Projekt UMA - dokumentacja wstępna

- Mikołaj Garbowski
- Michał Pałasz

Temat

Implementacja drzewa decyzyjnego, porównanie sposobu radzenia sobie z problemami wieloklasowymi, czyli porównanie jakości wyników typowej implementacji ID3 z jakością wyników dwóch podejść:

1) tworzymy osobny model binarny dla każdej klasy (jedna klasa traktowana jako pozytywna, wszystkie pozostałe jako negatywne), predykcja przez wybór klasy o maksymalnej wartości funkcji decyzyjnej (wymaga posiadania przez każdy klasyfikator „stopnia pewności siebie", co można zdefiniować na wiele sposobów).
2) tworzymy osobny model binarny dla każdej pary klas (jedna klasa traktowana jako pozytywna, druga jako negatywna), predykcja przez głosowanie.

Opis algorytmu

Algorytm ID3

Opracowane na podstawie wykładów z przedmiotów UMA (2024Z, Paweł Cichosz) i WSI (2023Z, Paweł Zawistowski)

Algorytm ID3 jest metodą indukcji drzew decyzyjnych. Drzewo budowane jest rekurencyjnie zaczynając od korzenia. Węzły nieterminalne odpowiadają podziałowi zbioru uczącego na podstawie wartości wybranego atrybutu. Gałęzie odpowiadają konkretnej wartości atrybutu użytego do podziału. Liście zawierają klasę (predykcję) i stopień pewności (rozkład prawdopodobieństwa klas).

Proces predykcji dla wybranego przykładu polega na przejściu od korzenia drzewa do liścia, w każdym węźle wybierając gałąź, która odpowiada wartości atrybutu użytego do podziału w przykładzie, wynikiem predykcji jest klasa w liściu.

Pseudokod

```
Wejścje:
    Y - zbiór klas
    D - zbiór atrybutów wejściowych
    U - zbiór par uczących

jeśli jednakowa klasa y dla wszystkich przykładów
    zwróć liść z klasą y

jeśli D jest pusty
    zwróć liść zawierający klasę większościową w T

d = argmax InfGain(d, T)
{T_1, T_2, ...} = podział zbioru T po atrybucie d
węzły potomne = {ID3(Y, D-{d}, T_1), ID3(Y, D-{d}, T_2), ...}

zwróć drzewo z korzeniem d i gałęziami prowadzącymi do węzłów potomnych
```

Podział zbioru uczącego w węźle Algorytm ID3 stosuje podziały wielowartościowe na podstawie wartości atrybutu - z węzła (nieterminalnego) wychodzi po 1 gałęzi dla każdej wartości atrybutu dyskretnego.

Do podziału zbioru atrybutów w węźle wybiera się ten atrybut, który daje największą zdobycz informacyjną przy podziale definiowaną jako:

$$InfGain(d, T) = I(T) - Inf(d, T)$$

$$Inf(d, T) = \sum_j \frac{|T_j|}{|T|} \cdot I(T_j)$$

$$I(T) = -\sum_i f_i \cdot \log(f_i)$$

Gdzie:

- d - atrybut użyty do podziału
- T - zbiór trenujący
- T_j - podzbiór T, gdzie każdy przykład ma j-tą wartość atrybutu d
- f_i - częstość i-tej klasy w zbiorze
- $InfGain$ - zdobycz informacyjna
- Inf - entropia zbioru T podzielonego na podzbiory przez atrybut d
- I - entropia zbioru T

Kryterium stopu W kroku algorytmu tworzony jest liść jeśli wszystkie przykłady podzbioru zbioru uczącego mają jednakową klasę (wybierana jest ta klasa z pewnością 1) lub jeśli nie ma już atrybutów do kolejnego podziału - wybierana jest klasa większościowa z pewnością równą częstości występowania tej klasy w podzbiorze uczącym (f_i).

W przypadku, kiedy w przykładzie dla którego wyznaczana jest predykcja pojawia się wartość atrybutu, dla której nie istnieje gałąź w drzewie, wynikiem predykcji będzie klasa większościowa w węźle nieterminalnym (tym z którego brakuje gałęzi).

Przykładowe obliczenia Jednolita klasa

Dla zbioru trenującego

x_1	x_2	y
A	1	0
B	2	0
C	3	0

Wszystkie przykłady mają jednakową klasę więc tworzony jest liść z klasą 0 i pewnością 1.

