Lab 3 - drzewa decyzyjne i lasy losowe Drzewa decyzyjne są klasyfikatorami dodatkowo ilustrującymi proces decyzyjny w sposób graficzny. Przykładowe drzewo decyzyjne Poniższe drzewo decyzyjne ilustruje proces podejmowania decyzji odnośnie gry w tenisa w zależności od warunków pogodowych.

Słoneczna Wilgotność Niska Wysoka

Pogoda

Pochmurna

Tak

n

 p_{i}

 $p(f_i)$

rano

rano

rano

rano

rano

rano

rano

rano

Gain(S, sezonletni) = 0.61

Gain(S, poradnia) = 0.2816

Deszczowa

 $S_{slonecznie}$

 $S_{slonecznie}$

 $S_{wietrznie}$

 $S_{slonecznie}$

 $Entropy(S_{slonecznie}) = 0.918$

 $Gain(S_{slonecznie}, sezonletni) = 0.918 - rac{|S_{tak}|}{|S|} Entropy(S_{tak}) - rac{|S_{nie}|}{|S|} Entropy(S_{nie}) = 0.918 - rac{|S_{nie}|}{|S|} Entropy(S_{nie})$

 $0.918 - \frac{1}{3} * 0 - \frac{2}{3} * 0 = 0.918$

 $Gain(S_{slonecznie}, poradnia) = 0$

 $S_{slonecznie}$

 $S_{slonecznie}$

pogoda

10 słonecznie nie

słonecznie tak

słonecznie nie

Nie

Dolina

Pogoda

Wietrznie

Sezon letni

Nie -

Pora dnia

- Rano **–**Popołudnie

no

no

yes

yes

yes

no

yes

no

yes

yes

yes

yes

no

Szczyt

Tak -

Morskie oko

Outlook Temp Humidity Windy Play

High

High

High

High

Normal

Normal

Normal

Normal

Normal

Normal

Normal

High

High

 $Precision = rac{TP}{TP + FP}$

 $Sensitivity = \frac{TP}{TP + FN}$

 $Accuracy = rac{TP + TN}{Total}$

Za pomocą precyzji można określić trafność klasyfikacji, czyli odsetek wszystkich pozytywnych obiektów, które zostały poprawnie sklasyfikowane. Czułość służy do określania trafności wyszukiwania

High

Hot

Hot

Hot

Mild

Cool

Cool

Cool

Mild

Cool

Mild

Mild

Mild

Hot

Mild

Szczyt

Rainy

Rainy

Sunny

Sunny

Sunny

Rainy

Rainy

Sunny

Rainy

Overcast

Overcast

Sunny

Overcast

Overcast

sezon letni pora dnia cel wycieczki

szczyt

dolina

dolina

Pogoda

Pochmurna

Deszczowa

Deszczowo

Sezon letni

Nie -

Krupówki

Tak ·

Szczyt

rano

rano

rano

pora dnia cel wycieczki

popołudnie szczyt

popołudnie szczyt

szczyt

dolina

szczyt

Krupówki

Morskie Oko

szczyt

szczyt

dolina

sezon letni

Deszczowa

Silny

Nie

Wiatr

Słaby

Tak

Tak Nie

1. Jaką decyzję podejmiemy przy słonecznej pogodzie i wysokiej wilgotności powietrza? 2. Jaką decyzję podejmiemy przy pogodzie pochmurnej i niskiej wilgotności powietrza? Uczenie (formowanie) drzewa decyzyjnego odbywa się na podstawie danych w formie systemu decyzyjnego.

Uczenie drzewa decyzyjnego

Metodę uczenia drzewa decyzyjnego można uogólnić następującymi krokami: • znalezienie atrybutu w systemie treningowym atrybutu niosącego najwięcej informacji, a następnie podzielenie systemu na kategorie według tego atrybutu. • powtarzanie procesu do osiągnięcia pewnego założonego progu Do podziału systemu decyzyjnego na podkategorie można wykorzystać następujące metryki:

 Zysk informacyjny • Niejednorodność Giniego

 $Gain(S,A) = Entropy(S) - \sum_{v \in Values(A)} rac{|S_v|}{|S|} Entropy(S_v)$

Zysk informacyjny Zysk informacyjny jest miarą zgodności badanego atrybutu z celem. Miarę zysku informacyjnego można wyrazić wzorem:

gdzie: • S oznacza zbiór obiektów w systemie decyzyjnym • A oznacza badany atrybut $Entropy(S) = \sum_{i=1}^n -p_i log_2(p_i)$, gdzie: oznacza liczbę klas atrybutów

oznacza prawdopodobieństwo wystąpienia wartości atrybutu w systemie decyzyjnym Niejednorodność Giniego Niejednorodność Giniego jest miarą probabilistyczną definiującą prawdopodobieństwo przydzielenia błędnej decyzji za pomocą badanego atrybutu, co można wyrazić następującym wzorem: $I_G(A)=1-\sum_{i=1}^n p(A_i)^2$ gdzie:

 f oznacza badany atrybut • n oznacza liczbę wartości w badanym atrybucie

oznacza prawdopodobieństwo wystąpienia wartości badanego atrybutu w i-tej klasie decyzyjnej Podczas wyznaczania podziału systemu decyzyjnego należy wybrać atrybut, dla którego wartośc zysku informacyjnego jest największa lub wartość niejednorodności Giniego jest najmniejsza.

