez CART) lub maksymalizację Information Gain (używanego przez ID3, C4.5) [13].

Przegląd istniejącej literatury pokazuje, że do najczęściej stosowanych algorytmów drzewa decyzyjnego należą algorytm Iterative Dichotomiser 3 (ID3), algorytm C4.5, CHAID oraz CART algorytm. Wśród tych algorytmów są pewne różnice, z których jedną jest możliwość modelowania różnych typów danych. Ponieważ zestaw danych może być skonstruowany na podstawie różnych typów danych, np. danych kategorycznych, danych numerycznych lub kombinacji obu, istnieje potrzeba użycia odpowiedniego algorytmu drzewa decyzyjnego, który może obsługiwać określony typ danych wykorzystywanych w zestawie danych. Wszystkie wyżej wymienione algorytmy mogą wspierać modelowanie danych jakościowych, podczas gdy tylko algorytm C4.5, C5.0 i algorytm CART mogą być używane do modelowania danych numerycznych (Tab. 1). Inną różnicą między tymi algorytmami jest proces opracowywania modeli, szczególnie na etapie budowania i przycinania drzew. Algorytmy ID3, C4.5 i C5.0 dzielą model drzewa tyle rozgałęzień ile można osiągnoć w danym momenci, podczas gdy algorytm CART obsługuje jedynie binarne podziały. Z kolei mechanizmy przycinania zlokalizowane w algorytmach C4.5, C5.0 i CART wspierają usuwanie nieistotnych węzłów i rozgałęzień. Natomiast algorytm CHAID nie wymaga dodatkowego etapu przycinania drzewa, z uwagi na fakt że CHAID stara się zapobiegać przeuczeniu od samego początku, wykorzystując pomysł wstępnego przycinania (węzeł jest dzielony tylko wtedy, gdy spełnione jest kryterium istotności) [9, 13].

5.1 Algorytm CART

Algorytm CART (ang. Classification and Regression Trees) jest popularnym algorytmem uczenia drzew decyzyjnych. W odróżnieniu od ID3 i C4.5, drzewo uczenia się w tym przypadku może być używane zarówno do klasyfikacji wieloklasowej, jak i do regresji w zależności od rodzaju zmiennej zależnej. Proces budowy drzewa składa się z rekurencyjnego dwójkowego podziału węzłów. Aby znaleźć najlepszy podział w każdym węźle, rozważane są wszystkie możliwe podziały wszystkich dostępnych atrybutów predykcyjnych. Najlepszy podział to taki, który maksymalizuje pewne kryterium podziału. Do zadań klasyfikacyjnych, tj. gdy atrybut zależny jest kategoryczny, jako kryterium podziału stosuje się indeks

Giniego. Do zadań regresyjnych, tj. gdy zmienna zależna jest ciągła, stosowane jest metoda najmniejszych kwadratów [3, 6, 12].

5.2 Algorytm CHAID

CHAID to algorytm do uczenia drzew decyzyjnych zaproponowany przez Kassa (1980). Działa podobnie do CART - oba mogą być używane zarówno do klasyfikacji, jak i regresji. Ale w przeciwieństwie do CART, CHAID wewnętrznie obsługuje tylko kategoryczne funkcje. Funkcje ciągłe są najpierw konwertowane na zmienne kategoryczne za pomocą grupowania/kubełkowania (ang. binning). Liczba kubełków (ang. bins) (K) musi być dostarczona przez użytkownika. Biorąc pod uwagę K, predykator jest podzielony w taki sposób, że wszystkie kubełki mają mniej więcej taką samą liczbę różnych wartości predykcyjnych. Maksymalna wartość funkcji w każdym pojemniku jest używana jako punkt przerwania [3, 14].

Ważnym parametrem w procesie wzrostu drzewa CHAID jest wartość p. Wartość p jest miarą używaną do decydowania o tym, które kategorie wartości predykcyjnych mają się łączyć podczas łączenia, a także do decydowania o najlepszym atrybucie podczas dzielenia. Wartość p jest obliczana przy użyciu różnych metod testowania hipotez w zależności od rodzaju zmiennej zależnej (nominalnej, porządkowej lub ciągłej) [3].

5.3 Algorytm ID3

ID3 jest prostym algorytmem uczenia drzewa decyzyjnego opracowanym przez Quinlana (1986). ID3 ma zastosowanie tylko w przypadkach, w których atrybuty (lub cechy) definiujące przykłady danych mają charakter kategoryczny, a przykłady danych należą do wcześniej zdefiniowanych, wyraźnie odróżnialnych (tj. dobrze zdefiniowanych) klas. ID3 to iteracyjny chciwy algorytm, który rozpoczyna się od węzła głównego i ostatecznie buduje całe drzewo. W każdym węźle wybiera się "najlepszy" atrybut do klasyfikacji danych. Atrybut "najlepszy" jest wybierany przy użyciu metryki Information gain. Po wybraniu atrybutu w węźle, przykłady danych w węźle są podzielone na podgrupy na podstawie wartości atrybutów, które mają. Zasadniczo wszystkie przykłady danych o tej samej wartości atrybutu są umieszczane w tej samej podgrupie. Podgrupy tworzą dzieci obecnego węzła, a algorytm jest powtarzany dla każdego z nowoutworzonych węzłów potomnych. Trwa to dopóki wszystkie elementy danych węzła nie należą do tej samej klasy lub wszystkie atrybuty zostaną wyczerpane [3, 4].