Brak atrybutów do podziału

y
0
1
1

Nie ma atrybutów do dalszego podziału, tworzony jest liść z klasą większościową 1 i pewnością 2/3

Obliczanie zdobyczy informacyjnej

Dla T:

x_1	x_2	y
A	1	0
B	1	1
B	2	1
B	2	0
B	3	1

Podział po atrybucie x_1

T_A:

x_1	x_2	y
A	1	0

T_B:

x_1	x_2	y
B	1	1
B	2	1
B	2	0
B	3	1

$$I(T) = -2/5 \log(2/5) - 3/5 \log(3/5) \simeq 0,67$$

$$I(T_A) = 0$$

$$I(T_B) = -1/4 \log(1/4) - 3/4 \log(3/4) \simeq 0,56$$

$$InfGain(x_1, T) = I(T) - \frac{|T_A|}{|T|} I(T_A) - \frac{|T_B|}{|T|} I(T_B)$$

$$InfGain(x_1, T) \simeq 0,67 - 0,2 \cdot 0 - 0,8 \cdot 0,56 \simeq 0,22$$

Wariant z n klasyfikatorów binarnych

Dla problemu klasyfikacji z liczbą klas równą n powstanie n modeli klasyfikacji binarnej według algorytmu ID3.

Dla pojedynczego modelu dla klasy c, modyfikujemy etykiety w zbiorze danych - przypisujemy klasę pozytywną w miejsce klasy c, przypisujemy klasę negatywną w miejsce wszystkich pozostałych.

Przy tworzeniu liścia, poza klasą większościową zapamiętujemy również częstość występowania przykładów klasy większościowej w zbiorze przykładów rozważanym w danym liściu. W przypadku jednakowej klasy dla wszystkich przykładów - wartość 1, w pozostałych przypadkach - wartość z przedziału $(0, 1)$. Częstość potraktujemy jako stopień pewności modelu co do decyzji.

Wynikiem predykcji zespołu modeli będzie ta klasa, którą model binarny zaklasyfikował pozytywnie z największą pewnością.

Przykładowo, dla klas A, B, C, D

Model binarny dla klasy	A	B	C	D
Predykcja (0/1)	1	0	1	0
Pewność	0,8	0,7	0,7	0,9

Predykcją zespołu modeli będzie klasa A, ponieważ zarówno model binarny dla klasy A i C dał pozytywny wynik klasyfikacji, ale model dla klasy A miał większą pewność.

Jeśli wszystkie modele binarne dadzą predykcję negatywną, predykcją zespołu modeli będzie ta klasa, która została zaklasyfikowana negatywnie z najmniejszą pewnością. Przykład:

Model binarny dla klasy	A	B	C	D
Predykcja (0/1)	0	0	0	0
Pewność	0,9	0,7	0,8	0,5

Predykcją zespołu modeli będzie klasa D, ponieważ wszystkie predykcje dają klasę negatywną, ale model dla klasy D ma najmniejszą pewność

Wariant z głosowaniem

Dla problemu klasyfikacji z liczbą klas równą n powstanie $n(n-1)/2$ modeli klasyfikacji binarnej - po 1 dla każdej pary klas. Klasyfikator binarny rozstrzyga do której klasy z pary należy przykład.

Do trenowania klasyfikatora binarnego dla pary klas A i B użyjemy takiego podzbioru zbioru trenującego, który zawiera tylko przykłady klas A i B.

Predykcja zespołu klasyfikatorów będzie wyznaczana przez głosowanie. Ze względu na możliwość remisu przy zliczaniu głosów jako liczby modeli które przewidują daną klasę, proponujemy poniższy sposób obliczania głosów ważonych stopniem pewności predykcji (stopień predykcji definiowany jak w poprzednim wariancie).

Przykład dla klas A, B i C:

Model binarny dla pary	A vs B	B vs C	C vs A
Predykcja	A	B	C
Pewność	0,99	0,8	0,7

Klasa	Ważony głos
A	1,29 = 0,99 + (1-0,7)
B	0,81 = (1-0,99) + 0,8
C	0,9 = (1-0,8) + 0,7

Dla powyższego przykładu predykcją zespołu klasyfikatorów byłaby klasa A, ponieważ choć wszystkie uzyskały po 1 głosie, dla każdej klasy obliczamy sumę pewności głosów *za* i dopełnienia do 1 pewności głosów *przeciw*.