Przykład uczenia drzewa decyzyjnego

Rozważmy następujący system decyzyjny obrazujący cele wycieczek w Tatrach pewnego turysty A: pogoda słonecznie tak słonecznie nie wietrznie

tak deszczowo tak deszczowo nie deszczowo tak nie wietrznie wietrznie wietrznie tak 10 słonecznie nie Na podstawie tej bazy wiedzy można zbudować drzewo decyzyjne obrazujace schemat podejmowania decyzji przez turystę A.

Pierwszym krokiem budowy drzewa decyzyjnego będzie wyznaczenie atrybutu zachowującego największy zysk informacyjny:

Zatem korzeniem drzewa zostanie atrybut pogoda i od tego węzła należy wyprowadzić tyle krawędzi ile jest wartości w atrybucie:

nie zakończy się liściem drzewa decyzyjnego. W tym celu należy wybrać kolejny atrybut, który rozdzieli obiekty należące do tego podzbioru.

Po utworzeniu korzenia drzewa należy przyjrzeć się klasom decyzyjnym przypisanym wartościom atrybutu

 $Entropy(S) = -p(szczyt)log_2p(szczyt) - p(dolina)log_2p(dolina) - p(MorskieOko)log_2p(MorskieOko) - p(Krup\acute{o}wki)log_2p(Krup\acute{o}wki) = -p(szczyt)log_2p(krup\acute{o}wki)log_2p(krup\acute{o}wki) - p(krup\acute{o}wki)log_2p(krup\acute{o}wki)log_2p(krup\acute{o}wki) - p(krup\acute{o}wki)log_2p(krup\acuteowki)log_2p(krup\acute$ $=-rac{6}{10}log_2rac{6}{10}-rac{2}{10}log_2rac{2}{10}-rac{1}{10}log_2rac{1}{10}-rac{1}{10}log_2rac{1}{10}=0$ $= -\frac{6}{10} * -0.737 - \frac{2}{10} * -2.322 - \frac{1}{10} * -3.322 - \frac{1}{10} * -3.322 =$ = 0.4422 + 0.4644 + 0.3322 + 0.3322 = 1.571 $Gain(S, Pogoda) = 1.571 - rac{|S_{slonecznie}|}{10} Entropy(S_{slonecznie}) - rac{|S_{wietrznie}|}{10} Entropy(S_{wietrznie}) - rac{|S_{deszczowo}|}{10} Entropy(S_{deszczowo}) = 0$ = 1.571 - 0.3 * 0.918 - 0.4 * 0.81125 - 0.3 * 0.918 = 0.70

Pogoda Słoneczna Wietrzna

Podobnie sytuacja wygląda w przypadku podzbioru W tym celu należy wyznaczyć zysk informacyjny dla obiektów znajdujących się w podzbiorach gałęzi oraz atrybutów z wyjątkiem użytego przy budowie korzenia drzewa decyzyjnego. Przy wyznaczaniu zysków informacyjnych należy brać pod uwagę tylko te atrybuty, które nie zostały wykorzystane w podziale wyższych poziomów drzewa decyzyjnego. Rozważmy podzbiór obiektów dla odgałęzienia

. Są to klasy {szczyt, dolina}, zatem krawędź

Zatem do podziału podzbioru

Tak

Szczyt Powtarzając powyższy schemat można zbudować pełne drzewo decyzyjne, które dla przedstawionego systemu decyzyjnego prezentuje się następująco:

Dolina Szczyt

Precyzja i czułość

gdzie:

Zadanie

Pakiet Scikit learn udostępnia implementacje drzewa decyzyjnego zarówno jako predyktora w procesie klasyfikacji, jak i regresji. Klasyfikacja przy użyciu drzewa decyzyjnego In [1]: import pandas as pd from sklearn.model_selection import cross_val_score from sklearn.tree import DecisionTreeClassifier Wczytanie systemu decyzyjnego df = pd.read_csv('data/data-samples.csv')

glucose bloodpressure diabetes

85

63

80

73

df.head()

40

40

45

45

40

Utworzenie obiektu klasyfikatora

clf = DecisionTreeClassifier()