5.4 Algorytm C4.5

Algorytm C4.5 jest rozszerzeniem algorytmu ID3. Ma dodatkową możliwość obsługi ciągłych atrybutów i atrybutów z brakującymi wartościami. Proces budowy drzew w przypadku C4.5 jest taki sam jak w przypadku ID3 - znajdowanie najlepszego podziału w każdym węźle przy użyciu metryki Information gain. Jednak w przypadku atrybutu ciągłego algorytm C4.5 musi wykonać dodatkowy krok przekształcania go w dwuwartościowy atrybut kategoryczny, dzieląc około odpowiedniego progu. Próg ten jest wybierany w taki sposób, że wynikowy podział daje maksymalne wzmocnienie informacji [3, 15].

5.5 Algorytm C5.0

C5.0 to najnowsza wersja Quinlana na podstawie licencji firmowej. Wykorzystuje mniej pamięci i buduje mniejsze zestawy reguł niż C4.5, a jednocześnie jest bardziej dokładna [16].

6 Przycinanie drzew decyzyjnych

Wydajność drzewa można dodatkowo zwiększyć przez przycinanie. Polega ono na usunięciu gałęzi, które korzystają z funkcji o niskim znaczeniu. W ten sposób zmniejszamy złożoność drzewa, a tym samym zwiększamy jego moc predykcyjną, zmniejszając przeuczenie. Przycinanie może rozpocząć się od korzenia lub liści. Najprostsza metoda przycinania rozpoczyna się na liściach i usuwa każdy węzeł z najbardziej popularną klasą w tym liściu, zmiana ta jest zachowana, jeśli nie pogarsza dokładności. Bardziej zaawansowaną metodą przycinania, jest przycinanie kosztów, gdy parametr uczenia (alfa) jest używany do ważenia, czy węzły mogą być usunięte w oparciu o rozmiar pod-drzewa. [6, 10].

7 Biblioteka scikit-learn

8 Opis programu

9 Wnioski

10 Opisanie danych wykorzystanych do testowania działania metody wybranej biblioteki

11 Omówienie głównych części programu/skryptu i wyników

12 Wnioski końcowe odnośnie biblioteki i projektu

Literatura

[1] B. W., Matching and prediction on the principle of biological classification, JRSS, Series C, Applied Statistics 8 (2).

[2] J. Morgan, J. Sonquist, Problems in the analysis of survey data, and a proposal, J. Amer. Statist. Assoc. 58 (1963) 415–434.

[3] M. P., Decision trees.
URL http://www.shogun-toolbox.org/static/notebook/current/DecisionTrees.html

[4] J. R. Quinlan, Induction of decision trees, Machine Learning 1 (1) (1986) 81–106. doi:10.1007/bf00116251.
URL https://doi.org/10.1007/bf00116251

[5] J. Quinlan, Decision trees and decision-making, IEEE Transactions on Systems, Man, and Cybernetics 20 (2) (1990) 339–346. doi:10.1109/21.52545.
URL https://doi.org/10.1109/21.52545

[6] B. L., F. J., O. R., S. C., Classification and Regression Trees, Taylor & Francis Ltd, Wadsworth, Belmont, CA, 1984.

[7] G. E., Symboliczne metody klasyfikacji danych, PWN, Warszawa, 1998.

[8] L. P., P.-P. G., T. R., Metody sztucznej inteligencji i ich zastosowania w ekonomii i zarządzaniu, Wydawnictwo Akademii Ekonomicznej, Kraków, 2007.

[9] A. V. C. Team, A complete tutorial on tree based modeling from scratch (in r & python) (2016).
URL https://www.analyticsvidhya.com/blog/2016/04/complete-tutorial-tree-based-modeling-scratch-in-python/

[10] L. H., Decision trees and random forests for classification and regression pt.1.
URL https://towardsdatascience.com/decision-trees-and-random-forests-for-classification-and-regression-pt-1-dbb65a458df

[11] AIspace, Tutorial 4: Splitting algorithms.
URL http://aispace.org/dTree/help/tutorial4.shtml

[12] B. J., How to implement the decision tree algorithm from scratch in python (2016).
URL https://machinelearningmastery.com/implement-decision-tree-algorithm-scratch-python/

[13] Decision tree applications for data modelling (artificial intelligence).
URL http://what-when-how.com/artificial-intelligence/decision-tree-applications-for-data-modelling-artificial-intelligence/

[14] G. V. Kass, An exploratory technique for investigating large quantities of categorical data, Applied Statistics 29 (2) (1980) 119. doi:10.2307/2986296.
URL https://doi.org/10.2307/2986296

[15] J. R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[16] scikit learn, Decision Trees.
URL http://scikit-learn.org/stable/modules/tree.html#tree-algorithms-id3-c4-5-c5-0-and-cart