Plan eksperymentów

Dla każdego z 3 klasyfikatorów porównamy miary jakości osiągane na każdym ze zbiorów danych (miary jakości i zbiory danych opisane w kolejnych sekcjach).

Miary jakości

W celu oceny jakości klasyfikacji zastosowane zostaną poniższe metryki, które pozwalają na dokładną analizę wyników modeli. Aby zastosować miary jakości *recall*, *precision*, *f-measure*, i *specificity* w przypadku drzewa decyzyjnego klasyfikującego do więcej niż dwóch klas, zostaną wykorzystane następujące podejścia:

Binaryzacja problemu wieloklasowego

Dla każdej klasy k, miary jakości zostaną obliczone przy zastosowaniu podejścia OvR (One vs Rest)

- klasa k jest traktowana jako pozytywna, a wszystkie inne jako negatywne

Definicje miar jakości

- TP - liczba próbek poprawnie zaklasyfikowanych do klasy k
- FP - liczba próbek błędnie zaklasyfikowanych do klasy k
- TN - liczba próbek poprawnie zaklasyfikowanych jako należące do innych klas
- FN - liczba próbek należących do klasy k, ale zaklasyfikowanych do innych klas

Odzysk (Recall)

Współczynnik prawdziwych pozytywnych, określający, jak dobrze klasyfikator potrafi znaleźć wszystkie pozytywne przypadki.

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

Im wyższy współczynnik, tym więcej przypadków pozytywnych zostało prawidłowo wykrytych.

Precyzja (Precision)

Stosunek liczby prawdziwie pozytywnych wyników do liczby wszystkich przypadków zaklasyfikowanych jako pozytywne.

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

Ocenia jak dobre są wyniki pozytywne, czyli ile spośród nich jest faktycznie prawdziwie pozytywnych.

Miara F (F-measure)

Średnia harmoniczna precyzji i odzysku. Balansuje oba wskaźniki, szczególnie w przypadku, gdy jeden z nich jest znacznie wyższy od drugiego.

$$F = \frac{2 * Recall * Precision}{Recall + Precision}$$

Specyficzność (Specificity)

Współczynnik prawdziwych negatywnych, określający jak dobrze klasyfikator potrafi zidentyfikować negatywne przypadki.

$$Specificity = \frac{TN}{TN + FP}$$

Im wyższa wartość, tym lepsze odróżnienie przypadków negatywnych od fałszywie pozytywnych.

Uśrednianie wyników

Aby uzyskać jedną wartość podsumowującą wydajność klasyfikatora dla wszystkich klas, zostaną zastosowane dwa podejścia do uśredniania wyników:

- **Makro-uśrednianie**
  - Miary jakości są obliczane osobno dla każdej klasy k, a następnie ich wyniki są uśrednianie arytmetycznie
  - Każda klasa ma równą wagę, niezależnie od liczby próbek w danych

    $$Precision_{makro} = \frac{1}{K} \sum_{k=1}^{K} Precision_k$$

    Analogicznie jak w powyższym wzorze dla Recall, F-measure i Specificity.

- **Mikro-uśrednianie**
  - Wszystkie wartości TP, FP, FN, TN są sumowane dla wszystkich klas
  - Na podstawie tych zsumowanych wartości obliczane są globalne miary jakości
  - Metoda ta przypisuje wagę klasom proporcjonalnie do liczby próbek

    $$Precision_{mikro} = \frac{\sum_{k=1}^{K} TP_k}{\sum_{k=1}^{K} (TP_k + FP_k)}$$

    Analogicznie jak w powyższym wzorze dla Recall, F-measure i Specificity.

Analiza ROC

Analiza ROC zostanie zastosowana w kontekście wieloklasowym poprzez podejście One-vs-Rest:

- Dla każdej klasy zostanie wygenerowana osobna krzywa ROC, traktując tę klasę jako pozytywną, a pozostałe jako negatywne
- Dla każdej krzywej ROC obliczone zostanie pole pod krzywą (Area Under the Curve)

Wybór i opis zbiorów danych

Car Evaluation (https://archive.ics.uci.edu/dataset/19/car+evaluation)

Cel zbioru: Celem zbioru jest klasyfikacja samochodów na podstawie ich cech, takich jak cena, koszty utrzymania czy pojemność pasażerska, do jednej z czterech kategorii akceptowalności: - `unacc` (nieakceptowalny), - `acc` (akceptowalny), - `good` (dobry), - `vgood` (bardzo dobry).