0

2

4

Out[3]:

In [8]: Regresja przy użyciu drzewa decyzyjnego

precyzja czułość 2. Korzystając ze źródła http://archive.ics.uci.edu/ml/datasets.php wybrać zbiór danych z przeznaczeniem regresyjnym. Sprawdzić przy użyciu 10-krotnej walidacji krzyżowej MAE modelu na wybranym zbiorze. Wizualizacja procesu decyzyjnego przy użyciu drzewa decyzyjnego Pakiet Scikit learn udostępnia interfejs służący wygodnej wizualizacji drzew decyzyjnych. Dane treningowe

from sklearn import datasets

from sklearn import tree

dokładność

import matplotlib.pyplot as plt iris = datasets.load_iris(as_frame=True) # zbior danych iris X = iris.data y = iris.target iris.frame sepal length (cm) sepal width (cm) petal length (cm) petal width (cm) target Out[9]: 0 5.1 3.5 1.4 4.9 3.0 1.4 2 4.7 3.2 1.3

6.5

6.2

5.9

147 148 149 150 rows × 5 columns Uczenie (formowanie drzewa decyzyjnego) In [15]:

clf, plt.plot() Out[16]: []

In [16]:

Zadanie Zwizualizować przy użyciu pakietu Scikit learn proces decyzyjny turysty A na urlopie w Zakopanem. Należy zwrócić uwagę na stosowanie wartości liczbowych w atrybutach systemu decyzyjnego, zatem każda wartość (np. słonecznie, rano, itp.) powinna otrzymać wartość liczbową.

In [20]:

Uczenie modelu lasu losowego Stworzymy model lasu losowego składający się z 32 drzew decyzyjnych (hiperparametr n_estimators) clf = RandomForestClassifier(n_estimators=32, n_jobs=-1) # n_jobs=-1 oznacza wykorzystanie wszystkich dostępnych rdzeni clf.fit(X_train, y_train) Przydzielanie decyzji z wykorzystaniem modelu lasu losowego In [22]: prediction = clf.predict([[5., 3.1, 1.2, 0.3]]) prediction Out[22]: array([0])

Zadanie

Zadanie

Lasy losowe w ocenie istotności atrybutów systemu decyzyjnego W wielu przypadkach nieistotne atrybuty (niosące niewielką ilość informacji) mogą wpływać negatywnie na jakość prognoz. Model lasu losowego umożliwia sprawdzenie istotności każdego z atrybutów poprzez sprawdzenie w jakim stopniu węzły każdego drzewa decyzyjnego wykorzystujące tę cechę zmniejszają zanieczyszczenie poprzez obliczenie średniej ważonej gdzie waga każdego węzła odpowiada liczebności powiązanych z nim obiektów z systemu decyzyjnego. Pakiet Scikit learn umożliwia wygodny interfejs do oceny istotności atrybutów, który można wykorzystać następująco: In [23]: for i, f_importance in enumerate(clf.feature_importances_): print(f'Atrybut: {iris.feature_names[i]}, istotność: {f_importance}') Atrybut: sepal length (cm), waga: 0.011587603285713902 Atrybut: sepal width (cm), waga: 0.0 Atrybut: petal length (cm), waga: 0.6570335847127355 Atrybut: petal width (cm), waga: 0.33137881200155045

wynik z modelem drzewa decyzyjnego.

użyjemy atrybutu "sezon letni", który zachowuje większy zysk informacyjny. Atrybut "pora dnia" dla tego podzbioru nie wnosi żadnej wartości. Drzewo decyzyjne po podziale podzbioru będzie wyglądało następująco: Słoneczna

Sezon letni

Słonecznie Sezon letni Tak Nie -

Do narysowania drzewa można wykorzystać narzędzie https://app.diagrams.net Miary precyzji i czułości pozwalają na określenie jak klasyfikator radzi sobie z trafnością przydzielanych decyzji, co można wyrazić następującymi wzorami:

• TP oznacza liczbę obiektów poprawnie sklasyfikowanych jako pozytywne

• TN oznacza liczbę obiektów poprawnie sklasyfikowanych jako negatywne

• FP oznacza liczbę obiektów fałszywie sklasyfikowanych jako pozytywne

• FN oznacza liczbę obiektów fałszywie sklasyfikowanych jako negatywne

Drzewo decyzyjne w pakiecie Scikit learn

0

wszystkich pozytywnych obiektów, czyli odsetka obiektów sklasyfikowanych jako pozytywne spośród wszystkich pozytywnych obiektów.

Zbudować drzewo decyzyjne na podstawie następującego systemu decyzyjnego:

5-krotna walidacja krzyżowa dokładności modelu drzewa decyzyjnego cross val score(clf, df.iloc[:, 0:2], df.iloc[:, 2], cv=5) array([0.96482412, 0.90954774, 0.91457286, 0.90452261, 0.92964824]) W celu zbudowania regresora metodą drzewa decyzyjnego należy utworzyć obiekt klasy DecisionTreeRegressor In [9]: from sklearn.tree import DecisionTreeRegressor regressor = DecisionTreeRegressor() Zadania 1. Korzystając ze źródła http://archive.ics.uci.edu/ml/datasets.php wybrać dowolny system decyzyjny z binarnym atrybutem decyzyjnym, a następnie dokonać klasyfikacji przy użyciu drzewa decyzyjnego z pakietu Scikit learn. Sprawdzić następujące parametry klasyfikacji za pomocą 5-krotnej walidacji krzyżowej:

3 4.6 3.1 1.5 4 5.0 3.6 1.4 145 6.7 3.0 5.2 6.3 2.5 5.0 146

3.0

3.4

3.0

5.2

5.4

5.1

0

0

0

2

2

petal length (cm) \leq 2.45 entropy = 1.585samples = 150value = [50, 50, 50]class = setosa

entropy = 0.0

samples = 50

value = [50, 0, 0]

class = setosa

entropy = 0.918

samples = 6

value = [0, 2, 4]

class = virginica

entropy = 0.0

samples = 2

value = [0, 2, 0]class = versicolor

Las losowy to klasyfikator składający się z drzew decyzyjnych uczony metodą agregacji, która polega na uczeniu składowych klasyfikatorów za pomocą obiektów pochodzących z systemu decyzyjnego

Korzystając ze źródła http://archive.ics.uci.edu/ml/datasets.php wybrać dowolny system decyzyjny, a następnie dokonać klasyfikacji przy użyciu modelu lasu losowego z pakietu Scikit learn. Porównać

Zbadać istotność atrybutów z systemu decyzyjnego użytego w poprzednim zadaniu i wybrać empirycznie tylko najbardziej istotne. Dokonać klasyfikacji metodą lasu losowego, a następnie porównać

dobieranych metodą losowania ze zwracaniem. Zatem każde z drzew decyzyjnych wchodzące w skład lasu losowego jest uczone na podstawie innych obiektów.

petal length (cm) \leq 5.45

entropy = 0.918

samples = 3

value = [0, 2, 1]

class = versicolor

entropy = 0.0

samples = 1value = [0, 0, 1]

class = virginica

petal width (cm) ≤ 1.75

entropy = 1.0

samples = 100

value = [0, 50, 50]

class = versicolor

petal length (cm) \leq 4.85

entropy = 0.151

samples = 46

value = [0, 1, 45]class = virginica

entropy = 0.0

samples = 2

value = [0, 0, 2]

class = virginica

entropy = 0.0

samples = 43

value = [0, 0, 43]

class = virginica

sepal length (cm) <= 5.95

entropy = 0.918

samples = 3

value = [0, 1, 2]

class = virginica

entropy = 0.0

samples = 1

value = [0, 1, 0]

class = versicolor

0.2

0.2

0.2

0.2

0.2

2.3

1.9

2.0

2.3

1.8

from sklearn.tree import DecisionTreeClassifier

clf = DecisionTreeClassifier(criterion='entropy') clf.fit(X, y) DecisionTreeClassifier(criterion='entropy') Wizualizacja drzewa decyzyjnego fig = plt.figure(figsize=(30,20)) tree.plot_tree(feature_names=iris.feature_names, class_names=iris.target_names, filled=True

petal length (cm) <= 4.95 entropy = 0.445samples = 54value = [0, 49, 5]class = versicolor petal width (cm) \leq 1.65 petal width (cm) <= 1.55 entropy = 0.146samples = 48value = [0, 47, 1]class = versicolorentropy = 0.0entropy = 0.0entropy = 0.0samples = 3samples = 47samples = 1value = [0, 0, 1]value = [0, 0, 3]value = [0, 47, 0]class = versicolor class = virginica class = virginica

Lasy losowe i uczenie zespołowe

from sklearn.model_selection import train_test_split from sklearn.ensemble import RandomForestClassifier

Dane treningowe i testowe

from sklearn import datasets

iris = datasets.load_iris()

X = iris.data y = iris.target

Finalna decyzja jest dominantą decyzji podjętych przez składowe klasyfikatory modelu.

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=.3)

Dzięki takiemu zabiegowi można wybrać atrybuty z systemu decyzyjnego o największej istotności.

wyniki z modelem drzewa decyzyjnego, naiwnego klasyfikatora bayesowskiego i modelu knn (z pakietu Scikit learn).

Przykład wykorzystania modelu lasu losowego pochodzącego z pakietu Scikit learn