Charakterystyka zbioru danych:

- **Liczba próbek:** 1 728
- **Liczba cech:** 6 atrybutów wejściowych + 1 cecha docelowa
- **Typy danych:** kategoryczne

Opis cech:

1. **buying** (cena zakupu):
   - Możliwe wartości: `vhigh`, `high`, `med`, `low`
2. **maint** (koszty utrzymania):
   - Możliwe wartości: `vhigh`, `high`, `med`, `low`
3. **doors** (liczba drzwi):
   - Możliwe wartości: `2`, `3`, `4`, `5more`
4. **persons** (liczba miejsc dla pasażerów):
   - Możliwe wartości: `2`, `4`, `more`
5. **lug_boot** (wielkość bagażnika):
   - Możliwe wartości: `small`, `med`, `big`
6. **safety** (poziom bezpieczeństwa):
   - Możliwe wartości: `low`, `med`, `high`

Nursery (https://archive.ics.uci.edu/dataset/76/nursery)

Cel zbioru: Celem zbioru jest klasyfikacja wniosków o przyjęcie dzieci do przedszkola na podstawie różnych kryteriów, takich jak sytuacja finansowa rodziny, liczba dzieci w rodzinie czy zdrowie dziecka. Każdy wniosek jest przypisywany do jednej z trzech kategorii akceptowalności: - `not_recom` (niezalecany), - `recommended` (zalecany), - `priority` (priorytetowy).

Charakterystyka zbioru danych:

- **Liczba próbek:** 12 960
- **Liczba cech:** 8 atrybutów wejściowych + 1 cecha docelowa
- **Typy danych:** kategoryczne

Opis cech:

1. **parents** (sytuacja rodziców):
   - Możliwe wartości: `usual`, `pretentious`, `great_pret`
2. **has_nurs** (potrzeba opieki pielęgniarskiej):
   - Możliwe wartości: `proper`, `less_proper`, `improper`, `critical`, `very_crit`
3. **form** (forma opieki):
   - Możliwe wartości: `complete`, `completed`, `incomplete`, `foster`
4. **children** (liczba dzieci w rodzinie):
   - Możliwe wartości: `1`, `2`, `3`, `more`
5. **housing** (warunki mieszkaniowe):
   - Możliwe wartości: `convenient`, `less_conv`, `critical`
6. **finance** (sytuacja finansowa rodziny):
   - Możliwe wartości: `convenient`, `inconv`
7. **social** (sytuacja społeczna):
   - Możliwe wartości: `nonprob`, `slightly_prob`, `problematic`
8. **health** (zdrowie dziecka):
   - Możliwe wartości: `recommended`, `priority`, `not_recom`

Balance Scale (https://archive.ics.uci.edu/dataset/12/balance+scale)

Cel zbioru: Celem zbioru jest klasyfikacja stanu równowagi szalki wagi na podstawie masy i odległości obiektów umieszczonych na jej lewym i prawym ramieniu. Każda próbka jest przypisywana do jednej z trzech kategorii: - `L` (szalka przechylona w lewo), - `B` (szalka w równowadze), - `R` (szalka przechylona w prawo).

Charakterystyka zbioru danych:

- **Liczba próbek:** 625
- **Liczba cech:** 4 atrybuty wejściowe + 1 cecha docelowa
- **Typy danych:** kategoryczne

Opis cech:

1. **Left-Weight** (waga na lewym ramieniu):
   - Możliwe wartości: liczby całkowite od `1` do `5`
2. **Left-Distance** (odległość na lewym ramieniu):
   - Możliwe wartości: liczby całkowite od `1` do `5`
3. **Right-Weight** (waga na prawym ramieniu):
   - Możliwe wartości: liczby całkowite od `1` do `5`
4. **Right-Distance** (odległość na prawym ramieniu):
   - Możliwe wartości: liczby całkowite od `1` do `5`

Żaden z powyższych zestawów danych nie posiada brakujących danych.